Mathematics in Industry 18

Evangelos Kranakis *Editor*

Advances in Network Analysis and its Applications

Mitacs

Springer

For further volumes:
http://www.springer.com/series/4650

Evangelos Kranakis
Editor

Advances in Network Analysis and its Applications

With 95 Figures and 51 Tables

Springer

Mitacs

Editor
Evangelos Kranakis
School of Computer Science
Carleton University
Ottawa, Ontario
Canada

ISSN 1612-3956
ISBN 978-3-642-30903-8 ISBN 978-3-642-30904-5 (eBook)
DOI 10.1007/978-3-642-30904-5
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2012950143

Mathematics Subject Classification (2010): 68M10, 68M11, 94A60, 94A62, 91D30

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Networks composed of interacting, communicating, and co-operating processes provide powerful models for understanding the behavior of complex systems. The explosive development of the Internet in the last decade has made them pervasive in all aspects of our lives. It has also made possible their emergence as new models for applying computational techniques for solving and providing understanding for old problems with new insights. As such they often provide the framework for new methodologies that lead to better decision making in many fields such as transportation, communication, health, finance, and social engineering. Solving the many emerging problems in this area has required the collaboration of researchers from fields as diverse as mathematics, computer science, biology, economics, sociology, management science, and engineering.

Papers included in this volume originate from participants in a sequence of seven workshops on mathematics of networking (FP-NETS) I organized on behalf of Mitacs in the period 2010–2012. One important area of Mitacs research was the mathematical study and analysis of complex systems as this relates to and is inspired by the study and research and development of information technologies in the scientific and engineering communities. A major idea underpinning many of Mitacs' past research projects was also related to that of *dynamic network analysis* whereby *interacting communicating entities* process, exchange, and compute in order to attain optimal design goals. Applications can be found in all scientific and engineering areas: from wireless communication to network security, from cooperative and large-scale computing to social networking, and from financial analysis and risk assessment to cyber-warfare and understanding of war.

Overall, it appears that the networking field is somewhat siloed, with research approaches to problems in one type of networks, say biological, having common elements with those in linguistic networks. For example, many questions can be rephrased as shortest path problems, routing problems, max flow problems, min cost flow problems, etc., and heuristic techniques developed in one sub-area may be applicable in others. At the same time, networking often transcends the scientific boundaries of traditional fields like biology, economics, physics, and computer science. For example, commonalities are easily found in the study of the formation of cellular systems by biologists, the origins of company networks by economists, alignment of atoms by physicists, and design of networks of computers by computer scientists.

In general, we are also interested in exploring how to organize and facilitate information acquisition, processing and acting in a large-scale system using adaptive techniques and local, sometimes restricted, channels of communication. One of the most powerful tools to emerge from the study of networking is computational methodologies that allow the testing and exploration of a wider range of more realistic models that may include dynamic parameters such as noise, motion, locality, etc. At the same time, models developed have been enriched by a vast wealth of applicable mathematical methodologies ranging from probability theory and statistics to graph theory, from combinatorial optimization to mathematical analysis and PDEs, from number theory to algebra, and from distributed computing to mechanism design that find applications in networking. The goal of this focus period on networking was to highlight application areas relevant to network analysis, identify new mathematical research areas that may provide insights, and enable cross-fertilization of ideas.

The focus period *FP-NETS: Focus Period on Recent Advances in Networking* organized and coordinated conferences, problem solving workshops, summer schools, plenary talks, and industrial academic panel discussions in selected, key areas of networking. The aim was to organize and run events pertinent to networking, promote the cross-fertilization of new ideas, as well as to support the participation of leading experts, faculty members, postdocs, and students from Canadian universities and international partner organizations. Activities included (1) tutorials that brought students and interested researchers up to speed, (2) invited talks by leaders in the field that illuminated state-of-the-art problems, (3) contributed talks by researchers, (4) panel discussions that elaborated and discussed important issues transcending current research problems, and (5) industrial and interdisciplinary exchanges. It also provided several opportunities for academics to brainstorm with research end users and identify relevant open problems. The outcome from each conference and workshop included the identification of open problems and ideas suitable for further exploration and collaboration. The focus period ended with a problem-solving interdisciplinary workshop with selected participants from all the networking workshops to interact and share expertise and ideas on important problems in networking.

Overall, the focus period FP-NETS attempted to provide a diverse and comprehensive forum to all interested researchers for understanding the most recent advances and developments in this important area. In particular, there were activities in the following networking themes: (1) Wireless Networking and Mobile Computing, (2) Network Security and Cryptography, (3) Social Networks, (4) Internet and Network Economics, (5) Biological Networks and Systems Biology, (6) Financial Networks and Risk Assessment, as well as a Problem solving workshop which concentrated on the solution of open problems resulting from the workshops. The current proceedings represent only samples of extensive discussions and scientific presentations from three of these workshops, namely, Financial Networks, Network Security and Cryptography, and Social Networks.

Needless to say, organizing all these events would have not been possible without the support and encouragement I received from several people and organizations. First of all, in the last 14 years, my involvement with Mitacs (when it managed an NCE grant for research in the mathematical sciences) has been pivotal in shaping and enhancing my evolving understanding of the nature and beauty of mathematics. Mitacs has been a truly transformative organization in its efforts to change the mathematical culture not only in Canada but globally. The scientific discussions with the other members of the Research Management Committee shaped my mathematical focus. My interactions with Arvind Gupta have been inspirational. The logistical support and efficiency of Olga Stachove were always truly amazing.

Nilima Nigam and Rebeccah Marsh were very helpful in the initial stages of the planning while Michael Lynch was always present in supporting and guiding all the organizational aspects of the events. Also many thanks to Oscar Morales Ponce for helping to integrate the electronic files into a single volume.

Ottawa, ON, Canada Evangelos Kranakis

Contents

Part I Financial Networks

Part II Network Security

Part III Social Networks

Part I
Financial Networks

Chapter 1
Mathematical Modeling of Systemic Risk

Hamed Amini and Andreea Minca

1.1 Introduction

Since the onset of the financial crisis in 2007, more than 370 of the almost 8,000 US banks insured by the Federal Deposit Insurance Corporation have failed. By comparison, between 2000 and 2004 there were around 30 failures and no failures occurred between 2005 and the beginning of 2007.

The subject of this chapter is the mathematical modeling of such episodes of default contagion, by which an economic shock causing initial losses and defaults of a few institutions is amplified due to complex financial linkages, leading to large scale defaults.

Drawing a parallel with single name credit risk models we can distinguish between two classes of default contagion models.

The first approach is represented by *reduced form models*. Here one regards firms as an ensemble of names in a portfolio and models the probability of defaults in this portfolio [15, 20, 25, 28, 42]. Defaults occur according to the arrival times of a marked point process, where the mark determines the loss in the portfolio upon default. Clearly, capturing contagion effects depends in reduced form models on the ability of the underlying point process to exhibit clusters.

The second approach is that of *structural models* of default risk. Here, one models specifically the economical linkages leading to contagion, building on the representation of the financial system as *a network* of counterparties with interlinked balance sheets. The main types of financial distress that cause financial failure are illiquidity and insolvency. Illiquidity occurs when the liquidity reserves at a

H. Amini
EPFL, Lausanne, Switzerland
e-mail: hamed.amini@epfl.com

A. Minca (✉)
ORIE Department, Cornell University, Ithaca, NY, USA
e-mail: acm299@cornell.edu

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_1,

certain time cannot cover the payment obligations at that time, whereas insolvency means that the total value of a bank's liabilities exceeds the total value of its assets. Propagation of financial distress can be modeled via domino effects: a shock (which may be a liquidity shock or a loss in the value of total assets) affecting balance sheets of a few institutions will propagate due to interconnectedness to neighboring institutions and may possibly affect an important fraction of the financial system. The acknowledgement of bank's interconnectedness and the associated contagion mechanisms has led to an increased advocacy to account for network effects when discussing regulatory requirements [12, 30, 31], be it for liquidity or capital.

The *difference between these classes of models* lies primarily in the information set available to the modeler. Structural models of contagion rely on a large set of information on balance sheets and the interrelations between those balance sheets. On the other hand, reduced form models rely on a much smaller information set, for example the market information. Therefore, the scope of these two classes of models is different. First, as argued in [34], for pricing and hedging of derivatives, the relevant set of information is the market information, since this set of information is used by market participants to determine prices. In this case, the reduced form modeling is appropriate. On the other hand, the relevant set of information available to a regulator is much more detailed, containing information on the composition of balance sheets, the degree of interconnectedness of each bank, etc. As such, for regulatory purposes – for example identifying sets of banks which pose the highest systemic risk, setting regulatory minimal ratios of liquidity and capital, rendering a network resilient to contagion – the network approach is natural.

These regulatory purposes are precisely the motivation of our work. Looking at the financial system as a network in which nodes are financial institutions and edges the linkages identified above, the central question regards the impact of the network features on the magnitude of contagion: is the underlying topology and the local properties of the nodes (i.e., balance sheets) such that the initial distress of several institutions can propagate to a large fraction of the system, or on the contrary the network is resilient, and the distress propagation will die out quickly?

For large networks (financial networks consist of several thousands of nodes), answering such questions by an exhaustive analysis of distress contagion is not possible. On the other hand, thanks to their size, such networks may be modeled by random graphs having the same asymptotic behavior and for which quantitative results on the spread of epidemics can be shown. Empirical studies like [10, 16] for interbank exposure networks, or [43] for interbank payment flows have pointed out the heterogenous nature of such network's features. First, both the in and out-degree of a node – its number of in-coming and out-going links – are characterized by a power law tail distribution. This is known as the *scale-free* property. Second, the weights on the edges – receivables or exposures – are deeply heterogenous.

The economics literature on domino effects in an economy of interlinked firms goes back to Kiyotaki and Moore [35], Hellwig [32] and Allen and Gale [3]. In [35], the authors investigate how liquidity shocks propagate across small entrepreneurial firms that lend and borrow from one another. They do not model the precise linkages in this network, but rather the behavior of a typical agent. Hellwig [32] points out the overall maturity mismatch of the financial system as a whole. Take the example of a firm i that funds a fixed-interest instrument with maturity $i+1$ by issuing an instrument with maturity i. For this firm alone, the maturity mismatch is small. Place now firm i in a chain of n firms, where firm i borrows from firm $i-1$ with maturity $i-1$ and lends firm $i+1$ with maturity $i+1$: The overall maturity mismatch scales linearly with the size of the system! Allen and Gale [3] model specifically a network of banks. Based on equilibrium models on stylized networks like the complete network and circular networks, this study points out the crucial role played by the network structure in the trade off between risk sharing and contagion. In the same sense, Stiglitz et al. [6] investigate the impact of connectivity on the spread of financial insolvency on a regular graph.

Building on economics literature [1, 11, 21] that described the mechanisms of contagion in the recent crisis, we consider a stylized network model which accounts for different types of linkages and in which one can model illiquidity cascades, insolvency cascades and price feedback effects. Indeed, insolvency cascades have been extensively investigated in the literature; Sect. 1.2.2 reviews the different contagion models and the assumptions of the respective approaches. Meanwhile, models that place the two types of cascades in relation have been lacking. Sections 1.2.3 and 1.2.4 attempt to fill this gap.

The rest of the chapter is organized as follows. Section 1.2 describes the economical mechanisms that can lead to a system level contagion like the financial crisis we have witnessed. We identify different types of linkages that transmit financial distress across institutions. In Sect. 1.2 we introduce a detailed model of balance sheets, that allows for joint modeling of insolvency and illiquidity cascades on the financial network. In Sect. 1.3 we introduce a weighted random graph model that will serve us as underlying model of a financial network. We finally give asymptotic results on the size of a default cascade. These results can be applied to several channels of distress propagation.

1.2 Financial Linkages and Contagion

The financial system acted during the recent financial crisis as an amplificator of initial losses in one asset class, mortgage backed securities, to losses that threatened the functioning of the system as a whole and spilled out into the global economy. The underlying mechanisms may be understood as modern counterparts of bank runs.

In the classical version of a bank run, depositors worried about the solvency of a bank, rush to withdraw their funds. The bank, unable to satisfy the liquidity withdrawals, fails and in turn, due to interconnectedness in the financial system, brings down other financial institutions with it, and also companies, which in absence of credit are unable to function. Bank runs have the following ingredients: an institution that holds debt with short maturity (like deposits that can be withdrawn at any moment), assets with long maturity (like long term loans) and depositors that are uninsured. Whereas classical bank runs have no longer occurred in the US since the introduction of federal insurance after the Great Depression (which eliminated the last ingredient above), the recent crisis can be deemed as a “modern bank run”.

Modern bank runs are complex and several mechanisms are at work. First, as explained in [1, 11, 21], modern financial institutions, depending more and more on short term financing via money markets, face a run from short-term lenders. These may decide to withdraw their funding, for example in anticipation of their own future needs of liquidity or because of counterparty risk. Even if a bank can still obtain funding of its less liquid assets, such funding bears the risk of increasing haircuts (the difference between the book value of the asset and the funding obtained when using it as collateral). Second, banks may face large liquidity demands, for example in the form of margin payments on outstanding derivatives. Such cases may be deemed as “margin runs” and arise from large jumps in the mark-to-market values of the derivatives. Credit default swaps are particularly prone to large jumps, even in absence of default of the reference entity. One can cite the example of leverage buyouts, i.e. the acquisition of a company using a significant amount of borrowed money, when the spreads of the acquiring company suffer large jumps. Third, when an illiquid portfolio of a defaulted bank is sold on the market, there is a price feedback effect on the portfolios of other banks holding similar assets. This can be seen as a shock that fragilizes the capitalization of the whole financial system. When the capital position of a bank no longer can withstand losses, it becomes insolvent. Its counterparties, with their already fragile capital positions, write off their exposures to the defaulted bank and in turn they may become insolvent, leading to a potential insolvency cascade.

The channels of contagion described above create **systemic risk**, defined as the risk that an initial shock is amplified by the way institutions respond and further transmit it to other institutions, such that the overall effect on the system goes largely beyond the initial shock. These contagion mechanisms rely on intricate network effects: financial institutions are interlinked by their mutual claims, be it on their balance sheets or not. Distress may propagate to neighboring institutions in a way depending solely on the local properties of the network.

The first kind of links are represented by cash flows between institutions, including margin calls and short term funding that is withdrawn. A node B is a out-neighbor of a node A if B has an immediate payment obligation to A (conversely we say that A is an in-neighbor of B). Depending on the set of out-neighbors that cannot meet their payment obligations, node A may become illiquid. But then, node A cannot meet its own payment obligations to its in-neighbors and so on. So the state of ‘illiquidity’ can spread in the network.

The second kind of links are balance sheet exposures of financial institutions to one another. By exposure we understand the expected loss on outstanding claims in case of counterparty default. A node B is a out-neighbor of a node A if A has a positive exposure to B. Depending on the set of out-neighbors in default, node A may in turn become insolvent if its capital buffer cannot withstand the losses due to direct exposures to these out-neighbors, and in this case the state of ‘insolvency’ may spread.

We identify a third kind of links, that do not represent direct claims, but relations of similarity between portfolios of banks. A node B is said to be a neighbor of node A if they hold similar assets in the portfolio. When node B becomes illiquid, its illiquid assets that were funded in the interbank market are sold at fire sale prices. When the liquidated portfolio is large, there are important price effects on the assets comprising that portfolio. Therefore, the value of the portfolio of any neighbor A will be negatively impacted. This last kind of linkages produce losses have similar economic effects as direct claims, while the size of the losses they induce can even be much larger.

1.2.1 Financial Networks

At a given point in time, a cross section of the financial system reveals a set of n financial institutions (banks) that are interlinked by their mutual claims. This cross section may thus be modeled by a weighted directed graph $g = (v, e)$, on the vertex set $v = [1, \ldots, n]$, where for any two institutions i and j, $e(i, j)$ represents the maximum loss related to direct claims incurred by i upon the default of j. We will call $e(i, j)$ the exposure of i to j, and this may include any kind of interbank loans of short or long maturities, or derivatives contracts, but also deposits held in custody by a dealer bank. If $e(i, j) > 0$, we also say that j has a liability or negative exposure to i.

In some cases, interbank contracts are placed under a netting agreement. Such an agreement specifies that, in case of default of one counterparty, the claims will net out. For example, if party j owes party i \$100M and party i owes party j \$50M, then if those claims are placed under a netting agreement, the exposure of i to j is equal to \$50M. From now on, we will understand exposures as exposures after netting if they are placed under such an agreement.

Another issue is the fact that some interbank exposures are collateralized with cash of cash equivalents, in the sense that the party with negative exposure posts collateral to its counterparty. When this collateral is deposited in a margin account, it is available to the party receiving it for its own purposes, so we will consider that the exposure $e(i, j)$ is net of collateral.

In addition to these interbank assets and liabilities, a bank holds a portfolio of non-interbank assets $\tilde{x}(i)$ and liabilities, such as deposits $D(i)$. Since we considered exposures net of collateral, we consider then that collateral received by bank i and placed in a margin account is included in $\tilde{x}(i)$. The reason for this is that, from a modeling point of view, receiving collateral against an exposure is equivalent to having reduced that exposure.

The total interbank assets of i are given by $A(i) = \sum_j e(i, j)$, whereas $L(i) = \sum_j e(j, i)$ represents the total interbank liabilities of i.

We denote by $\tilde{c}(i)$ the Tier I + Tier II capital of bank i which is the institution's buffer that absorbs losses.

Table 1.1a displays a stylized "balance sheet" of a financial institution i.

Table 1.1 Stylized balance sheet before and after shock

(a)

Interbank assets $A(i)=\sum_j e(i,j)$	Net worth $\tilde{c}(i)$
	Deposits $D(i)$
Other assets $\tilde{x}(i)$	Interbank liabilities $L(i)=\sum_j e(j,i)$
Assets	Liabilities

(b)

Interbank assets $A(i)=\sum_j e(i,j)$	$\varepsilon(i)$ – loss on capital Net worth $c(i)=(\tilde{c}(i)-\varepsilon(i))_+$
	Deposits $D(i)$
$\varepsilon(i)$ – loss on assets Other assets $x(i)=\tilde{x}-\varepsilon(i)$	Interbank liabilities $L(i)=\sum_j e(j,i)$
Assets	Liabilities

Now consider that a shock $\varepsilon(i)$ affects the non-interbank assets $\tilde{x}(i)$.

As shown in Table 1.1b, the shock $\varepsilon(i)$ is first absorbed by the capital $\tilde{c}(i)$. By the limited liability rule, the capital becomes

$$c(i) := (\tilde{c}(i)-\varepsilon(i))_+ = \max(\tilde{c}(i)-\varepsilon(i),0) \qquad (1.1)$$

after the shock. A bank is solvent while its (Tier I and Tier II) capital is positive, i.e., $c(i)>0$. An insolvent bank defaults.

From now on, we refer as time 1 to the time immediately after the shock. Our reference balance sheet is given then given by Table 1.1b.

1.2.2 Insolvency Cascades

A defaulted bank i is liquidated and the proceeds are redistributed among creditors. Let us denote by $R(i)$ the recovery rate for the debt of bank i: This represents the ratio of debt that is recovered. For simplicity, we disregard debt seniority and assume that all creditors lose a fraction $1-R(i)$ of their total exposure. If this loss is greater than their capital, then, in turn, some creditors may become insolvent and so on. It is clear that the impact of defaults on other institutions is highly dependent on recovery rates.

We now describe several cases treated in the literature and, for each case, we discuss its assumptions.

Case 1: Orderly liquidation. The model introduced by Eisenberg and Noe [22] endogenizes recovery rates. The model has been conceived for payment systems, in which assets and liabilities alike are short term.

When applying this model to a network of interbank exposures rather than a network of cash-flows, one should acknowledge the following implicit assumptions.

Assumption 1.1 (Orderly liquidation).

(i) All assets are liquid and demand for them is perfectly inelastic, i.e., during the liquidation of the portfolio there is no effect on its price.
(ii) There exists a clearing mechanism that redistributes the proceeds of defaulted banks among their creditors proportionally to their outstanding debt.

Let us denote by $(L^*(i))_{i\in v}$ the effective payable liability after all insolvent banks have been liquidated and the proceeds redistributed among creditors. Under assumptions of orderly liquidation, [22] show that L^* can be obtained as a solution of the fixed point equation $y = H(y)$, with the mapping H given by:

$$(y(i))_{i=1}^n \xrightarrow{H} (\max\{L(i), x(i) + \sum_j y(j)\frac{e(i,j)}{L(j)} - D(i) - y(i)\})_{i=1}^n. \tag{1.2}$$

If a fixed point L^* to the mapping H given by Eq. (1.2) exists, then it defines the set of insolvent banks by

$$\{i \mid L^*(i) < L(i)\}.$$

Eisenberg and Noe [22] proved that there exists a fixed point of the mapping above. They also show the uniqueness under some supplementary conditions that we briefly discuss. Let $C^-(i)$ denote the set of nodes reaching i by a directed path in the graph (v,e). Then uniqueness holds if, and only if, for every node i,

(i) No node in $C^-(i)$ has a liability to a node outside this set, and,
(ii) $C^-(i)$ has positive net external assets, i.e., $\sum_{j\in C^-(i)} x(j) > \sum_{j\in C^-(i)} D(j)$.

One example where these conditions hold is where the financial network is strongly connected: there is a directed path between any pair of two nodes. In this case, for all i, $C^-(i) = v$. The first condition above is trivially satisfied, while the second condition is equivalent to

$$\sum_j x(j) - D(i) > 0,$$

which can be interpreted as the positivity of the total equity in the system.

The recovery rate $R(i) := \frac{L^*(i)}{L(i)}$ can be understood as the recovery rate under orderly liquidation: all external assets have been liquidated at their book value $x(i)$ and the interbank assets of a defaulted bank have been redistributed at face value among the holders of the bank's liabilities according to the proportionality rule.

A crucial observation related to this model is the fact that, while initial losses are redistributed in the system, potentially causing subsequent defaults, there exists no mechanism that amplifies them.

This is the most likely cause for which some simulation studies conducted by central banks (see the survey by Upper [44] and the references therein) and based on this model dismiss the danger of contagion.

Case 2: The long term horizon. As Cifuentes et al. [13] point out, liquidation generally has feedback effects on the mark-to-market value of external assets. The fixed point of the mapping H given by Eq. (1.2) (assuming its uniqueness) depends on the external assets x. In reality, the mark-to-market values of x are affected by portfolios of external assets sold while liquidating insolvent banks. The model in [13] drops the second part of Assumption 1.1 and incorporates price feedback effects into the insolvency cascade. The resulting equilibrium point gives the long term recovery rates for the debt of defaulting firms.

Case 3: The short term horizon. In the short term, it has been argued in [16] that under assumptions of distressed liquidation, given below, recovery rates for exposures net of collateral can be approximated by zero. A non-exhaustive list of works investigating this model is [4–6, 26, 29].

Assumption 1.2 (Distressed liquidation).

(i) The insolvency cascade happens over a short time horizon.
(ii) A clearing mechanism that redistributes a bank's assets among creditors does not exist in the short term.

We consider the capital sequences given exogenously . We let $\mathbb{D}_0$ the set of initial defaults. Unlike in the previous cases, the set of initial defaults may be specified exogenously as a superset of the set of initially insolvent nodes:

$$\mathbb{D}_0 \supseteq \{i \in v \mid c(i) = 0\}, \tag{1.3}$$

allowing thus to account for defaults due to mechanisms other than insolvency.

The default of j induces a loss equal to $e(i, j)$ for its counterparty i. If this loss is greater then i's capital, then i defaults. The set of nodes which become insolvent due to their exposures to initial defaults is

$$\mathbb{D}_1(e,c) = \{i \in v \mid c(i) < \sum_{j \in \mathbb{D}_0} e(i, j)\}, \tag{1.4}$$

and generally $\mathbb{D}_r$ represents the set of nodes defaulting in round r due to exposures to nodes defaulted in rounds $0, \ldots, r-1$.

Definition 1.1 (Insolvency cascade). Starting from the set of fundamental defaults institutions $\mathbb{D}_0 \supseteq \{i \in [1, \ldots, n] \mid c(i) = 0\}$, define $\mathbb{D}_k(e,c)$ for $k = 1, \ldots, n-1$, as the set of institutions whose capital is insufficient to absorb losses due to defaults of institutions in $\mathbb{D}_{k-1}(e,c)$:

$$\mathbb{D}_k(e,c) = \{i \mid c(i) < \sum_{j \in \mathbb{D}_{k-1}(e,c)} e(i, j)\}. \tag{1.5}$$

It is easy to see that, if the size of the network is n, the cascade finishes at most in $n-1$ rounds. The final set of defaults is given by $\mathbb{D}_{n-1}(e,c)$.

To fix ideas, let us consider a simple example of a contagion starting by the default of node a on the graph illustrated in Fig. 1.1. In this simple example, contagion finishes in three rounds, node b defaults in the first round while nodes c and d default in the second round.

In the sequel, we will work under the zero recovery assumption.

1.2.3 Illiquidity Cascades

So far we discussed insolvency cascades that start from a set of exogenous initial bank defaults in a context where all balance sheets were observed at time 1 *after* an exogenous shock. The purpose of this section is to endogenize this shock as well as the emergence of initial defaults as resulting from other distress propagation mechanisms.

This brings us to the period before the shock, time 0. A snapshot of a bank's balance sheet before the shock has been shown in Table 1.1a. We now draw more detail into the balance sheet by decomposing assets according to their liquidity and maturity.

1.2.3.1 Distinguishing Short-Term and Long-Term Claims

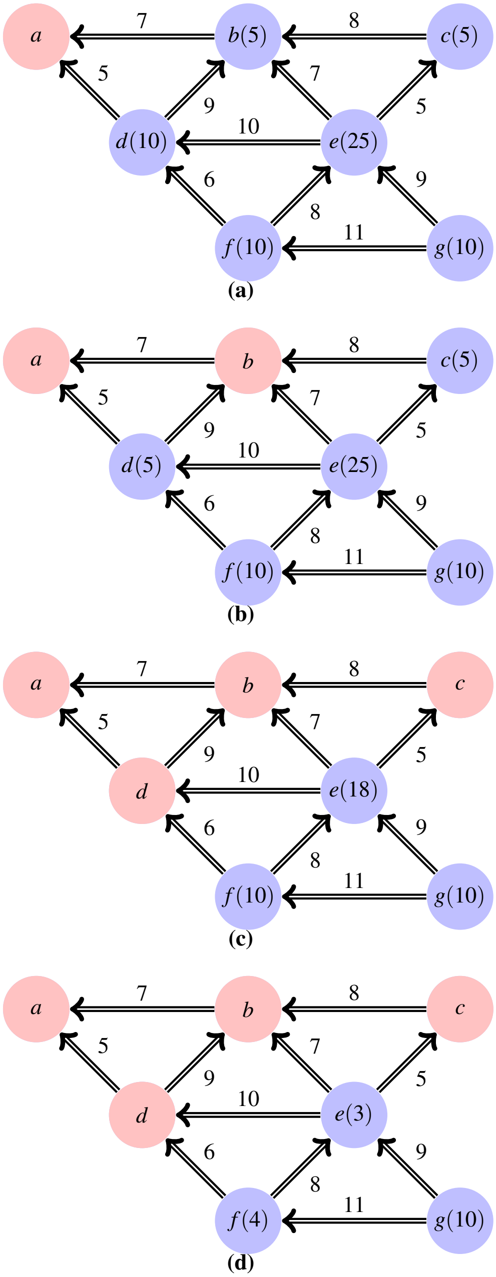
The non-interbank assets $\tilde{x}(i)$ are decomposed into highly liquid assets, that we assimilate to cash $m(i)$, and an illiquid portfolio for which the mark-to-market value at time 0 is given by $\phi(i)$. Thus,

$$\tilde{x}(i) = m(i) + \phi(i).$$

The illiquid portfolio is assumed to be funded by collateralized short term debt. However, debt cannot finance 100% of the illiquid portfolio. The difference between the market value of the illiquid asset and the value as collateral is called "haircut" and is funded by equity [11]. It follows that the exposure $e(i,j)$ of i to j includes the funding $f(i,j)$ of j's illiquid portfolio. Denoting by $H(i)$ the haircut applied to i's illiquid assets, we have that

$$\phi(i)(1 - H(i)) = \sum_j f(j,i). \tag{1.6}$$

We let $s(i,j)$ be the cash flow at time 0 from j to i. This may be a loan arriving at maturity, margin calls on derivatives, coupon payments or other contractual cash flows payable at time 0.


a
b(5)
c(5)
d(10)
e(25)
f(10)
g(10)
7
8
5
9
7
5
10
6
8
9
11
a
b
c(5)
d(5)
e(25)
f(10)
g(10)
7
8
5
9
7
5
10
6
8
9
11
a
b
c
d
e(18)
f(10)
g(10)
7
8
5
9
7
5
10
6
8
9
11
a
b
c
d
e(3)
f(4)
g(10)
7
8
5
9
7
5
10
6
8
9
11

(a)

(b)

(c)

(d)

Two cases are of particular interest:

First, the due cash-flows may be related to a shock in haircuts. If there is a (positive) jump in the haircut of bank i at time 0, equal to $\Delta H(i)$, it follows that the liquidity outflow of bank i includes $\frac{\Delta H(i)}{1-H(i)} \sum_j f(j,i)$. The situation where haircuts jump to 100% is equivalent to the situation where there is a run of short term creditors on bank j and its illiquid asset becomes unusable [11]. Such types of cascades, where runs of short term creditors prompt banks to hoard on liquidity, have been investigated in [27].

Second, the liquidity outflow of a bank i may be related to collateral demands from counterparties on OTC derivatives. The liquidity outflow in this case may be particulary large if a bank has net unidirectional positions with negative mark-to-market. A famous example is the failure of AIG in 2008, who had large net seller positions on CDS contracts.

We can write the net interbank liquidity outflow of bank i as the difference between the total liquidity outflow and the total liquidity inflow of bank i:

$$\Delta m(i) = \sum_j s(j,i) - \sum_j s(i,j). \tag{1.7}$$

Table 1.2b shows a balance sheet of a bank, where these details have been added.

We consider that banks with low liquidity have no incentive to sell the illiquid portfolio on the market in a fire sale rather than funding it [19] and that, at our observation time 0, the funding capacity of illiquid portfolios has been attained. Otherwise said, we are in the phase after balance sheets have expanded. Thus, no supplementary liquidity enters the market, but rather banks, anticipating difficulties ahead, start hoarding on liquidity and applying higher margins.

In this case, the liquidity condition of bank i is given by

$$m(i) - \Delta m(i) \geq 0. \tag{1.8}$$

Remark 1.1. Let us compare our liquidity Condition (1.8) with the liquidity condition given in the literature that investigates illiquidity due to withdrawal of short term funding alone. If a bank i suffers a liquidity shock in the form of an increase in haircuts (with the convention that haircuts increase to 1 at full funding withdrawal), we have $s(i,j) = \frac{\Delta H(i)}{1-H(i)} f(j,i)$. Condition (1.8) becomes

Fig. 1.1 Contagion on a toy financial network. Links represent exposures net of collateral. Nodes' labels represent capital buffers. (**a**) Bank a defaults exogenously. This is called a fundamental default. (**b**) First round of defaults: bank bs capital cannot withstand the loss due to the exposure to a. Bank d writes down the exposure to a from its capital. (**c**) Second round of defaults: banks c and d default due to their respective exposures to b. Bank e writes down the exposure to b from its remaining capital. (**d**) Round 4: no other defaults occur. Bank e writes down the exposure to c and d from its remaining capital, and bank f writes down its exposure to d from its capital. Contagion ends here

Table 1.2 Balance sheet with short-term and long-term claims

(a)

Net cash outflow $\Delta m(i)$	Interbank inflows $\sum_j s(i,j)$
Interbank outflows $\sum_j s(j,i)$	

(b)

Interbank assets $\sum_j e(i,j)$ including Short term collateralized lending $\sum_j f(i,j)$	Net worth $c(i)$
	Deposits $D(i)$
Liquidity $m(i)$	Interbank liabilities $\sum_j e(j,i)$ Short term collateralized borrowing $\sum_j f(j,i)$
Illiquid assets $\phi(i)$	
Assets	Liabilities

$$m_t(i) - \frac{\Delta H(i)}{1-H(i)} \sum_j f(j,i) > 0, \text{ which can be written as}$$

$$m_t(i) + (1 - H(i) - \Delta H(i)) \cdot \phi(i) > \sum_j f(j,i).$$

The second inequality is obtained by applying Eq. (1.6). This condition is equivalent to the absence of a run of short term creditors in [38].

1.2.3.2 Cash Flows and Illiquidity Cascades

If a bank j is illiquid because it does not satisfy Condition (1.8), i.e.,

$$m(i) - \Delta m(j) < 0,$$

then we will call this bank fundamentally illiquid. If there exists a set of fundamentally illiquid banks, than an illiquidity cascade might ensue. Indeed, the net liquidity outflow $\Delta m(i)$ was given in Eq. (1.7) as if all due liquidity inflows were actually received. But if a counterparty of i, say j, is illiquid, then it will default on its due payments to i. As such, the liquidity inflow of bank i must in fact be diminished by $s(i,j)$. But in this case, bank i may turn illiquid. We can now define an illiquidity cascade similarly to the insolvency cascade in the short term given by Definition 1.1. Keeping the same notations as in Definition 1.1, the final set of illiquid banks is given by

$$\mathbb{D}_{n-1}(s, m - \Delta m),$$

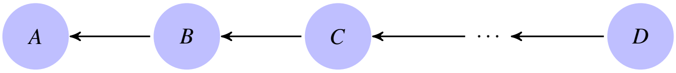


Fig. 1.2 Chains of intermediaries in OTC markets (Source: Cont and Minca [14])

while the total liquidity net outflow is given by

$$\Delta\tilde{m}(i) = \Delta m(i) + \sum_{j \in \mathbb{D}_{n-1}(s,m-\Delta m)} s(i,j). \tag{1.9}$$

Note that this liquidity net outflow appears in the balance sheet of the bank under the form of profit and loss, so it is immediately deduced (or added in case it is negative) from capital. Such network effects, are investigated in [14] in the context of OTC derivatives cash flows. Consider for example an institution A that buys protection from an institution B. Institution B will hedge its exposure to the default of the reference entity by buying protection from an institution C, and so on, until reaching an institution D which is a net seller of protection. This is pictured in Fig. 1.2. All the intermediary institutions appear well hedged and have little incentive to keep a large liquidity position. On the other hand, margin calls may be particulary large following jumps in the spread of the reference entity. If the end net seller of protection defaults, then there is potential of domino effects along the aforementioned chain of intermediaries.

1.2.4 Liquidation and Price Feedback Effects

Upon the default of bank j, the holders of its secured debt liquidate the portfolio of illiquid assets. This might trigger important price feedback effects [11, 17].

Consider that there is a finite set of assets on the market whose prices before the distressed selling are given by $(S_k)_{k\geq 1}$. Then the portfolio of bank j can be written as a vector product $\phi(j) = S \cdot \beta(j)$, with $\beta(j)$ the positions vector of bank j.

Following [17], we let λ specify the vector of market depths: The price of asset k moves by 1% when the net supply is equal to $\frac{\lambda}{100}$.

Due to liquidations, the price of asset k becomes

$$S'_k = S_k \left(1 - \frac{1}{\lambda_k} \sum_{j \in \mathbb{D}_{n-1}(s,m-\Delta m)} \beta_k(j) \right). \tag{1.10}$$

This change in price induces a change in the value of the portfolio of bank i

$$\begin{aligned} \phi'(i) = \sum_k \beta_k(i) S'_k &= \phi(i) - \sum_{j \in \mathbb{D}_{n-1}(s,m-\Delta m)} \beta_k(i) \sum_k \beta_k(j) S_k \frac{1}{\lambda_k} \\ &= \phi(i) - \sum_{j \in \mathbb{D}_{n-1}(s,m)} \rho(i,j), \end{aligned} \tag{1.11}$$

where the quantity

$$\rho(i,j) := \sum_k \beta_k(i)\beta_k(i)S_k\frac{1}{\lambda_k} \tag{1.12}$$

can be understood as the impact on the portfolio of i of liquidating the portfolio of j.

As we have seen in the previous subsection, the illiquidity cascade starting from several banks that default on their payables leads to a final set of illiquid banks given by $\mathbb{D}_{n-1}(s,m-\Delta m)$.

Now, taking into account the price feedback effects, we have that for every bank i, there will be a supplementary capital loss equal to $\sum_{j\in\mathbb{D}_{n-1}(s,m-\Delta m)}\rho(i,j)$. The shock $\varepsilon(i)$ from Sect. 1.2.1 is now endogenized:

$$c(i) = \tilde{c}(i) - \Delta m(i) - \sum_{j\in\mathbb{D}_{n-1}(s,m-\Delta m)} (\rho(i,j)+s(i,j)). \tag{1.13}$$

The set of banks $\mathbb{D}_{n-1}(s,m-\Delta m)$ can be seen as the fundamental set of defaults at time 1 when we consider the insolvency cascade.

An important observation is the effective appearance of the exposures $\rho(i,j)$, which contrary to all other exposures encountered so far, are not related to contractual claims. A bank i has a hidden exposure to j because they hold similar assets in their portfolio. These exposures reveal themselves at liquidation time and are likely to just as sizeable, or probably even more than exposures related to contractual claims [2].

We have thus seen that distress propagates in a financial network through a sequence of mechanisms. Starting from a liquidity shock, banks may become illiquid. The initial illiquidity and default on payments may transmit to counterparties, which in absence of receivables from their illiquid counterparties cannot meet their payments and default. A cascade of illiquidity may thus ensue. At the end of this cascade, we obtain a set of illiquid banks. At this point, liquidation of the portfolios belonging to defaulted banks generate a loss in the capital of all banks holding similar assets, irrespective of them having direct claims on defaulted banks or not. With the set of fundamental defaults given by the set of illiquid banks and the shock on the capital arising from fire sales of illiquid bank's portfolios, we have the premises for an insolvency cascade. This is pictured in Fig. 1.3. It is then obvious that a necessary condition for the financial network to be resilient to the initial shocks is that both the network of payments s endowed with the liquidity buffer $m-\Delta m$ and the network e endowed with the capital buffer c, considered after a shock coming from fire sales, are resilient to contagion.

1.3 Random Financial Network Models

We have shown in the previous section that various channels of distress propagation – insolvency, illiquidity and price feedback effects – may be modeled as some kind of network epidemics, in the web of interbank exposures, interbank short term

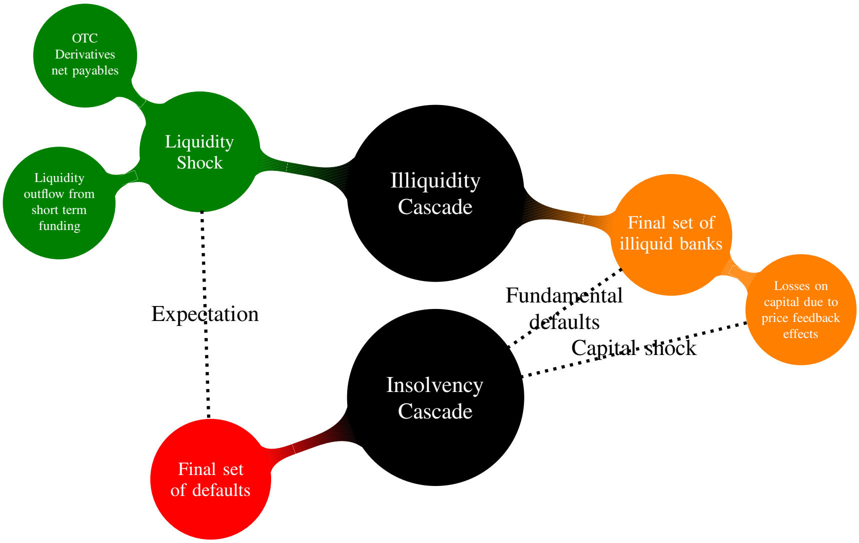

OTC Derivatives net payables
Liquidity outflow from short term funding
Liquidity Shock
Illiquidity Cascade
Final set of illiquid banks
Losses on capital due to price feedback effects
Expectation
Fundamental defaults
Capital shock
Insolvency Cascade
Final set of defaults

Fig. 1.3 Financial distress propagation

lending, the network of derivative cash flows or the network describing the degree of similarity between banks' portfolios. Generally, financial networks consist of several thousands of nodes, so an exhaustive analysis of distress propagation in such large networks is not possible. On the other hand, thanks to their size, the behavior of cascades on financial networks can be studied using a probabilistic approach: we can introduce a random network of which the financial network is a typical sample and analyze, under some mild conditions, the cascading behavior of this random counterpart. The question is then what type of random graphs are suitable models for financial networks. Empirical studies like [10, 16] for interbank exposure networks, or [43] for interbank payment flows have pointed out the heterogenous nature of these network's features. First, both the in and out-degree of a node – its number of in-coming and out-going links – are characterized by a power law tail distribution. More precisely, $\mu^+(k)$, defined as the probability that a node chosen uniformly at random has a number k of incident links, is such that for a parameter $\gamma > 0$, $\mu^+(k) \sim k^{-\gamma}$ for k larger than a given constant. A similar empirical result holds for the out-degree. This is known as the *scale-free* property and is a property shared with a plethora of other networks, arising in completely different contexts [39]. Second, the weights on the edges – receivables or exposures – also have skewed distributions.

These networks are structurally different from the classical *Erdős-Rényi random graphs* [23]. Indeed, in the classical random graphs, each pair of nodes is linked with probability p independently of everything else. If $p = c/n$, the sequence of degrees has an asymptotic Poisson distribution of average c, which is a homogenous distribution. In order to account for the scale free properties of real networks, Newman et al. [40, 41] proposed to use as an underlying graph model the so called random graph with fixed degree distribution. Some of the properties of this random graph had been previously investigated by Molloy and Reed [36, 37]. Their version of the model is in fact different from [40, 41] in the sense that they look at graphs with prescribed degree sequences rather than prescribed degree distributions. The sequences of degrees can be any integers that satisfy certain conditions.

So ideally, we are interested in (uniformly chosen) random graphs having a prescribed degree sequence. But it is difficult to directly examine these random graphs, so instead, we introduce a model that produces a multigraph with the prescribed degrees, and which, when conditioned on simplicity of the multigraph, becomes uniform over all simple graphs with the prescribed degree sequence. This random multigraph is called the *configuration model* (or '*CM*'). The configuration model was originally developed by Bender and Canfield [7] and Bollobás [8] as a mean for generating a random graph with a prescribed sequence of vertex degrees (its earliest applications were in the study of random regular graphs).

For $n \in \mathbb{N}$, let $(d_i)_1^n$ be a sequence of non-negative integers such that $\sum_{i=1}^n d_i$ is even. By means of the configuration model, we define a random multigraph with given degree sequence $(d_i)_1^n$, denoted by $G^*(n,(d_i)_1^n)$ as follows. To each node i, we associate d_i labeled half-edges. All half-edges need to be paired to construct the multigraph, this is done by randomly matching them. When a half-edge of i is paired with a half-edge of j, we interpret this as an edge between i and j. The graph $G^*(n,(d_i)_1^n)$ obtained following this procedure may not be simple, i.e., may contain self-loops due to the pairing of two half-edges of i, and multi-edges due to the existence of more than one pairing between two given nodes. Note that $G^*(n,(d_i)_1^n)$ does not have exactly the uniform distribution over all multigraphs with the given degree sequence; there is a weight with a factor $1/j!$ for every edge of multiplicity j, and a factor $1/2$ for every loop. However conditional on the multigraph $G^*(n,(d_i)_1^n)$ being a simple graph, we obtain a uniformly distributed random graph with the given degree sequence, which we denote by $G(n,(d_i)_1^n)$.

One specific example is when the degrees are all equal, in which case we speak of a random regular graph.

Although the classical model of Erdős-Rényi cannot capture the properties of real networks, the most important finding in the seminal papers [23, 24] – namely the fact that many graph properties undergo a phase transition with a rather small change in the parameters [33] – has been shown to have corresponding results on the configuration model.

If we denote by $\mathrm{ER}(n,c/n)$, the Erdős-Rényi random graph of size n where edges are present independently with probability c/n, the following result holds [9]: If $c<1$ then with high probability (i.e., with probability tending to 1 as $n\to\infty$), every component of $\mathrm{ER}(n,c/n)$ has order $O(\log n)$. If $c>1$ then with high probability (w.h.p.) $\mathrm{ER}(n,c/n)$ has a component with $(\alpha(c)+o(1))n$ vertices, where $\alpha(c)>0$, and all other components have $O(\log n)$ vertices.

So, with a slight increase in the average connectivity (which is the only parameter governing the law of Erdős-Rényi random graphs), we pass to a regime where a giant component (i.e., a connected component representing a positive fraction of the vertices) exists. In the case of $G(n,(d_i)_1^n)$, the corresponding question of the existence of a giant component was answered by Molloy and Reed [37] who show that under some regularity conditions on the degree sequence (such as the convergence of the empirical degree distribution to some finite-mean probability distribution μ) $G(n,(d_i)_1^n)$ contains a giant component w.h.p. if and only if $\sum_k \mu(k)k(k-2) > 0$.

Now consider the case of a simple epidemics on the random graph $\mathrm{ER}(n,c/n)$. Assume that each infected node with probability p infects its neighbours. The question of whether it is possible for a single node to infect a positive fraction of all nodes, is in fact equivalent to the existence of a giant component in the random graph $\mathrm{ER}(n,pc/n)$. Indeed, it is easy to see that the random graph $\mathrm{ER}(n,pc/n)$ has the same distribution as the random graph obtained from $\mathrm{ER}(n,c/n)$ by removing any edge with probability $1-p$ independently of everything else. (The model in which edges are removed independently with a given probability is known as bond percolation.) Then the spread of the above epidemics on the random graph $\mathrm{ER}(n,c/n)$ presents a phase transition stemming from the emergence of the giant component in the random graph $\mathrm{ER}(n,pc/n)$: with a slight increase in the 'contagiousness' p of the infection we pass to a regime where a single node can infect a positive fraction of the network.

It is clear from this simple example that the dynamic properties of networks, e.g., the spread of epidemics, are closely related to their geometrical properties, e.g., does a giant component exist.

Bearing this in mind, we turn our attention to the configuration model. The version of Molloy and Reed [36] has been extended by Cooper and Frieze [18] to allow for prescribed sequences of directed degrees. This model has been further extended in [4] to allow for a prescribed sequences of weights, while relaxing the conditions on the degree sequence given in [18]. The Weighted Configuration Model will be the basis of our model of financial network, on which we will study distress propagation.

Definition 1.2 (Weighted Configuration Model). Given a set of nodes $[1,\dots,n]$ and a degree sequence (d_n^+, d_n^-), we associate to each node i two sets, $H_n^+(i)$ representing its out-going half-edges and $H_n^-(i)$ representing its in-coming half-edges, with $|H_n^+(i)| = d_n^+(i)$ and $|H_n^-(i)| = d_n^-(i)$. Let $H_n^+ = \bigcup_i H_n^+(i)$ and $H_n^- = \bigcup_i H_n^-(i)$. A prescribed set of weights $E_n(i)$ with $|E_n(i)| = d_n^+(i)$ is assigned in an arbitrary order to i's out-going half edges. A *configuration* is a matching of H_n^+ with H_n^-. To each configuration we assign a graph. When an out-going half-edge of node i is matched with an in-coming half-edge of node j, a directed edge from i to j appears in the graph. The *configuration model* is the probability space in which all configurations, as defined above, have equal probability. We denote the resulting random directed multigraph by $G_n^*(d_n^-, d_n^+)$, shown in Fig. 1.4.

The previous definition does not account directly for dependence between the weight of an edge and the node in which the edge ends, which may be unrealistic in applications.

Fortunately, we can easily extend the previous model in this sense. The intuition behind our construction can be given by rephrasing the Pareto principle: 20% of the links carry 80% of the weights. Therefore, we will distinguish between two types of links, type A and type B. In applications, we will think of links of type A as those

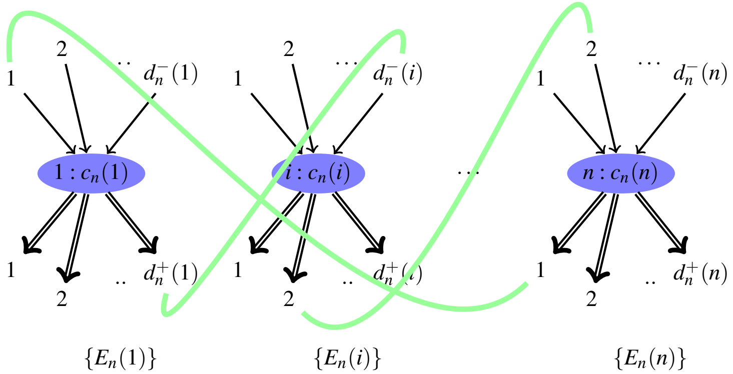


Fig. 1.4 Weighted configuration model

representing a small fraction of links but carrying the largest weights. We can prescribe, for each node the proportion of in-coming and out-going edges of each type.

We can now define Two-Tier Weighted Configuration Model.

Definition 1.3 (Two-Tier Weighted Configuration Model). Let (d_n^+, d_n^-) a prescribed sequence of in and out-degrees. For every node i, its $d_n^+(i)$ in-coming links are partitioned into $d_n^{+,A}(i)$ links of type A and $d_n^{+,B}(i)$ links of type B:

$$d_n^+(i) = d_n^{+,A}(i) + d_n^{+,B}(i). \tag{1.14}$$

Similarly, the out-going half edges are partitioned, for any node i, into $d_n^{-,A}(i)$ links of type A and $d_n^{-,B}(i)$ links of type B, with $d_n^-(i) = d_n^{-,A}(i) + d_n^{-,B}(i)$. We assume the following conditions: $\sum_{i=1}^n d_n^{+,A}(i) = \sum_{i=1}^n d_n^{-,A}(i) =: m^A$ and $\sum_{i=1}^n d_n^{+,B}(i) = \sum_{i=1}^n d_n^{-,B}(i) =: m^B$, where we denoted by m^A and m^B the total number of links of type A and type B respectively. We let $F^A : \mathbb{R}_+^{m^A} \to [0,1]$ and $F^B : \mathbb{R}_+^{m^B} \to [0,1]$ the joint probability distributions functions for links of type A and B respectively. The probability distribution functions F^A and F^B are assumed to be invariant under permutation of their arguments. The random graph is generated then as follows:

- Draw m^A random variables from the joint distribution F^A. Assign these exchangeable variables in an arbitrary order to the out-going half edges of type A;
- Generate the weighted subgraph of links of type A by from the Weighted configuration model with prescribed degree sequence $(d_n^{+,A}(i), d_n^{-,A}(i))_{i=1}^n$;
- Proceed similarly for the links of type B.

1.4 Asymptotic Analysis of Default Cascades

We overview here the main results which were given in the context of insolvency cascades in financial networks in [4], but can be used for other types of cascades on a network. We consider a directed network in which nodes can be in one of two states, say 0 and 1. Starting from a set of nodes initially in the state 1, other nodes switch to state 1 according to the weighted influence of their neighbors and a personal threshold.

More precisely, we define the network w_n on the vertex set $v = \{1, \ldots, n\}$, whereby $w_n(i, j)$ weighs the influence of node j on the state of node i. Each node i has a threshold $q_n(i)$ which determines its capacity to withstand the influence of other nodes. Denoting by $X(j) \in \{0,1\}$ the state of a node j, node i switches to state 1 the first time the following condition is met

$$\sum_j X(j) w_n(i, j) > q_n(i). \tag{1.15}$$

The out-neighbors of a node i are given by the set of nodes having an influence on i and its in-neighbors are given by the set of nodes on which i has an influence. Their respective numbers represent node i's out-degree $d_n^+(i) := \#\{j \mid w_n(i, j) > 0\}$, and respectively in-degree $d_n^-(i) := \#\{j \mid w_n(j,i) > 0\}$. The empirical distribution of the degrees is given by

$$\mu_n(j,k) := \frac{1}{n}\#\{i : d_n^+(i) = j, d_n^-(i) = k\}.$$

We assume that the degree sequences d_n^+ and d_n^- satisfy the following regularity conditions.

Assumption 1.3. *For each $n \in \mathbb{N}$, $d_n^+ = \{(d_n^+(i))_{i=1}^n\}$ and $d_n^- = \{(d_n^-(i))_{i=1}^n\}$ are sequences of nonnegative integers with $\sum_{i=1}^n d_n^+(i) = \sum_{i=1}^n d_n^-(i)$, and such that, for some probability distribution $\mu(j,k)$, the following hold.*

1. *The degree density condition: the proportion $\mu_n(j,k)$ of nodes with degree (j,k) tends to $\mu(j,k)$, i.e.,*
$$\mu_n(j,k) \overset{n\to\infty}{\to} \mu(j,k)$$
2. *Finite expectation property: $\sum_{j,k} j\mu(j,k) = \sum_{j,k} k\mu(j,k) =: \lambda \in (0,\infty)$;*
3. *Second moment property: $\sum_{i=1}^n (d_n^+(i))^2 + (d_n^-(i))^2 = O(n)$.*

We turn now our attention to the role of continuous weights and thresholds in the spread of the epidemics. We denote by $\Sigma^w(i)$ the set of permutations of i out-neighbors and let $\tau \in \Sigma^w(i)$ specify the order in which i out-neighbors switch to state 1. Then Condition (1.15) is equivalent to saying that node i switches to state 1 precisely after a certain number of its out-neighbors have switched to state 1, where this number is given by

$$\Theta(i, w, q, \tau) := \min\{k \geq 0, \sum_{j=1}^{k} w(i, \tau(j)) > q(i)\}. \tag{1.16}$$

The map Θ gives the *discretized thresholds* that govern the spread of epidemics.

We let

$$p_n(j,k,\theta) := \frac{\#\{(i,\tau) \mid 1 \leq i \leq n,\ \tau \in \Sigma^{e_n},\ d_n^+(i) = j,\ d_n^-(i) = k,\ \Theta(i, w_n, q_n, \tau) = \theta\}}{n\mu_n(j,k) j!}, \tag{1.17}$$

for which we make the following assumption:

Assumption 1.4. *There exists a function* $p:\mathbb{N}^3\to[0,1]$ *such that for all* $j,k,\theta\in\mathbb{N}$ *(*$\theta\le j$*)*

$$p_n(j,k,\theta)\overset{n\to\infty}{\to}p(j,k,\theta).$$

Definition 1.4 (Contagious links). We say that a link is 'contagious' if it represents an influence on a node larger than its threshold.

It is easy to see that $p_n(j,k,1)$ represents the proportion of 'contagious' links leaving nodes with degree (j,k). The limit $p(j,k,1)$ also represents the fraction of nodes with degree (j,k) that switch to 1 as soon as one out-neighbor has switched to 1.

We now define the random network with prescribed degree and weights.

Definition 1.5 (Random network ensemble). Let $\mathcal{G}_n(w_n)$ be the set of all weighted directed graphs with degree sequence (d_n^+,d_n^-) such that, for any node i, the set of weights is given by the non-zero elements of line i in the matrix w_n. On a probability space $(\Omega,\mathcal{A},\mathbb{P})$, we define W_n as a random network uniformly distributed on $\mathcal{G}_n(w_n)$.

Then for all $i=1,\dots,n$,

$$\{W_n(i,j),\quad W_n(i,j)>0\}=\{w_n(i,j),\quad w_n(i,j)>0\}\quad \mathbb{P}-a.s.$$

$$\#\{j\in v,\ W_n(j,i)>0\}=d_n^+(j),\quad \text{and}\qquad \#\{j\in v,\ W_n(i,j)>0\}=d_n^-(i).$$

We denote by $\alpha_n(W_n,q_n)$ the set of defaults at the end of the cascade generated by the set of nodes $\{i\mid q_n(i)=0\}$.

The following theorems give the asymptotic behavior of this quantity.

Theorem 1.5. *Define the function*

$$I(\pi):=\sum_{j,k}\frac{k\mu(j,k)}{\lambda}\sum_{\theta=0}^{j}p(j,k,\theta)\mathbb{P}(\mathrm{Bin}(j,\pi)\ge\theta), \tag{1.18}$$

where $\mathrm{Bin}(j,\pi)$ *denotes a binomial variable with parameters j and* π.

Consider a sequence of weights and thresholds $\{(w_n)_{n\ge1},(q_n)_{n\ge1}\}$ *satisfying Assumptions 1.3 and 1.4 and the corresponding sequence of random matrices* $(W_n)_{n\ge1}$ *defined on* $(\Omega,\mathcal{A},\mathbb{P})$ *as in Definition 1.5. Let* π^* *be the smallest fixed point of I in* $[0,1]$*, i.e.,*

$$\pi^*=\inf\{\pi\in[0,1]\mid I(\pi)=\pi\}.$$

1. *If* $\pi^*=1$*, i.e., if* $I(\pi)>\pi$ *for all* $\pi\in[0,1)$*, then asymptotically all nodes switch to state* 1*:*

$$\alpha_n(W_n,q_n)\xrightarrow{p}1.$$

2. *If* $\pi^*<1$ *and furthermore* π^* *is a stable fixed point of I* $(I'(\pi^*)<1)$*, then the asymptotic fraction of nodes in state* 1 *at the end of the cascade satisfies:*

$$\alpha_n(W_n, q_n) \xrightarrow{p} \sum_{j,k} \mu(j,k) \sum_{\theta=0}^{j} p(j,k,\theta)\mathbb{P}(\mathrm{Bin}(j,\pi^*) \geq \theta).$$

Definition 1.6 (Resilience measure). We define as the resilience measure the following function of the network's features, which takes values in $(-\infty, 1]$:

$$1 - \sum_{j,k} \frac{jk}{\lambda} \mu(j,k) p(j,k,1).$$

Theorem 1.6. *Under Assumptions 1.3 and 1.4;*

- *If the resilience measure is positive, i.e.,*

$$1 - \sum_{j,k} \frac{jk}{\lambda} \mu(j,k) p(j,k,1) > 0, \tag{1.19}$$

then for every $\varepsilon > 0$, there exists N_ε and ρ_ε such that if the initial fraction of nodes in state 1 is smaller than ρ_ε, then $\mathbb{P}(\alpha_n(W_n, q_n) \leq \varepsilon) > 1 - \varepsilon$ for all $n \geq N_\varepsilon$.

- *If the resilience measure is negative, i.e.,*

$$1 - \sum_{j,k} \frac{jk}{\lambda} \mu(j,k) p(j,k,1) < 0, \tag{1.20}$$

then there exists a connected set C_n of nodes representing a positive fraction of the network, i.e., $|C_n|/n \xrightarrow{p} c > 0$ such that, with high probability, any node in the set switching to state 1 activates the whole set: for any sequence $(q_n)_{n\geq 1}$ such that $\{i, q_n(i) = 0\} \cap C_n \neq \emptyset$,

$$\liminf_n \alpha_n(W_n, q_n) \geq c > 0.$$

These results can be applied to the following current issues in finance:

- The problem of short term funding in money markets; the weights are represented by short term funding, i.e., $f(i,j)$ and the thresholds are represented by the liquidity reserves $m(i)$ (see Table 1.2b). Banks that do not satisfy condition (1.8) withdraw any of the funds they lent on the market [27], and thus other banks may become illiquid.
- The problem of margin calls in over the counter markets; the weights are represented by derivatives payables and the thresholds are represented by liquidity reserves. This problem is investigated in [14]
- The problem of insolvency cascades, in which weights represent exposures and the threshold the capital. This problem is investigated in [4] and the above results are given in this context.

Conditions for stability of random networks with respect to contagion had been anticipated in the literature [26, 45], using mean field approximations or heuristic methods, in terms of the *expected* size of a cascade starting from a randomly chosen node, where the expectation was taken over the law of the random graph with given degree distribution. The results in [4] represent stronger statements: the final fraction of defaults is shown to converge in probability as the size of the network tends to infinity, to a limit which depends explicitly on the fundamentals of the model: degree sequence, interbank exposures, capital ratios, liquidity reserves, payables, short term debt etc. All this is crucial if one wants to identify and monitor the nodes posing the largest systemic risk in a given network.

Acknowledgements We would like to thank Rama Cont and Damir Filipović for helpful comments and discussions that improved the presentation of this chapter. Andreea Minca would like to thank the Natixis Foundation for Quantitative Research who supported this work. Hamed Amini gratefully acknowledges financial support from the Austrian Science Fund (FWF) though project P21709.

References

1. T. Adrian and H. S. Shin. Financial intermediary leverage and value-at-risk. *Federal Reserve Bank of New York Staff Reports*, (338), 2008.
2. T. Adrian and H. S. Shin. The changing nature of financial intermediation and the financial crisis of 2007–2009. *Annual Review of Economics*, 2(1):603–618, 2010.
3. F. Allen and D. Gale. Financial contagion. *Journal of Political Economy*, 108(1):1–33, 2000.
4. H. Amini, R. Cont, and A. Minca. Resilience to Contagion in Financial Networks. *SSRN eLibrary*, 2010.
5. H. Amini, R. Cont, and A. Minca. Stress testing the resilience of financial networks. *To appear in the International Journal of Theoretical and Applied Finance*, 2011.
6. S. Battiston, D. D. Gatti, M. Gallegati, B. Greenwald, and J. E. Stiglitz. Liaisons dangereuses: Increasing connectivity, risk sharing, and systemic risk. *Preprint available at http://www.nber.org/papers/w15611*, 2009.
7. E. A. Bender and E. R. Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24:296–307, 1978.
8. B. Bollobás. The asymptotic number of unlabelled regular graphs. *J. London Math. Soc. (2)*, 26(2):201–206, 1982.
9. B. Bollobás and O. M. Riordan. Mathematical results on scale-free random graphs. In *Handbook of graphs and networks*, pages 1–34. Wiley-VCH, Weinheim, 2003.
10. M. Boss, H. Elsinger, M. Summer, and S. Thurner. The network topology of the interbank market. *Quantitative Finance*, (4):677–684, 2004.
11. M. K. Brunnermeier. Deciphering the liquidity and credit crunch 2007–2008. *Journal of Economic Perspectives*, 23(1):77–100, 2009.
12. J. A. Chan-Lau, M. Espinosa, K. Giesecke, and J. A. Sole. Assessing the systemic implications of financial linkages. In *Global Financial Stability Report*. International Monetary Fund, 2009.
13. R. Cifuentes, G. Ferrucci, and H. Shin. Liquidity risk and contagion. *Journal of the European Economic Association*, 3:556–566, 2005.
14. R. Cont and A. Minca. Credit default swaps and systemic risk. *Working Paper*, 2011.
15. R. Cont and A. Minca. Recovering portfolio default intensities implied by CDO quotes. *Mathematical Finance*, 2011.

16. R. Cont, A. Moussa, and E. B. Santos. Network Structure and Systemic Risk in Banking Systems. *Preprint available at http://papers.ssrn.com/sol3/id=1733528*, 2010.
17. R. Cont and L. Wagalath. Running for the exit: distressed selling and endogenous correlation in financial markets. *Mathematical Finance*, 2011.
18. C. Cooper and A. M. Frieze. The size of the largest strongly connected component of a random digraph with a given degree sequence. *Combinatorics, Probability & Computing*, 13(3):319–337, 2004.
19. D. W. Diamond and R. G. Rajan. Fear of Fire Sales and the Credit Freeze. *SSRN eLibrary*, 2009.
20. X. Ding, K. Giesecke, and P. I. Tomecek. Time-changed birth processes and multiname credit derivatives. *Oper. Res.*, 57(4):990–1005, 2009.
21. D. Duffie. The failure mechanics of dealer banks. *Journal of Economic Perspectives*, 24(1):51–72, 2010.
22. L. Eisenberg and T. H. Noe. Systemic Risk in Financial Systems. *Management Science*, 47(2):236–249, 2001.
23. P. Erdős and A. Rényi. On random graphs. I. *Publ. Math. Debrecen*, 6:290–297, 1959.
24. P. Erdős and A. Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5:17–61, 1960.
25. E. Errais, K. Giesecke, and L. R. Goldberg. Affine point processes and portfolio credit risk. *SIAM J. Financial Math.*, 1:642–665, 2010.
26. P. Gai and S. Kapadia. Contagion in Financial Networks. *Proceedings of the Royal Society A*, 466(2120):2401–2423, 2010.
27. P. Gai and S. Kapadia. Liquidity hoarding, network externalities, and interbank market collapse. *Mimeo, Bank of England*, 2010.
28. K. Giesecke. Portfolio credit risk: top-down vs bottom-u. In *Cont, R. (ed.) : Frontiers in quantitative finance: credit risk and volatility modeling*. Wiley, 2008.
29. J. Gleeson, T. R. Hurd, S. Melnik, and A. Hackett. Systemic risk in banking networks without monte carlo simulation. In E. Kranakis, editor, *Advances in Network Analysis and its Applications*, Mathematics in Industry. Springer Verlag, Berlin Heidelberg New York, 2011.
30. A. G. Haldane. Rethinking the financial networks. Speech delivered at Financial Student Association in Amsterdam, http://www.bankofengland.co.uk/publications/speeches/2009/speech386.pdf, 2009.
31. A. G. Haldane and R. M. May. Systemic risk in banking ecosystems. *Nature*, 469:351–355, 2011.
32. M. Hellwig. Systemic aspects of risk management in banking and finance. *Swiss Journal of Economics and Statistics*, 131:723–737, 1995.
33. S. Janson, T. Łuczak, and A. Rucinski. *Random graphs*. Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley-Interscience, New York, 2000.
34. R. Jarrow and P. Protter. Structural versus reduced form models: a new information based perspective. *Journal of Investment Management*, 2(2):34–43, 2004.
35. N. Kiyotaki and J. Moore. Credit chains. *Mimeo, Walras-Bowley Lecture to the North American Meeting of the Econometric Society, Iowa City, IA*, 1996.
36. M. Molloy and B. Reed. A critical point for random graphs with a given degree sequence. *Random Structures Algorithms*, 6(2–3):161–179, 1995.
37. M. Molloy and B. Reed. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing*, 7:295–305, 1998.
38. S. Morris and H. S. Shin. Illiquidity component of credit risk. *Working paper*, 2009.
39. M. Newman, Albert-László Barabási, and D. J. Watts. *The Structure and Dynamics of Networks*. Princeton University Press, 2006.
40. M. E. J. Newman. Spread of epidemic disease on networks. *Phys. Rev. E*, 66(1):016128, Jul 2002.
41. M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64:026118, 2001.

42. P. Schönbucher. Portfolio losses and the term structure of loss transition rates: a new methodology for the pricing of portfolio credit derivatives. *Working paper*, 2005.
43. K. Soramaki, M. L. Bech, J. Arnold, R. J. Glass, and W. E. Beyeler. The topology of interbank payment flows. *Physica A: Statistical Mechanics and its Applications*, 379(1):317–333, 2007.
44. C. Upper. Simulation methods to assess the danger of contagion in interbank markets. *Journal of Financial Stability*, 7:111–125, 2011.
45. D. J. Watts. A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences of the USA*, 99(9):5766–5771, 2002.

Chapter 2
Systemic Risk in Banking Networks Without Monte Carlo Simulation

James P. Gleeson, T.R. Hurd, Sergey Melnik, and Adam Hackett

Abstract An analytical approach to calculating the expected size of contagion events in models of banking networks is presented. The method is applicable to networks with arbitrary degree distributions, permits cascades to be initiated by the default of one or more banks, and includes liquidity risk effects. Theoretical results are validated by comparison with Monte Carlo simulations, and may be used to assess the stability of a given banking network topology.

2.1 Introduction

The study of contagion in financial systems is currently very topical. “Contagion” refers to the spread of defaults through a system of financial institutions, with each successive default causing increasing pressure on the remaining components of the system. The term “systemic risk” refers to the contagion-induced threat to the financial system as a whole, due to the default of one (or more) of its component institutions, and it has become a familiar term since the failure of Lehman Brothers and the rescue of AIG in the autumn of 2008.

Interbank (IB) networks in the real world are financial systems that range in size from dozens to thousands of institutions [6,26,28]. An IB network may be modelled as a (directed) graph; the *nodes* or *vertices* of the network are individual banks, while the *links* or *edges* of the network are the loans from one bank to another. Such systems are vulnerable to contagion effects, and the importance of studying these complex networks has been highlighted by Andrew Haldane, Executive director of

J.P. Gleeson (✉) • S. Melnik • A. Hackett
MACSI, Department of Mathematics and Statistics, University of Limerick, Limerick, Ireland
e-mail: james.gleeson@ul.ie; sergey.melnik@ul.ie; adam.hackett@ul.ie

T.R. Hurd
Department of Mathematics and Statistics, McMaster University, Hamilton, ON, Canada
e-mail: hurdt@mcmaster.ca

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_2,

Financial Stability at the Bank of England in his speech [15], in which he posed the following challenge: ‘Can network structure be altered to improve network robustness? Answering that question is a mighty task for the current generation of policymakers’.

The study of complex networks has advanced rapidly in the last decade or so, with large-scale empirical datasets becoming readily available for a variety of social, technological, and biological networks (see [19, 23, 24] for reviews). By virtue of their size and complexity, such networks are amenable to statistical descriptions of their characteristics. The *degree distribution* p_k of a network, for example, gives the probability that a randomly-chosen node of the network has degree k, i.e., that it is connected by k edges to neighbours in the network. While classical random graph models of networks [10] have Poisson degree distributions, many empirical networks have been found to possess “fat-tailed” or “scale-free” degree distributions, where the probability of finding nodes of degree k decays as a power law in k ($p_k \propto k^{-\beta}$) for large k, in contrast to the exponential decay with k of the Poisson distribution [23].

This structural (topological) aspect of real-world networks has important implications for dynamical systems which run on the nodes of the network graph, see Barrat et al. [3] for a review. For example, the rate of disease spread on networks depends crucially on whether or not they have fat-tailed degree distributions. As a consequence, there is considerable interest in the effect of network structure on a range of dynamics. Cascade-type dynamics occur whenever the switching of a node into a certain state increases the probability of its neighbours making the same switch. Examples include cascading failures in power-grid infrastructure [22], congestion failure in communications networks [21], the spread of fads on social networks [27], and bootstrap percolation problems [5], among others [17]. Building on earlier work on the random field Ising model of statistical physics [7], the expected size of cascades has recently been determined analytically for a range of cascade dynamics and (undirected) network topologies [13, 14]. Our goal in this paper is to extend and develop these methods for application to default contagion on (directed) interbank networks.

Although the importance of network topologies has been recognized for many years in the finance and economics literature (e.g., [1]), it is only following the publication of empirical studies for large-scale interbank networks [6,9,26,28] that theoretical models have moved beyond small networks and simple topologies. In this paper we focus on “deliberately simplified” models for contagion on interbank networks exemplified by those of Gai and Kapadia [11] (“GK” for short) and of Nier et al. [25] (“NYYA” for short), which have attracted significant recent attention [16, 18]. We develop an analytical approach to calculating the expected size of contagion events in networks of a prescribed topology.

The calculation is “semi-”analytical because it requires the iteration of a nonlinear map to its fixed point, but it nevertheless offers significantly faster calculation than Monte Carlo simulation. This reduces the computational burden of interbank network simulations, hence making network theory more useful for practical applications. Moreover, the analytical approach gives insights into the mechanisms of contagion transmission in a given network topology, and enables formulas relating critical parameter values to be derived.

Our work extends the seminal paper of May and Arinaminpathy [18] by moving beyond their assumption that every bank in the network is identical (i.e., that all banks have the same numbers of debtors and creditors). As shown by May and Arinaminpathy, this "mean-field" assumption gives reasonably accurate results for Erdös-Rényi random networks, which have independent Poisson distributions for in- and out-degrees. This means that each bank in such a network is similar to the "average" bank. However, real-world banking networks often have fat-tailed degree distributions [6], meaning that there is a significant probability of finding a bank with in-degree (or out-degree) very different to the mean degree. To analyze contagion on such networks we need to move beyond the mean-field assumption. Moreover, unlike May and Arinaminpathy, our formalism allows us to consider how the extent of the contagion is affected by the size of the bank which initiates the cascade, and so to inform the question of which banks are 'too big to fail'.

The remainder of this paper is structured as follows. In Sect. 2.2 we review the models of GK and NYYA. Sections 2.3 and 2.4 develop a general theoretical framework for analyzing such models, while in Sect. 2.5 we compare the results of our analytical approach with full Monte-Carlo simulations, and discuss conclusions in Sect. 2.6. Three appendices give details of several results that are not crucial to the main flow of the paper.

2.2 Models of Contagion in Banking Networks

We consider simplified models of banking networks, as introduced by GK and NYYA. As noted in May and Arinaminpathy [18], such "deliberately oversimplified" mathematical models are caricatures of real banking networks, but may nevertheless lead to useful insights. These model networks can be considered as generated in two steps. First, a "skeleton" structure of N nodes (representing banks) and directed edges (to represent the interbank positions) is created. This structure should be a realization from the ensemble of all possible directed networks which are consistent with the joint probability p_{jk} (the probability that a randomly chosen node has j in-edges and k out-edges). We choose the following convention for the direction of edges: an arrow on an edge representing an interbank position ("loan" for short) points from the debtor bank to the creditor bank, see Fig. 2.1. This convention ensures that shocks due to defaults on loans travel in the direction of the arrows on the edges. Thus p_{jk} is the probability that a randomly-chosen bank in the system has j debtors (or, more strictly, that it has j asset loans, since multiple links are possible) and k creditors (strictly speaking, k liability loans).

In the second step, each node (bank) of the skeleton structure is endowed with a balance sheet and the edges between banks are weighted with loan magnitudes. This process is performed in such as way as to ensure the banking system so represented is fully in equilibrium (i.e., assets exceed liabilities for each bank) in

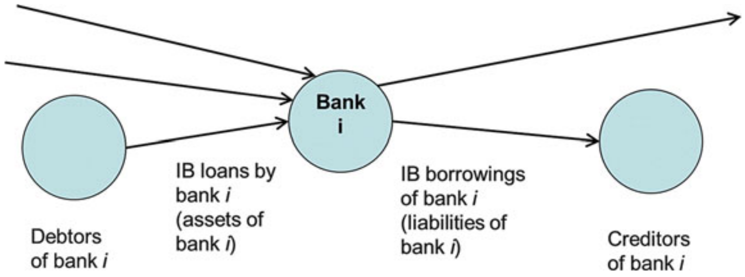


Fig. 2.1 Skeleton structure of the network locality of bank i. Bank i is in the $(j,k) = (3,2)$ class, since it has three debtors and two creditors in the interbank (*IB*) network

the absence of exogenous shocks. Once the banking networks are generated, the cascade dynamics can be implemented to examine the effects of various types of shocks. In Monte Carlo implementations, each step of the process (skeleton structure/balance sheets/dynamics) is repeated many times to simulate the ensemble of possible systems. The most common output from such simulations is the expected fraction of defaulted banks in steady-state (i.e., when all cascades have run their course) for the prescribed p_{jk} network topology.

We stress that this two-step procedure is only one of many possible alternatives for generating an ensemble of random networks. However, it is easily explained and reproducible by other researchers, and proves amenable to analysis. As a "deliberately oversimplified" model of the true complexities of banking networks, it is not suitable for calibration to real network data in its current form, but may nevertheless provide a starting point for improving our understanding of the interplay between network topology and default contagion cascades.

2.2.1 Generating Model Networks

We first discuss the creation of the skeleton structure for N banks (or nodes) consistent with a prescribed p_{jk} distribution. It is usually assumed that N is large (indeed theoretical results are proven only in the $N \to \infty$ limit), but in practice values of N as low as 25 have been successfully examined (see Results section). In each realization, N pairs of (j,k) variables are drawn from the p_{jk} distribution. For each pair (j,k), a node is created with j in-edge stubs and k out-edge stubs. Then a randomly-chosen out-stub is connected to a randomly-chosen in-stub to create a directed edge of the network. This process is continued until all stubs are connected. Note it is possible under this process for multiple edges to exist between a given pair of nodes, or for a node to be linked to itself, but both these likelihoods become negligibly small (proportional to $1/N$) as $N \to \infty$. Note also that interbank positions are not netted, so directed edges may exist in both directions between any two nodes of the banking network.

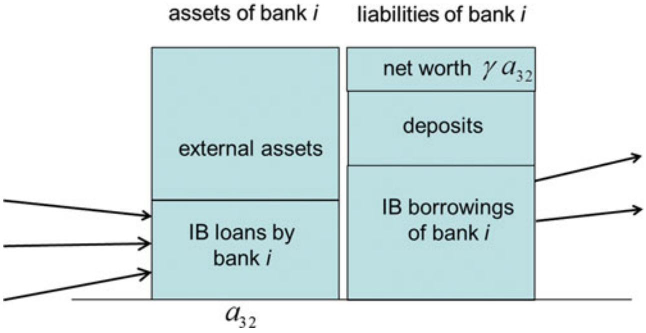


Fig. 2.2 Schematic balance sheet of banks in the $(j,k) = (3,2)$ class

The second step of the network generation process, the creation of balance sheets for each bank node, can vary considerably from model to model. In both the GK and NYYA models, the balance sheet quantities of a node depend on its in-degree (number of debtors) j and out-degree (number of creditors) k; we collectively refer to all banks with j debtors and k creditors as the "(j,k)-class". The total assets a_{jk} of a (j,k)-class bank are the sum of its external assets e_{jk} (such as property assets), and its interbank assets, i.e., the sum of its j loans to other banks, see Fig. 2.2. The liabilities side of the balance sheet is composed of the interbank liabilities (sum of the k loans taken from other banks) and customer deposits. The amount by which the total assets exceed the total liabilities is termed the *net worth* of the bank, and is denoted c_{jk} for banks in the (j,k) class. Within both the GK and NYYA models the net worth c_{jk} is assumed (in the initial, shock-free, state) to be proportional to the total assets a_{jk} of the bank:

$$c_{jk} = \gamma a_{jk}, \tag{2.1}$$

where the constant of proportionality γ is termed the "percentage net worth" or "capital reserve fraction". Note that shareholders' funds and subordinated debt are not considered here as useful to the loss absorption capacity; thus only three categories (interbank, customer deposits, and capital) appear on the liabilities side of the balance sheets.

An important difference between the GK and NYYA models is in how they assign values to loans, see Fig. 2.3. Recall the number of loans is determined by the number of directed edges in the skeleton structure of the first step, but there remains considerable freedom in allocating the weight to each edge. In the GK model (Fig. 2.3a), each bank is assumed to have precisely 20 % of its assets as interbank assets, and all in-edges to a (j,k)-class node (i.e. all asset loans of a (j,k) bank) are assigned equal weight $0.2/j$ (in units where the total assets of every bank equals unity):

$$a_{jk} = 1, \quad e_{jk} = 0.8 \quad \text{for all } (j,k) \text{ classes.} \tag{2.2}$$

This case represents a maximum-diversity lending strategy, where banks give loans of equal size to all their debtors [11].

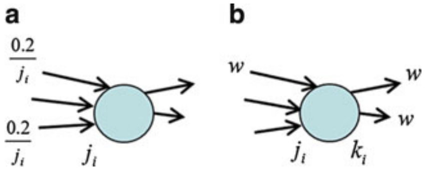


Fig. 2.3 Loan sizes in each of the models for a bank in the (j_i, k_i) class. In the GK model, all asset loans are of size $0.2/j_i$; liability loans are determined endogenously (by the random linking of in-stubs to out-stubs described in Sect. 2.2.1). In the NYYA model, every loan in the network is of equal size w. (**a**) GK model. (**b**) NYYA model

Table 2.1 Summary of main balance sheet quantities within the GK and NYYA models (see Gai and Kapadia [11] and Nier et al. [25] for details)

	GK	NYYA
Total assets of a (j,k)-class bank	$a_{jk} = 1$	$a_{jk} = \tilde{e} + w\max(j,k)$
Net worth of a (j,k)-class bank	$c_{jk} = \gamma a_{jk}$	$c_{jk} = \gamma a_{jk}$
Size of asset loans of (j,k)-class bank	$\frac{0.2}{j}$	w
External assets of (j,k)-class bank	$e_{jk} = 0.8$	$e_{jk} = \tilde{e} + w\max(0, k-j)$

In the model of NYYA, on the other hand, the same weight w is assigned to all directed edges in the network (Fig. 2.3b). A (j,k)-class node therefore has interbank assets of jw, and interbank liabilities of kw. To ensure all banks are initially solvent, NYYA describe a process for distributing a pool of external assets over the banks (see Nier et al. [25] for details). As a consequence, the resulting total assets and external assets may respectively be written as

$$a_{jk} = w\max(j,k) + \tilde{e}, \quad e_{jk} = a_{jk} - jw \quad \text{for all } (j,k) \text{ classes}, \tag{2.3}$$

where $\tilde{e}$ is related to the pool of external assets. The balance sheet quantities and their definitions within the two models considered are summarized in Table 2.1.

2.2.2 Contagion Mechanisms

Having generated the banking system via the network skeleton structure and balance sheet allocations, the dynamics of cascading defaults can then be investigated. Recall that the banks' balance sheet have been set up so that the system is initially in equilibrium, i.e., total assets for each bank equals its total liabilities plus its net worth. The effect of an exogenous shock is simulated, typically by setting to zero the external assets of one (or more) banks. The shocked bank(s) may be chosen randomly from all banks in the simulation, or a specific (j,k)-class may be targeted—the latter case allows us to investigate the impact of the size of the initially shocked bank upon the final cascade size (see Results section). The initial exogenous shock is intended to model, for example, a sudden decrease in the market value of the external assets held by the bank. The decrease may lead to a situation where the total liabilities of the bank now exceed the total assets: in this case, the bank is deemed to be in default As a consequence, the bank will be unable to repay its creditors the full values of their loans; the loans from these creditors to the defaulted bank are termed "distressed". The creditors (in network terminology, the out-neighbors of the original "seed" bank) experience a shock to their balance sheets at the next timestep due to writing-down the value of the distressed loans. If at any time the total of the shocks received by a bank (i.e. the total losses to date on its loan portfolio) exceeds the net worth of the bank, then its liabilities exceed its assets, and it is deemed to be in default. The defaulted bank then passes shocks to its creditors in the system, and so the cascade or contagion may spread through the banking network. Timesteps are modelled as being discrete, with possibly many banks defaulting simultaneously in each timestep, and with the shocks transmitted to their creditors taking effect in the following timestep.

The mechanism of shock transmission is treated differently by GK and by NYYA, and this is an important distinction between the models.

2.2.2.1 Shock Transmission in the GK Model

In the GK model, defaulted banks do not repay any portion of their outstanding interbank debts because the timescale for any recovery on these defaulted loans is assumed to exceed the timescale of the contagion spread in the system. Consequently, all creditors of a bank which defaulted in timestep n receive, at timestep $n+1$, a shock of magnitude equal to the total size of their loan to the defaulted bank. If multiple banks defaulted at timestep n, then a bank which is a creditor of several of these will receive multiple shocks at timestep $n+1$. Specifically, if the creditor bank is in the (j,k) class, then it receives a total shock of size $0.2\mu/j$, where μ is the number of its asset loans which defaulted at timestep n (since each loan is of size $0.2/j$, see Table 2.1). This process of shock transmission continues until there are no new defaults, at which point the cascade terminates.

2.2.2.2 Shock Transmission in the NYYA Model

The NYYA model allows for the possibility of non-zero recovery on defaulted loans. Suppose the total shock received by a (j,k)-class bank from all its defaulted debtors is of size σ, and this shock is sufficient to make the bank default, i.e., $\sigma > c_{jk}$. The amount $\sigma - c_{jk}$ by which total liabilities now exceed total assets for the bank is distributed evenly among the k creditors of the bank, with the proviso that no creditor can lose more than the size w of its original loan (recall every loan in the NYYA system is the same size w, see Table 2.1). Thus the shock transmitted to each creditor of the defaulted bank is

$$\min\left(\frac{\sigma - c_{jk}}{k}, w\right). \tag{2.4}$$

As in the GK model, shocks transmitted from banks which default at timestep n will affect the creditor banks at timestep $n+1$, and a cascade of banks failures may ensue. This cascade mechanism bears some resemblance to the "fictitious default" cascade used by Eisenberg and Noe [8] ("EN" for short) to determine the clearing payment vector in a system with defaults, see Appendix A. However, the NYYA cascades are not identical to the EN cascades. When a bank defaults in the NYYA model, it transmits a once-off shock to each of its creditors, but then plays no further role in the dynamics of the system. In particular, any shocks received by this bank subsequent to its default do not affect its creditors. In contrast, the EN clearing algorithm effectively requires defaulted banks to transmit newly-received shocks to their creditors. Although the EN algorithm is not the main focus of this paper, we present in Sect. 2.5 (see Figs. 2.5a and 2.6a) numerical results for the fraction of defaults in EN cascades. The results are qualitatively similar, though not identical, to those obtained using the NYYA contagion dynamics, the difference being most notable in cases where a large fraction of the network is in default.

2.2.3 Liquidity Risk

In both the GK and NYYA dynamics, it is possible to include liquidity risk effects in a simple fashion. Suppose that at timestep n, a fraction ρ^n of all banks in the system have already defaulted. It is plausible that the market value of external assets (e.g., property) will be adversely affected by the weakened banking system. A bank needing to liquidate its external assets may, for example, find it difficult to realise the full value in a "fire sale" scenario. To model the effects of this system-wide effect, we assume that at timestep n the external assets of a (j,k)-class bank are marked-to-market as

$$e_{jk}\exp\left(-\alpha\rho^n\right). \tag{2.5}$$

The liquidity risk parameter α measures the influence of the system contagion upon asset prices; note when $\alpha = 0$ the external asset values are constant over time, but for $\alpha > 0$ the asset values decrease with increasing contagion levels. This effect is included in the dynamics of the GK and NYYA models by subtracting the quantity $e_{jk}\left[1-\exp(-\alpha\rho^n)\right]$ from the net worth c_{jk} of the (j,k)-class banks. Thus, for example, banks default in the NYYA model if the incoming shock s is bigger than $c_{jk} - e_{jk}\left[1-\exp(-\alpha\rho^n)\right]$ (the fire-sale adjusted net worth), and the shock transmission equation (2.4) is generalized to

$$\min\left(\frac{\sigma - c_{jk} + e_{jk}\left[1-\exp(-\alpha\rho^n)\right]}{k}, w\right), \tag{2.6}$$

for $\alpha \geq 0$. A similar modification applies in the GK model. Interestingly, if α is sufficiently large, the liquidity risk effect can lead to banks defaulting even if they receive no shocks from debtors, because their net worth is obliterated by the fall in market value of their external assets. Consequences of this are explored in the Results section.

2.2.4 Monte Carlo Simulations

The steps needed to study the models using Monte Carlo simulation are now clear. In each realization a skeleton structure for a network of N nodes with joint in- and out-degree distribution p_{jk} is first constructed. Then balance sheets are assigned to each node, consistent with the specific model chosen (see Table 2.1). The cascade of defaults initiated by an exogenous shock to one (or more) banks proceeds on a timestep-by-timestep basis, following the dynamics of either the zero recovery (GK) or non-zero recovery (NYYA) prescription for shock transmission. When no further defaults occur, the fraction of defaulted banks (the "cascade size") is recorded, and then another realization may begin. When a sufficiently large number of realizations are recorded, the average cascade size (and potentially further statistics, i.e., the variance, of the cascade size) may be calculated in a reproducible (up to statistical scatter) manner. Monte Carlo simulations of this type were implemented in GK and NYYA; our focus in the remainder of this paper is on analytical approaches to predicting the average size of cascades, and so avoiding the need for computationally expensive numerical simulations.

2.3 Theory

In this section we derive analytical equations which allow us to calculate the expected fraction of defaults in a banking network with a given topology (defined by p_{jk}). Like related approaches for cascades on undirected networks [13, 14], the method is only approximate for finite-sized networks because it assumes the $N \to \infty$ limit of infinite system size. However, in practice we find it nevertheless gives reasonably accurate results for networks as small as $N = 25$ banks (see Sect. 2.5).

2.3.1 Thresholds for Default

We begin by defining the threshold level M_{jk}^{n} as the maximum number m of distressed loans that can be sustained by a (j,k)-class bank at timestep n without the bank defaulting at timestep $n+1$. If a (j,k)-class bank has m defaulted debtors, with $m > M_{jk}^{n}$, then it will default in the subsequent timestep, otherwise it will remain solvent. As we show below, the GK model is easily expressed in terms of thresholds, but thresholds can be defined for the NYYA model only under an approximating assumption.

In the GK model a bank in the (j,k) class has total assets of unity ($a_{jk} = 1$), net worth of $c_{jk} = \gamma a_{jk} = \gamma$, and each distressed loan carries a shock of $0.2/j$. In the absence of a liquidity risk (fire sale) factor, the (j,k) bank would then default if the sum of the shocks it receives from its m defaulted debtors exceeds its net worth, i.e., if $0.2m/j > \gamma$, giving $M_{jk}^n = \lfloor 5jc_{jk} \rfloor$, where $\lfloor \cdot \rfloor$ is the floor function (returning the greatest integer less than or equal to its argument). Liquidity risk may also be included in models of this type by appropriately reducing the effective net worth, and we can write the threshold levels in their most general form as

$$M_{jk}^n = \min\left\{ j, \max\left\{ \lfloor 5jc_{jk} - 5je_{jk}\left(1 - e^{-\alpha\rho^n}\right) \rfloor, -1 \right\}\right\}. \tag{2.7}$$

Here e_{jk} is the value of external assets for (j,k)-class banks, α is the liquidity risk parameter introduced in Sect. 2.2 and we constrain M_{jk}^n to be between -1 and j. Note that this expression for M_{jk}^n is constant over time n if $\alpha = 0$, and is decreasing in time if α is positive and ρ^n is increasing.

In the NYYA model the size of the write-down shock on a newly-distressed loan depends on how large the shock received by the debtor bank was compared to its net worth. This means that there will, in general, be a distribution of shocks of various sizes in the system, and this distribution will change in time. Denoting the distribution of shock sizes by $S^n(\sigma)$—so that at timestep n a randomly-chosen distressed loan (i.e. an out-edge of a defaulted bank node) carries a shock of size σ with probability $S^n(\sigma)$—we would require m-fold convolutions of $S^n(\sigma)$ to correctly describe the shock received by a bank with m distressed asset loans (as the sum of m independent draws of shock values from $S^n(\sigma)$). It is clearly desirable to find a simple parametrization of $S^n(\sigma)$ to make the model computationally tractable, even at the loss of some accuracy. With this in mind, we approximate the true value of the shock received by a bank with m distressed loans at timestep n by ms^n, where s^n is the average shock on all distressed loans in the system at that timestep. Effectively we are replacing the true distribution $S(\sigma)$ of shock sizes by a delta function distribution: $S^n(\sigma) \mapsto \delta(\sigma - s^n)$, where s^n is the average shock $s^n = \int \sigma S^n(\sigma) d\sigma$; in other words, every distressed loan at timestep n is assumed to have equal recovery value $w - s^n$. This approximation turns out to work rather well because in cases where many debtors are in default, the total shock received by a creditor is well approximated by m times the average shock. However we will also show examples (in the Results section) where the approximation of the shock distribution $S^n(\sigma)$ by a delta function leads to less accurate results.

Using this approximation, the NYYA threshold levels are:

$$M_{jk}^n = \min\left\{ j, \max\left\{ \lfloor \frac{1}{s^n}\left[c_{jk} - e_{jk}\left(1 - e^{-\alpha\rho^n}\right)\right] \rfloor, -1 \right\}\right\}. \tag{2.8}$$

The time dependence of the thresholds in this case is due to both liquidity risk ($\alpha > 0$), and to the time-varying nature of the (mean) shock size s^n. In Appendix B we derive an iteration equation for s^n, consistent with the general model (2.12) and (2.13) below and based on the approximation of the true shock size distribution by a delta function.

2.3.2 General Theory

We consider (j,k)-class banks, of which there are approximately Np_{jk} in any given network realization (for sufficiently large N). Each bank in the (j,k) class has j debtors, each of which may be either solvent or in default at a specific time. Given that a bank is in the (j,k) class, we define $u^n_{jk}(m)$ as the probability that the bank (1) is solvent at timestep n and (2) has m distressed asset loans (due to the default of the corresponding debtors in earlier cascades). According to its definition, the sum of $u^n_{jk}(m)$ over all m gives the fraction of (j,k)-class banks which are solvent at timestep n:

$$\sum_{m=0}^{j} u^n_{jk}(m) = 1 - \rho^n_{jk}, \tag{2.9}$$

where ρ^n_{jk} denotes the fraction of (j,k)-class banks which are in default at timestep n. In a slight abuse of mathematical terminology we will refer to $u^n_{jk}(m)$ as a "distribution", but note from (2.9) that the sum of $u^n_{jk}(m)$ over all m is not unity.

We consider how the states of the banks change from timestep n to timestep $n+1$, and update the $u^n_{jk}(m)$ distribution accordingly. The update occurs in two stages: first a "node update" stage, where the states of the banks are updated, followed by an "edge update", where the $u^n_{jk}(m)$ distribution is updated to give $u^{n+1}_{jk}(m)$. In the node update stage, banks in the (j,k) class default if their number of distressed loans m at timestep n exceeds their threshold M^n_{jk} (see Sect. 2.3.1). Thus the newly defaulting fraction of (j,k)-class banks is made up of those who were previously solvent but now have m values above threshold. These newly defaulted banks increase the total default fraction of the (j,k) class by the amount:

$$\rho^{n+1}_{jk} - \rho^n_{jk} = \sum_{m=M^n_{jk}+1}^{j} u^n_{jk}(m). \tag{2.10}$$

Each newly defaulted (j,k)-class bank is a debtor of k other banks in the system and correspondingly triggers k newly-distressed loans: this is the edge update stage between timestep n and timestep $n+1$. The number of newly-distressed loans in the network due to defaults in the (j,k) class of banks is approximately $Np_{jk}k(\rho^{n+1}_{jk} - \rho^n_{jk})$ (since there are Np_{jk} such banks, each newly-defaulted with probability $\rho^{n+1}_{jk} - \rho^n_{jk}$, and each with k creditors). Summing over all classes gives

$$N\sum_{j,k} kp_{jk}\left(\rho^{n+1}_{jk} - \rho^n_{jk}\right) \tag{2.11}$$

as the number of newly-distressed loans in the system. The total number of loans which were not distressed at timestep n is similarly calculated as $N\sum_{j,k} kp_{jk}(1-\rho^n_{jk})$. So the probability that a previously-undistressed loan will be distressed at timestep $n+1$ is given by

$$f^{n+1} = \frac{\sum_{j,k} k p_{jk} \left(\rho_{jk}^{n+1} - \rho_{jk}^{n} \right)}{\sum_{j,k} k p_{jk} \left(1 - \rho_{jk}^{n} \right)} = \frac{\sum_{j,k} k p_{jk} \sum_{m=M_{jk}^{n}+1}^{j} u_{jk}^{n}(m)}{\sum_{j,k} k p_{jk} \sum_{m=0}^{j} u_{jk}^{n}(m)}. \tag{2.12}$$

Consider a (j,k)-class bank which remains solvent and has exactly m distressed asset loans at timestep $n+1$. This bank was also solvent at timestep n and had some number $\ell \leq \min(m, M_{jk}^{n})$ of distressed asset loans at timestep n. Amongst the remaining $j-\ell$ asset loans of this bank, exactly $m-\ell$ of the loans must have become newly distressed due to the debtor bank having defaulted in the first stage of the update: this happens independently to each of the $j-\ell$ loans with probability f^{n+1}. If we introduce the convenient notation $B_{i}^{k}(p)$ for the binomial probability $\binom{k}{i} p^{i}(1-p)^{k-i}$, the probability that a (j,k)-class bank remains solvent and has exactly m distressed asset loans at timestep $n+1$ can be written as

$$u_{jk}^{n+1}(m) = \sum_{\ell=0}^{\min(m, M_{jk}^{n})} B_{m-\ell}^{j-\ell}\left(f^{n+1}\right) u_{jk}^{n}(\ell). \tag{2.13}$$

Equations (2.12) and (2.13) together define the updating of the state variables $u_{jk}(m)$ and f in terms of the $u_{jk}(m)$ distribution at timestep n. Given the initial condition—for instance, if a randomly-chosen fraction ρ^{0} of all banks are initially subject to default-causing shocks, this is $u_{jk}^{0}(m) = \left(1-\rho^{0}\right) B_{m}^{j}\left(\rho^{0}\right)$—it is straightforward to iterate the system given by (2.12) and (2.13) forward through the discrete timesteps until it converges to a steady state. The total fraction of defaulted banks in the system at timestep n is given by summing (2.9) over all (j,k) classes:

$$\rho^{n} = 1 - \sum_{j,k} p_{jk} \sum_{m=0}^{j} u_{jk}^{n}(m), \tag{2.14}$$

and the steady-state value of this quantity (as $n \to \infty$) is reported for various cases in Sect. 2.5 below.

In Sect. 2.4 we prove that a certain class of models, including GK, admits an exact reduction of the system described here to just two state variables. In the GK model, and for the case where a fraction ρ^{0} of the banks are chosen at random to be the seed defaults, the fraction of bank defaults ρ^{n} and the fraction of edge defaults g^{n} are given by the recurrence

$$\rho^{n+1} = \rho^{0} + \left(1-\rho^{0}\right) \sum_{j,k} p_{jk} \sum_{m=M_{jk}^{n}+1}^{j} B_{m}^{j}\left(g^{n}\right) \tag{2.15}$$

$$g^{n+1} = \rho^{0} + \left(1-\rho^{0}\right) \sum_{j,k} \frac{k}{z} p_{jk} \sum_{m=M_{jk}^{n}+1}^{j} B_{m}^{j}\left(g^{n}\right), \tag{2.16}$$

with the initial condition $g^{0} = \rho^{0}$.

For the NYYA model, we use the mean-shock-size approximation discussed in Sect. 2.3.1, so the thresholds M_{jk}^n are given by Eq. (2.8). Then the iteration equation for s^n (see Appendix B), along with Eqs. (2.12) and (2.13), gives us a system of equations for $u_{jk}^{n+1}(m)$, f^{n+1}, and s^{n+1} in terms of the values of these quantities at the previous timstep. Results for both models are compared with Monte Carlo simulations in Sect. 2.5.

2.4 Simplified Theory

In this section we show that the iteration of the system defined by Eqs. (2.12) and (2.13) in order to obtain the expected fraction of defaulted banks (as given by Eq. (2.14)) may be dramatically simplified in certain cases. A sufficient condition for this simplified theory to exactly match the full theory of Eqs. (2.12) and (2.13) is:

Condition 1. For every (j,k) class with $p_{jk} > 0$, the threshold level M_{jk}^n is a non-increasing function of n.

This condition holds if the threshold levels for each (j,k) class are constant, or decreasing with time, as in the GK model. For the NYYA model, cases where the shock size decreases over time may have thresholds M_{jk}^n which increase with n, and so this model does not satisfy Condition 1.

2.4.1 Simplified Theory for GK

Focussing now on the GK model, whose thresholds (2.7) satisfy Condition 1, we claim that at timestep n, the distribution for the number m of distressed loans of solvent banks is a binomial distribution, at least for m values below the threshold:

$$u_{jk}^n(m) = \left(1 - \rho_{jk}^0\right) B_m^j\left(g^n\right) \qquad \text{for } m \le M_{jk}^n, \tag{2.17}$$

and the fraction of distressed edges is

$$g^n = \sum_{j,k} \frac{k}{z} p_{jk} \rho_{jk}^n. \tag{2.18}$$

Here ρ_{jk}^0 is the initially defaulted fraction of (j,k)-class banks and ρ_{jk}^n is the defaulted fraction of (j,k)-class banks at timestep n. For the case $m > M_{jk}^n$, the values $u_{jk}^n(m)$ are slightly more complicated in form: they are given by the update Eq. (2.13) for level n, with the right-hand side given using (2.17) at the level $n-1$. As we show below, the result (2.17) is sufficient to determine the expected fraction of defaulted banks at any timestep n.

To prove our claim, we use an induction argument, showing that if the subthreshold distribution at timestep n is assumed to take the form (2.17) and (2.18) then the distribution at timestep $n+1$ (as given by Eq. (2.13) of the full theory) is also of the form (2.17) and (2.18). Substituting for $u^n_{jk}(\ell)$ in (2.13) using (2.17) yields

$$u^{n+1}_{jk}(m) = \left(1-\rho^0_{jk}\right) \sum_{\ell=0}^{\min\left(m,M^n_{jk}\right)} B^{j-\ell}_{m-\ell}\left(f^{n+1}\right) B^j_\ell\left(g^n\right). \tag{2.19}$$

To satisfy (2.17) at timestep $n+1$ we need only consider values of m between 0 and M^{n+1}_{jk}, and by Condition 1 we have $M^{n+1}_{jk} \leq M^n_{jk}$, so that $0 \leq m \leq M^{n+1}_{jk} \leq M^n_{jk}$, and thus the upper limit on the summation in (2.19) is $\min\left(m,M^n_{jk}\right) = m$. The sum in (2.19) is therefore a convolution sum of two binomial distributions, which is itself a binomial distribution:

$$u^{n+1}_{jk}(m) = \left(1-\rho^0_{jk}\right) B^j_m\left(g^{n+1}\right) \qquad \text{for } m \leq M^{n+1}_{jk}, \tag{2.20}$$

Here g^{n+1} is given by $g^{n+1} = g^n + (1-g^n) f^{n+1}$. One can now use (2.12) and (2.18) to verify that

$$g^{n+1} = \sum_{j,k} \frac{k}{z} p_{jk} \rho^{n+1}_{jk}. \tag{2.21}$$

By assuming the form (2.17) and (2.18) at timestep n we have shown the full theory yields the corresponding result (2.20) and (2.21) at timestep $n+1$. The induction proof is completed by verifying that the initial condition is given by

$$u^0_{jk}(m) = \left(1-\rho^0_{jk}\right) B^j_m\left(g^0\right) \qquad \text{for } m = 0 \text{ to } j, \tag{2.22}$$

$$gamma^0 = \sum_{j,k} \frac{k}{z} p_{jk} \rho^0_{jk} \tag{2.23}$$

which is of the form (2.17) and (2.18).

Using the binomial distribution for u^n_{jk} in (2.9) and (2.10) gives the update equations for ρ^{n+1} and g^{n+1} in terms of the parameter g^n only:

$$\begin{aligned}
\rho^{n+1} = \sum_{j,k} p_{jk}\rho^{n+1}_{jk} &= 1 - \sum_{j,k} p_{jk}\left(1-\rho^0_{jk}\right) \sum_{m=0}^{M^n_{jk}} B^j_m\left(g^n\right) \\
&= 1 - \sum_{j,k} p_{jk}\left(1-\rho^0_{jk}\right)\left(1 - \sum_{m=M^n_{jk}+1}^{j} B^j_m\left(g^n\right)\right) \\
&= \rho^0 + \sum_{j,k} p_{jk}\left(1-\rho^0_{jk}\right) \sum_{m=M^n_{jk}+1}^{j} B^j_m\left(g^n\right),
\end{aligned} \tag{2.24}$$

and

$$g^{n+1} = \sum_{j,k} \frac{k}{z} p_{jk} \rho_{jk}^{n+1} = \sum_{j,k} \frac{k}{z} p_{jk} \left[\rho_{jk}^{0} + \left(1 - \rho_{jk}^{0}\right) \sum_{m=M_{jk}^{n}+1}^{j} B_m^j \left(g^n\right) \right]$$

$$= \rho^0 + \sum_{j,k} \frac{k}{z} p_{jk} \left(1 - \rho_{jk}^{0}\right) \sum_{m=M_{jk}^{n}+1}^{j} B_m^j \left(g^n\right), \qquad (2.25)$$

where $\rho^0 = \sum_{j,k} p_{jk} \rho_{jk}^0$ is the overall fraction of initially defaulted banks. In the case where a fraction ρ^0 of the banks are chosen at random to be the seed defaults we have $\rho_{jk}^0 = \rho^0$ for all (j,k) classes, and Eqs. (2.24) and (2.25) reduce to Eqs. (2.15) and (2.16).

The expected size of global cascades in a given GK-model network has essentially been reduced to solving the single Eq. (2.16), since ρ^{n+1} can be immediately determined by substituting g^n into (2.15). Equation (2.16) is of the form $g^{n+1} = J\left(g^n\right)$, and the function $J(\cdot)$ is non-decreasing on $[0,1]$. It follows that $g^{n+1} \geq g^n$ for all n, and iteration of the map leads to the solution g^∞ of the fixed-point equation $g^\infty = J\left(g^\infty\right)$. The corresponding steady-state fraction of defaulted banks is determined by substituting g^∞ for g^n in (2.15).

Equations of this sort, giving the expected size of cascades on directed networks, have been previously derived in various contexts [2, 12]. In Gleeson [12], the main focus is on percolation-type phenomena (see also the undirected networks case Gleeson [13]), while Amini et al. [2] consider more complicated dynamics but take the limit $\rho^0 \to 0$. The general case (2.24) and (2.25) where initial default fractions can be different for each (j,k) class has not, to our knowledge, been considered previously, even in Monte Carlo simulations.

In the limit $\rho^0 \to 0^+$, the scalar map $g^{n+1} = J\left(g^n\right)$ has a fixed point at $g^n = 0$, but it is an unstable fixed point if $J'(0) > 1$, where J' is the derivative of the function J. Thus the condition for an infinitesimally small seed fraction to grow to a large-scale cascade may, using (2.16), be written as

$$J'(0) = \sum_{j,k} \frac{jk}{z} p_{jk} \Theta \left[\tfrac{0.2}{j} - c_{jk} \right] > 1, \qquad (2.26)$$

where the GK threshold (2.7) for $m = 1$ and $\rho^0 = 0$ has been used, and Θ is the Heaviside step function ($\Theta(x) = 1$ for $x > 0$; $\Theta(x) = 0$ for $x \leq 0$). This “cascade condition” has been derived in a rather different fashion by GK; they extend Watts’ (2002) percolation theory approach from his work on undirected networks to the case of directed networks considered here. In Gleeson and Cahalane [14] and Gleeson [13], the generalization of this result to cases where ρ^0 is finite but small has been given for cascades on undirected networks. Similar “higher-order cascade conditions” may similarly be derived for this directed-network case, but are beyond the scope of the present paper.

2.4.2 Frequency of Contagion Events

The simplified Eqs. (2.15) and (2.16), and indeed the more general method of Sect. 2.3, allow the specification of a fraction ρ^0 (or ρ^0_{jk} in the case of targeted (j,k) classes) of initially defaulted bank nodes. This fraction need not be small, and this feature allows us to investigate features of systemic risk due to simultaneous failure of more than one bank (see Results section). However, most work to date has focussed exclusively on the case where a single initially defaulted bank leads to a cascade of defaults through the network. Because our theory assumes an infinitely large network, some special attention must be paid to the case of a single "seed" default in the GK model. As we show in Appendix C, in this model the locality of the seed node determines whether, in a given realization, a cascade will reach global size, or remain restricted to a small neighborhood of the seed. The distribution of cascade sizes observed in single-seed GK simulations is thus typically bimodal: only a certain fraction (termed the *frequency*) of cascades reach a network-spanning size, the remainder remain small (typically only a few nodes). The average *extent* (i.e. size) of the global cascades is given by Eqs. (2.15) and (2.16), whereas the frequency of cascades which escape the neighborhood of the seed may be expressed in terms of the size of connected components for a suitable percolation problem, see Appendix C and the Results section. The NYYA model does not exhibit this sensitivity to the details of the neighborhood of the seed node(s), so its distribution of cascade sizes is quite narrowly centered on the mean cascade size given by theory; the same comment applies to the GK model with multiple seed nodes.

2.5 Results

2.5.1 GK Model

Figure 2.4a compares results of Monte Carlo simulations of the GK model (symbols) with the results of the simplified theory of Eqs. (2.15) and (2.16). As in Fig. 2.1 of the GK paper, we show the extent and frequency (see Appendix C) of contagion resulting from a single seed default in Erdös-Rényi directed random graphs with $N = 10^4$ nodes. The mean degree z of the network is varied to investigate the effects of connectivity levels upon the contagion spread. In such networks the in- and out-degree of a node (i.e., the number of debtors and creditors of a bank) are independent, and the joint distribution p_{jk} is a product of Poisson distributions:

$$p_{jk} = \frac{z^j}{j!}e^{-z}\frac{z^k}{k!}e^{-z}. \tag{2.27}$$

The formula for the contagion window derived in Gai and Kapadia [11] (which is the same as our Eq. (2.26)) predicts that cascades occur for z values between

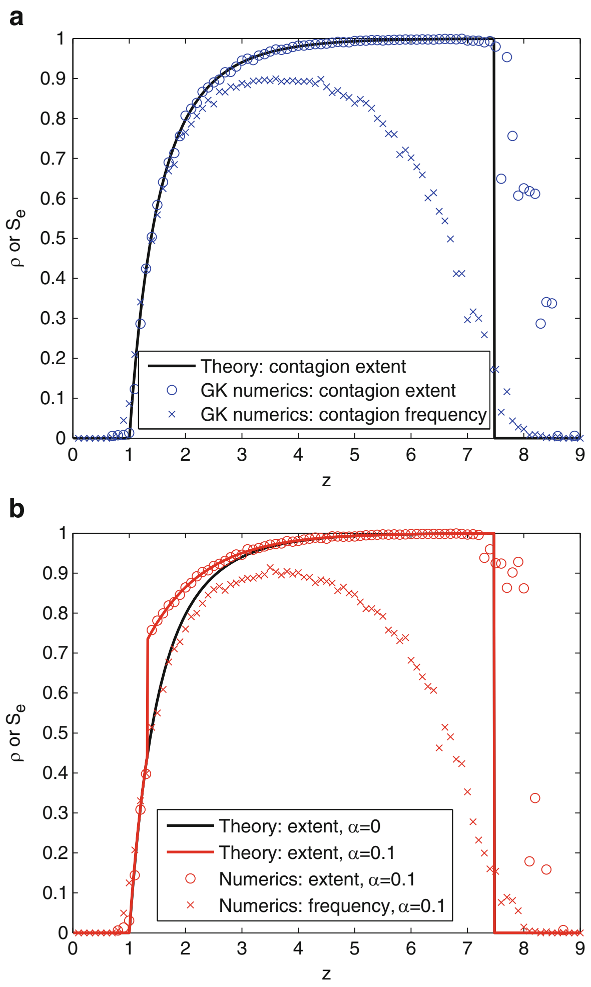


Fig. 2.4 Theory and Monte Carlo simulation results for GK model on Erdös-Rényi networks with $N = 10^4$ nodes and mean degree z. The percentage net worth is set to $\gamma = 3.5\,\%$ for all cases. Cascades which exceed 0.5 % of the network are considered as "global" cascades; the "extent" of contagion is the average size of these global cascades, while the "frequency" is the fraction of all cascades that become global cascades. In (**b**), the effects of non-zero liquidity risk are clearly seen for lower z values, and cause the appearance of a discontinuous transition which is not present in the $\alpha = 0$ case of (**a**). Monte Carlo numerical results are averages over 5,000 realizations

1 and 7.477, but our theory also accurately predicts the expected magnitude of these events. Moreover, as shown in Fig. 2.4b, our theory also accurately incorporates the effects of the liquidity risk model (2.5), capturing the discontinuous transition in cascade size which appears above $z = 1$ for the case $\alpha = 0.1$.

2.5.2 NYYA Benchmark Case

Figure 2.5a examines the benchmark case of NYYA; note our Monte Carlo simulation results match those presented in Chart 1 of Nier et al. [25]. The fraction of defaults (extent of contagion) is here plotted as a function of the percentage

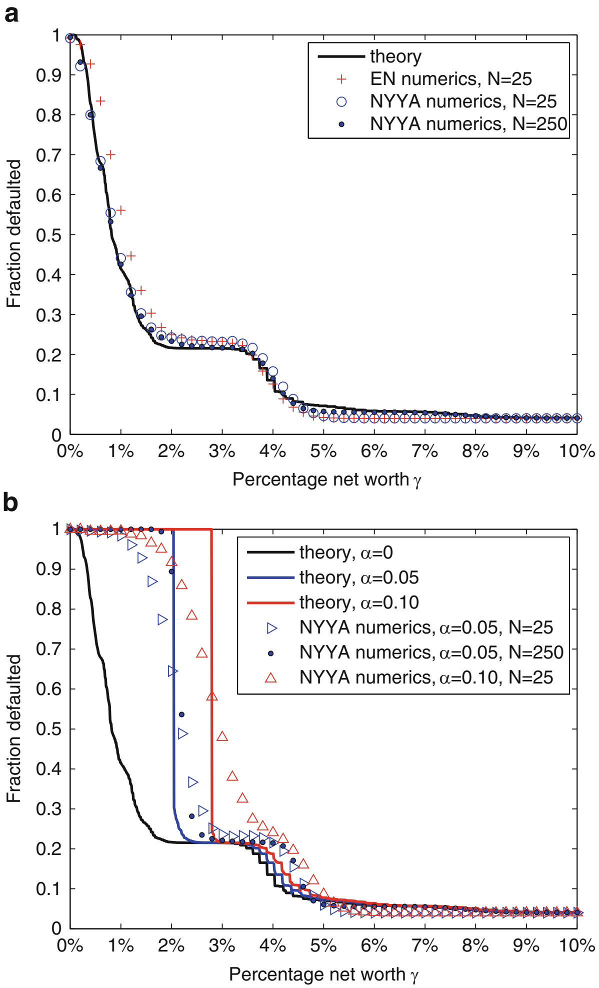


Fig. 2.5 Expected steady-state default fraction in Erdös-Rényi random graphs with mean degree $z = 5$. Monte Carlo numerical simulation results are averages over 5,000 realizations. In the networks with $N = 25$ nodes, cascades are initiated by the default of a single randomly-chosen node; in the larger networks with $N = 250$, ten randomly-chosen nodes are defaulted to begin the cascade; theory uses $\rho^0 = 1/25$

net worth parameter γ, as defined in Eq. (2.1). The network structure is again Erdös-Rényi, with p_{jk} given by (2.27), and mean degree $z = 5$. We also show Monte Carlo results for the default fraction resulting from the clearing vector algorithm of Eisenberg and Noe (see Appendix A). This algorithm gives results which are qualitatively similar in behavior (though not identical) to those generated by the NYYA shock transmission dynamics described in Eq. (2.6). As in the NYYA paper, our Monte Carlo simulations use $N = 25$ nodes (banks) in each realization, and cascades are initiated by a single randomly-chosen bank being defaulted by an exogenous shock. Despite this relatively small value of N, we find very good agreement between the theoretical prediction (which assumes the $N \to \infty$ limit) from Eqs. (2.12) and (2.13), and the Monte Carlo simulation results. The theory also enables us to examine the case where multiple banks are defaulted to begin the cascade. We demonstrate this by also showing numerical results for a larger Erdös-Rényi network of $N = 250$ nodes, with the same mean degree $z = 5$. In order to match the seed fraction of defaults, cascades in the larger networks are initiated by simultaneously shocking ten randomly-chosen banks (each shock being calibrated to wipe out the external assets of the bank), so $\rho^0 = 1/25 = 0.04$. The numerical results for this case are almost indistinguishable from the $N = 25$ case, and both cases match very well to the theory curve.

In Fig. 2.5b we increase the liquidity risk parameter from $\alpha = 0$ (as in Fig. 2.5a) to $\alpha = 0.05$ and $\alpha = 0.1$. For clarity, the results of the Eisenberg-Noe dynamics are not shown here, but as in Fig. 2.5a, they are qualitatively similar to the simulation results using the NYYA shock transmission dynamics. The theory predicts a discontinuous transition in ρ at γ values between 2 and 3 % for the $\alpha = 0.05$ and $\alpha = 0.1$ cases, but this is not well reproduced in Monte Carlo simulations with $N = 25$ nodes and $\rho^0 = 1/N$ (triangles). However, this is due to finite-N effects (i.e., due to having a finite-sized network whereas theory assumes the $N \to \infty$ limit), as can be seen by the much closer agreement between the theory and the $N = 250$ (with ten seed defaults) case (filled circles) for $\alpha = 0.05$.

A more serious discrepancy between theory and numerics can be seen in the γ range 4–5 %. Here the theory underpredicts the cascade size, and the difference is unaffected by increasing the size of the network. Detailed analysis of this case reveals that the root of the discrepancy is in fact the simplifying assumption made for the shock size distribution $S^n(\sigma)$ in the NYYA case (see Sect. 2.3.1). By replacing all shocks with the mean shock size we are underestimating (at timestep $n > 1$) the residual effects of the large shock which propagated from the first defaulted node(s) at timestep $n = 1$. Indeed, if we modify the Monte Carlo simulations to artificially replace all shocks at each timestep by their mean, we find excellent agreement between theory and numerics over all γ values. We conclude that the simplifying assumption $S^n(\sigma) \to \delta(\sigma - s^n)$ of the shock size distribution may lead to some errors, and further work on approximating $S^n(\sigma)$ by analytically tractable distributions is desirable. Despite this caveat, overall the theory works very well on the Erdös-Rényi random graphs studied by NYYA.

2.5.3 Networks with Fat-Tailed Degree Distributions

As noted in May and Arinaminpathy [18], empirical data on banking networks indicates that their in- and out-degree distributions are fat-tailed, and so it is important that theoretical approaches not be restricted to Erdös-Rényi networks. Accordingly, for Fig. 2.6 we generate a network with joint in- and out-degree

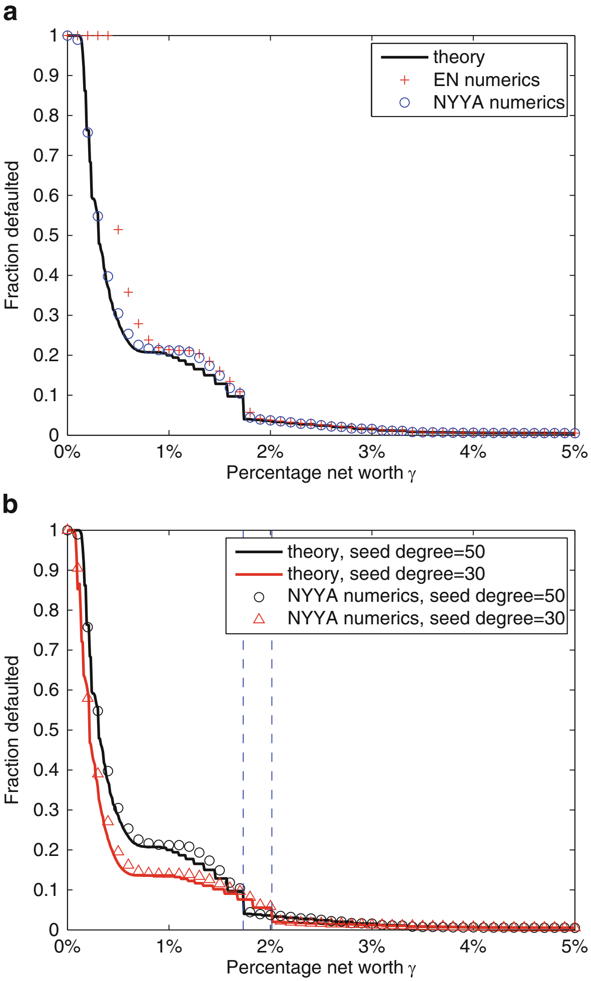


Fig. 2.6 Comparison of theory and Monte Carlo numerical simulations for banking networks with joint in- and out- degree distribution (2.28), with $N = 200$ nodes. Cascades are initiated by targeting a single node of a specific (j,k) degree class. Monte Carlo simulation results are averages over 5,000 realizations. The *dashed lines* in (**b**) mark the critical γ values given by (2.33)

distribution given by

$$p_{jk} = C\delta_{jk}k^{-1.7} \quad \text{for } k = 5, 10, 15, \ldots, 50. \tag{2.28}$$

Here C is a normalization constant (so that $\sum_{j,k} p_{jk} = 1$), and the exponent 1.7 has been chosen to be similar to that found for the in-degree distribution in the empirical data set of Boss et al. [6]. The Kronecker delta δ_{jk} appears in (2.28) to give our networks very strong correlations between in- and out-degrees: in contrast to the independent j and k distributions of (2.27), here we set the in- and out-degree of every node to be equal (i.e., each bank has equal numbers of debtors and creditors). We also consider for the first time the effect on the contagion of the size of the initially defaulting bank. If the single bank to be defaulted by the initial exogenous shock is chosen randomly from a specific (j,k) class, denoted (j',k'), then the initial values of ρ^n_{jk} are

$$\rho^0_{jk} = \begin{cases} \frac{1}{Np_{j'k'}} & \text{for } (j,k) = (j',k') \\ 0 & \text{for all other } (j,k) \text{ classes.} \end{cases} \tag{2.29}$$

The corresponding initial conditions for $u_{jk}(m)$ are:

$$u^0_{jk}(m) = \begin{cases} \left(1 - \frac{1}{Np_{j'k'}}\right) B^{j'}_m(g^0) & \text{for } (j,k) = (j',k') \\ B^j_m(g^0) & \text{for all other } (j,k) \text{ classes,} \end{cases} \tag{2.30}$$

where

$$g^0 = \sum_{j,k} \frac{k}{z} p_{jk} \rho^0_{jk} = \frac{k'}{Nz} \tag{2.31}$$

is the fraction of loans (edges) in the network which are initially distressed (i.e. have their debtor bank in default). We use $N = 200$ banks and ignore liquidity effects: $\alpha = 0$. All other parameters are as in the benchmark case of NYYA [25].

Figure 2.6a shows the theoretical and numerical results for the case where one of the largest banks in the network (i.e., with $j' = k' = 50$) is targeted initially. Note that the theory accurately matches to the NYYA Monte Carlo simulation results; also note that the Eisenberg-Noe clearing vector case is (at low γ values) somewhat further removed from the NYYA dynamics than in previous figures.

Figure 2.6b compares the results of Fig. 2.6a to the case where the targeted bank is from the class with $j' = k' = 30$, i.e., a mid-sized bank in this network. Theory and numerics again match well, and over most of the γ range the smaller target bank leads to smaller cascade sizes. Interestingly however, near $\gamma = 2\,\%$ is a range where the smaller target bank actually generates a larger cascade than the bigger target bank—this phenomenon is clearly visible in both numerical and theoretical results. To explain it, we consider the threshold levels at timestep $n = 0$ (and with $\alpha = 0$). The initially-targeted bank was subject to an exogenous shock that wiped out its external assets and each of its out-edges (liability loans) carries a residual shock (cf. (2.4)) of magnitude

$$s^0 = \min\left(\frac{e_{j'k'} - c_{j'k'}}{k'}, w\right), \tag{2.32}$$

where (j',k') denotes the class of the targeted bank. If a single such shock is to cause further defaults, say of a (j,k)-class node, then the threshold M^0_{jk} must be zero (cf. Eq. (2.8)). This requires $c_{jk} < s^0$ (note $\alpha = 0$ here), or, using (2.1),

$$\gamma < \frac{s^0}{a_{jk}}. \tag{2.33}$$

The largest critical value s^0/a_{jk} for γ occurs for the lowest $j = k$ value (because of the dependence of a_{jk} on degree, see Table 2.1) and this γ value for each case is marked by the vertical dashed lines in Fig. 2.6b—note in each case it matches the location of the sudden change in the contagion size. Essentially, this is the level of γ below which a single shock of magnitude s^0 can cause further defaults (moreover, our argument indicates that these further defaults will be among the smallest banks in the system). The shock magnitude s^0 given by (2.32) (see Table 2.1 for details of e_{jk} and c_{jk}) is a non-increasing function of k', and in the crucial γ range the value of s^0 is less for $k' = 50$ than for $k' = 30$. This is reflected in the respective critical values for γ, and allows the $k' = 30$ case to cause larger cascades than the $k' = 50$ case, at least while these cascades are relatively small.

2.6 Discussion

In this paper we have introduced an analytical method for calculating the expected size of contagion cascades in the banking network models of Gai and Kapadia [11] and Nier et al. [25]. Our method may be applied to cases with:

- An arbitrary joint distribution p_{jk} of in- and out-degrees (i.e., numbers of debtors and creditors) for banks in the network. This includes fat-tailed distributions; see Eq. (2.28) and Fig. 2.6;
- Arbitrary initial conditions for the cascade, including the targeting of one or more banks of a specified size (see Fig. 2.6);
- Liquidity risk effects (see Figs. 2.4 and 2.5).

In the general case, the theory gives a set of discrete-time update equations (2.12), (2.13), and (2.42) for a vector of unknowns $\mathbf{g}^n$, which is composed of the state variables f^n, $u^n_{jk}(m)$, and s^n. The update equations may be written in the form $\mathbf{g}^{n+1} = \mathbf{H}(\mathbf{g}^n)$ and this vector mapping is iterated to steady-state to find the fixed point solution $\mathbf{g}^\infty = \mathbf{H}(\mathbf{g}^\infty)$, hence giving the expected fraction of defaults ρ^∞, see Figs. 2.5 and 2.6 for examples. Under certain conditions it proves possible to simplify the equations to be iterated: as shown in Sect. 2.4, this reduces the vector $\mathbf{g}^n$ to a scalar g^n, with iteration map $g^{n+1} = J(g^n)$. The GK model is of this type, and the simplified Eqs. (2.15) and (2.16) were used to generate the theoretical results in Fig. 2.4. In all cases we find very good agreement between Monte Carlo simulations and theory, even on relatively small ($N = 25$) networks.

We expect it will prove possible to improve and extend these results in several ways. As noted in Sect. 2.5.2, the approximation of the shock size distribution in the NYYA model leads to some loss of accuracy, and this merits further attention. It is also desirable to develop analytical methods for calculating the frequency of cascades caused by single seeds in the GK model (see Appendix C). Even in its current form, however, the theory presented here is ideally suited to the study of some policy questions. For example, suppose the models are modified so that the capital reserve fraction γ is not the same for all banks in the system, instead depending on the size of the bank (i.e. $\gamma \mapsto \gamma_{jk}$). This requires only a slight modification of the existing equations. The question is then: how should γ_{jk} depend on the (j,k) class in order to optimally reduce the risk of contagion-induced systemic failure? Other possible extensions, such as allowing for the existence of subgroups of banks with different levels of interbank assets or with interbank loans/liabilities drawn from a prescribed distribution, are required to begin modelling the important non-homogeneities that are seen in the real banking system, and these will be the subject of future work.

For these and similar questions, it is likely that a general cascade condition (or "instability criterion"), analogous to Eq. (2.26) for the GK model, will prove very useful. Cascade conditions for dynamics with vector mappings have been derived for undirected networks (see Gleeson [13] and references therein), so we believe that similar methods may be applied to the directed networks analyzed here.

Finally, it is hoped that the methods introduced here will prove extendable beyond the stylized models of Gai and Kapadia [11] and Nier et al. [25], and in particular that related methods will be applicable to datasets from real-world banking networks. Ideally, such datasets would include information on bank sizes, connections, and the sizes of loans [4]. Modelling the distribution of loan sizes within a semi-analytical framework will be challenging, but the understanding gained of how network topology affects systemic risk on toy models will no doubt prove important to finding the solution.

Acknowledgements We acknowledge the work of undergraduate students Niamh Delaney and Arno Mayrhofer on an early version of the simulation codes used in this paper. Discussions with the participants at the Workshop on Financial Networks and Risk Assessment, hosted by MITACS at the Fields Institute, Toronto in May 2010 (particularly Rama Cont and Andreea Minca) are also gratefully acknowledged, as are the comments of Sébastien Lleo and Mark Davis. This work was funded by awards from Science Foundation Ireland (06/IN.1/I366, 06/MI/005 and 11/PI/1026), from an INSPIRE: IRCSET-Marie Curie International Mobility Fellowship in Science Engineering and Technology, and from the Natural Sciences and Engineering Research Council of Canada.

Appendix A: Generalized Eisenberg-Noe Clearing Vector Cascades

This Appendix provides a summary of the financial cascade framework of Eisenberg and Noe [8], placed in a slightly more general context. Extending their work somewhat ([8] combine the quantities Y_i and D_i into a single quantity $e_i = \mathrm{Y}_i - \mathrm{D}_i$), we identify the following stylized elements of a financial system consisting of N "banks" (which may include non-regulated leveraged institutions such as hedge funds). The assets A_i of bank i at a specific time consists of *external assets* Y_i (typically a portfolio of loans to external debtors) plus *internal assets* Z_i (typically in the form of interbank overnight loans). The liabilities of the bank includes *external debts* D_i (largely in the form of bank deposits, but also including long term debt) and *internal debt* X_i. The bank's *equity* is defined by $\mathrm{E}_i = \mathrm{Y}_i + \mathrm{Z}_i - \mathrm{D}_i - \mathrm{X}_i$ and is constrained to be non-negative.

The amounts $\mathrm{Y}, \mathrm{Z}, \mathrm{D}, \mathrm{X}$ refer to the notional value, or face value, of the loans, and are used to determine the relative claims by creditors in the event a debtor defaults. Internal debt and assets refer to contracts between the N banks in the system. Banks and institutions that are not part of the system are deemed to be part of the exterior, and their exposures are included as part of the external debts and assets. Let $\bar{L}_{ij}$ denote the notional exposure of bank j to bank i, that is to say, the amount i owes j. Note the constraints that hold for all i

$$\mathrm{Z}_i = \sum_j \bar{L}_{ji}, \quad \mathrm{X}_i = \sum_j \bar{L}_{ij}, \quad \sum_i \mathrm{Z}_i = \sum_i \mathrm{X}_i, \quad \bar{L}_{ii} = 0,$$

and that the matrix of exposures $\bar{L}$ is not symmetric.

A.1 Default Cascades

A healthy bank manages its books to maintain mark-to-market values with sufficient "economic capital" to provide an "equity buffer" against shocks to its balance sheet. This means that the bank maintains its asset-to-equity ratio A_i/e_i above a fixed threshold Λ_i (a typical value imposed by regulators might be 12.5).

Following a bank-specific catastrophic event, such as the discovery of a major fraud, or a system wide event, the assets of some banks may suddenly contract by more than the equity buffer. Assets are then insufficient to cover the debts, and these banks are deemed insolvent. The assets of an insolvent bank must be quickly liquidated, and any proceeds go to pay off that bank's creditors, in order of seniority. We now discuss three simple settlement mechanisms for how an insolvent bank i is removed from the system.

- Version A, the original mechanism of [8], supposes that external debt is always senior to internal debt. We define fractions $\pi_{ij} = \bar{L}_{ij}/\mathrm{X}_i$. If p_i denotes the amount available to pay i's internal debt, this amount is split amongst creditor banks in proportion to π_{ij}, that is bank j receives $\pi_{ij} p_i$. Given $\mathbf{p} = [p_1, \ldots, p_N]$, the clearing conditions are $p_i = 0$ if $\mathrm{Y}_i - \mathrm{D}_i + \sum_j \pi_{ji} p_j < 0$ and $p_i = \min(\mathrm{Y}_i - \mathrm{D}_i + \sum_j \pi_{ji} p_j, \mathrm{X}_i)$ if $\mathrm{Y}_i - \mathrm{D}_i + \sum_j \pi_{ji} p_j \geq 0$. We can write this as

$$p_i = F_i^{(A)}(\mathbf{p}) := \min(\mathrm{X}_i, \max(\mathrm{Y}_i + \sum_j \pi_{ji} p_j - \mathrm{D}_i, 0)), \quad i = 1, \ldots, N \qquad (2.34)$$

- Version B supposes that external and internal debt have equal seniority. We define fractions $\tilde{\pi}_{ij} = \bar{L}_{ij}/(\mathrm{D}_i + \mathrm{X}_i)$. If $\tilde{p}_i$ denotes the amount available to pay i's total debt, creditor bank j receives $\tilde{\pi}_{ji} \tilde{p}_i$ while the external creditors receive $\mathrm{D}_i \tilde{p}_i/(\mathrm{D}_i + \mathrm{X}_i)$. The clearing conditions are:

$$\tilde{p}_i = F_i^{(B)}(\tilde{\mathbf{p}}) := \min(\mathrm{D}_i + \mathrm{X}_i, \mathrm{Y}_i + \sum_j \tilde{\pi}_{ji}\tilde{p}_j), \quad i = 1,\ldots,N.$$

- Most simply, Version C supposes as in the GK model that the recovery from any insolvent bank is zero. That means the amount p_i available to pay i's internal debt is simply
$$p_i = F_i^{(C)}(\mathbf{p}) := \mathrm{X}_i\Theta(\mathrm{Y}_i - \mathrm{D}_i + \sum_j \pi_{ji}p_j)$$
where Θ denotes the Heaviside function.

Under each of these settlement mechanisms, any solution $\mathbf{p} = (p_1,\ldots,p_N) \in \mathbb{R}_+^N$ of the clearing conditions is called a "clearing vector". In the subsequent discussion we consider only version A. The existence result extends easily to versions B and C by considering fixed points of the monotonic mappings $F^{(B)}, F^{(C)} : \mathbb{R}_+^N \to \mathbb{R}_+^N$.

Proposition 2.1. *Consider a financial system with* $\mathrm{Y} = [\mathrm{Y}_1,\ldots,\mathrm{Y}_N], \mathrm{D} = [\mathrm{D}_1,\ldots,\mathrm{D}_N]$ *and matrix* $\bar{L} = (\bar{L}_{ij})_{i,j=1\ldots,N}$. *Then the mapping* $F^{(A)} : \mathbb{R}_+^N \to \mathbb{R}_+^N$ *defined by* (2.34) *has at least one clearing vector or fixed point* $\mathbf{p}^*$. *If in addition the system is "regular" (a natural economic constraint on the system), the clearing vector is unique.*

Proof. Existence is a straightforward application of the Tarski Fixed Point Theorem to the mapping F acting on the complete lattice

$$[\mathbf{0}, \bar{X}] := \{\mathbf{x} = [x_1,\ldots,x_N] \in \mathbb{R}_+^N : 0 \le x_i \le \bar{X}_i, i = 1,\ldots,N\}.$$

One simply verifies the easy monotonicity results that for any vectors $mathbf{f}0 \le \mathbf{p} \le \mathbf{p}' \le \mathrm{X}$ one has

$$\mathbf{0} \le F^{(A)}(\mathbf{0}) \le F^{(A)}(\mathbf{p}) \le F^{(A)}(\mathbf{p}') \le F^{(A)}(\mathrm{X}) \le \mathrm{X}$$

(where $\mathbf{a} \le \mathbf{b}$ for vectors means $a_i \le b_i$ for all $i = 1,\ldots,N$). For the definition of "regular" and the uniqueness result, please see [8].

A.2 Clearing Algorithm

Cascades of defaults arise when primary defaults trigger further losses to the remaining banks. The above proposition proves the existence of a unique "equilibrium" clearing vector that characterizes the end result of any such cascade. The following algorithm for version A of the settlement mechanism resolves the cascade to the fixed point $\mathbf{p}^*$ in at most $2N$ iterations by constructing an increasing sequence of defaulted banks $A^k \cup B^k, k = 0,1,\ldots$. Analogous (but simpler) algorithms are available for settlement mechanisms B and C.

1. **Step 0** Determine the primary defaults by writing a disjoint union $\{1,\ldots,N\} = A^0 \cup B^0 \cup C^0$ where

$$\begin{aligned} A^0 &= \{i|\mathrm{Y}_i + \mathrm{Z}_i - \mathrm{D}_i < 0\} \\ B^0 &= \{i|\mathrm{Y}_i + \mathrm{Z}_i - \mathrm{D}_i - \mathrm{X}_i < 0\} \setminus A^0 \\ C^0 &= \{1,\ldots,N\} \setminus (A^0 \cup B^0). \end{aligned}$$

2. **Step** k, $k = 1,2,\ldots$ Solve the $|B^{k-1}|$ dimensional system of equations:

$$p_i = \mathrm{Y}_i - \mathrm{D}_i + \sum_{j \in C^{k-1}} \pi_{ji} \mathrm{X}_j + \sum_{j \in B^{k-1}} \pi_{ji} p_j,\ i \in B^{k-1}$$

and define result to be $\mathbf{p}^{k*}$. Define a new decomposition

$$\begin{aligned} A^k &= A^{k-1} \cup \{i \in B^{k-1} | p_i^{k*} \le 0\} \\ B^k &= (B^{k-1} \setminus A^k) \cup \{i \in C^{k-1} | \mathrm{Y}_i - \mathrm{D}_i + \sum_{j \in C^{k-1}} \pi_{ji} \mathrm{X}_j + \sum_{j \in B^{k-1}} \pi_{ji} p_j^{k*} \le \mathrm{X}_i\} \\ C^k &= \{1,\ldots,N\} \setminus (A^k \cup B^k) \end{aligned}$$

and correspondingly

$$p_i^k = \begin{cases} 0 & i \in A^k \\ \mathrm{Y}_i + \sum_{j \in C^k} \pi_{ji} \mathrm{X}_j + \sum_{j \in B^k} \pi_{ji} p_j^{k*} - \mathrm{D}_i & i \in B^k \\ \mathrm{X}_i & i \in C^k. \end{cases} \tag{2.35}$$

If $A^k = A^{k-1}$ and $B^k = B^{k-1}$, then halt the algorithm and set $A^* = A^k, B^* = B^k, \mathbf{p}^* = \mathbf{p}^{k*}$. Otherwise perform step $k+1$.

Appendix B: Updating of Average Shock Strength for NYYA Model

Assuming a delta function distribution approximating $S^n(\sigma)$ as in Sect. 2.3.1, we need to count the number of loans (edges in the directed network) which link defaulted banks to solvent banks. In the notation of Sect. 2.3.2, the number of such “d-s” (for “defaulted-to-solvent”) edges in the network at timestep n is

$$N \sum_{j,k} p_{jk} \sum_{m=0}^{j} m\, u_{jk}^n(m), \tag{2.36}$$

since each solvent bank with m defaulted debtors contributes m d-s edges to the total. We assume that all these d-s edges at timestep n carry an equal shock s^n.

Now consider the situation at timestep $n+1$. Some of the d-s edges from timestep n are still d-s edges, although others will have become d-d ("defaulted-to-defaulted") edges. We count the number of d-s edges which remained as d-s from timestep n to timestep $n+1$ as

$$A^{\text{old}} = N \sum_{j,k} p_{jk} \sum_{m=0}^{M_{jk}^n} m\, u_{jk}^n(m). \tag{2.37}$$

Note the upper limit of M_{jk}^n for the sum over m (cf. Eq. (2.36)); this arises because the creditor banks in question remain solvent at timestep $n+1$.

The other mechanism generating d-s edges at timestep $n+1$ is the default of the debtor end of a timestep-n s-s (solvent-to-solvent) edge. Similar to (2.36), we can count the number of s-s edges at timestep n as

$$N \sum_{j,k} p_{jk} \sum_{m=0}^{j} (j-m) u_{jk}^n(m), \tag{2.38}$$

since each (solvent) (j,k)-class bank with m defaulted debtors must also have $j-m$ solvent debtors. Each of the s-s edges at timestep n becomes an d-s edge at timestep $n+1$ if (1) the debtor bank defaults during the timestep, and (2) the creditor bank remains solvent to at least timestep $n+1$. Noting that (1) occurs with probability f^{n+1} (see Eq. (1.12) of the main text), and that (2) requires $m \leq M_{jk}^n$, we obtain the number of new d-s edges at timestep $n+1$ as

$$A^{\text{new}} = f^{n+1} N \sum_{j,k} p_{jk} \sum_{m=0}^{M_{jk}^n} (j-m) u_{jk}^n(m). \tag{2.39}$$

The total number of d-s edges at timestep $n+1$ is then $A^{\text{old}} + A^{\text{new}}$, while the cumulative total of the shock sizes transmitted by these edges is

$$s^n A^{\text{old}} + \widetilde{s} A^{\text{new}}, \tag{2.40}$$

where $\widetilde{s}$ is the average shock on each newly-distressed loan (using (1.6) of the main text):

$$\widetilde{s} = \frac{\sum_{j,k} k p_{jk} \sum_{m=M_{jk}^n+1}^{j} u_{jk}^n(m) \min\left(\frac{ms^n - c_{jk} + e_{jk}[1-\exp(-\alpha\rho^n)]}{k}, w\right)}{\sum_{j,k} k p_{jk} \sum_{m=M_{jk}^n+1}^{j} u_{jk}^n(m)}. \tag{2.41}$$

Thus, under the simplifying assumption on the shock size distribution $(S^n(\sigma) \mapsto \delta(\sigma - s^n))$, we model the shocks on d-s edges at timestep $n+1$ to each be of equal size s^{n+1}, where

$$s^{n+1} = \frac{s^n A^{\text{old}} + \widetilde{s} A^{\text{new}}}{A^{\text{old}} + A^{\text{new}}}, \tag{2.42}$$

with A^{old}, A^{new}, and $\widetilde{s}$ given in terms of u_{jk}^n by Eqs. (2.37), (2.39), and (2.41), respectively. This gives an update equation for s^n in terms of known quantities from timestep n.

Appendix C: Frequency of Cascades for Single-Seed Initiation in GK Model

In this Appendix we consider the frequency of cascades in the GK model when initiated by a single seed node. Mathematically, our theory applies to the limiting case $N \to \infty$ of a sequence of networks of size N, with $\lfloor \rho^0 N \rfloor$ seed nodes. In Monte Carlo simulations of real banking networks, the size N of the system is fixed, and the case of a single seed corresponds to a fraction $\rho^0 = 1/N$ of initial defaults. The mechanism of cascade initiation in the infinite-N network may be understood as follows. As in [27], we call bank nodes *vulnerable* if they default due to a single defaulting loan. When the cascade condition (2.26) is satisfied, a giant connected cluster of vulnerable nodes exists in the network. The fractional size of this vulnerable cluster is denoted S_v, and it may be calculated by solving a site percolation problem for the directed network (see [20]) in a similar fashion to the calculation for undirected networks in [27]:

$$S_v = \sum_{jk} p_{jk} \left[1-(1-\phi)^j\right] \Theta\left[\frac{0.2}{j} - c_{jk}\right], \tag{2.43}$$

where ϕ is the non-zero solution of the equation

$$\phi = \sum_{jk} \frac{k}{z} p_{jk} \left[1-(1-\phi)^j\right] \Theta\left[\frac{0.2}{j} - c_{jk}\right]. \tag{2.44}$$

Here, as in [27], the Θ term plays the role of a degree-dependent site occupation probability: sites (nodes) are deemed occupied if they are vulnerable in the sense defined above, and this happens if the shock due to a single defaulting loan $(0.2/j)$ exceeds their net worth c_{jk}. In Fig. 2.7 we directly calculate the size of the largest

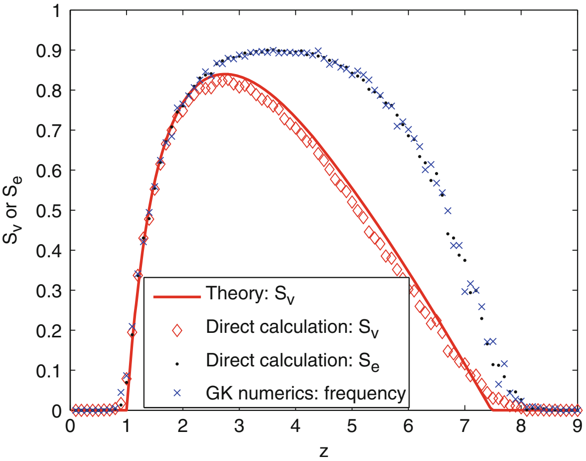


Fig. 2.7 Sizes of vulnerable cluster (S_v) and of extended vulnerable cluster (S_e) as calculated directly from (for each value of mean degree z) a single Erdös-Rényi network with $N = 10^4$ nodes. The vulnerable cluster size is compared with the analytical result of Eq. (2.43), while the extended vulnerable cluster is shown to closely match the frequency of global cascades in the single-seed GK model (cf. Fig. 2.4)

vulnerable cluster in a single realization of an Erdös-Rényi network with $N = 10^4$ nodes and mean degree z (cf. Fig. 2.4) and show that it closely matches to the analytical result (2.43).

The *extended vulnerable cluster* [27], which takes up a fraction S_e of the network, consists of nodes which are debtors of at least one bank in the vulnerable cluster. If a seed node is part of the extended vulnerable cluster, it immediately causes the default of its creditor in the vulnerable cluster, which in turn leads to default of other nodes in the vulnerable cluster, and so on until the entire vulnerable cluster is in default. Nodes outside the vulnerable cluster (i.e. banks which can withstand the default of a single asset loan) may also be defaulted later on in this cascade as the percentage of defaulted banks increases; the result is a global cascade of expected size ρ^∞, given by Eq. (2.15). On the other hand, if no seed node is part of the extended vulnerable cluster, then no further defaults will occur and the cascade immediately terminates. Thus, if only a single seed node is used in each realization, we expect cascades of size ρ^∞ to occur in a fraction S_e of realizations (corresponding to cases where the seed node lies in the extended vulnerable cluster), and no cascades to occur in the remaining fraction $1-S_e$ of realizations. The size S_e of the extended vulnerable cluster thus determines the frequency of global cascades among the set of single-seed realizations. The size of S_e was calculated analytically in [13] for the undirected networks case, but the corresponding derivation for directed networks is non-trivial. Instead, we directly calculate the size of the largest extended vulnerable cluster in the network, and show in Fig. 2.7 that it corresponds very closely to the frequency of global cascades in the large ensemble of Monte Carlo simulations of Fig. 2.4 in the main text.

As argued in [13], the frequency of cascades increases with the number $\lfloor \rho^0 N \rfloor$ of seed nodes used as

$$1-(1-S_e)^{\lfloor \rho^0 N \rfloor}, \tag{2.45}$$

which reduces to S_e for the single-seed case ($\rho^0 = 1/N$) and to 1 for the case where ρ^0 remains a finite fraction as $N \to \infty$. The frequency of cascades (of size ρ^∞) in the GK model initiated by a single default is thus S_e, whereas if multiple seeds (say, 10 initial defaults among 1,000 banks) are used we find that almost all cascades are of size ρ^∞.

References

1. Allen, F. and Gale, D. 2000 Financial contagion, *Journal of Political Economy*, **108**, 1–33.
2. Amini, H., Cont, R., and Minca, A. 2010 Resilience to contagion in financial networks, working paper.
3. Barrat, A., Vespignani, A., and Barthélemy, M. 2008 *Dynamical Processes on Complex Networks*, Cambridge University Press.
4. Bastos, E., Cont, R., and Moussa, A. 2010 The Brazilian banking system: network structure and systemic risk analysis, working paper.

5. Baxter, G. J., Dorogovtsev, S. N., Goltsev, A. V., and Mendes, J. F. F. 2010 Bootstrap percolation on complex networks, *Phys. Rev. E*, **82**, 011103.
6. Boss, M., Elsinger, H., Thurner, S. & Summer, M. 2004 Network topology of the interbank market. *Quantitative Finance* **4**, 677–684.
7. Dhar, D., Shukla, P., and Sethna, J. P. 1997 Zero-temperature hysteresis in the random-field Ising model on a Bethe lattice, *J. Phys. A: Math. Gen.* **30**, 5259–5267.
8. Eisenberg, L. and Noe, T. H. 2001 Systemic risk in financial systems,*Management Science,* **47**, (2) 236–249.
9. Elsinger, H., Lehar, A. & Summer, M. 2006 Using market information for banking system risk assessment, *International Journal of Central Banking*, **2**, (1) 137–165.
10. Erdös, P. and Rényi, A. 1959 On random graphs, *Publicationes Mathematicae*, **6**, 290–297.
11. Gai, P. and Kapadia, S. 2010 Contagion in financial networks, *Proc. R. Soc. A*, **466** (2120) 2401–2423.
12. Gleeson, J. P. 2008a Mean size of avalanches on directed random networks with arbitrary degree distributions, *Phys. Rev. E*, **77**, 057101.
13. Gleeson, J. P. 2008b Cascades on correlated and modular random networks, *Phys. Rev. E*, **77**, 046117.
14. Gleeson, J. P. and Cahalane, D. J. 2007 Seed size strongly affects cascades on random networks, *Phys. Rev. E*, **75**, 056103.
15. Haldane, A. G. 2009 Rethinking the financial network, online: http://www.bankofengland.co.uk/publications/speeches/2009/speech386.pdf.
16. Haldane, A. G. and May, R. M. 2011 Systemic risk in banking ecosystems, *Nature*, **469** (7330) 351–355.
17. Lorenz, J., Battiston, S., and Schweitzer, F. 2009 Systemic risk in a unifying framework for cascading processes on networks, *Eur. Phys. J. B*, **71** (4) 441–460.
18. May, R. M. and Arinaminpathy, N. 2010 Systemic risk: the dynamics of model banking systems, *J. R. Soc. Interface,* **7**, (46) 823–838.
19. May, R. M., Levin, S. A., and Sugihara, G. 2008 Ecology for bankers, *Nature*, **451**, 893–895.
20. Meyers, L. A., Newman, M. E. J., and Pourbohloul, B. 2006 Predicting epidemics on directed contact networks, *J. Theor. Biol.*, **240**, 400–418.
21. Moreno, Y., Paster-Satorras, R., Vázquez, A., and Vespignani, A. 2003 Critical load and congestion instabilities in scale-free networks, *Europhys. Lett.*, **62**, 292–298.
22. Motter, A. E. and Lai, Y.-C. 2002 Cascade-based attacks on complex networks, *Phys. Rev. E*, **66**, 065102(R).
23. Newman, M. E. J. 2003 The structure and function of complex networks, *SIAM Review*, **45** (2), 167–256.
24. Newman, M. E. J. 2010 *Networks: An Introduction*, Oxford University Press.
25. Nier, E., Yang, J., Yorulmazer, T., and Alentorn, A. 2007 Network models and financial stability, *J. Economic Dynamics & Control*, **31**, 2033–2060.
26. Upper, C. & Worms, A. 2004 Estimating bilateral exposures in the German interbank market: is there a danger of contagion? *European Economic Review* **48**, 827–849.
27. Watts, D. J. 2002 A simple model of global cascades on random networks, *Proc. Nat. Acad. Sci.*, **99**, (9) 5766–5771.
28. Wells, S. 2002 UK interbank exposures: systemic risk implications. *Bank of England Financial Stability Review* December, 175–182.

Chapter 3
Systemic Valuation of Banks: Interbank Equilibrium and Contagion

Grzegorz Hałaj

Abstract The aim of the chapter is to propose the new approach to valuation of individual banks which takes into account the risk of the whole interbank market network. We show that the value of the bank is equal to the value of the call option on the bank's debt which is the standard step in the valuation theory. However, the underlying value process depends on the possible interbank payments the bank expects to receive from other participants of the interbank market. In this way valuation theory originated to Black and Scholes (J Polit Econ 81:637–653, 1973) is embedded into the systemic framework *a la* (Cifuentes et al. (2004) Liquidity risk and contagion. London School of Economics) and we are able to prove that the value of a bank should not only depend on its internal financial standing but on the ability of their interbank counterparties to repay their debts. Our model has two unique features. Firstly, we demonstrate how losses originated to the interbank exposures can be reflected into the valuations of the banks even if it is extremely difficult to estimate the default probabilities of the interbank deposits. Secondly, liquidity of the bank and marketability of the bank's counterbalancing capacity is an outcome of the interbank market equilibrium. We apply the developed theory to study the relationship between the US banking system structure and the valuations of the US banks. We solely use publicly available data: the financial statements of the US banks provided by FDIC and the stock exchange quotes.[1]

[1] DISCLAIMER: The chapter presents views of the author which are not necessarily those of the ECB. Most of the results presented in the chapter were completed when the author was working for Bank Pekao SA, Warsaw (UniCredit Group).

G. Hałaj (✉)
ECB (Frankfurt am Main), Frankfurt, Germany
e-mail: grzegorz.halaj@n-s.pl

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_3,

3.1 Introduction

A classical approach to the bank valuation dates back to the research of [4] and [19]. The value of a bank can be obtained from the theoretical call option with the bank assets evolution as the underlying stochastic process and the strike equal to the volume of the bank's indebtedness. The detail can be found in the corporate finance books like [20]. This approach most important features are the linking of the valuation to the volatility of the assets and to the bank's ability to pay back their debts, and to the amount of bank's capital which is the risk absorption capacity. The asset prices are usually modeled as the diffusion processes and this allows for embedding the bank valuation into the well established framework of the no-arbitrage paradigm. Moreover, the parameters of the diffusions can accurately be estimated based on the time series of asset prices.

What is a comfortable assumption for practical implementations turns out to inadequately describe the dynamics of banks' balance sheets and their funding needs. We claim that banks' ability to pay back their obligations is not only the function of return and risk inferred from statistical analysis of trends and volatilities of their assets but on the financial standing of their counterparties and their capability of paying back placements among themselves. We address this issue in this chapter where we build a model of valuation taking into account the interbank market for funding and liquidity.

Apart from a very positive influence that the interbank market exerts on liquidity it can also contribute to the amplification of the insolvency problems in the banking system. Insolvency means the inability to pay back all the debts after liquidation of assets. Liquidity refers to the short-term assets and obligations. That is why solvency and liquidity are usually considered separately. However, if a bank does not hold cash or securities that can be immediately sold on the market it may not be able to give back money to other in which case it may be lead to insolvency. The propagation of the financial problems among banks was studied by Elsinger et al. [11,12], Degryse and Nguyen [8], Wells [23] and Cifuentes et al. [5].

Those interbank placements at risk should diminish the value of the bank. If there are premises that a bank is not going to receive demanded payments then this bank should immediately set aside additional capital or report extra loss. Usually these interbank payments are large and concentrated exposures. Their credit risk features are rather difficult to model and their impact on the financial results and even solvency is substantial. That is why Basel II committee [3] decided that in the case of those bulky exposures no internal rating-based approaches to model their credit risk are viable; they are outside the standard econometric framework for assessing credit risk of granular portfolios. Unlike in the case of the liquid securities or the retail loans (consumer or mortgage loans) with well reported history of prices and risk factors their risk and return cannot be estimated with the reasonable statistical confidence. We endogenize the risk of those placements studying the relationship between the balance sheet structures of banks.

Our approach captures the possible influence of the transferred credit risk on banks valuation. We define the aggregate balance sheet risk as the volatility of the net income of banks which we embed into the option-based valuation of banks. However, we are also interested in the following credit transfer mechanism. Banks sell the credit-linked securities originated to their loan portfolios among other intermediaries. The amount of the transferred risk is disclosed as the securitization activity. If the loan losses occur in the loan portfolio of a bank that issued MBSs (or ABSs in general) then the credit risk protection sellers may expect not to receive the full amount of proceeding from loans linked to the corresponding MBSs what may manifest itself in the devaluation of these securities. Thus, the quality of the counterbalancing capacity declines.

We propose to study the relationship between valuation and liquidity in the banking system starting from the funding structure of banks. There are two reasons for this approach. Firstly, funding sources and their availability determine liquidity positions of banks. Funding is the way banks raise financial resources (e.g. deposits, debt issuance, securitization) to invest on the market (e.g. granting loans, buying securities but also underwriting). While expanding their business, banks have to match their assets and liabilities with the available liquidity profiles. The resultant balance sheet structure may be thought of as a result of the optimal risk sharing and interbank allocation of liquidity described by Allen and Gale [1]. Some of the sources are long-term and stable, but other categories are short-term and volatile, e.g. placements of financial institutions. In the case of a liquidity shortage banks have to accept those short-term types of liabilities to continue doing business. Hence, the possibly widening liquidity mismatch exposes banks to liquidity risk on the interbank market [9]. Secondly, we can get rid of arbitrariness of the choice of interbank networks which vary in time (problem already noticed by van Lelyveld and Liedorp [21]).

Since the valuation of banks should be determined by the interbank funding structure and the transfer of risk from the loan portfolios the question arises how to find the appropriate interbank network and the channels of the credit risk transfer in the pricing exercise. Taking into account the fact that no distribution of the interbank connections can be estimated with a reasonable accuracy and that it is practically impossible to link a portfolio of ABSs to the particular securitied loans we propose to follow the idea of Epstein and Schneider [14]. They introduced the concept of ambiguity in decision making and pricing theory as well. Ambiguity refers to the economic circumstances in which it is impossible to assign the specific probability to the states of the world. To get rid of ambiguity the bank valuation is performed for the most unfavorable structure of the interbank connections. Epstein and Schneider [14] justifies this approach showing the so-called *Ellsberg Paradox*, i.e. a psychological experiment proving that if priors are ambiguous then economic agents make choices comparing the worst case scenarios.

Our model of interbank liquidity was inspired by market equilibrium approach of [5] which we modified (see also [15, 18]). We use their concept of the *clearing payments vector* to study liquidity in a network of banks but we add two components. Firstly, the marketability of the securities portfolio is concerned. However, as other theoretical results and market practice show we claim that a bank that cannot raise enough cash to meet its obligations may only sell part of its securities portfolio on the interbank market if there in not enough buyers willing to purchase it. That part that can be sold is an outcome of the interbank equilibrium. We assume in our model that banks can cover liquidity gap by selling securities only to those banks with enough liquidity surplus. It is a unique feature of our model and it seems to be very intuitive. Secondly, we model the credit risk transfer mechanism which may devalue the ABS portfolios if losses materialize in the securitized portfolios. In this way, we study the propagation of the loan portfolio losses in the banking system and its potential influence on the valuations.

We verify the assumptions looking into the US banking system. A large, publicly available database of the Federal Deposit Insurance Corporation (FDIC) provides detailed information about the balance sheet structures and the income statements of almost all banks in the US. Based on that information from just before the liquidity crisis (June 30, 2007) and from Dec 31, 2008 we calibrate the model and perform simulations of the equilibrium interbank payments, its defaulted fractions that adversely affect the valuations and the equilibrium liquidation of the securities. In order to study the credit risk transfer mechanism the market equilibrium and the valuations in equilibrium are calculated in two different settings: taking and not taking into account possible defaults in the securitized loans which may adversely affect the value of ABS securities. The outcomes of our model were compared with the market performance of banks. We find evidence that funding profile of the banks (i.e. composition of the securities portfolio and leverage) explains the observed deepness of devaluations during the financial crisis. It shows that only a few largest banks exposed to the contagion effect related to the interbank payments needs could not find the buyers for a part of the MBS securities portfolio to fully meet their obligations. However, the credit risk transfer that does not substantially aggravate the valuations of the US banks and does not contribute to the amplification of the contagion effect in the US banking system.

Summing up, the model helps in answering to the question about how a large interbank exposures can be incorporated into the bank valuation model without the history-based estimation of their risk.

The chapter is structured as follows.[2] The valuation model is introduced in Sect. 3.2. We recall the option-based valuation (Sect. 3.2.1), build the model of interbank payments related to the direct interbank placements (Sect. 3.2.2) and we embed the credit transfer mechanism in it (Sect. 3.2.3). We define the equilibrium vector of payments (*clearing payments vector*) in Sect. 3.2.4 and the secondary (contagious) liquidity default equivalent to the domino effect of illiquidity in Sect. 3.2.5.

[2] We will recall the following operators "$\wedge$", "$\vee$", "$^+$" and "$^-$" useful in the notation of equilibrium, i.e. $x \wedge y :\ = \min\{x,y\}$ and $x \vee y :\ = \max\{x,y\}$, $x^+ :\ = \max\{x,0\}$ and $x^- :\ = -\min\{x,0\}$. By $\prod_{j \in J} A_j$ we denote the Cartesian product of sets A_j indexed by the set J. By $\bar{\mathbb{N}} :\ = \{1,\ldots,N\}$ we denote the set of cardinal numbers of banks in the banking system. Symbol "$^\top$" denotes transposition. By $\mathcal{P}$ we denote $\prod_{i \in \bar{\mathbb{N}}} [0, \bar{p}_i]$.

In Sect. 3.3 we prove that a proper equilibrium exists. Section 3.3.1 is devoted to the four-bank toy model the aim of which is to build intuition about the interbank equilibrium and contagion. Finally in Sect. 3.4 the standard option pricing based valuation formula for bank is modified to take into account the equilibrium interbank payments. Ultimately, in Sect. 3.5, we perform simulations of the US banks valuations and their sensitivity to the funding structure and the risk of contagion. Conclusions are gathered in Sect. 3.6.

3.2 The Model

In the proposed modelling framework we show that there is a link between interbank liquidity and valuation of banks. Liquidity of a bank depends on the relationship between liquid assets and liabilities that the bank has to pay back to its creditors, i.e. it depends on the financial situation of the bank. The liquidity also hinges on two systemic factors – firstly, apart from the ability of the debtors to repay their obligations and the propensity of depositors to roll over their deposits, it depends on the accessibility of liquid funds on the market e.g. those that can be borrowed on the interbank market or obtained by selling the marketable securities. Hence, it is reasonable to assume that the value of the bank should be related to the interbank conditions as well.

3.2.1 Valuation Fundamentals

Let us assume that the risk on the market is described by a probability space $(\Omega, \mathcal{F}, \mathbf{P})$. The time dimension in the model is represented by two periods – 0 and 1. The information on the market at time $t = 0$ is trivial, i.e. $\mathcal{F}_0 \equiv \{\emptyset, \Omega\}$ and in time $t = 1$ is denoted $\mathcal{F}_1$. Its composition is related to the revelation of uncertain risk factors that will be introduced later. Thus, the evolution of information on the market is described by filtration $\mathbb{F}\colon = (\mathcal{F}_0, \mathcal{F}_1)$. Additionally, probability measure $\mathbf{P}$ is supposed to be the risk neutral measure. It means, that each discounted price or valuation process is a martingale with respect to $(\mathbb{F}, \mathbf{P})$. This is the standard no-arbitrage valuation setting.

According to the classical result of [4] the value of a bank can be thought of as the call option on the bank assets with the strike equal to the debt volume. Let us assume that a given bank (indexed with symbol i) grants L_i^0 loans to the non-banking sectors (retail customers, production companies, insurance companies, pension funds, etc.) at time 0 and LB_i loans to the credit institutions (predominantly banks but also other firms granting loans like trusts and savings associations). The value of loans, i.e. the expected payments that bank can receive back from the borrowers and the volatility of those payments is influenced by a risk factor ε_i, which is a given $\mathcal{F}_1$-measurable random variable in $(\Omega, \mathcal{F}, \mathbf{P})$. Thus, loans at time 1 are denoted $\mathrm{L}_i(\varepsilon_i)$.

For simplicity, we do not consider any common (systemic) risk factors (like in [11, 22]) but this still allows to study contagious devaluations of the banks.[3]

The remaining volume of the bank's assets comprises of the securities $\mathrm{S}_i(\varepsilon_{-i})$. We consider two types of securities: the very liquid Treasury securities denoted $\bar{\mathrm{S}}_i$ and other securities $\underline{\mathrm{S}}_i(\varepsilon_{-i})$ which have limited liquidity. These less liquid or even illiquid securities mostly consist of Mortgage Backed Securities (MBS) guaranteed by the state[4] and other asset backed securities, not guaranteed MBSs in practice. Latter in Sect. 3.2.3, $\underline{\mathrm{S}}(\varepsilon_{-i})$ value is linked to the credit risk accumulated in the other banks (i.e. the banks $\{j \in \bar{\mathbb{N}} | j \neq i\}$) that securitized part of their loans and transformed them into the credit-linked securities. By (ε_{-i}) we denote the dependence of the liquidity of the securities on the other banks financial standing.[5] Thus, $\mathrm{S}_i(\varepsilon_{-i}) := \bar{\mathrm{S}}_i + \underline{\mathrm{S}}(\varepsilon_{-i})$.

Banks activity is funded by equity (capital) E and deposits from the non-banking firms and the retail sector (D_i) and from the credit institutions (DB_i). Let us suppose that there are N banks in the system indexed by $\{1, 2, \ldots, N\}$ with assets and liabilities satisfying identity $\mathrm{L}_i(\varepsilon_i) + \mathrm{LB}_i + \mathrm{S}_i(\varepsilon_{-i}) = \mathrm{E}_i + \mathrm{D}_i + \mathrm{DB}_i$. In this setting the value of bank i is given by the formula:

$$V_i^0 = \mathbf{E}\left[\mathrm{LB}_i + \mathrm{S}_i(\varepsilon_{-i}) + \mathrm{L}_i(\varepsilon) - (\mathrm{DB}_i + D_i)\right]^+ \tag{3.1}$$

The exposures LB_i and DB_i represent the aggregate interbank market. Obviously, if the banks lend and borrow on the interbank market, they may default on the interbank obligations. The goal of Sects. 3.2.2–3.2.4 is to present the modeling framework which helps to find the fractions of the payments LB_i and DB_i that may not be returned to the creditors.

3.2.2 *Interbank Liquidity and Funding*

The liquidity model combines the two most essential components influencing liquidity condition of banks [6]: the relationship between obligations to be met and assets providing liquidity but carrying risk of devaluation, and between demand and supply of the securities on the interbank market.

The question is who can buy those liquid securities? We assumed that bank can sell securities remitting liquidity only to other banks with liquidity surplus. In

[3] The risk factors are univariate random variables instead of being multivariate, correlated factors. This assumptions can be easily relaxed in this setting to account for the dependance structure of the banking income.

[4] The guarantees should be thought of as MBS issuance of Government National Mortgage Association (Ginnie Mae), Federal Home Loan Mortgage Corporation (Freddie Mac) and Federal National Mortgage Association (Fannie Mae).

[5] Here, we apply the standard game-theoretical notation for the "other players" than i putting $-i$.

practice, banks look for liquidity in other banks. The central banks are requested to bring liquidity support only in case of the intraday settlements, market disruptions or adverse signals about commercial banks' liquidity position. Usually, this support has very short tenors. Sometimes it even carries reputation stigma. In our approach, we focus on the market sources of liquidity.

We postulate that banks supplement liquidity only by selling securities to other banks or by lending from other banks if they do not have enough liquid securities but some of the banks have excess liquidity.

We define a couple of helpful notions. The bilateral exposures of the banks on the interbank market are described by a matrix P with N columns and N rows. The matrix P is called the *matrix of exposures*. Element P_{ij} denotes the amount of placements of bank j in bank i. In other words, P_{ij} stands for obligation of bank i against bank j. By means of the so-called *matrix of relative expositions* π we described ratio of placements in bank i received from j to the total placements received by i on the interbank market (i.e. $\pi_{ij} = P_{ij}/\sum_{j=1}^{N} P_{ij}$). The sum of the interbank obligations of i is denoted $\bar{p}_i$ (ie. $\bar{p}_i :\ = \sum_{j=1}^{N} P_{ij}$).[6]

We assume that all the interbank placements have equal maturities. It simplifies the calculations but may distort the results. However, there are reasons to treat the interbank placement as having the same maturities. If an interbank deposit, which is critical for creditor's liquidity, becomes due other placements that are important for other parties may be perceived as becoming due as well. The expected chain reaction that may occur in the banking system justifies the supposition. Namely, if the debtor is expected to return a placement to the creditor and has liquidity shortage then it could negotiate a prolongation of this placement until it receives deposits with longer maturities from other banks. This can trigger a whole "wave of negotiations" spreading across the system and encompassing other banks with tight liquidity conditions.

In particular circumstances banks may not have enough resources to satisfy its creditors with the whole amount of interbank deposits $\bar{p}$. The following sections introduce the framework to study the realized interbank payments.

3.2.3 Transfer of Credit Risk

Transfer of credit risk from banks' loan portfolios to other banks' securities portfolios is another important source of contagion. Banks sell part of their loan portfolios usually in the form of the Asset Backed Securities which can be purchased by many market participants spreading the risk across the financial – and in particular banking – system. In this way worsening quality of loan portfolios in banks-originators affects the profit and loss distributions of bank-buyers of the credit risk.

[6] Symbol $:\ =$ means "by definition".

Let us suppose that each bank i securitized loans with aggregate volume amounting to L_i^S. It means that the Special Purpose Vehicles (SPV) collecting loans for securitization issue $\sum_{i\in\bar{\mathbb{N}}}\mathrm{L}_i^S$ of ABS securities. For simplicity, we assume that all the securities are bought by banks. Proceedings from these securitized loans will then be transferred from the originators to the buyers of ABSs, de facto mainly MBSs. So will the credit risk shocks from the securitized loan portfolios.

The main idea to securitize part of loan portfolio is to "slice" the risk related to the loans in this portfolio in order to obtain sources for further credit expansion or to improve the solvency ratios. The sliced risk is then spread across the financial system depending on the demand for ABS securities. So the risk related to the particular securitized loan can be distributed to many banks. The channels of the distribution can be represented by the matrix σ, $\dim\sigma = N\times N$. The entry σ_{ij} indicates the fraction of the loan portfolio securitized by bank i the risk of which is bought by bank i in its ABS portfolio.

Given that the matrix σ is known the propagation of a default shock in the securitized loan portfolio to other banks in the system can be traced. Namely, Δ fraction of the portfolio default in bank i (with loss given default equal to 100% for simplicity) causes the devaluation of

$$\begin{bmatrix}\Delta\mathrm{L}_i^S\sigma_{i1}\\ \Delta\mathrm{L}_i^S\sigma_{i2}\\ \vdots\\ \Delta\mathrm{L}_i^S\sigma_{iN}\end{bmatrix}_{N\times 1} = \left(\begin{bmatrix}0 \; 0 \ldots \Delta\mathrm{L}_i^S \ldots 0\end{bmatrix}_{1\times N}\begin{bmatrix}\sigma_{11} & \ldots & \sigma_{1N}\\ \vdots & \ddots & \vdots\\ \sigma_{N1} & \ldots & \sigma_{NN}\end{bmatrix}_{N\times N}\right)^\top$$

in all the banks.

In general, the credit risk transfer channels are extremely difficult to follow. It is practically impossible to identify the loan portfolios from which the credit risk originates by just observing the balance sheet statements of banks. To capture the potential influence that the credit risk transfer can exert on valuation of banks we apply the idea of the worst case scenario of σ, as in the case of the interbank structure π. We leave the details to the Sect. 3.4.

Having constructed the interbank market of the direct exposures and credit risk transfer we proceed to the definition of the interbank equilibrium.

3.2.4 The Equilibrium

The main question we pose related to the interbank payments p is the following: are all the banks capable of paying back their interbank obligations $\bar{p}$? We give the answer through the equilibrium clearing payments [10].

Let us assume that banks declare to pay $p:=[p_1,\ldots,p_N]^\top\in\mathcal{P}$. We define the *conditional inflow from payments* as a vector $\mathrm{IP}(p)\equiv[\mathrm{IP}_1(p),\ldots,\mathrm{IP}_N(p)]$ with components $\mathrm{IP}_i(p):=\mathrm{L}_i(\varepsilon_i)+\mathrm{S}_i^T-\mathrm{D}_i+\sum_{j=1}^N\pi_{ji}p_j$ being the function of the interbank payments p. That formula takes into account possible rolling-over of deposits, the fact that some regular assets (especially loans) and interests from these assets may migrate to irregular ones and that the credit losses on the loan portfolios of banks may be transferred to other banks through the ABS exposures. It includes the treasury securities which can immediately be liquidated on the market or *repo*ed in the central banks since they are the most eligible collateral. Consequently, the *conditional funding gap* – conditional on the flow of p – equals to $\mathrm{Gap}(p) :\; = \mathrm{IP}(p) - p$ and *conditional gap to be covered by securities selling* is $\overline{\mathrm{Gap}}(p) :\; = \mathrm{IP}(p) - \bar{p}$. The later formula measures the balance of funding sources and expenses on investments given that banks receive only p of repaid placements which may not necessarily be equal to $\bar{p}$.

We assume that banks can only cover their negative liquidity gap if they have liquid securities to sell and there are banks with free cash resources to buy these securities. Banks are selling the non-treasury securities with a haircut $h \in [0,1]$. We define two auxiliary functions of p:

- *Aggregate disposable and necessary securities* $\mathrm{DS}^{(a)} : \mathcal{P} \to \mathbb{R}_+ \cap \{0\}$ of banks with negative gap i.e. $\mathrm{DS}^{(a)}(p) :\; = \sum_{i \in \bar{\mathbb{N}}} (\overline{\mathrm{Gap}}_i(p))^- \wedge (1-h)\underline{S}_i(\varepsilon_{-i})$. $\mathrm{DS}^{(a)}$ is the sum of securities in all banks which can be used to cover the negative funding gap;
- *Aggregate free cash* $\mathrm{FC}^{(a)} : \mathcal{P} \to \mathbb{R}_+ \cap \{0\}$ of bank with positive gaps i.e. $\mathrm{FC}^{(a)}(p) :\; = \sum_{i \in \bar{\mathbb{N}}} (\overline{\mathrm{Gap}}_i(p))^+$. $\mathrm{FC}^{(a)}$ is the sum of liquidity surpluses in all banks given that banks pay back p.

The functions can be thought of as a measures of the interbank supply and demand for securities, which forces drive the value of these securities.

Ultimately, we can define the function describing payments received by banks from the real sector and from the other banks. This is the mapping $\Psi : \mathcal{P} \to \mathcal{P}$ defined for the component i of the vector $\Psi(p) :\; = [\Psi_1(p), \ldots, \Psi_N(p)]^\top$ as

$$\Psi_i(p) = \begin{cases} \left[\left[\mathrm{IP}_i(p) + [\overline{\mathrm{Gap}}_i(p)^- \wedge (1-h)\underline{S}_i(\varepsilon_{-i})\right] g_1\right] \vee 0\right] \wedge \bar{p}_i, \\ \qquad \overline{\mathrm{Gap}}_i(p) < 0 \\ \left[\left[\mathrm{IP}_i(p) - \overline{\mathrm{Gap}}_i(p)^+ g_2\right] \vee 0\right] \wedge \bar{p}_i, \\ \qquad \overline{\mathrm{Gap}}_i(p) \geq 0 \end{cases} \tag{3.2}$$

where $g_1(p) :\; = g\left(\mathrm{FC}^{(a)}(p), \mathrm{DS}^{(a)}(p)\right)$ and $g_2(p) :\; = g\left(\mathrm{DS}^{(a)}(p), \mathrm{FC}^{(a)}(p)\right)$ for $g : \mathbb{R}_+ \cap \{0\} \times \mathbb{R}_+ \cap \{0\} \to [0,1]$ defined as

$$g(x,y) = \begin{cases} 1 \wedge \frac{x}{y}, \; y \neq 0 \\ 0, \quad y = 0 \end{cases}.$$

We want to find $p^* \in \mathcal{P}$ s.t. $\Psi(p^*) = p^*$. The vector p^* *clears* the interbank market reflecting the propagation of liquidity in the banking network. Why is this an interesting equilibrium? Let us suppose that banks declare to pay p. $\Psi(p)$ indicates how much funds banks receive if all banks would pay p (perhaps different from the required $\bar{p}$). If $p_i > \Psi_i(p)$ then bank i cannot meet all of its interbank obligations. If on the other hand $\Psi_i(p) > p_i$ then bank i declared less than it is able to pay and it should pay $\bar{p}_i$.

Why do we use $\overline{\text{Gap}}(p)$ instead of $\text{Gap}(p)$ in the definition of Ψ? We assume that each bank i wants timely to give back all interbank loans; prudent banks compare declared (expected) incoming payments $\text{IP}(p)$ with its interbank deposits $\bar{p}_i$. If the gap is negative then that bank sells as much securities as it needs to cover the liquidity gap.

Summing up, the pair (p^*, S^*), $\text{S}^* := g_1 \cdot \underline{\text{S}}$, of the interbank payments and the effective counterbalancing capacity define the equilibrium on the market for a given realization of the shocks ε. The deficiency of liquidity $\bar{p}_i - p_i^*$ translates into losses of banks $\{j|\pi_{ij} > 0\}$.

Before showing that p^* exists we demonstrate how to use p^* and the function Ψ to define the liquidity contagion.

3.2.5 Secondary Defaults: Domino Effect

A bank can have liquidity problems stemming from other banks' inability to pay back their liabilities. Transmission of liquidity shortages between banks is a central issue in the studies of how banks fulfil their obligations and – as we described it in the introduction of this chapter – the resultant valuation of banks. The equilibrium payments vector p^* helps to address this issue.

A liquidity shortfall resulting only from liquidity problem of other banks is called a secondary liquidity default (shortfall), domino effect of illiquidity or just *contagion effect*. Using the notation from our model we can formally define the secondary default of bank i as the event when:

$$\Psi_i(\bar{p}) = \bar{p}_i,$$

$$p_i^* < \bar{p}_i.$$

The first line of the formula indicates that bank i would completely pay back its liabilities if other banks returned all their interbank placements. The second one expresses the condition that in the clearing payments equilibrium bank i fails to meet all the obligations since some other banks defaulted on their liabilities against bank i. In Sect. 3.4 we show how to measure the impact of these secondary defaults on the valuations of banks.

3.3 Existence of Equilibrium and Numerical Procedure

Definitions from the previous section may not be intuitive as far as the existence of the equilibrium p^* is concerned. Function (3.2), central to define equilibrium, is nonlinear and quite complex. We do not know the closed form formula for fixed points of it. It may not even be obvious that Ψ has any fixed point. However, the famous Knaster-Tarski theorem guarantees that the maximum equilibrium payments vector exists.

Elsinger et al. [12] and Cifuentes et al. [5] and others had an advantage to work in a framework of the so-called contracting transforms. In their case, the Banach theorem guarantees a unique fixed point. Unfortunately, Ψ is not a contraction. But it is continuous and since $\mathcal{P}$ is compact, the set of the fixed points of Ψ is not empty. It means that there may be many equilibria.

However, Ψ is the *isotone mapping*, i.e. mapping $\Psi\colon \mathcal{P} \to \mathcal{P}$ satisfying for each pair $(p_1, p_2) \in \mathcal{P}^2$, such that $p_1 \succ p_2$ the condition $\Psi(p_1) \succ \Psi(p_2)$. The Tarski-Knaster theorem allows for finding the *maximal* (in the order sense) fixed point. The maximum means that there is a fixed point $p^*_{\max}$ such that any other fixed point p^* satisfies $p^* \preceq p^*_{\max}$. This maximal fixed point can be understood as the one that maximizes the payments related to the interbank obligations. We focus on this particular equilibrium since it is very desirable that all creditors receive the maximal possible payments. Let us sum up this result in the theorem.

Theorem 3.1. *There exists the maximal clearing payments vector p^*.*

Proof. The maximal fixed point of the mapping Ψ is guaranteed by Knaster-Tarski fixed point theorem since Ψ is the isotone (order-preserving) mapping on the bounded, complete lattice (see [16]). Since the proof of the isotone property of Ψ is just technical we postpone it to the Sect. 3.7. □

The fixed points of Ψ can only be found in a numerical, approximate procedure. We implemented the iterative procedure proposed in the constructive proof of the Knaster-Tarski Theorem by Cousot and Cousot [7] and relying on the fact that for a given $p \in \mathcal{P}$ satisfying $\Psi(p) \succ p$, $\lim_n \Psi^n(p) \in \mathrm{Fix}(\Psi)$.

3.3.1 How Does the Equilibrium Work? An Example

We illustrate the interbank equilibrium in a very simplistic example. Let us assume that there are four banks in the system. Each bank's total assets are equal to 100 units and equity amounts to 20 units.[7] Banks' exposures on the interbank market – loans LB and deposits DB, and the portfolio of securities with the limited liquidity consisting of the ABSs ($\underline{\mathrm{S}}(\varepsilon_{-i})$) – are given by the following vectors

[7] For example USD bn. We skip "units" for brevity.

$$\mathrm{LB} = \begin{bmatrix}10\ 10\ 20\ 10\end{bmatrix}^{\top},$$

$$\mathrm{DB} = \begin{bmatrix}10\ 20\ 10\ 10\end{bmatrix}^{\top},$$

$$\underline{\mathrm{S}}(\varepsilon_{-i}) = \begin{bmatrix}20\ 20\ 10\ 20\end{bmatrix}^{\top}.$$

Consequently, the sum of the interbank obligations of the four banks is equal to $\bar{p} = \begin{bmatrix}10\ 20\ 10\ 10\end{bmatrix}^{\top}$.

As already noticed in Sect. 3.2.2, the balance sheet structures of the banks imply infinitely many matrices of the relative exposures π. The following two matrices of the interbank structure are possible:

$$\pi^1 = \begin{bmatrix}0.00 & 0.50 & 0.50 & 0.00\\ 0.35 & 0.00 & 0.65 & 0.00\\ 0.00 & 0.00 & 0.00 & 1.00\\ 0.30 & 0.50 & 0.20 & 0.00\end{bmatrix}, \pi^2 = \begin{bmatrix}0.00 & 0.10 & 0.90 & 0.00\\ 0.45 & 0.00 & 0.05 & 0.50\\ 0.15 & 0.85 & 0.00 & 0.00\\ 0.00 & 0.05 & 0.95 & 0.00\end{bmatrix}.$$

Firstly, let us assume that the market is described by the matrix π^1. Let us also assume that loans satisfy the additive parametrization $\mathrm{L}(\varepsilon) = \mathrm{L}^{const} + \varepsilon$ and the following profits and losses $\hat{\varepsilon}$ are realized by the banks: $\hat{\varepsilon} = \begin{bmatrix}-5\ -25\ 10\ 5\end{bmatrix}^{\top}$. After that shock the funding gap changes to $\begin{bmatrix}-5\ -15\ 10\ 5\end{bmatrix}^{\top}$.

If all the banks receive exactly what other banks should pay back (i.e. $\bar{p}$) then the second bank defaults. It is indicated by value of the function Ψ at point $\bar{p}$, i.e. $\Psi(\bar{p}) = [10\ \ 15\ \ 10\ \ 10]^{\top}$. The second component of $\Psi(\bar{p})$ is less than the second component of $\bar{p}$. To check whether there are contagion effects of the second bank's default we calculate vector p^*. We assume haircut $h = 0.0$ for simplicity. Iterating randomly selected starting points $\tilde{p}$ by the mapping Ψ we obtain $p^* = [0.77\ 0.00\ 10.00\ 10.00]^{\top}$. The size of contagion equals $10.00 - 0.77 = 9.23$ units for the first bank. It happens that the first bank is not capable of meeting all its interbank obligations. In spite of having a high positive gap it strongly relies on the flow of payments from the other banks – it receives 3 units instead of 10. That -7 unit deficiency plus -5 funding gap gives -12 units of the gap that could, however be covered by the liquidation of ABS portfolio. Unfortunately, the supply for that securities surpasses the demand and only $g_1 = 26\%$ of the securities needed to restore the liquidity is marketable meaning that the first bank can sell only 2.77 units. The market conditions allow for liquidating such the amount of securities that the overall funding gap equals to $-12 + 2.77 = -9.23$. It implies the interbank payments $p^*(1) = 10 - 9.23 = 0.77$ instead of required $\bar{p}(1) = 10$.

However, the interbank market can also be described by the matrix π^2. In that case $\Psi(\bar{p})$ remains unchanged if the same profits and losses $\hat{\varepsilon}$ are realized. What changes is the propagation of the contagion effect related to the exposure $\pi^2_{24} = 0.50$ of bank 4 against bank 2. The defaulting bank 2 does not pay back its debts to the bank 4 and that bank, in turn, can only pay back 5.74 units (instead of 10).

The credit risk transfer in this example of the banking system is illustrated assuming that:

- The interbank market is defined by the matrix π^1 of the relative exposures,
- The default of 10 units takes place in the securitized loans portfolio of bank 3,
- Concentrating on the two possible matrices of the credit risk transfer from loans to ABS portfolios i.e.

$$\sigma^1 = \begin{bmatrix} 0.00 & 1.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 1.00 \\ 1.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 1.00 & 0.00 \end{bmatrix}, \sigma^2 = \begin{bmatrix} 0.00 & 1.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 1.00 \\ 0.20 & 0.80 & 0.00 & 0.00 \\ 0.00 & 0.00 & 1.00 & 0.00 \end{bmatrix}.$$

The credit transfer mechanism may have surprising consequences for the scale of liquidity contagion. If the transfer is described by σ^1 then the shock in bank 3 only effects the ABS portfolio in bank 1 and diminishes the counterbalancing capacity from 20 to 10 units. Therefore, bank 1 can only return 0.40 units of the interbank placements (instead of 0.77 units in the case of no credit risk transfer). However, if the channels along which the shocks from loan portfolios propagate to ABS portfolios of other banks are defined by σ^2 than the 10 unit shock effects mainly the ABSs in bank 2 ($\sigma^2_{32} = 0.80$). This has the opposite impact on the first bank capability to pay back their debts. The shock diminishes the overall supply of ABS securities on the market and bank 1 can sell more securities since $g = 33\%$ instead of 24% in the case of σ^1. That is why $p^*(1)$ increases from 0.40 to 2.00 units. The contagion effect transmitted by π^2 does not disappear but is evidently smaller.

3.4 Back to Bank Valuation Formula

The interbank payments in equilibrium p^* allow for revision of the classical bank valuation formula (3.1). Firstly, deposits from the financial institutions of bank i can be decomposed into two categories: placements of the other banks ($\bar{p}_i$) and deposits of other financial institutions. However, for a buyer of the stocks of bank i it is important to know what is the ability of i to pay bank its whole debts. Thus, the liability part of the formula remains unchanged. Secondly, loans to the financial institutions can be similarly decomposed but in their case the possible equilibrium payments indicate the true inflows of cash that can be used by i to meet their obligations. The available interbank sources of liquidity in bank i are equal to $\sum_j p^*_i \pi_{ji}$.

The choice of an interbank structure that matches a given set of banks with a certain composition of balance sheets is not unique. It is illustrated in the four-bank example in the previous Sect. 3.3.1. As such, a variety of possible interbank funding connections leads to ambiguity in banks' valuation. Let Pi denote the set of all possible matrices of exposures P for given balance sheet structures of N banks.

Equivalently, it can be pairs of matrices π describing the relative exposures and vectors p indicating the interbank obligations of all banks.

The same line of argumentation applies to the credit risk transfer mechanism. It is practically impossible to specify the true matrix σ describing the distribution of credit risk originated to the loan portfolios and transferred to the ABS portfolios of other banks, therefore it is worth considering a matrix that gives the lowest valuation. The selection depends on the balance sheet structure of a bank that is subject to the valuation. Let us denote by Sigma the set of all possible matrices σ.

Consequently, following [14] the value V_i of a bank i is given by

$$V_i = \min_{\substack{\pi\in\mathrm{Pi}\\ \sigma\in\mathrm{Sigma}}} \mathbf{E}\left[\sum_{j=1}^{N} p_j^*\pi_{ji} + \bar{\mathrm{S}}_i + \underline{\mathrm{S}}_i^{(*)}(\varepsilon_{-i}) + \mathrm{L}_i(\varepsilon_i) - (\mathrm{DB}_i + \mathrm{D}_i)\right]^+ \tag{3.3}$$

The value of a bank that has the net interbank exposure $\sum_{j=1}^{N}\bar{p}_i\pi_{ji} - \bar{p}_i$ but operating on the interbank market which has ambiguous structure of placements is equal to the value of the option on its assets given that structure is the least favorable, i.e. can lead in bank i to the highest losses incurred on the interbank exposures.

In order to assess the effect of the interbank liquidity contagion on valuations we suggest to compare V_i with the valuation calculated under the assumption that:

- Each bank j receives $\Psi_j(\bar{p})$ of the interbank placements, i.e. assuming that all the other banks pay back their interbank debts,
- Banks can sell $g_1(\Psi(\bar{p}))\underline{\mathrm{S}}$ securities.

It means that the valuation V_i is compared with V_i^{Ψ}, where

$$V_i^{\Psi} = \min_{\substack{\pi\in\mathrm{Pi}\\ \sigma\in\mathrm{Sigma}}} \mathbf{E}\left[\sum_{j=1}^{N} \Psi_j(\bar{p})\pi_{ji} + \bar{\mathrm{S}}_i + g_1(\Psi(\bar{p}))\underline{\mathrm{S}}_i(\varepsilon_{-i}) + \mathrm{L}_i(\varepsilon_i) - (\mathrm{DB}_i + \mathrm{D}_i)\right]^+ \tag{3.4}$$

The difference $V_i^{\Psi} - V_i$ is non-negative. Intuitively, since $\Psi_j(\bar{p}) \geq p^*$ for all banks j, the bank i would receive higher or equal inflow of payments if banks paid $\Psi(\bar{p})$ instead of p^*. The positive difference for bank i indicates that the interbank contagion has an impact on the valuations.

3.5 Valuation of US Banks

The empirical analysis shows that liquidity and credit risk accumulated in the banks' portfolios of securities and their distribution across US banking sector may have important impact on the valuations of banks. The proposed valuation model helps to study the channels and scale of this impact.

3.5.1 Data

The Federal Deposit Insurance Corporation provides a vast database about the banking sector in the United States. It collects the detailed financial statements of all FDIC insured credit institutions in US covering almost 100% of assets. The reports on the balance sheet structures, the income statements and auxiliary tables on the off-balance sheet items, the past due receivables, securitization servicing and regulatory capital are downloadable from FDIC webpage[8] on a quarterly basis.

We limit the set of banks taken for the analysis to a group of 200 most significant institutions in terms of the total assets. These 200 largest banks account for 8.8 bn USD out of 10.4 bn USD total assets of US banks in June 2007 (85%) and are most active on the interbank market. The set is sufficiently wide to prove the relationship between liquidity and valuation and possible size of the impact that contagion may exert on the valuations.

We feed the valuation model with the FDIC data selecting just a few items from the broad collection of banking data. The structure of assets is measured by total loans, liquid treasury securities, less liquid guaranteed mortgage backed securities (MBS) and the least liquid non-guaranteed asset backed securities (MBS and other ABS). The liability side is split into total liabilities including deposits and equity. The detailed parametrization is presented in Table 3.1.

Even though we extract the fair values of the MBSs from the FDIC reports we apply an additional haircut for them. If the MBSs are to accept as a collateral in the short term FED operation they are charged with a standard liquidity cost established by the central bank[9] depending on the class of securities they belong to. It means that 100 units of the fair value of MBSs may be exchanged into $(1-h)\cdot 100$ of cash. For simplicity, we prudentially apply 10% haircut irrespective of the type of the loans underlying the MBS.

We assume that the risk bearing by banks is fully reflected into their income variability [17]. We collect a time series of quarterly net income of the selected banks between March 2006 and June 2009 (14 points of time) and calculate its mean ($I^m(i)$) and the standard deviation ($I^{sd}(i)$) for each bank i. Since income exhibits substantial left skewness we assumed that the income process of the banks follows random variable which is a combination of Gaussian and Cauchy distributions constructed in the following arbitrary way. Let $B(i)$ be a random variable taking values 0 and 1 with equal probability 1/2. Income $I(i)$ of bank i is equal to

$$I(i) = \begin{cases} |Gauss(I^m(i), I^{sd}(i))|, & B(i) = 0 \\ -\min(|Cauchy(I^m(i), I^{sd}(i))|, E_i), & B(i) = 1 \end{cases}$$

where $Gauss(m,\sigma)$ is the normal distribution with mean m and standard deviation σ and $Cauchy(m,s)$ is Cauchy distribution with location parameter m and s specifying

[8] http://cdr.ffiec.gov/public/PWS/DownloadBulkData.aspx

[9] http://www.ny.frb.org/markets/talf_faq.html#10

Table 3.1 Parametrization of the model based on the FDIC reports

Model category	Components description	FDIC table	Fields
TA	TOTAL ASSETS	RC	RCON2170
S^T	US TREAS HTM	RCB	RCON0213; RCFD0213
	US TREAS AFS	RCB	RCON1287; RCFD1287
	US BONDS	RCB	RCFD1290; RCFD1293; RCFD1295; RCFD1298; RCON1290; RCON1293; RCON1295;RCON1298
	US STATE SEC	RCB	RCFD8497; RCFD8499
LB	BANK LOANS	RC-CI	RCONB531; RCFDB532; RCFDB533; RCFDB534; RCONB532; RCONB533; RCONB534
LT	TOTAL LOANS	RC	RCFDB528; RCONB528
S^{ABS}	MBS GRNTED	RCB	RCON1699; RCON1702; RCON1705; RCON1707; RCON1715; RCON1717; RCON1719; RCON1732
	MBS NOT GRNTED	RCB	RCFD1699; RCFD1702; RCFD1705; RCFD1707; RCFD1715; RCFD1717; RCFD1719; RCFD1732
	ABS	RCB	RCONC027; RCONC988; RCFDC027; RCFDC988
TL	TLIAB	RC	RCON2948
LB	BDEPO	RCEI	RCONB551; RCONB552; RCFNB554
L^S	SCRTIZE	RCS	RCFDB705; RCFDB706; RCFDB707; RCFDB708; RCFDB709; RCFDB710; RCFDB711; RCONB705; RCONB706; RCONB707; RCONB708; RCONB709; RCONB710; RCONB711
DT	TDEPOS	RC	RCON2200

Source: Selection based on FDIC data

half of the interquartile range.[10] The min of the realization of Cauchy random variable and equity refers to the fact that losses cannot exceed the amount of equity.[11] In the calibration process we take m equal to the mean of the time series of the reported income and σ and s equal to the standard deviation of that income.

A relationship between theoretical and market prices of banks would verify the assumptions of the model. We test the model by collecting market prices of 30 largest banks in our sample.

[10] The assumption about independent income processes for banks is very much simplified. In this way, rather lower bound of the interbank contagious losses can be captured – positive correlation of the income processes would probably amplify the losses. Generalization of the model setting to the case of income correlation is straightforward.

[11] In this way all moments of $I(i)$ are finite.

3.5.2 Simulation

We are forced to look for the model-based valuations of the banks numerically. This is a result of a complexity that interbank market brings to the model and of the option pricing formula which requires to calculate expectations with respect to the complicated distribution of losses and incomes. However, the main source of complexity is related to the equilibrium of the interbank payments.

The are two core components of the numerical procedure we propose. Firstly, it is drawing of the net income I for each bank. The definition of I straightforwardly translates into the algorithm – drawing 0 or 1 from two-point distribution B and then drawing *Gauss* or *Cauchy* respectively which is an elementary process in any programming language (e.g. in *Octave* that we use). Secondly, for given balance structures, interbank structure (matrix π) and the realized income in the banking system we find equilibrium payments vector by iterating of the transform Ψ (see Sect. 3.3). We specify 1,000 income scenarios for all banks, drawn independently across banks. That sequence of realized income and calculating of the interbank equilibria was the main building block of the valuation procedure. As defined in valuation formula (3.3), the equilibrium payments adjust the classical option price and the expectation operator in the option price formula has to be treated numerically as well.

The final step in our framework leading to the valuation of banks is the selection of the interbank structure π which minimizes the option value. We searched among 2,000 possible structures which are feasible given the aggregate interbank exposures reported by banks. Those structures are the outcome of the following *random matching scheme* (vectors LB and DB of interbank loans and deposits respectively are given):

1. Set the initial matrix of the interbank exposures P with all entries equal to 0 ($P_{ij} := 0$) and auxiliary $\mathrm{LB}^a := \mathrm{LB}$ and $\mathrm{DB}^a := \mathrm{LB}$
2. Draw randomly a pair of banks (i, j) such that $\mathrm{LB}^a_i > 0$ and $\mathrm{DB}^a_j > 0$
3. Draw a fraction α from uniform distribution on unit interval $[0.1, 1]$ and choose the incremental interbank placement $P^\delta := \alpha * \min(\mathrm{LB}^a_i, \mathrm{DB}^a_j)$
4. Assign $P_{ij} := P_{ij} + P^\delta$ and pull interbank loans and deposits off the stock, e.g. assign $\mathrm{LB}^a_i = \mathrm{LB}^a_i - P^\delta$ and $\mathrm{DB}^a_j = \mathrm{DB}^a_j - P^\delta$
5. *IF* $\sum_n \mathrm{DB}^a_n > 0$ and $\sum_n \mathrm{LB}^a_n > 0$ *THEN GOTO 2 ELSE END.*

There is s direct intuition behind the scheme. The pairs of banks reporting the accepted and granted interbank placements are drawn until the whole amount of either the interbank loans or deposits is distributed across the banking system. Thus, an interbank structure is incrementally constructed and practically any structure can be obtained in this way (Fig. 3.1). The parameter α allows for controlling of a number of the interbank debtors or creditors that banks may have.

The mechanism of the credit risk transfer is structured as follows. The matrix σ representing the propagation of the credit risk across the system is calculated in the matching procedure of securized loans and non guaranteed ABS securities as in the

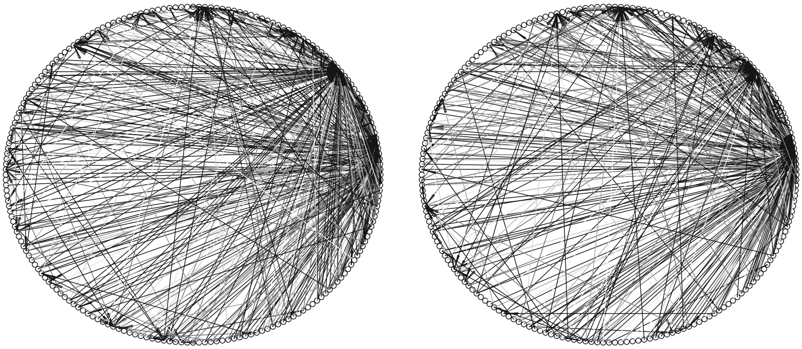


Fig. 3.1 Graphical representation of the matrix of the interbank exposures P – two possible outcomes of the random matching scheme (Note: *Small circles* on the *big circle* represent 200 largest banks in the US. *Arrows* correspond to P_{ij} for all pairs of banks (i, j). The *darker* the *arrow* the higher is the exposure (logarithmic scale of colors))

case of the interbank placements. Namely, in order to get a matrix Σ we match part of the securitized portfolios with the ABSs in the portfolios of the other banks until either the whole stock of the securitized loans is assigned or there are no more ABSs that can be matched with the securitized portfolio. The relative matrix σ is the direct consequence of the composition of L^S. Then, for each realization of the net income process we select banks that incur 1% highest loss (or the lowest net income which is negative). We check whether those banks securitized part of their loan portfolios. If so, then we assume 25% shock on that securitized part. The matrix σ propagates the shock across the system as described in Sect. 3.2.3. It means that a 25% shock in bank i that is equal to $\Delta \mathrm{L}_i^S$ leads to the devaluation of ABSs in bank j by $\sigma_{ij}\Delta \mathrm{L}_i^S$.

3.5.3 Discussion

There are various aspects of US banks' valuation test introduced in the previous section. The following three the most important. Firstly, liquidity of the interbank market affects the valuation of banks through the marketability of the counterbalancing capacity. Secondly, the application of the proposed valuation framework suggests which banks can be affected by the credit risk transfer mechanism. Thirdly, the contagious defaults have a very little impact on the valuations. This observation is coherent with the research of [11, 13, 21, 23] who argued that the network models are not capable of explaining the whole possible contagion of illiquidity among the financial institutions leading to losses (even insolvencies) of banks. We elaborate about those results in this section.

The liquidity was one of the main drivers of the financial crisis that wiped out a substantial part of capitalization of the financial market, esp. the US financial market. Hence, a desirable feature of our model would be to link the excessive banking assets devaluation to a distressed liquidity conditions in the system. The liquidity impact on valuation is compared with the observed changes of the market

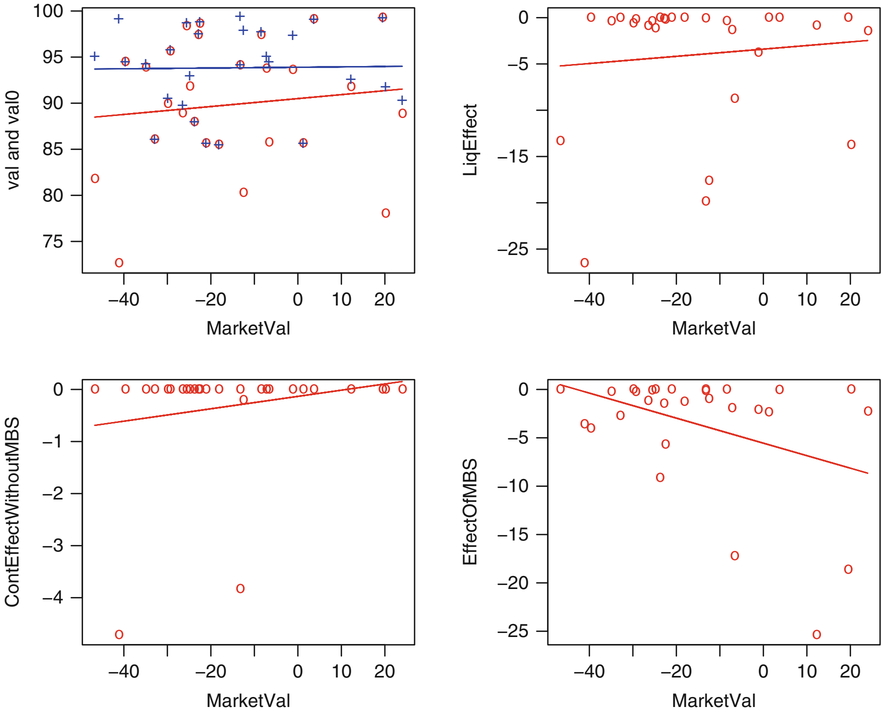


Fig. 3.2 The valuation – model vs. market-based performance (changes June 2007–Dec 2007) (Note: (**val and val0**) "+" valuation assuming that the interbank placements are paid back and "o" valuation in the market equilibrium; (**LiqEffect**) difference between val0 and val; (**ContEffectWithoutMBS**) $V_i^{\Psi} - V_i$ – measure of contagion impact on valuations; (**EffectOfMBS**) difference between val calculated without and with shocks on MBS. Source: finance.yahoo.com and the model)

valuations between June 2007 and the end of 2007 and between June 2007 and the end of 2008 in the group of 30 largest banks in terms of total assets. It has been evidenced that the deepness of the devaluation suggested by the model and resulting from insufficient liquidity or the quality of the counterbalancing capacity coincides with the magnitude of the market devaluation. All the analyzed banks exhibited the negative change of market valuation and two banks with the most profound liquidity shock (over 25%) had one of the strongest devaluations (over 60% during 1.5 years of the crisis). In general, we find a positive relationship between the valuations in the equilibrium and the market valuations. As on Figs. 3.2 and 3.3, the lower the valuation suggested by the model, the deeper devaluation of banks occurred, especially for the most negative changes of the adjusted valuations. Thus, the model gives one appealing explanation of linkages between liquidity and valuation.

The way the interbank equilibrium is defined points out to two mechanisms of how problems with liquidity may adversely affect the valuation. Both are observed in the data. Firstly, if loan portfolio shocks materialized then maximally ten banks would have excessively wide funding gap to be covered by the equity and

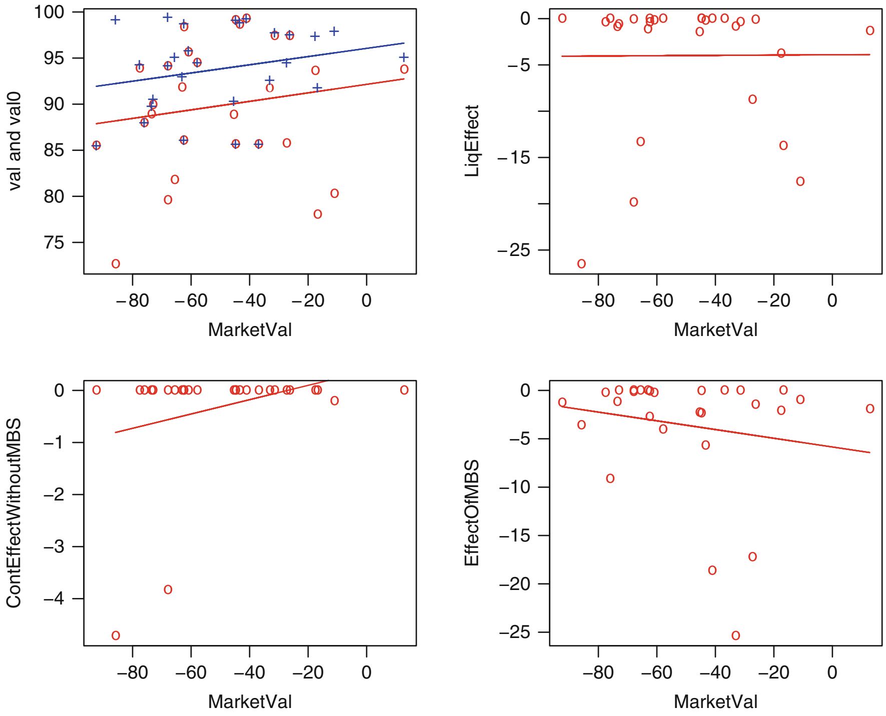


Fig. 3.3 The valuation – model vs. market-based performance (changes June 2007–Dec 2008) (Note: see Fig. 3.2. Source: finance.yahoo.com and the model)

counterbalancing capacity they possessed. Secondly, given the amount of the stock of the treasuries even though the banks kept enough MBS in the nominal terms, they would not find buyers for them. The parameter g_1 (see Eq. 3.2) reflecting the fraction of securities needed to restore liquidity that can be purchased by banks with liquidity surplus oscillates around 80%. Although, the counterbalancing capacity seems to cover the negative funding gap there is not enough potential on the buyers side of the interbank market. On average, but depending on the risk scenario and the interbank structure π, the volume of 110 USD mn of MBS would not find a purchaser on the interbank. Nevertheless, the model does not tell anything about other financial institutions like insurance companies or pension funds that could provide liquidity. However, banks usually provide liquidity to each other or obtain it from central banks.

We cannot find any reliable pattern between theoretical valuation and the contagion effect or the credit risk transfer. The outcomes of the simulation indicate that the contagion of the liquidity problems could effect two of the largest banks in the US. We observe that the market valuations of those two banks were hit the hardest in the sample during the crisis. However, according to the model maximally around 5% additional decrease of the valuation of one bank could happen if the debtors of two banks did not meet their interbank obligations. The direct interbank exposure is small enough to have only insignificant impact on the valuations.

Table 3.2 Comparing the effect of MBS devaluation on the valuation of some biggest US banks in the model (EFF. ON VAL.) with facts about the market performance of those banks

Name	EFF. ON VAL. (%)	Facts related to exposure
PNC Bank NA	20	Large MBS exposure proved to be secure, however large unrealised losses on MBS portfolio, bank assessed by Weiss Research as on borderline in stress test excersise
State Street Bank	11	6.28 bn loss in inv. portfolio (eoy 2008), 20% share prices decline in Jan–Apr 2009
Merrill Lynch	39	Mortgage related investments were detrimental, not only misrepresentation of risk in MBS portfolio but unhedged CDOs all triggering bankruptcy
The Bank of NY	32	Reported substantial loss in 2008 due to residential MBS portfolio[a]
Commerce Bank NA	26	Bought by TD Bank in 2007 and early 2008
Morgan Stanley	11	Bank withstood the crisis but reported huge losses (over 11 bn) related to its buyout financing activity, MBS and CDO exposure
Mellon Bank	26	Merged with The Bank of NY
Investors Bank and Trust	17	Acquired by State Street
Charles Schwab Bank NA	49	Its MBS portfolio is considered as secure, however substantial unrealised losses in 2008; bank did not avoid the stock prices decline at the end of 2008
Met Life Bank	4	Investment portfolio losses but well-capitalised passing Federal Government Stress-Test
Signature Bank	24	Almost 40% write-downs on the ABS portfolio but the bank remained sound

Source: yahoo.finance; the annual financial reports; Times

Note: Except for Charles Schwab the banks for which the devaluation caused by MBS shock was the most severe in the model (Merrill, Bank of NY, Signature) reported substantial ABS (mainly MBS) losses

[a] Press release of the banks on the financial results

The possible credit risk transfer through MBS exposures only directly amplifies the devaluation of banks. Namely, the banks having large, not guaranteed MBS portfolios may report losses on that portfolios translating into banks' devaluations (see val-valMBS graph on Fig. 3.4). The comparison of the market performance of a given bank and its MBS portfolio exposure confirm in most cases that those exposures contributed to the substantial devaluations of US banks during the turmoil (Table 3.2). It has to be mentioned that in some cases (PNC Bank, Charles Schwab Bank and Metlife Bank) it is difficult to disentangle the overall (systemic) and portfolio specific impact of the materialized credit risk on the valuations. The results of the simulation indicate that MBS did not aggravate the contagious diminishing of the valuations (compare valContMin-val and valContMinMbs-valMBS graphs on

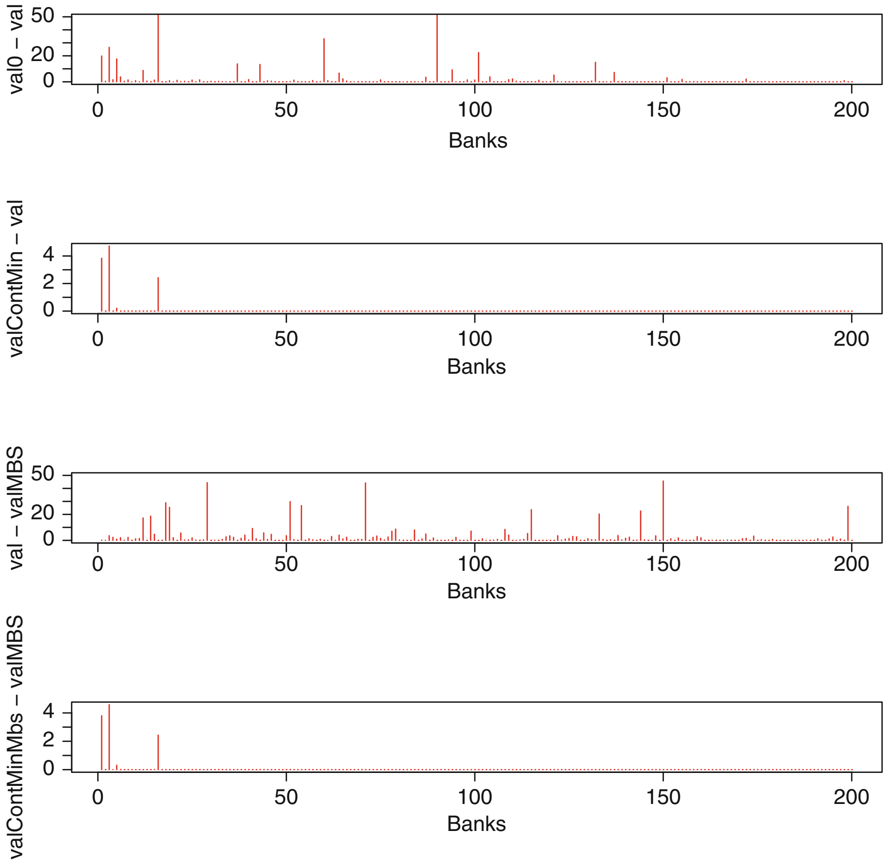


Fig. 3.4 The influence of liquidity and credit transfer on the valuations in the interbank equilibrium (Note: (**val0-val**) the difference between valuation assuming that the interbank loans and deposits are paid back *and* valuation in the market equilibrium assuming that the credit shocks do not affect ABS; (**valContMin-val**) difference between valuations in the system where it is assumed that each bank receives all the interbank placements *and* val; (**val-valMBS**) difference between val *and* valuation in the equilibrium with credit risk transfer shocks (**valContminMBS-valMBS**) difference between valuation assuming that the transfer of the credit risk shock takes place and the banks repay their interbank debts as if they receive back all the interbank placements *and* valMBS. Source: Model)

Fig. 3.4). So, the leverage effect described by Shin [22] which led to the inflation of MBS portfolios cannot explain the widespread losses observed in the system and related to the interplay of the credit risk and liquidity without referring to the correlation in balance sheets and to the bank runs.

Although the suggested size of contagion is rather small the defaults of banks in the model point out correctly to the banks that had serious financial problems during the crisis. The results confirm the doubts expressed by van Lelyveld and Liedorp [21] about the potential that direct interbank large exposure studies have in

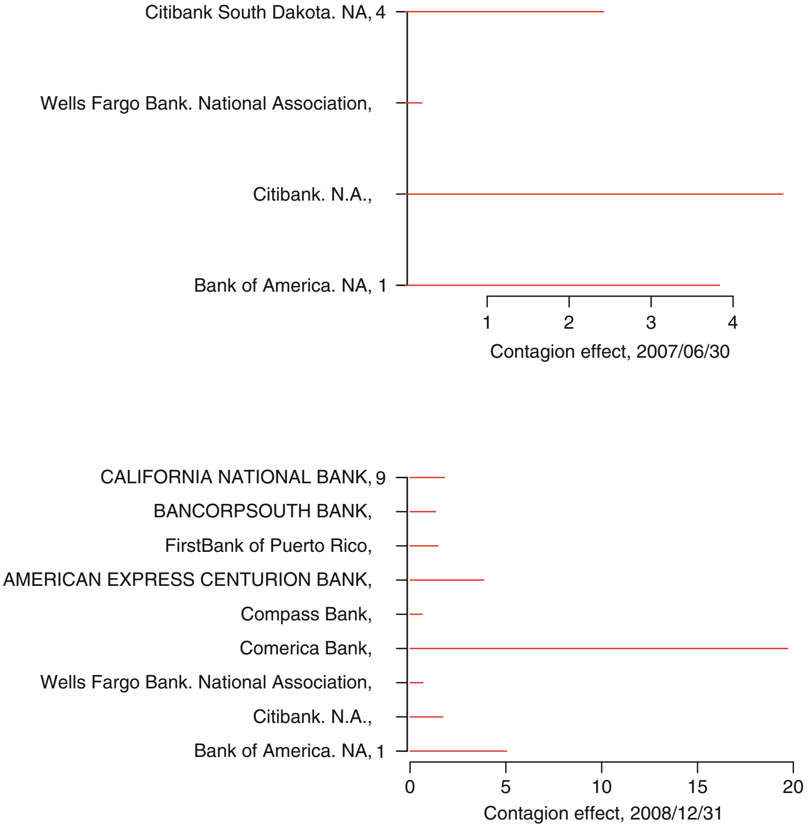


Fig. 3.5 Contagion effect of illiquidity (Note: Contagion for a given bank $= V_i^{\Psi} - V_i$. Source: The model)

explaining the magnitude of the contagion effect. However, the list of banks that may be effected by the liquidity problems of other banks contains: Bank of America, Citibank and Wells Fargo (see Fig. 3.5), i.e. the names whose financial strength was seriously undermined during the crisis.

3.6 Conclusions

The model we propose allows for calculating the value of banks that operate on the interbank market and whose risk related to the interbank exposures is difficult to assess applying historical, time series analysis. We extend the classical theory of valuation saying that the value of a bank depends, on one hand, on the risk of investment assets kept by the bank in its balance sheet, i.e. market risk of securities, the credit risk of loans granted to customers or the credit-linked securities and, on the other hand, is related to the leverage ratio of the bank defined as debt versus equity. The price of the bank is then formulated as the European call option with the strike equal to the debt volume. However, we claim the dynamics of the banks' assets is driven by the other banks financial capability to pay back their debts. It is universal observation that does not pertain only to the recent financial crisis. If it is expected that a given bank may not receive back its interbank placements then the valuation of that bank should take this expectation into account. The serious problem arises how to reflect the possible default of the interbank debtor into the valuation of the creditor if it is impossible to assign any probability to that defaults and to the interbank market structures. We propose to consider all theoretically feasible interbank placements related to the given structure of bank balance sheets and to calculate the price as the minimum value of the call option among the set of interbank structures. We show that the valuation of the bank depends on the interbank payments in equilibrium. The equilibrium describes the ability of banks to pay back their interbank debts given their balance sheet structures, volume of their portfolio of liquid securities and other banks ability to purchase them (i.e. liquidity surplus of other banks). This approach of the minimum value call option is justified by the theory of the decision making under ambiguity.

The proposed valuation framework gives an interesting insight into the US banking market. It explains the link between bank and market liquidity that is not an exogenous parameter but an outcome of the interbank equilibrium in which banks are trying to liquidate their portfolios of securities. The most profound market devaluations of the largest US banks after the outburst of the subprime crisis coincide with the predictions of our model. The framework provides an estimate for the valuations of banks if natural limitations to the marketability of the counterbalancing capacity may occur. It also indicates the banks with the highest exposure to the credit risk that can be transferred from securitized loan through ABS portfolios. And in order to obtain the results we only use public data.

Notwithstanding, further research is needed to introduce more strategic behavior of banks into the contagion and network models what has already been postulated by Allen and Gale [1,2] and van Lelyveld and Liedorp [21]. We believe that the models of the equilibrium payments based on the random graphs of different linkages in the financial system, complemented by more explicit treating of bank strategies, can further contribute to better understanding of extreme and unexpected behavior of the economy.

3.7 Isotone Ψ: The Proof

In order to show that Ψ is isotone it is sufficient to prove that, in its region of differentiability, Ψ has positive partial derivatives. Since function g used to define Ψ is not even continuous we decided to show the detailed proof of this key property of Ψ that guarantees the existence of the maximal fixed point.

We rewrite the function Ψ in a slightly different but an equivalent form. For each bank i define $\mathrm{ND}_i(p) \colon = \overline{\mathrm{Gap}}_i(p)^- \wedge (1-h)\bar{S}_i$ and $\mathrm{FC}_i(p) \colon = \overline{\mathrm{Gap}}_i(p)^+$. Let for each $i \in \bar{\mathbb{N}}$ a function $G_i \colon \left(\mathbb{R}_+ \cap \{0\}\right)^{N+1} \to \mathbb{R}_+ \cap \{0\}$ be defined as

$$G_i(x_1,\ldots,x_N,y) = \begin{cases} x_i \left(1 \wedge \frac{y}{x_1+\cdots+x_N}\right) & x_i \neq 0 \\ \\ 0 & x_i = 0 \end{cases}$$

and let us denote $G \equiv [G_1,\ldots,G_N]^\top$. It is straightforward that G is continuous. Hence,

$$\Psi_i(p) = \begin{cases} \left[\left[\mathrm{IP}_i(p) + G_i\Big(\mathrm{NS}_1,\ldots,\mathrm{NS}_N,\mathrm{FC}^{(a)}(p)\Big)\right] \vee 0\right] \wedge \bar{p}_i \\ \\ \left[\left[\mathrm{IP}_i(p) - G_i\Big(\mathrm{FC}_1,\ldots,\mathrm{FC}_N,\mathrm{NS}^{(a)}(p)\Big)\right] \vee 0\right] \wedge \bar{p}_i \end{cases}$$

is continuous.

Firstly, we show that G_i is Lipschitz continuous with the constant 1. Without loss of generality, we can show that $G_1(x_1,\ldots,x_N,y) = x_1(1 \wedge \frac{y}{x_1+\cdots+x_N})$ has Lipschitz constant equal to 1. A deserved Lipschitz continuity for y is obvious. Let us concentrate on argument x_1. We have: $G_i(x_1,\ldots,x_N,y) = x_1 \wedge \frac{x_1 y}{x_1+\cdots+x_2}$. If $x_1 < \frac{x_1 y}{x_1+\cdots+x_N}$ then the property is trivial. If $x_1 > \frac{x_1 y}{x_1+\cdots+x_N} \iff x_1+\cdots+x_N > y$ then $\frac{\partial}{\partial x_1} G_i(x_1,\ldots,x_N,y) = \frac{y(x_1+\cdots+x_N)-x_1 y}{(x_1+\cdots+x_N)^2} = \frac{y(x_2+\cdots+x_N)}{(x_1+\cdots+x_N)^2} < \frac{(x_1+\cdots+x_N)(x_2+\cdots+x_N)}{(x_1+\cdots+x_N)^2} = \frac{x_2+\cdots+x_N}{x_1+\cdots+x_N} < 1$. Hence, the derivative $\frac{\partial}{\partial x_1} G_i(x_1,\ldots,x_N,y) \in [0,1]$ and Lipschitz constant is equal to 1. Analogously, we can prove that for all $1 < j \leq N$,

$$\frac{\partial}{\partial x_j} G_i(x_1,\ldots,x_2,y) \in [-1,0].$$

Finally, we can prove the thesis. It is sufficient to show that in the region of differentiability the function $P : [0,\bar{p}_k] \to [0,\bar{p}_m]$,

$$P(z) = \Psi_m([\hat{p}_1,\ldots,\hat{p}_{k-1},z,\hat{p}_{k+1},\ldots,\hat{p}_N]^\top)$$

has the derivative higher than 0.

Take $\hat{p} \colon = [\hat{p}_1,\ldots,\hat{p}_{k-1},\hat{p}_k,\hat{p}_{k+1},\ldots,\hat{p}_N] \in \mathcal{P}$. Let us put for brevity:

$$\begin{aligned} a_i &: = \mathrm{Gap}_i^{(-interB,-\underline{S})}, \\ X_i(\hat{p}) &: = a_i + \sum_{j=1}^{N} \pi_{ji}\hat{p}_j - \bar{p}_i \\ S_i^h &: = (1-h)\underline{S}_i. \end{aligned}$$

Let us assume that $\overline{\mathrm{Gap}}_m(\hat{p}) < 0$. Define two sets:

$$\mathcal{J}_1 = \left\{ j \in \bar{\mathbb{N}} \middle| a_j + \sum_{n=1}^{N} \pi_{nj} p_n < p_j \right\},$$

$$\mathcal{J}_2 \colon = \bar{\mathbb{N}} / \mathcal{J}_1.$$

Then in a vicinity of $\hat{p}_k$

$$\begin{aligned} P(\hat{p}_k) = a_m + \sum_{j=1}^{N} \pi_{jm} \hat{p}_j + \\ + G_m\Big(X_1(\hat{p})^- \mathbf{1}_{\{1 \in \mathcal{J}_1\}} \wedge \underline{S}_1^h, \dots, X_N(\hat{p})^- \mathbf{1}_{\{N \in \mathcal{J}_1\}} \wedge S_N^h, \sum_{j \in \mathcal{J}_2} X_j(\hat{p})^+ \Big). \end{aligned}$$

Put $A \colon = (X_1(\hat{p})^- \mathbf{1}_{\{1 \in \mathcal{J}_1\}} \wedge \underline{S}_1^h, \dots, X_N(\hat{p})^- \mathbf{1}_{\{N \in \mathcal{J}_1\}} \wedge \underline{S}_N^h, \sum_{j \in \mathcal{J}_2} X_j(\hat{p})^+)$. Hence, differentiating P,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}z} P(\hat{p}_k) = \pi_{km} + \sum_{j \in \mathcal{J}_1 / \{m\}} \frac{\partial}{\partial x_j} G_m(A) \cdot (-\pi_{kj}) + \frac{\partial}{\partial x_k} G_m(A) \pi_{kk} \\ + \sum_{j \in \mathcal{J}_2 / \{k\}} \frac{\partial}{\partial y} G_m(A) \pi_{kj}. \end{aligned}$$

Taking into account upper and lower bounds for derivatives of G and $\pi_{mm} = 0$ we obtained the coefficients standing at $\pi_{k\cdot}$ that are positive (and not higher than 1 except for the case of π_{km}). Thus, $0 \le \frac{\mathrm{d}}{\mathrm{d}z} P(\hat{p}_k) \le \sum_{j \in \mathcal{J}_1 / \{k\}} \pi_{kj} + 2\pi_{km} + \sum_{j \in \mathcal{J}_2} \pi_{kj}$. The reasoning in the case of $\overline{\mathrm{Gap}}_m(\hat{p}) \ge 0$ is similar.

Acknowledgements The author would like to express his gratitude to the participants of the research seminar in the National Bank of Belgium (September 2009) and of the Hurwicz Workshop in Warsaw (October 2009) and the anonymous referee for very valuable comments on the interbank liquidity part of this chapter. The content of the chapter was essentially improved during the author's postdoc fellowship at the Fields Institute (Toronto) in 2010.

References

1. Allen, F., Gale, D.: Financial Contagion. Journal of Political Economy **108**, 1, 1–33 (2000).
2. Allen, F., Gale, D.: Financial Fragility, Liquidity, and Asset Prices. Journal of the European Economic Association **2**, 1015–1048 (2004).
3. Basel Committee on Banking Supervision, Basel II: International Convergence of Capital Measurement and Capital Standards: A Revised Framework – Comprehensive Version. Bank for International Settlements (2006) http://www.bis.org/publ/bcbs128.htm

4. Black F., Scholes M.: The Pricing of Options and Corporate Liabilities. Journal of Political Economy **81**, 637–653 (1973).
5. Cifuentes R., Ferrucci G., Shin, H.S.: Liquidity Risk and Contagion. London School of Economics (2004).
6. Chorafas, D., Liabilities, Liquidity, and Cash Management: Balancing Financial Risk. John Wiley & Sons Inc, (2002).
7. Cousot, P., Cousot, R.: Constructive versions of Tarski's fixed point theorems. Pacific Journal of Mathematics **82** 1, 43–57 (1979).
8. Degryse, H., Nguyen, G.: Interbank exposures: an empirical examination of systemic risk in Belgium banking system. International Journal of Central Banking **3** 2, 123–171 (2007).
9. Diamond D., Rajan R.: Liquidity Shortages and Banking Crises. The Journal of Finance **60** 2, 615–647 (2005).
10. Eisenberg L., Noe T.: Systemic risk in financial systems. Management Science **47**, 236–249 (2001).
11. Elsinger, H., Lehar, A., Summer, M.: Risk Assessment for Banking Systems. Vienna University, Working Paper (2003).
12. Elsinger, H., Lehar, A., Summer, M.: Analyzing Systemic Risk in the European Banking System: A Portfolio Approach. Vienna University Working Paper (2004).
13. Elsinger, H., Lehar, A., Summer, M.: Using Market Information for Banking System Risk Assessment. International Journal of Central Banking **2**, 1–29 (2006).
14. Epstein, L., Schneider, M.: Ambiguity, Information Quality, and Asset Pricing. The Journal of Finance, **63** 1, 197–228 (2008).
15. Estrada, D., Osorio, D.R.: A Market Risk Approach to Liquidity Risk and Financial Contagion. Department of Financial Stability in Banco de la República de Colombia (preliminary version of the paper presented on the seminar "Research Forum: Micro-models of systemic risk", May 25–26 London (2006).
16. Granas A., Dugundji J.: Fixed Point Theory. Springer-Verlag, New York (2003).
17. Hałaj, G.: Contagion Effect in Banking System — Measures Based on Randomised Loss Scenarios. Bank and Credit **6** (journal of the National Bank of Poland), 69–80 (2007).
18. Iori, J., Saqib, J., Padilla, F.: Inter Bank Lending, Reserve Requirements and Systemic Risk. Department of Mathematics of King's College, London (2004).
19. Merton, R.: On the Pricing of Corporate Debt: The Risk Structure of Interest Rates. The Journal of Finance **29** 2, 449–470 (1974).
20. Ross, S.A., Westerfield, R.W., Jordan, B., Jaffe, J.: Corporate Finance: Core Principles and Applications. McGraw-Hill (2007).
21. Lelyveld, I van., Liedorp, L.: Interbank Contagion in the Dutch Banking Sector: A Sensitivity Analysis. The International Journal of Central Banking **2** 2, 99–133 (2006).
22. Shin, H.S.: Securitisation and financial stability. The Economic Journal **119**, 309–332 (2009).
23. Wells S.: Financial interlinkages in the United Kingdom's interbank market and the risk of contagion. Working Paper of the Bank of England **230/2004** (2004).

Chapter 4
An Open Problem

John B. Walsh

Mathematicians delight in pointing out how much the sciences owe to mathematics, but it is only fair to record the converse: mathematics owes just as much to the sciences. Indeed, many mathematicians regard the sciences as subjects which, though useful in themselves, are mainly there to provide interesting new mathematical problems.

The purpose of this note is to suggest one of these, an open problem brought up by recent economic and financial events.

After the sub-prime crisis and the bursting of the associated financial bubble, it was decided that the economy urgently needed a stimulus. This brought up two key questions:

"Where?" and **"How much?"**

Ideally, if we had a good enough mathematical model of the economy, these questions would be answered by solving an extremal problem somewhat like this: among all scenarios with acceptable outcomes, choose "where" to minimize "how much."

Since nobody announced such a solution, I conclude the problem is open.

In any case, the stimulus money has already been spent, so this particular question is moot. But it likely to arise again, and the problem is important enough in its own right that even partial solutions should lead to both interesting mathematics and interesting economics.

Of course, we are not in a position to solve it at the minute. Both economic and mathematical theories need to be developed.

We must also decide what constitutes an acceptable outcome. This question is as much political as mathematical or economic. Cynics might suggest that it is that the institutions too large to fail, don't. A more likely aim would be that the system ends

J.B. Walsh (✉)
Department of Mathematics, The University of British Columbia, Vancouver, BC, Canada
e-mail: walsh@math.ubc.ca

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_4,

up in a certain desirable statistical equilibrium. In any case, we would like to be able to handle this. So we can state the first open problem:

> What do we need to develop in order to state and solve this extremal problem?

Before we can even state it clearly, we must identify the economic variables, understand how they effect each other, how changes in one leads to changes in the others, and what variables are controllable. These are primarily problems for economists. But we can make some general remarks on the mathematics this implies.

It appears that the system, particularly the financial part, can be modelled as a network. Then the above extremal problem becomes a problem in optimal control of (probably stochastic) economic/financial networks.

Notice that the problem is decidedly non-linear: lending a financial institution enough money to stave off bankruptcy is more than twice as effective as lending it half enough. Moreover, the system will be more than a coupled system of differential equations, one for each node. Indeed, one cannot ignore the possibility of defaults, and these, coupled with possible mergers, suggest that the solutions may be discontinuous, and that the network itself may change with time.

The first step is probably to understand cascades of bankruptcies, and the second is to understand how to block such a cascade. This is the simplest case because the immediate result of a bankruptcy is fairly straightforward: the remaining assets of the bankrupt firm are distributed among its creditors according to well-defined laws. But the stimulus problem includes the influx of money—in some sense the opposite of a bankruptcy—and this is much deeper. This is because an influx of money requires a choice: what should be done with it? How should it be invested? Choices require humans to make them, and wherever humans enter, randomness cannot be far behind. In other words, the problem is fundamentally stochastic.

It should be clear from the above that the problem is immensely, but perhaps not hopelessly, complex. So let us say a few words about cutting it down to size.

First, a problem for the economists: how small a part of the economic system can one treat and still get believable results? Is it enough to treat the financial system alone? If so, is it enough to treat a skeleton consisting of the biggest institutions? Or must one include the manufacturing sector and questions of employment? In fact, can one even isolate—just for the purposes of this problem, of course—the US and Canada from, say, Europe and Asia?

Here is a related problem for mathematicians: find results on the approximation of bigger financial networks by smaller ones, and find bounds on the effect of one part of the network on another that would help to justify the approximation.

However, the most feasible approach to this would appear to be the usual one: construct toy models which share important characteristics of the full-scale system, and which are mathematically tractable. Then search these for significant behavior. One might anticipate some counter-intuitive results, and these might very well lead to some rules of thumb which would help improve the answers to the two questions, "Where?" and "How much?"

For example, Rama Cont pointed out the following possibility. In the study of the security of computer networks, in order to limit the spread of virus, it can be more effective to inoculate the first-order clients of a big server than to just treat the server itself. There is a well-developed theory on this. He conjectured that this might apply to financial networks, and that the financial regulators could perhaps concentrate on the health of the clients of a large central financial institution, not just on the institution itself.

Let's make one final observation, which leads to an amusing irony. It may turn out to be useful to have something in the model to quantify what is called investor confidence. It often happens in financial bubbles that there is an instance, totally unpredictable but inevitable, when some child cries, "But the emperor has no clothes." This triggers a change, seemingly en masse, of investors' strategies, from risk-taking to extremely risk-averse, whereupon the bubble collapses. The irony is this: if it were possible to actually solve these problems, and if it were generally known that the method was sound and that the regulators and the government were using it, this would greatly increase the investor confidence. In other words, it is possible that the theory contains a variable whose value depends on the validity of that very theory.

Part II
Network Security

Chapter 5
Dynamic Trust Management: Network Profiling for High Assurance Resilience

Mike Burmester and W. Owen Redwood

Abstract Trust Management (TM) systems are infrastructures that support efficient and secure access to resources in large decentralized systems. They provide a language for expressing authorizations and access control policies as well as a trust management engine that processes requests, to automatically address access requests. Traditionally, the enforcement of Trust Management decisions is static and involves the use of appropriate cryptographic mechanisms. However, recently two TM systems were proposed for which the enforcement is dynamic.

Dynamic TM systems expand, (i) the expressibility of a system language to capture anomaly-triggered access control policies, and (ii) the enforcement capabilities via graduated response mechanisms such as Rollback Access control. These mechanisms are proactively triggered under the perceived potential of an attack: they selectively disrupt the TM-granted access to a resource temporarily, to mitigate the system threat.

In this Chapter we discuss the use of real-time stochastic analyzers and graduated response security mechanisms to detect/prevent anomalies in TM systems, and propose an architecture for dynamic Trust Management that tolerates 0-day attacks and insider attacks.

5.1 Introduction

Network profiling allows networks to deploy graduated security mechanisms to respond dynamically to 0-day attacks and potential insider threats. While most network threats appear statistically anomalous, it is important to note that not all anomalies are malicious. A separate, modular network layer is required to filter

M. Burmester (✉) • W.O. Redwood
Florida State University, Tallahassee, FL, 32306-4530, USA
e-mail: burmester@cs.fsu.edu; redwood@cs.fsu.edu

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_5,

out significant anomalies that are more likely to indicate a potential threat, from other anomalies. Filtering is done by using a graduated response mechanism. In this chapter we present a dynamic Trust Management (TM) architecture that combines a network profiler with a graduated response mechanism to combat and mitigate damage caused by such threats.

The outline of the chapter is as follows. In Sect. 5.2 we discuss related work, Trust Management, Intrusion Detection/Prevention and graduated response mechanisms. In Sect. 5.3 we present a dynamic TM architecture and in Sect. 5.4 we cover the mathematical concepts and notation used in our approach. Finally, in Sect. 5.5 we present two network application scenarios prone to insider attacks and describe a dynamic TM infrastructure that will tolerate such attacks.

5.2 Overview of Related Work

The research presented in this chapter is closely related to many areas of network security and extends work in [7, 8, 17, 20, 24, 33, 44, 45].

In this section we discuss various access control and Trust Management systems, Intrusion Detection (signature and anomaly based) and Prevention systems, and graduated response security mechanisms.

5.2.1 Access Control and Trust Management

With the advent of the computer security modeling era in the 1970s–1980s, the concepts of the Mandatory Access Control (MAC) and Discretionary Access Control (DAC) were first envisioned by David Elliott Bell and Leonard J. La Padula [4] as a tool for managing the resources of computer and network systems. The Bell-LaPadula confidentiality model established the foundation for modeling secure multi-user computing, and network file management. Several integrity models emerged, most notably Biba, Lipner, and Clark-Wilson [23].

Steve Lipner conceived the use of access control matrices for file system protection and integrity in his famous integrity model that built on the Bell-LaPadula model. However, matrix models do not scale well with the number of agents in a network, and this motivated Role Based Access Control (RBAC) [13, 39]. RBAC systems assign different roles with specific permissions to users, and can be combined with MAC and DAC to offer a more efficient platform for managing system resources. Many systems today use RBAC mechanisms. A notable weakness of these systems, originally, was the requirement of some form of centralized architecture to manage access control.

Trust Management (TM) systems provide a unified approach in specifying and interpreting security policies, credential and relationships. They support scalable and efficient access control management for network resources in decentralized environments. In these systems the trust is modeled by the graph whose nodes correspond to the TM users (agents) and whose (directed) edges correspond to the trust between the users. Authorization is captured by (directed) trust flow paths that link users, and access requests are established through policies [44].

Various TM systems have been developed over the past 20 years. Some focus on the logic of the TM engine [1, 2, 26], others on general purpose authorization [5, 6, 25], while others combine specific authorization mechanisms such as Access Control Lists (ACL) and Public Key Infrastructures (PKI) [3, 9, 18].

Attribute Based Access Control (ABAC) systems are better suited for need-to-share centered policies, rather than a need-to-know policies [30]. In ABAC systems, access is granted based on attributes or characteristics of the user, the environment and of the resource itself, not based on the rights of the user. Consequentially, ABAC systems can theoretically offer a finer grained access control model than RBAC systems. In ABAC systems, a user must prove any attributes he/she claims to have in order to gain access to a resource requiring those attributes, and this can be performed anonymously. XACML (eXtensible Access Control Markup Language) is a common XML-based standard for ABAC systems with a number of implementations [31]. It is important to note that most available operating systems do not inherently support the ABAC model, and that its major limitation is that it suffers from scalability issues in large environments with many resources, users, and attributes [30].

However network administrators often choose to implement simpler TM systems such as PGP (Pretty Good Privacy) [46] and variants of SDSI (Simple Distributed Security Infrastructures) [37]. Today, the systems that have attracted the most attention are SPKI (Simple Public Key Infrastructures) [25], KeyNote [5], RT (Role based Trust management) [27], and RBAC based systems [13, 39].

The current paradigm shift in Trust Management systems is Risk-Adaptive Access Control (RAdAC) [30]. RAdAC extends upon earlier access control models by primarily introducing environmental conditions and attributes (i.e. from ABAC), risk levels into the access control decision process (as seen in our model with threat levels). RAdAC systems incorporate information about an entity's trustworthiness, information about the network and extra-organizational networks, environmental risk factors (i.e. Federal or agency threat level), and RAdAC systems use all of these factors to establish a quantifiable risk metric [30]. The work presented in this paper can be considered a theoretical extension of RAdAC systems.

5.2.2 Intrusion Detection/Prevention Systems

While the systems discussed above may provide adequate protection for resource confidentiality, integrity, and availability with "normal" users, the need to protect them against *adaptive malicious* outsiders (outside users) sparked the proliferation of Intrusion Detection Systems, Intrusion Prevention Systems and anti-virus/malware software. These systems offer *reactive* defenses against outsider threats.

The National Institute of Standards and Technology (NIST) formally defines *intrusion detection* as [29]:

> The process of monitoring the events occurring in a computer system or network and analyzing them for signs of possible incidents, which are violations or imminent threats of violation of computer security policies, acceptable use policies, or standard security practices.

NIST defines *intrusion prevention* as:

> The process of performing intrusion detection and attempting to stop detected possible incidents.

Both processes focus on identifying, (a) possible incidents, logging information about these, attempting to stop these, and report them to security administrators as well as, (b) problems with security policies, documenting existing threats, and deterring individuals from violating security policies [29].

There are two types of intrusion detection modes: (a) *signature based* (aka *pattern matching based*, *misuse detecting*, *rule based*, and *knowledge based*) and (b) *anomaly based* (aka *statistical based*). Furthermore, intrusion detection can be *host based* (HIDS) and *network gateway based* (NIDS). HIDS and NIDS traditionally focus on incoming attacks against the host or network.

Eventually, attack detection mechanisms may become sufficiently accurate and responsive to develop automated response mechanisms. However, these mechanisms will only respond to signature/anomaly based attacks.

5.2.3 Signature Detection Systems

Signature-based detection is the primary mode of most Intrusion Detection Systems (IDS) that are publicly available. This mode offers high-accuracy in detecting attacks for which the system has the signature base to detect them. A typical signature base may include: packet header information, payload information, network statistics, user-login behavior, operating system calls used by programs, and so on.

However the effectiveness of signature-based IDS is limited by the IDS's signature knowledge. By definition all signature-based IDS will fail to detect attacks for which they do not have a signature to match the attack, e.g., novel attacks, 0-day attacks, or tweaked attacks [29]. Indeed sophisticated attackers can fool almost all known IDS systems by altering the signature of the traffic via computationally cheap obfuscation techniques. For example a simple compression of packet payloads will fool most existing IDS. Attackers primarily use such techniques to obfuscate the incoming and outgoing network traffic to an infected terminal. Another common problem is that most IDS solutions fail to implement Deep Packet Inspection (DPI) mechanisms that reconstruct the traffic flow at the application layer.

Signature based detection is closely related to *specification-based misuse detection*. Specification-based misuse detection relies on network administrators to carefully define normal parameters for various factors in the network (e.g., the percentage of traffic that is normally ICMP). The process of specifying parameters is often handled by technicians with expert knowledge, and is considered more of an art than a science [41]. A key challenge with deploying a specification-based solution is achieving "good" feature selection. However, by definition this method cannot automatically adapt to changes in user behavior, technology, and applications, and thus does not scale well.

5.2.4 Anomaly Detection Systems

NIST defines *anomaly-based intrusion detection* as [29]:

> The process of comparing definitions of what activity is considered normal against observed events to identify significant deviations.

To accomplish this, IDS forms a *profile* (or *baseline* of normal user/system activity). The first comprehensive anomaly-based intrusion detection models date back to 1987 [10]. Other early anomaly-based intrusion detection research explored various statistical methods and models (e.g., Bayesian, Markov chain, and Hidden Markov Models) for anomaly detection with high accuracy and low false-positive rates in attack reporting [11, 14, 21, 28, 40].

The computational overhead of anomaly-based detection systems includes the time it takes to develop the profile and the number of passes the detector should perform while processing the input. A sufficient, but not necessary property of an anomaly detector to perform in real-time, is that processing requires only one pass over the input to be analyzed. Most anomaly detection systems can achieve this [14].

Anomaly-based attack detection has been popular in *operating system call based intrusion detection* [14, 15]. Developing the normal behavior profile for system call based detectors is relatively simple, as a typical program makes over a million system calls per execution. The attack detection mechanisms in IDS have recently become fairly sophisticated, with the leading contemporary systems using a combination of signature-based and anomaly-based detection. However, the problem of protecting a network and its resources against (malicious) insiders is still open. Indeed insiders require little sophistication to defeat most existing security mechanisms. Insider attacks can be quite subtle and may also involve infected terminals or hacked accounts of normal users, which are preprogrammed or remotely controlled in an undetected manner.

5.2.5 Binary Versus Graduated Response Mechanisms

During the early era of computer security research, biological immune systems were used to model security systems. This gave rise to the concept of *graduated response* mechanisms, as opposed to binary response mechanisms.

Binary action/response events are ubiquitous in the field of computer security: e.g., access to system resources is either permitted or forbidden (access control); files are either protected or not (privacy/integrity); connections are either permitted or blocked (firewalls and gateways); and so on. However computer viruses exhibit biological behavior [42] for which the natural defenses are graduated and not binary. Naturally, the human immune system inspired much research in the areas of virus/intrusion detection and network security [14].

Some of the most cited security mechanisms that employ graduated responses are *throttle traffic* [22] and *process-priority* [15]. For these, small system disturbances trigger small responses, while large disturbances trigger large responses. Architectures that employ such mechanisms will tolerate imperfect threat detection.

5.3 Threat Management

Network profiling allows networks to deploy graduated security mechanisms and respond dynamically to potential threats. While most network threats appear statistically anomalous, not all anomalies are malicious. To filter out significant anomalies that are more likely to indicate a potential threat a separate network layer is used, which we call the *threat level control* (TLC) layer.

This section details the TLC layer (expanding upon [7, 8, 35, 36]) and presents an architecture for dynamic Trust Management that combines a network profiler and a graduated response mechanism to manage threats. Two graduated response mechanisms are discussed in detail, Rollback-Access control (Sect. 5.3.5) and a new mechanism called *equivocated sanitazation* (Sect. 5.3.6).

5.3.1 A Dynamic Trust Management Infrastructure

Threat levels are a construct of the network's security policy that describe the degree of perceived threat posed by entities to the trust management infrastructure. Threat levels can be *local* or *global* and have a *linear ordering*, or more generally a *partial ordering*. For instance, the United States Department of Homeland Security implemented a five-stage linear order color coded terrorist threat advisory system (low/green, guarded/blue, elevated/yellow, high/orange, and severe/red), for which each stage restricts the possible authorizations available (in addition to the implied increased security measures and threat awareness)—see Fig. 5.1.

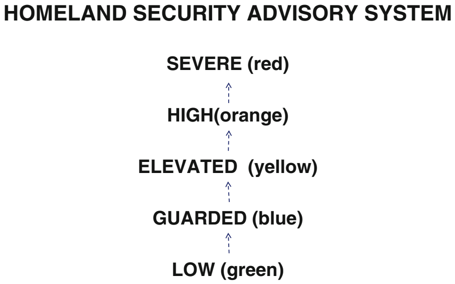


Fig. 5.1 The Department of Homeland Security five-stage threat advisory system

We shall model our threats by a partial order $(\Theta, \succeq)$.[1] To get a dynamic TM infrastructure, each threat $\theta \in \Theta$ will restrict the authorizations of TM to mitigate the associated risks. Let TM^{auth} be the *authorizing functionality* of the Trust Management system and TM_{θ}^{auth} the *restricted authorizing functionality*. We denote the set of all restricted authorizing functionalities by TM_{Θ}^{auth}, that is:

$$TM_{\Theta}^{auth} = \{TM_{\theta}^{auth}\}_{\theta \in \Theta}.$$

For graduated response access control each user (client), or group of users (for Role-Based Access Control), is assigned a *threat clearance* $\theta^* \in \Theta$ based on profile behavior: a high threat clearance is assigned for typical behavior and a low threat clearance is assigned for anomalous behavior. Authorization requires that the threat clearance θ^* dominates the system threat level. That is, an access request by a user U with threat clearance θ^* is authorized by TM_{θ}^{auth} only when $\theta^* \succeq \theta$.

This induces a domination relation "$\succeq_{auth}$" on the authorizing functionality infrastructure TM_{Θ}^{auth} for which: $TM_{\theta_2}^{auth} \succeq_{auth} TM_{\theta_1}^{auth}$, if every action that is authorized by $TM_{\theta_1}^{auth}$ is also authorized by $TM_{\theta_2}^{auth}$. We then get,

$$\theta_1 \succeq \theta_2 \quad \Rightarrow \quad TM_{\theta_2}^{auth} \succeq_{auth} TM_{\theta_1}^{auth}. \tag{5.1}$$

Consequently by lowering the threat level, authorization is extended until eventually it is fully restored. Conversely by raising the threat level, authorization is restricted until eventually it is reduced to "vanilla" authorization.

For an illustrating example, consider a small network with two groups G_1, G_2. The network administrator can establish a threat clearance for each group and an individual threat clearance for each user in the groups G_1, G_2. If the threat clearance of G_1 is $\theta(G_1) = \theta_1$ and the system threat level is raised to $\theta_2 \succ \theta_1$, then all actions of the users U_1 of G_1 will be suspended, regardless of any other security policies of the TM system for these users (e.g., their security clearances). Actions of the users

[1] The pair $(X, \succeq)$ is a *partial order* if "$\succeq$" is a binary relation on the set X with: (a) $x \succeq x$ for all $x \in X$ (reflexive), (b) if $x \succeq y$ and $y \succeq x$ then $x = y$, for $x, y \in X$ (antisymmetric), and (c) if $x \succeq y$ and $y \succeq z$ then $x \succeq z$, for $x, y, z \in X$ (transitive).

U_2 in G_2 that are authorized by the TM infrastructure will not be suspended if their individual threat clearance $\theta(U_2)$ and the group threat clearance $\theta(G_2)$ dominate θ_2 (that is: $\theta(U_2) \succeq \theta_2$ and $\theta(G_2) \succeq \theta_2$).

Threat clearances have an inversely proportional impact on the trust levels of the TM system. For example, a high threat clearance for a user means that the user is not a threat to the system and therefore the user can access the resources permitted by the TM system; conversely a low threat clearance for a user means that the user is a threat to the system and therefore that user's access to resources may be suspended when the threat level is raised, even if such access is permitted by the TM system.

Therefore, when the threat level is raised from θ to θ^+, the resulting functionality $TM_{\theta^+}^{auth}$ may no longer authorize a current or ongoing action of the user that was authorized by TM_{θ}^{auth}. Such (unauthorized) actions are suspended, or *rolled-back*. This occurs independently of other trust mechanisms that may apply.

5.3.2 How the Threat Level Changes

In our dynamic TM infrastructure TM_Θ, the threat clearances of users (clients) can be manipulated manually and/or automatically. Intrusion Detection/Prevention Systems, network administrators, supervisors, and anomaly threat level control (TLC) analyzers can all influence the threat clearance.

The particular factors that influence the threat clearance depend on whether this applies to an individual user or a group. For an individual user the input factors involve observations of that user's actions. The actions of other users are not taken into account. For a group of users, the actions of that group influence the group threat clearance.

Existing Intrusion Detection systems and Intrusion Prevention Systems can be configured to flag the threat clearance to be changed. Certain signatures (e.g., signatures for the latest 0-day exploit) can be flagged to automatically increase the threat clearance, independently of other monitors. One or more features may be analyzed for anomaly detection. To process output from multiple anomaly analyzers in a consistent manner, a *threat level dial* that fluctuates between two thresholds t_{lower}^{θ} and t_{raise}^{θ} is used. The least (minimum) threat clearance θ_0 in Θ has no lower threshold $t_{lower}^{\theta_0}$ and likewise the greatest (maximum) threat clearance θ_{max} has no higher threshold $t_{raise}^{\theta_{max}}$. These thresholds are independent across threat clearances.

When a threat clearance is initialized (e.g., when the threat clearance is changed, or monitoring begins), the threat level dial is set at zero. When the anomaly TLC analyzer observes a change in user (or group) behavior then the threat level dial increases (decreases) by $\varepsilon \cdot c$, where $\varepsilon \in [0,1)$ is a system parameter and c is the confidence level of the analyzer (if available).

The anomaly TLC analyzer is responsible for processing the readings of profiled behavioral features (e.g., typical user file access behavior), and often these features correlate under simulated attacks. If certain attacks demonstrate detectable correlation among monitored features then the TLC analyzer focuses on these features. If there are several features that are monitored, then one can use principal component analysis algorithms to select those features with the highest correlation. We shall discuss this further in the following section.

Upon a threat clearance change for a user (group), the system allows for the supervisor (administrator) of a user to be notified. The event trace, outlining exactly how the threat level has increased for that user (group), is provided for forensic purposes. Furthermore, the system supervisor (administrator) is provided with an override mechanism that effectively lowers and temporarily disengages the threat level of users (groups), in cases when there is evidence that the actions that triggered the threat level to rise were authorized.

The system threat level is established in a similar way, with input factors based on system behavior.

5.3.3 Feature Selection

The stochastic features for anomaly profiling are captured by a *multivariate data set*, whose variables correspond to the profiled features. Selecting those features that contribute most to the variability of the data set is a major task.

Principal Component Analysis (PCA) is one technique for doing this. PCA replaces the original variables (features) by a smaller number of derived variables, called *principal components*, which are linear combinations of the original variables. Often, it is possible to retain most of the variance of the original variables by using a much smaller set of variables. In such cases one can reduce the number of features needed for profiling.

The principal components are usually ordered so that their variance is decreasing, with the first component having the largest variance (see Sect. 5.4.7). This component is called *the principal component*. Sometimes one may also add some of the lower components to capture more variance. The features that correspond to these components are then used to profile the normal (or typical) behavior of users (or groups).

5.3.4 Threat Level Policies

When the threat level changes to θ, the authorization policies specified by TM_{θ}^{auth} are invoked. Naturally the lowest threat level authorizes a superset of actions that contains as subsets the authorized actions of higher threat levels. When the threat level rises, a subset of the actions of the previous threat level will be unauthorized. Likewise when the threat level is lowered, a superset of the actions of the previous threat level will be subsequently authorized.

Establishing effective policies for the threat level model can be a challenge in itself. The policies depend on the scope of the threat level and the corresponding permissions/clearances of the user (group).

In the following section we discuss a particular TLC policy analyzer mechanism presented in [7,8,36], called *Rollback Access control*, and outline a process that can be used to establish policies for a threat level model.

5.3.5 Rollback Access

This section explains the Rollback Access mechanism outlined in [7]. Each access action α of an entity (or program) U is assigned an *access threat classification* $\theta(\alpha)$ that is the threat level clearance of U; $\theta(\alpha)$ is the highest threat level at which access to α is authorized (this is independent of the authorizations of the underlying TM system). When the threat level is elevated to $\theta^+ \succ \theta(\alpha)$, the functionality $TM^{auth}_{\theta^+}$ is invoked and Rollback Access is triggered. Current actions that are not authorized under $TM^{auth}_{\theta^+}$ (such as α) get suspended (termed *rollback: withdrawal mode*), and a record of their partially executed state is temporarily stored. The record(s) for α are retrieved when (later) the threat level is lowered to $\theta \preceq \theta(\alpha)$, allowing for the state of α to be restored.

The key characteristics of a Rollback Access mechanism are:

(a) In withdrawal mode, action α is suspended;
(b) Rollback Access is transitory;
(c) Rollback Access is segregated (from other rollbacks); and
(d) Action α is restored when it gets authorized by a lower threat level (under *rollback: restore mode*).

Records of partially executed actions are stored in memory banks called *information compartments*. For each threat level $\theta \in \Theta$ there is an information compartment IC_θ. If the execution of an action α with threat classification $\theta(\alpha)$ is suspended because the threat level is raised from θ to $\theta^+ \succ \theta(\alpha)$, then a record of the state of α is stored in the information compartment $IC_{\theta(\alpha)}$. Intermediary information compartments are allowed, since several actions may be rolled-back upon a threat level elevation, allowing for Rollback Access segregation.

Any resource β that is created by a user U with threat clearance θ is automatically assigned a threat classification θ, that is: $\theta(\beta) = \theta(U) = \theta$, in addition to the other underlying policies (security classifications) specified by the TM system. This mechanism requires that the threat level $\theta \in \Theta$ is *inversely* related to the TM-dominance "$\succeq_{auth}$" in $TM_{\Theta}{}^{auth}$ as noted earlier in Eq. (5.1). The justification for this requirement is that our threat model supports the *principle of controlling threat flows*: information can only flow to entities whose threat clearance dominates the threat level. We call this, the *simple threat property*.

For the Bell-LaPadula confidentiality model the *simple security property* [4] does not allow information to flow *down* (for read-access). If the underlying

Table 5.1 Rollback Access mode behavior. Initially $IC_\theta = \emptyset$ for all $\theta \in \Theta$

Threat level action	Restore mode action	Withdrawal mode action	IC action
θ is raised to θ^+	No action	Invoke $TM^{auth}_{\theta^+}$. Unauthorized actions α are suspended. $IC_{\theta(\alpha)}$ is allocated as necessary. For every suspended α a record is put in $IC_{\theta(\alpha)}$ (partial execution state)	A new partition IC_θ, and a partial-execution state record of each suspended α is logged in $IC_{\theta(\alpha)}$
θ is lowered to θ^-	Invoke $TM^{auth}_{\theta^-}$. Authorized actions α under $TM^{auth}_{\theta^-}$ have their suspended state loaded from the corresponding $IC_{\theta(\alpha)}$	No action	$IC_{\theta(\alpha)}$, $\theta^- \leq \theta(\alpha)$, is cleared after each action α is restored

TM infrastructure uses the Bell-La Padula model then these requirements can be combined. For example, a user with clearance S (secret) can access a resource with classification S when the threat level is U (unclassified). If the threat level is raised to S, then access to the resource is suspended: only users with TS (top secret) clearance can access it. In Table 5.1 we illustrate the actions enforced by the Rollback Access mechanism.

5.3.6 Equivocated Sanitization

Data sanitization is the process of censoring/removing sensitive information from a document or medium so that it can be safely distributed. With classified information, this typically involves *redacting* the classified content of the document. For non-classified scenarios, there are two alternate versions for the definition of data sanitization. The first one involves the use of *anonymization*, *scrubbing*, *generation of gibberish*, *encryption*, and other techniques to purge the data of personally-identifiable or otherwise sensitive information in order to protect user privacy. The second version involves the removal of malicious data/code from user input in form submissions (e.g., on websites). Whichever definition is used, the goal of data sanitization is to prevent the adversary from accessing private or sensitive information.

Desanitization refers to the process of attempting to reverse the redaction process of sanitization. Given a sanitized document, desanitization can involve replacing the redacted portions with either incorrect or correct text, adding extra information, or other forms of integrity corruption. Malicious desanitization can sabotage a mission, mislead, and waste resources.

Automated data sanitization can be used as a graduated security mechanism for dynamic Trust Management. For example, anomalous user behavior can lower that user's threat clearance which may engage the redaction mode; then the file system/server will only provide a redacted file to that user. When the behavior of that user returns to normal, the redaction mode is disengaged. Theoretically this is equivalent to Rollback Access security, and provides no appreciable advantages to the existing work. However, this leads to a new security approach that focuses on *disinformation*, that we call *equivocated sanitization*.

Equivocated sanitization (ES) is a graduated response mechanism geared towards potential mass document leaks, and involves the generation of disinformation via a natural language generating engine and data redaction (partial or full). The objective is to mitigate damage from a potential leak attack by providing inaccurate data and/or disinformation to user(s) whose behavior is anomalous. Effectively, ES implements *controlled data sanitization*.

As with Rollback Access, ES can be controlled and managed at the TLC layer. It can be automatically generated with a natural language generation engine, given seed-parameters such as fake agency names, project titles, mission code-names, bogus operative code-names, etc. Natural language generators are known to produce imperfect results containing grammatically incorrect sentences, and this will allow raw output from the natural language generator to be easily detected. However redacted documents commonly have incorrect grammar, code-names, actions, and broken sentences. Thus one would expect that a document produced by: first adding to it text that is generated by a natural language generator and then, partially redacting the resulting document—called *post-processing redaction*, is indiscernible from a typical redacted document. Consequently, ES is capable of disseminating actionable disinformation and combating insider leaks and espionage.

However equivocated sanitization is applicable only to data/media resources. The key implementation issue here is the engage/disengage mechanism which differs if, (a) it is kept secret from the user base (which is unlikely to fool users for long), or (b) it is public. In the first case the supervisor is responsible for the timing of the engage mechanism (replacing the document by an obfuscated sanitized version), and the disengage mechanism. For this case, the state of suspended actions (fully or partially) is not stored in information compartments (and restored later), because we are mainly concerned with documents/media. In the second case the engage/disengage mechanism is automatic, and suspended actions get restored on disengagement.

Theoretically it is possible to include the offending user's account information into the natural language generation engine as a steganography uniquely-identifiable watermark, allowing the network to quickly identify and prosecute the offending party, should the leaked information be published.

ES is related to *chaffing* and *winnowing*, a cryptographic technique proposed by Ronald Rivest [38] for hiding data in plaintext without using encryption. The basic concept of this technique is quite simple: the sender obfuscates the original message (plaintext) with gibberish (while still in plaintext), and only the receiver is able to filter out (winnow) the original message from the data sent. With ES the gibberish is replaced by disinformation, and the message is partly or wholly redacted.

5.4 Mathematical Background

This section provides a brief mathematical overview and notation explanation. Markov chains (Sect. 5.4.1), Hidden Markov models (Sect. 5.4.5), and Bayesian approaches (Sect. 5.4.6) are discussed as tools for anomaly detection. Principal component analysis (Sect. 5.4.7) is discussed for feature discovery/selection.

5.4.1 An Introduction to Markov Chains

Definition 5.1 ([12, 32]). A discrete-time random process $X = \{X_1, X_2, \ldots\}$ is a *Markov chain* if it satisfies the Markov property:

$$\mathsf{Pr}(X_{t+1} = x_{t+1} | X_t = x_t) = \mathsf{Pr}(X_{t+1} = x_{t+1} | X_t = x_t, \ldots, X_1 = x_1),$$

where the values $x_{t+1}, x_t, \ldots, x_1$ belong to the state space $\mathcal{S}$.

The *transition probability* (aka *conditional discrete density function*)

$$p_{ij}(t, t+1) = \mathsf{Pr}(X_{t+1} = x_j \mid X_t = x_i)$$

is the probability that the state of the process at time $t+1$ will be x_j, given that at time t it was x_i. If the transitional probabilities do not depend on time, the Markov chain process is *time-homogeneous*. We will *not* be dealing with time-homogeneous Markov chains.

More generally, if we denote by $p_{ij}^{(n)}$ the transitional probability that the next state after n-steps will be x_j, given that the previous state was x_i, then it is easy to see that:

$$p_{ij}^{(n)} = \sum_{x_r \in \mathcal{S},\ 0<k<n}^{n} p_{ir}^{(k)} p_{rj}^{(n-k)}. \tag{5.2}$$

This formula is derived from the *Chapman-Kolmogorov* equations [17]. Using (5.2), one can easily calculate the n-step transition probability matrix P^n from the 1-step probability matrix,

$$P^1 = [p_{ij}^{(1)}] = \begin{pmatrix} p_{00} & p_{01} & \cdots & p_{0n} \\ p_{10} & p_{11} & \cdots & p_{1n} \\ \cdots & \cdots & \cdots & \cdots \\ p_{n0} & p_{n1} & \cdots & p_{nn} \end{pmatrix},$$

where for convenience we take $p_{ij} = p_{ij}^{(1)}$, the probability that the next state is x_j given that the previous state was x_i.

Markov chains can be modeled by finite state machines since all that is necessary to predict the next state is the current state. In many applications (in particular ours) random process models have memory. A Markov chain *of order m* predicts the next state based on the past m states.

5.4.2 Properties of Markov Chains

There are several key properties of Markov chains that require discussion. Markov chains can be decomposable or non-decomposable, periodic or aperiodic. The state space can be finite or countably infinite. These properties determine how transition probabilities are calculated, the calculation of P^n, as well as the convergence of P^n as $n \to \infty$ [32] (convergence is also referred to as the *long-run distribution*).

A Markov chain is said to be *decomposable* if it has a state from which it cannot reach another state in a finite number of steps. It is *non-decomposable* if from every one of its states it can reach any other state in a finite number of steps. Using our earlier notation, non-decomposable Markov chains satisfy: for every $x_i, x_j \in \mathcal{S}$ there is an integer $n \geq 1$ such that $p_{ij}^{(n)} > 0$. An important property of non-decomposable Markov chains is that all states are *homogeneous*. In particular, if one state is aperiodic then all states are aperiodic [17,32]. A Markov chain is *finite* if the number of its states is finite. For finite Markov chains the calculation of the transition matrix P^n is simple. However for Markov chains with a countably infinite number of states, obtaining P^n requires asymptotic analysis [17, 32]. We will not be dealing with countably infinite state spaces in this chapter.

5.4.3 Markov Chains of Order *m*

Definition 5.2 ([12]). A discrete-time random process $X = \{X_1, X_2, \ldots\}$ is a *Markov chain of order m* if:

$$\mathsf{Pr}(X_n = x_n \mid X_{n-1} = x_{n-1}, \ldots, X_1 = x_1) = \mathsf{Pr}(X_n = x_n \mid X_{n-1} = x_{n-1}, \ldots, X_{n-m} = x_{n-m})$$

for any integer $n \geq m$.

In our variable-threat application, the size of m is determined by the security policies of the TM system: when the threat level is high, the value of m should be raised; whereas for low threat levels m can be lowered. Additionally, m is influenced by the resources currently accessed by the user.

For Markov chains of order m, we can use the transition n-step probability given by Eq. (5.2) to compute the *temporal Markov distribution* $\mathsf{Pr}(X_t)$ for the system states at time t, given an initial distribution $\mathsf{Pr}(X_1)$. Markov chains of order m are best modeled by Turing machines, since they require the use of memory tape.

Table 5.2 A toy-example of a server-based Markov profiler for the day-to-day activity of a user

		Today's period of most activity			
		Morning	Afternoon	Evening	Late night
Yesterday's period of most activity	Morning	0.5	0.3	0.1	0.1
	Afternoon	0.6	0.2	0.1	0.1
	Evening	0.3	0.2	0.2	0.3
	Late night	0.1	0.3	0.2	0.4

In Table 5.2 we illustrate a toy example of a basic server-based Markov chain that profiles the day-to-day activity of a user. For this example the state space is:

$$\mathcal{S} = \{\text{Morning, Afternoon, Evening, Late Night}\}.$$

The values in the matrix correspond to transition probabilities for the periods of most activity. For example,

$$p_{(\text{afternoon, morning})} \; = \; \mathsf{Pr}(X_n = \text{Morning} \mid X_{n-1} = \text{Afternoon}) = 0.6,$$

means that the probability that the user interacted the most during the morning given that the previous day the user interacted the most in the afternoon is 0.6. While this example is not immediately useful for network security, it demonstrates how Markov chains model normal behavior.

5.4.4 The Markov Evolution Function

For dynamic Markov chains, the transition probabilities are updated by an *evolution function* upon an observed state change. Any state change can trigger the evolution. In our model evolution is triggered after a variable length time interval, and the previous process state is saved in a log so that it can be restored should the new profile prove problematic. This interval and the evolution function are crucial in maintaining the accuracy of a Markov analyzer.

The evolution of the process for a 1-step transition is given by:

$$\mathsf{Pr}(X_1 = x_j) \; = \; \sum_{x_r \in \mathcal{S}} p_{rj} \mathsf{Pr}(X_0 = x_r),$$

in terms of the earlier transition probabilities, where p_{rj} is the probability that the next state will be x_j given that the previous state was x_i. For an *n-step* transition we use the transition probabilities in Eq. (5.2).

Markov chains are commonly described by directed graphs in which the edges are labeled by the probabilities of going from one state to the other states. The following result is given without further discussion, to support our case for Markov profiling [17]:

Theorem 5.1. *Non-decomposable, aperiodic, Markov chains typically converge to a fixed profile. That is,*

$$\lim_{n\to\infty} p_{ij}^{(n)} = q_{ij},$$

regardless of the initial probability distribution.

This property is proven in [32].

5.4.5 Hidden Markov Models

Hidden Markov models (HMM) are Markov processes for which an observed event (output) depends on an unobserved or hidden state. Each hidden state has a probability distribution over the possible output events. Thus a chain of events generated by a HMM gives some information about the sequence of states.

Many researchers use Hidden Markov models for a vast range of behavioral analysis, such as speech [34], handwriting, music, networks, etc. [16, 20]. HMM capture processes in which hidden actions result in observable events. They can be considered as a special case of Markov chain processes. However with Markov processes there is no hidden state information, so the task of understanding and interpreting observed information is much easier. In particular Markov processes require fewer parameters to be setup, thus simplifying the training phase.

5.4.6 Bayesian Inference

Bayesian inference is a method by which some observation or evidence is used to: (a) calculate the probability that a hypothesis regarding an event may hold true with a degree of confidence, and (b) update the previously calculated probability for the event [12, 32, 40].

For anomaly detection applications, Bayesian inference describes the use of a learned or *prior* probability over a hypothesis, to determine the chance that hypothesis holds true given observed events. Specifically, for our applications the hypothesis is that the behavior of the entity generating the event is either normal or anomalous. The chance of the hypothesis holding true given the observed events and the prior probability, is termed the *posterior probability of the hypothesis* [12, 40]. This can also be seen as the confidence that the Bayesian application has in the given hypothesis.

Following the calculation of the posterior probability of the hypothesis, the prior probability is updated by using the following formula:

$$\mathsf{Pr}[H_j|X] = \frac{\mathsf{Pr}[X|H_j]\mathsf{Pr}[H_j]}{\mathsf{Pr}[X]},$$

where:

- H_j refers to hypothesis j;
- X refers to the observed event;
- $\mathsf{Pr}[X|H_j]$ refers to the conditional probability of X occurring when H_j is true;
- $\mathsf{Pr}[X]$ refers to the a priori probability of observing X under all hypothesis, as given by the following:

$$\mathsf{Pr}[X] = \mathsf{Pr}[X|H_1]\mathsf{Pr}[H_1] + \ldots + \mathsf{Pr}[X|H_n]\mathsf{Pr}[H_n].$$

One can model an access control system by a stochastic finite state machine $\mathcal{F}$ with state set $\mathcal{S} = \{s_j\}$. A Bayesian behavior model for anomaly detection is captured by the triple:

$$< \mathcal{S}, \mathcal{P}, \mathcal{T} >,$$

where $\mathcal{P} = \{\mathsf{Pr}(s_j|s_i), s_i, s_j \in \mathcal{S}\}$ is the set of transition probabilities and $\mathcal{T}$ a set of thresholds for the transition probabilities (used to determine anomalous behavior). In any realization of the system, the transition probabilities can only be approximated over a learning period, and then used for future behavior analysis, which will involve the thresholds in $\mathcal{T}$.

With this approach the Bayesian analyzer averages out over time the transient probabilities and then compares the average with the sampled behavior using the thresholds to decide whether the system behavior (or client behavior, if the analysis focuses on clients) is anomalous.

Without a *prior probability* updater, the Bayesian analyzer can be quite crude in its decisions, in the sense that actual behavior is compared with a fixed expected "normal" behavior. In fact any statistical method that is static will fail to adapt to changes in network behavior, and thus grow inaccurate.

5.4.7 Principal Component Analysis

Principal Component Analysis (PCA) is a mathematical procedure that is used to identify the principal feature among several features of an observed event [19]. It uses a linear orthonormal mapping to transform observations of selected features (trials) to a new coordinate system for which, the first coordinate describes the feature with the greatest variance—called the *principal component.*

We briefly describe this procedure for the case when m features of an event are monitored and there are n trials (measurements taken). Let X be a mean-centered $m \times n$ data set (matrix) whose rows correspond to the features and whose columns correspond to trials. The entries of X are the trial values. The *covariance* of the data set X is:

$$cov(X) = \frac{1}{n-1} XX^T.$$

$cov(X)$ is a symmetric $m \times m$ matrix whose (i,j)-th entry is the covariance between the i-th and j-th features when $i \neq j$, and the variance of the i-th feature when $i = j$. The constant $\frac{1}{n-1}$ is for normalization. The goal of PCA is to find an orthogonal transformation that diagonalizes $cov(X)$.

Let $A = XX^T$. A is a symmetric $m \times m$-matrix with $r \leq m$ orthonormal eigenvectors,[2] where r is the rank of A. If $r < m$, then we select a further $(m-r)$ orthonormal vectors from the nullspace[3] $null(A)$ of A to get an orthonormal basis,

$$\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_r, \ldots, \mathbf{v}_m$$

with corresponding eigenvalues

$$\lambda_1, \lambda_2, \ldots, \lambda_r, \ldots, \lambda_m [43].$$

We order the orthonormal basis so that the eigenvalues have decreasing order, with λ_1 the largest. Note that the last $(m-r)$ eigenvalues are 0 because they correspond to eigenvectors in the null space of A. Let

$$P = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_m \end{bmatrix}$$

be the matrix with rows the orthonormal eigenvectors. P is an orthonormal matrix (that is $PP^T = I$, and $det(P) = 1$). We shall show that the mapping P diagonalizes the covariance. Suppose that P takes X into $Y = PX$. Then,

$$cov(Y) = \frac{1}{n-1} YY^T = \frac{1}{n-1} (PX)(PX)^T = \frac{1}{n-1} P(XX^T)P^T = \frac{1}{n-1} PAP^T,$$

which is a diagonal matrix because its non-diagonal entries are:

$$\frac{1}{n-1} (\mathbf{v}_i A) \mathbf{v}_j^T = \frac{1}{n-1} \lambda_i (\mathbf{v}_i \cdot \mathbf{v}_j) = 0.$$

This solves the PCA problem: the principal component of the data set X is the eigenvector $\mathbf{v}_1$.

Another algebraic solution for the PCA problem can obtained by using the *singular value decomposition* (SVD). We do not discuss this procedure here because it is more mathematically involved, and refer the reader to [19].

[2] Vectors $\mathbf{u}, \mathbf{v}$ are orthonormal if $\mathbf{u} \cdot \mathbf{v} = \delta_{ij}$, where δ_{ij} is 1 for $i = j$ and 0 otherwise. The vector $\mathbf{u} \neq \mathbf{0}$ is an eigenvector of A if: $\mathbf{u}A = \lambda_u \mathbf{u}$, where λ_u is a constant called the *eigenvalue* of $\mathbf{u}$.

[3] $null(A) = \{\mathbf{x} \in A | \mathbf{x}A = \mathbf{0}\}$. The dimension of $null(A)$ is $(m-r)$ [43].

PCA is a powerful tool for analyzing statistical data. However it is computationally expensive, prohibitively so for real-time anomaly detection. For dynamic TM applications it is used to initialize the selection of the most significant features of statistical profilers, and then for resetting these features at given intervals.

5.5 Understanding Network Profiling

In Sect. 5.3 we presented a dynamic Trust Management infrastructure TM_{Θ} that uses a graduated response mechanism to protect system resources by monitoring anomalous behavior and temporarily disrupting granted accesses to mitigate system risks. In this section we consider two network application scenarios and show how these can be secured by using such a dynamic TM infrastructure.

5.5.1 *Scenario A: Defending an Enterprise Network Against Insider Privacy Attacks*

Recent high profile data leaks caused either by malicious insiders on closed networks, or by foreign based hackers have forced an enterprise/agency to deploy countermeasures to such attacks. In our first example we consider such a scenario and describe a graduated response threat management mechanism to secure it.

Operational environment. The network is set up so that security-critical resources (intelligence, banking info, etc.) are centralized on server clusters. Network administrators and security personnel have a reasonable level of physical access to network machines, to prevent tampering. Login requires strong authentication and access to resources is secured by providing the necessary credentials. A well defined Role-Based Access Control model is used and users are grouped into permission categories.

The approach. A role based dynamic TM model is established to monitor the behavior of users and groups of users (with the same role) with similar behavior, and establish a profile. For each user and profiled group a threat level is established. The threat levels operate independently, while the profiles evolve simultaneously.

The chief security officers have decided that two features, when analyzed, correlate strongly with various simulated leak attacks:

Feature 1A. The resource download/access rate from the server(s) of users.
Feature 2A. The client-side operating-system-call activity of users.

To defend the system against privacy attacks that exploit these features, two discrete-time Markov profilers are used. The first one monitors the file access behavior of a user, while the second one monitors the system-call activity rate of the user.

Table 5.3 A Markov profiler for monitoring a user's file access rate behavior with the server(s)

		Current file access rate			
		x_1	x_2	$\dots$	x_n
Previous file access rate	x_1	$p_{1,1}$	$p_{1,2}$	$\dots$	$p_{1,n}$
	x_2	$p_{2,1}$	$p_{2,2}$	$\dots$	$p_{2,n}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	x_n	$p_{n,1}$	$p_{n,2}$	$\dots$	$p_{n,n}$

Table 5.4 A Markov profiler for monitoring the client-side operating-system-call activity of a user

		Current operating system call			
		y_1	y_2	$\dots$	y_m
Previous operating system call	y_1	$p_{1,1}$	$p_{1,2}$	$\dots$	$p_{1,m}$
	y_2	$p_{2,1}$	$p_{2,2}$	$\dots$	$p_{2,m}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	y_m	$p_{m,1}$	$p_{m,2}$	$\dots$	$p_{m,m}$

Table 5.3 illustrates a profiler for the file access rate behavior of a user with the server(s), and addresses Feature 1A. In this table the values, $x_1, x_2, \dots, x_n$ are the possible rate-based events. For example x_1 can be 0–5 files per time-frame. Profiling for Feature 1A is server based.

Table 5.4 illustrates a profiler for the client-side operating-system-call activity of a user, and addresses Feature 2A. In this table $y_1, y_2, \dots, y_m$ are the monitored system calls. For example y_1 can be mmap(), y_2 can be open(), and so on. Various implementations of system-call monitoring applications are presented in [14]. The system does not profile every possible system call, and instead is limited to relevant system calls. Profiling for Feature 2A is client based and can be automatically coordinated by the operating system.

A training period is necessary for each profile, during which the users of each role are monitored, profiles are formed, and anomaly detection thresholds are established. Both feature profiles are discrete-time Markov chains and sampling is performed on a fixed time interval.

Alternatively a Hidden Markov model can be established to infer between normal and malicious users. The HMM identifies possible malicious users by checking the covariance between the two feature profiles and becomes high, in conjunction with anomalous readings from both features.

Threat level control. The TLC is responsible for processing the anomaly analyzer readings for both features. Thus it is configured to increase the threat level dial towards t^{θ}_{raise} when both features read anomalously, because of their correlation. Normal events decrease the dial towards t^{θ}_{lower}.

Policy selection. A key to effective graduated response lies within the overlying policy. In this scenario many policies are possible, but we will discuss the application of Rollback Access (Sect. 5.3.5) and Equivocated Sanitization (Sect. 5.3.6).

Rollback Access here can penalize a user by one clearance level, per threat level past the initial θ_0. It is triggered by raising the threat level and will cause an authorized file access/interaction to be barred/suspended (as explained in Table 5.1).

Equivocated Sanitization (ES) is applicable only for text-based resources, and when triggered causes *controlled data sanitization* and automated redaction. The key implementation issue here is the disengage mechanism, which differs if the mechanism is intended to be kept secret from the user base (which is unlikely to fool users for long), or publicly acknowledged as a countermeasure for malicious insiders. In the former case, the supervisor is responsible for the timing of the ES mechanism disengage (replacing the ES-manipulated documents in the user's possession with the genuine version); however, when the threat level subsides, ES will partially-disengage. In partial-disengage, ES will cease manipulating files but leave manipulated documents untouched. In the latter case when ES is publicly acknowledged, ES can be fully-disengaged automatically by the threat level layer.

5.5.2 Scenario B: Defending an Open Network Against Insider Privacy Attacks

This is based on a scenario discussed in [7]. The U.S. President through the Department of Homeland Security (DHS) and the Office of the Director of National Intelligence (DNI) have established a policy where counter-terrorism information is to be shared to the greatest extent possible. The point of the policy is to ensure that all participants in the national counter-terrorism effort are provided the most accurate and current information available. Meanwhile non-governmental organizations (NGOs) that are involved in a stabilization and humanitarian relief efforts in Orange Land are also provided with reasonable access to intelligence in order to safely avoid insurgent territory, recent improvised explosive device (IED) sightings, and firefight-prone areas. The Orange Land government is generally pro-west, but there are at least two factions within the government that have ties to terrorist organizations through their rhetoric and tribal affiliations.

Military and government networks need to be able to automatically facilitate such rapid information sharing of *fresh intelligence* without sacrificing the traditional adherence to the need-to-know information sharing policy.

Operational environment. A tactical wide area network is in place to support mission-critical information sharing. Resources such as recent IED maps, insurgent activity maps, and safe travel routes are provided on a need-to-know basis. Full disclosure of all resources is not permitted, as NGO personnel might have ties with insurgent or terrorist forces. The military provides time-lines for general NGO operations in order to ensure that non-military efforts are not caught up in military operations, that could result in casualties. Military liaison(s) are typically stationed at each NGO operational headquarters/network to ensure the prevention of physical tampering, unauthorized data sharing, and any other security-policy violations.

Table 5.5 A Markov profiler for monitoring a user's access rate behavior to files of type *Z*

		Current access rate to files of type *Z*			
		z_1	z_2	$\cdots$	z_n
Previous access rate for files of type *Z*	z_1	$p_{1,1}$	$p_{1,2}$	$\cdots$	$p_{1,n}$
	z_2	$p_{2,1}$	$p_{2,2}$	$\cdots$	$p_{2,n}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	z_n	$p_{n,1}$	$p_{n,2}$	$\cdots$	$p_{n,n}$

Table 5.6 A Markov profiler for monitoring a user's quantity access behavior to file type *Z*

		Current quantity access for files of type *Z*			
		w_1	w_2	$\cdots$	w_m
Previous quantity access for files of type *Z*	$\lvert w_1 \rvert$	$p_{1,1}$	$p_{1,2}$	$\cdots$	$p_{1,m}$
	w_2	$p_{2,1}$	$p_{2,2}$	$\cdots$	$p_{2,m}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	w_m	$p_{m,1}$	$p_{m,2}$	$\cdots$	$p_{m,m}$

Paramount to the success of the information sharing operation is the trust established between all organizations that information would be available to each group but that groups would not share between themselves the information. Thus an information channel is established for each organization that prevents cross-talk, yet allows for coordinated efforts.

The approach. The decision is to use a dynamic threat based TM infrastructure for the Orange Land government as well as for those NGOs that are overtly sympathetic to the insurgents. This system adaptively monitors the resource access behavior of users in NGOs, and establishes profiles. For each user, and role of users that share behavior profiles, a threat level is determined. The Military have decided that two features, when analyzed, correlate strongly with various simulated target-gathering attacks:

Feature 1B. The access *rate* of targeted files by users.
Feature 2B. The *quantity* of targeted files accessed by users.

Since the attacks target specific file types, such as IED maps, insurgent maps, safe-route maps, etc., Markov profilers are used to monitor user behavior for each file type and each feature. The profilers will provide targeted information-gathering, which is forensically useful for pinpointing malicious insiders within trusted organizations. Profiles can be templated beforehand, given the similarity between user roles within NGO operations, reducing deployment time for the network.

Table 5.5 illustrates a typical profiler for monitoring access rates of files of a given type Z. The values $z_1, z_2, \ldots, z_n$ are the possible rate based events. For example z_1 can be 0–5 files per time-frame. Profiling for Feature 1B is server based. Table 5.6 illustrates a profiler for monitoring the quantity of files of type Z accessed by a user,

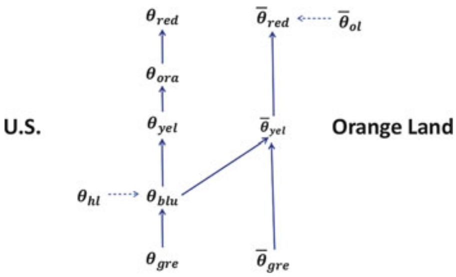


Fig. 5.2 A threat level structure for Scenario *B*: when the threat level in Orange Land is severe ($\bar{\theta}_{red}$) then access in that domain is restricted to "vanilla", while in Home Land it is less restrictive (θ_{blue})

and addresses Feature 2B. In this table $w_1, w_2, \ldots, w_m$ are the monitored quantities of files of type Z accessed by a user. For example w_1 can be 0–10 files accessed over a given time period. Profiling for Features 2A, 2B is server based.

As in the previous scenario a training period is needed to establish the access rate/quantity of targeted file types by users. During this period profiles are formed and anomaly detection thresholds are established. Sampling is performed on a fixed time interval.

Threat level interaction. The threat level controller (TLC) is responsible for processing the anomaly analyzer readings for both features. Thus it is configured to increase the threat level towards t^{θ}_{raise} when both features read anomalously, because of their correlation. Anomalies from single features move the dial positively to a lesser extent. Meanwhile, normal events move the threat level dial negatively.

In Fig. 5.2 we illustrate the flows of a threat level structure Θ appropriate for Scenario B. Θ is a partially ordered set with two linear suborders corresponding to the domains Home Land and Orange Land, respectively. Each domain has its own threat level, θ_{hl}, θ_{ol}, respectively, and the threat flows are linked by the requirement that:

$$\bar{\theta}_{ol} \succeq \bar{\theta}_{yel} \quad \Rightarrow \quad \theta_{hl} \succeq \theta_{blu}.$$

That is, when the threat level in Orange Land is "*elevated*" then in the U.S. the threat level must be "*guarded*", or higher.

Policy selection. In this scenario Rollback Access (RA) control is most applicable. RA here may penalize a user by one clearance level, per threat level past the initial θ_0. It is triggered when the threat level is raised, and will suspend access to resource that are at risk.

Conceptually, when users significantly deviate from the normal behavior for users of that role, the system engages in graduated response via the threat level rising. In this scenario, users attempting to download significantly more files than is typical for their role will trigger this defense mechanism, effectively enforcing the traditional need-to-know policy. This mechanism can disengage automatically when the threat level subsides, at the end of the current intelligence cycle, or after a fixed interval (e.g., 24 h). Should a sufficient number of users from any single NGO trigger threat level increases, it is possible to raise the network's threat level for the entire domain of that NGO—reflecting an overall decrease in trust.

5.6 Conclusion

In this chapter we presented a dynamic TM infrastructure for network profiling that will resist insider attacks and 0-day attacks. We discussed a threat level control architecture designed for graduated security mechanisms such as Rollback Access and Equivocated Sanitization. These mechanisms are geared towards combating and mitigating damage caused by insider attacks through the use of network profiling, and filtering out significant anomalies that are that are more likely to indicate a potential threat, from the static-like anomalies.

We also discussed the mathematical tools needed to capture network profiling and graduated response mechanisms that respond dynamically to insider threats and 0-day attacks.

Finally we presented two network application scenarios involving insider attacks and described a dynamic TM infrastructure, its profilers and a threat level controlling mechanism, that will tolerate such attacks.

Acknowledgements This material is based in part upon work supported by the National Science Foundation under Grant No. DUE 1027217.

References

1. M. Abadi, M. Burrows, B. Lampson, and G. Plotkin. A calculus for access control in distributed systems. In *Advances in Cryptology - CRYPTO '91: 11th Annual International Cryptology Conference*, pages 1–23. LNCS 576, 1991.
2. A. W. Appel and E. W. Felten. Proof-carrying authentication. In *6th ACM conference on Computer and Communications Security*. ACM, 1999.
3. D. Balfanz, D. Dean, and M. Spreitzer. A security infrastructure for distributed Java applications. In *21st IEEE Symposium on Security and Privacy*, 2000.
4. David Elliott Bell and Leonard J. La Padula. Secure Computer Systems: Mathematical Foundations. Technical report, MITRE Corporation, Bedford, Mass, 1973. MTR-2547.
5. M. Blaze, J. Feigenbaum, and A. D. Keromytis. *KeyNote: Trust management for public-key infrastructures*. 1999.
6. M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized trust management. In *Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on*, pages 164–173, may. 1996.
7. Mike Burmester, Prasanta Das, Martin Edwards, and Alec Yasinsac. Multi-domain Trust Management in Variable Threat Environments Using rollback-access. In *Proc. Military Communications Conference (MILCOM 2008)*. IEEE, 2008.
8. Mike Burmester, Prasanta Das, Martin Edwards, and Alec Yasinsac. Multi-domain Trust Management in Variable Threat Environments—a user-centric model. In *Proc. Military Communications Conference (MILCOM 2009)*. IEEE, 2009.
9. Yang-Hua Chu, Joan Feigenbaum, Brian LaMacchia, Paul Resnick, and Martin Strauss. REFEREE: trust management for Web applications. *Computer Networks and ISDN Systems*, 29(8–13):953–964, 1997. Papers from the Sixth International World Wide Web Conference.
10. D.E. Denning. An Intrusion-Detection Model. In *IEEE Transactions on Software Engineering*, volume 13, Issue:2, pages 222–232, Februrary 1987.
11. D. Endler. Intrusion detection Applying machine learning to Solaris audit data. In *Proceedings of the Computer Security Applications Conference*, 1998.

12. W. Feller. *An Introduction to Probability Theory and its Applications*. John Wiley & Sons, 1968.
13. D.F. Ferraiolo and D.R. Kuhn. Role Based Access Control. In *15th National Computer Security Conf*, pages 554–563, Oct 13–16. 1992.
14. Stephanie Forrest, Steven Hofmeyr, and Anil Somayaji. The Evolution of System-Call Monitoring. In *ACSAC '08: Proceedings of the 2008 Annual Computer Security Applications Conference*, pages 418–430, Washington, DC, USA, 2008. IEEE Computer Society.
15. Stephanie Forrest, Steven A. Hofmeyr, Anil Somayaji, and Thomas A. Longstaff. A Sense of Self for Unix Processes. In *SP '96: Proceedings of the 1996 IEEE Symposium on Security and Privacy*, page 120, Washington, DC, USA, 1996. IEEE Computer Society.
16. Sachin Shetty Gayathri Shivaraj, Mia Song. A Hidden Markov Model Based Approach to Detect Rogue Access Points. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1–7, Piscataway, NJ, USA, 2008. IEEE.
17. Gary D. Hachtel, Enrico Macii, Abelardo Pardo, and Fabio Somenzi. Markovian Analysis of Large Finite State Machines. *IEEE Transactions on CAD*, 15:1479–1493, 1996.
18. Amir Herzberg, Yosi Mass, Joris Michaeli, Yiftach Ravid, and Dalit Naor. Access Control Meets Public Key Infrastructure, Or: Assigning Roles to Strangers. *Security and Privacy, IEEE Symposium on*, 0:2, 2000.
19. I. T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics, 2002.
20. Shrijit S. Joshi and Vir V. Phoha. Investigating hidden Markov models capabilities in anomaly detection. In *ACM-SE 43: Proceedings of the 43rd annual Southeast regional conference*, pages 98–103, New York, NY, USA, 2005. ACM.
21. Y. F. Jou, F. Gong, C. Sargor, S. F. Wu, H.C. Chang, and F. Wang. Design and Implementation of a Scalable Intrusion Detection System for the Protection of Network Infrastructure. In *DARPA Information Survivability Conference and Exposition*, 2000.
22. Justin Lewis Balthrop. RIOT: A Responsive System for Mitigating Computer Network Epidemics and Attacks. Master's thesis, University of New Mexico, 2005.
23. Micki Krause and Harold F. Tipton. *Handbook of Information Security Management*. CRC Press LLC, Auerbach Publications, 1997.
24. Mihails Kulikovs and Ernests Petersons. Real-Time Traffic Analyzer for Measurement-Based Admission Control. *Advanced International Conference on Telecommunications*, 0:72–75, 2009.
25. B. Lampson, R. Rivest, B. Thomas, and T. Ylonen. SPKI Certificate Theory. 2007.
26. Ninghui Li, Benjamin N. Grosof, and Joan Feigenbaum. Delegation logic: A logic-based approach to distributed authorization. *ACM Trans. Inf. Syst. Secur.*, 6(1):128–171, 2003.
27. Ninghui Li and John C. Mitchell. RT: A Role-based Trust-management Framework, 2003.
28. P. G. Neumann and P. A. Porras. Experience with EMERALD to Date. In *1st SENIX Workshop on Intrusion Detection and Network Monitoring*, 1999.
29. NIST. *Guide to Intrusion Detection and Prevention Systems (IDPS)* . Recommendations of the National Institute of Standards and Technology. Special Publication 800–94, Gaithersburg, MD. February 2007.
30. NIST. A survey of access control models. Technical report, 2009.
31. OASIS. Oasis extensible access control markup language (xacml) tc, 2010. Developed by the P1363 Working Group.
32. E. Parzen. *Stochastic Processes*. Holden-Day, 1962.
33. Vern Paxson. Bro: a system for detecting network intruders in real-time. *Computer Networks*, 31(23–24):2435–2463, 1999.
34. Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.
35. W. O. Redwood. APECS: A Dynamic Framework for Preventing and Mitigating Theft, Loss, and Leakage of Mission Critical Information in Trust Management Networks. Master's thesis, Florida State University, FL, USA, 2010.

36. W. O. Redwood and M. Burmester. Markov anomaly modeling for Trust Management in variable threat environments. In *ACM-SE 2010: Proceedings of the 48th annual Southeast regional conference*, New York, NY, USA, 2010. ACM.
37. R. L. Rivest and B. Lampson. *SDSI A simple distributed security infrastructure*. http://theory.lcs.mit.edu/cis/sdsi.html., 1996.
38. Ronald L. Rivest. Chaffing and Winnowing: Confidentiality without Encryption. Technical report, MIT Lab for Computer Science, March 18 1998.
39. R. Sandhu, E.J. Coyne, H.L. Feinstein, and C.E. Youman. Role-Based Access Control Models. In *IEEE Computer (IEEE Press) 29*, pages 38–47, August 1996.
40. S. Scott. *A Bayesian paradigm for designing intrusion detection systems*. Computational Statistics and Data Analysis, 2003.
41. R. Sekar, A. Gupta, J. Frullo, T. Shanbhag, A. Tiwari, H. Yang, and S. Zhou. Specification-based anomaly detection: a new approach for detecting network intrusions. In *Proceedings of the 9th ACM conference on Computer and communications security*, CCS '02, pages 265–274, New York, NY, USA, 2002. ACM.
42. E. H. Spafford. Virus. *Encyclopedia of Software Engineering*, 1994.
43. G. Strang. *Linear Algebra and its Applications*. Thomson Learning, 1988.
44. Stephen Weeks. Understanding Trust Management Systems. In *SP '01: Proceedings of the 2001 IEEE Symposium on Security and Privacy*, page 94, Washington, DC, USA, 2001. IEEE Computer Society.
45. Nong Ye. A Markov Chain Model of Temporal Behavior for Anomaly Detection. In *Proceedings of the 2000 IEEE Workshop on Information Assurance and Security*, pages 171–174. IEEE, 2000.
46. P. Zimmerman. *The Official PGP User's Guide*. MIT Press, Cambridge, 1995.

Chapter 6
Security Issues in Link State Routing Protocols for MANETs

Gimer Cervera, Michel Barbeau, Joaquin Garcia-Alfaro, and Evangelos Kranakis

Abstract In link state routing networks, every node has to construct a topological map through the generation and exchange of routing information. Nevertheless, if a node misbehaves then the connectivity in the network is compromised. The proactive Optimized Link State Routing (OLSR) protocol has been designed exclusively for Mobile Ad Hoc Networks (MANETs). The core of the protocol is the selection of Multipoint Relays (MPRs) as an improved flooding mechanism for distributing link state information. This mechanism limits the size and number of control traffic messages. As for several other routing protocols for MANETs, OLSR does not include security measures in its original design. Besides, OLSR has been extended to address a number of problems in MANETs. For example, Hierarchical OLSR (HOLSR) has been proposed to address scalability and Multipath OLSR (MP-OLSR) to address fault tolerance. However, these OLSR extensions can be affected either by inheriting or adding new security threats. In this chapter, we present a review of security issues and countermeasures in link state routing protocols for MANETs.

6.1 Introduction

The design of a secure and efficient routing protocol for Mobile Adhoc Networks (MANETs) is a challenging problem. Routing protocols proposed for MANETs assume a trusted and cooperative environment. Therefore, several mechanisms to

G. Cervera (✉) • M. Barbeau • E. Kranakis
School of Computer Science, Carleton University, Ottawa, ON, Canada, K1S 5B6
e-mail: gcevia@scs.carleton.ca; barbeau@scs.carleton.ca; kranakis@scs.carleton.ca

J. Garcia-Alfaro
Institut Telecom, Telecom Bretagne Cesson-Sevigne, 35576, France
e-mail: joaquin.garcia-alfaro@acm.org

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_6,

enhance security in MANETs have been proposed. The proactive Optimized Link State Routing (OLSR) [10] protocol has been designed exclusively for MANETs. The core of the protocol is the concept of Multipoint Relay (MPR). A valid MPR set, is defined as a subset of one-hop neighbors, such that all two-hop neighbors are covered through at least one node in the MPR set. In OLSR, every node has to select a valid MPR set. This mechanism allows to flood the network with control traffic information. OLSR comprises Hello and Topology Control (TC) messages. Every node periodically generates Hello messages. Within each Hello message a node reports its one-hop neighbors. Receiver nodes learn about its one and two hop neighbors. TC messages are used to discover nodes at more than two hops away. TC messages are generated and retransmitted exclusively by the MPRs. Unlike other link state routing protocols (e.g., OSPF [24]), the MPRs report partial link state information. Therefore, the MPR mechanism reduces the size and amount of control traffic information flooded in the network.

OLSR is defined in RFC 3626 [10]. A second version of the protocol, i.e., OLSRv2, is presented by Clausen et al. as an Internet-Draft in [11]. OLSRv2 implements the same basic mechanisms and algorithms for distributing control traffic (i.e., MPR-based flooding). As many other routing protocols for MANETs, OLSR and OLSRv2 are not secure by design. The selection of the MPRs and exchange of topology control information are important vulnerability targets. In this context, a malicious node is defined as a node that interrupts the flooding of control traffic information or does not obey the rules of the protocol. The terms: malicious, misbehaving, attacker and intruder are equivalent. Therefore, several authors proposed countermeasures to prevent or mitigate security threats in link state routing protocols for MANETs. For instance, in [2, 25, 26], Raffo et al., reviewed vulnerabilities in OLSR. In [16, 17], Clausen et al., studied security risk in OLSRv2. The authors proposed cryptographic mechanisms to enhance: integrity, confidentiality, reliability and service availability (fault-tolerance). Countermeasures to secure OLSR can be classified in two categories: cryptographic mechanisms to avoid impersonation or replay attacks, and Intrusion Detection Systems (IDS) [2] to prevent altered information from an authenticated node. Nevertheless, cryptographic models are challenging because in MANETs there is no centralized authority. The network performance drops due to additional computation. Reputation models or IDS mechanisms are designed to detect malicious behavior. Nevertheless, they increase the network traffic and need time to detect misbehaving nodes. Additionally, when a malicious behavior is detected, an efficient method to report untrusted nodes is needed. Moreover, *flooding disruption* [8] attacks can be perpetrated in networks with cryptographic capabilities. For instance, if a node refuses to retransmit TC messages on behalf of other nodes (e.g., to save energy), then the connectivity is disrupted.

In this chapter, we present a review of security issues in OLSR networks, existing solutions and our proposed countermeasures. In addition to OLSR, we review the Hierarchical OLSR (HOLSR) [30] protocol proposed by Villasenor et al. to address scalability and the Multipath OLSR (MP-OLSR) [33–36], proposed by Yi et al., to address security, fault tolerance and reliability. This chapter is based on the work presented in [7–9]. In [7], we analyzed the effect of control traffic attacks in OLSR networks and the selection of MPR sets with additional coverage to mitigate their effect. The MPR selection with additional coverage is presented in RFC 3626 [10], we name it k-Covered-MPR selection. However, additional coverage reduces the performance of the network due to additional control traffic information (i.e., TC messages). We proposed a k-Robust-MPR selection. In a k-Robust-MPR selection a node selects, when possible, $k+1$ disjoint MPR sets to guarantee that even if k of the selected MPR sets become invalid, the remaining set is still a valid MPR set. Our proposed MPR selection offers equivalent protection against control traffic attacks but reducing the overhead generated by additional control traffic information.

In [8], we presented a taxonomy of flooding disruption attacks and their effect in HOLSR networks. HOLSR uses TC messages for intra-cluster communications and implements Hierarchical TC (HTC) messages for inter-cluster communications. HOLSR implements the MPR flooding mechanism for distributing control traffic information. HTC messages are flooded exclusively by the MPRs. Therefore, the inter-cluster communications are also affected by flooding disruption attacks. In [8], we proposed to mitigate the effect of the attacks against HTC messages by selecting MPR sets with additional coverage (i.e., k-Robust-MPR and k-Covered-MPR selections). Additionally, the cluster formation phase in hierarchical OLSR networks can be disturbed. In [9], we presented an algorithm based on hash chains to enforce the cluster formation phase in HOLSR networks. In HOLSR, Cluster ID Announcement (CID) messages are implemented to organize the network in clusters. A misbehaving node may maliciously alter mutable fields (e.g., hop count) in CID messages to unbalance the distribution of nodes in clusters. Our solution allows a node to detect and discard invalid CID messages. Our algorithm can be implemented in other hierarchical approaches that use messages with mutable fields to organize the network in clusters. Finally, we analyze vulnerabilities in multipath OLSR-based networks. MP-OLSR is based on the MPR flooding mechanism to distribute control traffic information in the network. The construction of multiple paths in MP-OLSR has two phases: topology discovery and route computation. In the first phase, the nodes obtain information about the network topology through the exchange of Hello and TC messages. In the second phase, the nodes compute multiple paths to a particular destination in the network based on the information gathered during the first phase. These two phases are affected by flooding disruption attacks. Additionally, MPRs report partial link state information. Therefore, MP-OLSR nodes only acquire a partial view of the network. We analyze how the construction of multiple paths in MP-OLSR networks is affected by flooding disruption attacks and incomplete view of the network topology.

We describe different link state routing protocols for MANETs, their specific vulnerabilities and proposed countermeasures. The chapter is organized as follows: in Sect. 6.2, we review the OLSR protocol, flooding disruption attacks and related work. HOLSR, other OLSR-based hierarchical approaches and their vulnerabilities are described in Sect. 6.3, MP-OLSR and its security risks are presented in Sect. 6.4 and finally, Sect. 6.5 concludes the chapter.

6.2 Optimized Link State Routing (OLSR)

This section presents an overview of the OLSR protocol and its vulnerabilities. OLSR is a proactive routing protocol designed for MANETs. The core of the protocol is the selection, by every node, of MPRs among their one-hop neighbors. The MPR set is selected such that all two-hop neighbors are reachable through at least one MPR. Figure 6.1 compares the MPR mechanism and classical flooding. In Fig. 6.1a, control traffic information is retransmitted by all the one-hop neighbors. In Fig. 6.1b, control traffic information is retransmitted exclusively by the MPRs. This optimization improves the network performance by reducing the size and number of control traffic messages in the network. OLSR is defined in RFC3626 [10]. A second version of the protocol, i.e., OLSRv2, is presented by Clausen et al. in an Internet-Draft [11]. OLSRv2 uses and extends: the MANET Neighbor Discovery Protocol (NHDP) [14], RFC5444 – Generalized MANET Packet/Message Format [15], RFC5497 – Representing Multi-Value Time in MANETs [12] and RFC5148 – Jitter Considerations in MANETs [13] (optional). These protocols were all originally created as parts of OLSRv2, but have been specified separately for wider use. OLSRv2 retains the same basic mechanisms and algorithms for distributing control traffic (i.e., MPR-based flooding) but provides a more efficient signaling framework and implements some message simplifications.

OLSR nodes flood the network with link state information messages. The link state information is constructed by every node and involves periodically sending Hello and TC messages. This information is used to determine the best path to every destination in the network. Due to the proactive nature, the routes are immediately available when needed. The OLSR protocol is based on hop by hop routing, i.e., each routing table lists, for each reachable destination, the address of the next node along the path to that destination. To construct a topology map, every node implements a topology discovery mechanism leveraging the periodic exchange of control traffic messages. Topology discovery includes: link sensing, neighbor detection and topology sensing. In the first phase, every node populates its local link information base (link set) and establishes communication with their symmetric neighbors, i.e., nodes with bidirectional communication. This phase is exclusively concerned with the OLSR interface addresses and ability to exchange packets between such OLSR interfaces. During the neighbor detection phase, every node

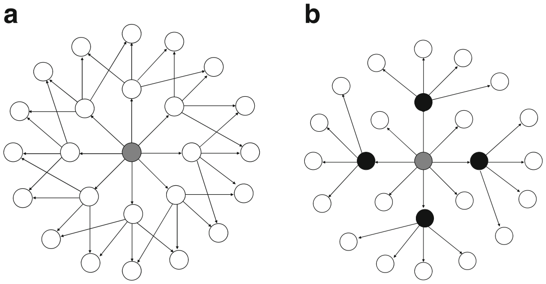


Fig. 6.1 MPR based mechanism against the classical flooding. Consider *gray nodes* as the originators of a TC message and *black nodes* as MPRs. (**a**) Classical flooding. (**b**) MPR mechanism

populates its neighborhood information base (i.e., one-hop and two-hop neighbor set). The link sensing and neighbor detection phases are based on the periodic exchange of Hello messages. Hello messages are solely transmitted to one-hop neighbors. In every Hello message, the nodes report their one-hop neighbors. This information allows every node to construct and maintain neighbor tables, as well as to select its MPR set. In the neighbor table, each node records the information about the one-hop neighbor link status (i.e., unidirectional, bidirectional or MPR), with this information every node builds its MPR selector set, i.e., the neighbors that selected that node as their MPR. OLSR detects and eliminates duplicate messages. OLSR keeps track of recently received messages by using a duplicate table. Therefore, when a message has been received and included in the duplicate table, the payload is not examined and the message is automatically discarded.

Topology sensing is achieved through the exchange of TC messages. TC messages are generated and retransmitted exclusively by the MPRs. TC messages have a Time-to-Live (TTL) field that is decremented every time an MPR retransmits the message. These messages allow each node to construct its topology table and to declare its MPR selector set. A TC message contains the MPR selector set of its originator. A node that has an empty MPR selector set does not send or retransmit any TC message. An MPR forwards a message only if it comes from a node in its MPR selector set (i.e., a source-dependant mechanism). This forwarding algorithm is defined in RFC 3626 [10]. Using the information from TC messages, each node maintains a topology table where each entry consists of:

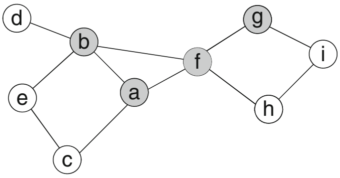
- An identifier of a possible destination, i.e., an MPR selector in a TC message,
- An identifier of a last-hop node to that destination, i.e., the originator of the TC message, and
- An MPR selector set sequence number [21].

It implies that a possible destination (i.e., an MPR selector) can be reached through the originator of the TC message. If there is an entry in the topology table whose last-hop address corresponds to the originator of a new TC message and the MPR selector set sequence number is greater than the sequence number in the received message, then the new message is discarded. Routing tables are constructed using the information from the one-hop neighbor, two-hop neighbor and topology tables.

OLSR implements two optional messages: Multiple Interface Declaration (MID) and Host and Network Association (HNA). They are exclusively retransmitted by the MPRs following the default forwarding algorithm defined in RFC 3626 [10]. MID messages are used to declare the presence of multiple interfaces on a node. HNA messages are employed to inject external routing information into an OLSR network and provide connectivity to nodes with non-OLSR interfaces (e.g., Internet). MID messages are implemented in a network with multiple interface nodes. Additional information is necessary in order to map interface addresses to main addresses. In OLSR, the main address is defined as the OLSR interface address. A node with multiple interfaces must generate periodically MID messages announcing all its interfaces to other nodes in the network. Thus, every node in an

Table 6.1 Summary of control traffic messages in OLSR networks. MID and HNA messages are optional

Messages	Generated by	Retransmitted by	Reported information
Hello	Every node	N/A	One-hop neighbors
TC	MPRs	MPRs	MPR selector set
MID	Nodes with more than one interface	MPRs	All available interfaces
HNA	Nodes with external access	MPRs	External routing information



MPRs	Selector Set
a	b,c,f
b	a,e,f,d
f	a,b,h,g
g	f,i,

Fig. 6.2 Example of an OLSR network

OLSR network will associate multiple interfaces to a node's main address. Nodes with just one interface do not generate MID messages and their main address is the OLSR interface address. A node with several interfaces, where only one of them is participating in an OLSR network must not generate MID messages. Upon receiving a MID message, the information is stored in an Interface Association table. This information is used to construct the routing tables. When a node misbehaves and does not retransmit TC, HNA or MID messages, the proper construction of the routing tables is compromised. Table 6.1 presents all the messages implemented in OLSR. In summary, the network topology discovery process is performed as follows:

1. First, every node periodically generates Hello messages to advertise itself and establish bidirectional links with its one-hop neighbors. Hello messages are not retransmitted. Figure 6.2 shows an example of an OLSR network. Node a includes nodes b,c and f in its one-hop neighbor set after exchanging Hello messages and establishing bidirectional links.
2. In subsequent Hello messages, every node reports its one-hop neighborhood. Receiver nodes identify their two-hop neighbors and compute their MPR set. In Fig. 6.2, nodes d,e,g and h are included in node a's two-hop neighbor table. Node a selects nodes b and f as its MPRs. Nodes a,b,f and g are selected as MPRs.
3. Nodes report their MPR set within their following Hello messages. If the receiver node was selected as an MPR, then it includes the sender node in its selector set, e.g., node b includes a in its selector set.
4. Nodes with a non empty selector set periodically generate TC messages advertising all nodes within their selector set. TC messages are retransmitted exclusively by the MPRs. To reach nodes more than two hops away, node a depends on the TC messages generated by all the MPRs. For instance, node g must periodically generate TC messages advertising its selector set, i.e., nodes f and i. TC messages generated by node g are retransmitted exclusively by nodes f, a and b.

5. When a node receives a TC message, it includes the contained information in its topology table. In Fig. 6.2, after receiving TC messages from node *g*, node *a* identifies node *g* as the last hop to reach node *i*. Note that node *b* receives TC messages from nodes *a* and *f*. However, node *b* stores the recently received TC messages in its duplicate table and discards future copies of the same message.
6. Finally, routing tables are constructed using information from the one-hop and two-hop neighbors and the topology table. Every node executes the Dijkstra's algorithm to obtain the shortest path to every other node more than two hops away. For instance, to reach node *i*, node *a* constructs a path trough nodes *f* and *g*. The shortest path to reach every other node in the network is always composed by MPRs. For example, to reach node *d*, node *i* constructs a path composed by nodes *g*, *f* and *b*.
7. Routing tables include the next node and number of hops to reach every other node in the network. Node *i* stores in its routing table only the next hop to reach node *d* (i.e., node *g*) and the number of hops (i.e., four hops). Thanks to the MPR mechanism, the nodes are aware of every other node in the network but some links are never advertised. For instance, node *a* never receives information about the link between nodes *h* and *i*, or between nodes *e* and *c*.
8. Optionally, a node with more than one interface generates MID messages. A node with access to an external network generates HNA messages. Information contained in MID and HNA messages is loaded in routing tables.

6.2.1 Related Work

As many other routing protocols for MANETs, OLSR is not secure by design. Vulnerabilities in OLSR have been studied extensively. For instance, in [2], Adjih et al. present security risks in the OLSR protocol and countermeasures based on cryptographic mechanisms to secure the protocol with or without compromised nodes in the network. The authors claim that an efficient securing mechanism should ensure the network integrity even when the network is subject to attacks that interrupt the connectivity. In [16, 19] Clausen and Herberg review security issues in OLSRv2. The authors analyze the basic algorithms that constitute the OLSRv2, and identify possible vulnerabilities and attacks.

Several authors have contributed with cryptographic mechanisms to secure OLSR. Cryptographic mechanism are proposed to enforce: integrity, authentication and confidentiality. Thus, public-key encryption is used for confidentiality, digital signature for integrity of the messages and digital certificates for authentication. However, the implementation of a Public Key Infrastructure (PKI) in MANETs is difficult due to the lack of a central authority (CA). Additionally, the efficient distribution of public and private keys is a challenging problem. Timestamps are implemented with digital signatures to assure the freshness of the message. However, time synchronization is difficult to achieve particularly in MANETs.

According to Adjih et al. [2], a *cryptographic capable* node is a node that has received valid keys to sign and verify messages. A misbehaving node can be also a cryptographic capable node. For example, in Fig. 6.2, node *g* may decide not to forward TC messages to node *i* or refuse to select an MPR set. In both cases, the connectivity of the network is compromised. Intrusion Detection Systems (IDS) are implemented to analyze malicious behavior in the network. However, once a misbehaving node has been detected, an efficient reputation model is needed to convey to other nodes the results observed by the IDS. In this chapter, we focus on attacks that prevent a node to acquire a complete network topology map. These attacks can be launched even in networks with cryptographic capabilities. In Sect. 6.2.2, we review them more precisely. In the following, we present some contributions to secure the OLSR protocol. We classify them in cryptographic mechanisms and IDS systems.

6.2.1.1 Cryptographic Mechanisms

In this section, we describe proposed solution based on cryptographic mechanisms. In [17], Clausen et al. present a digital signature mechanism for authentication and authorization in OLSRv2. The authors introduce the concept of admittance control for OLSRv2 networks and suggest a security extension based on digital signatures. They compare several standard digital signature algorithms such as: RSA, DSA, ECDSA and HMAC. The goal is to enable trusted nodes and to disable non-trusted nodes from participating in the control message exchange between routers, thereby providing a mode-of-operation similar to traditional mechanism employed for preserving network integrity in routed networks. Additionally, a performance study of the propose extension is presented to quantify the impact of increased control traffic overhead and increased message generation as well as processing time. The authors observed that HMAC requires significantly less time than ECDSA, DSA and RSA for generating a message signature. For the verification of a message signature, HMAC likewise spends substantially less time than ECDSA and DSA, whereas RSA is close to HMAC. Verification of RSA signatures has much greater overhead but is faster than both ECDSA and DSA.

In [26], Raffo et al., examined security issues related to the OLSR protocol, and enumerate a number of possible attacks against the integrity of the OLSR routing infrastructure. In particular, authors study attacks when a mechanism of digitally signed routing messages is deployed and an attacker may have taken control over trusted nodes. Their solution is based on inclusion of the geographical position of the sending node in control messages and on evaluation of reliability of links; this is accomplished using a GPS device and a directional antenna embedded in each node. Signatures with timestamps are sufficient to thwart attacks such as incorrect traffic generation and incorrect traffic relaying, when only legitimate nodes can sign control packets. Adding the node location in signature messages allows the network to avoid wormhole attacks and false messages generated by misbehaving nodes.

Raffo also presented in his Ph.D. thesis [25], a classification of possible attacks in OLSR networks. The author proposed a security architecture based on digital signatures. Additionally, the author proposed other techniques such as: reuse of previous topology information to validate the actual link state, cross-check of advertised routing control data with the node's geographical position, and intra-network misbehavior detection and elimination via flow coherence control or passive listening. Countermeasures in case of compromised nodes are also considered. Furthermore, the author assesses practical problems concerning the choice of a suitable symmetric or asymmetric cipher, alternatives for the algorithm of cryptographic key distribution, and the selection of a method for signature time stamping. In summary, the author presented an outline of different signature algorithms. The author suggested the study and design of better cryptographic algorithms, i.e., algorithms that use a smaller signature size to reduce computation complexity would increase the suitability of his proposed OLSR security architectures.

In [22], Khakpour et al., aboarded the access control problem in MANETs. The authors proposed a hierarchical distributed AAA (Authentication, Authorization, and Accounting) architecture for proactive link state routing protocols. This proposal contains a lightweight and secure design of an overlay authentication and authorization paradigm for mobile nodes as well as a reliable accounting system to enable operators to charge nodes based on their connection time. The authors also suggest a hierarchical distributed AAA server architecture with a resource and location aware election mechanism. Moreover, this proposal mitigates the OLSR security issues and eventually defines a node priority-based quality of service. The design of the architecture targets a minimum signaling overhead as well as calculation cost. In fact, different tasks are fairly distributed among distributed AAA servers. The calculation cost and overhead signaling is trivial compared to OLSR signaling and routing computations.

6.2.1.2 Intrusion Detection Systems

In this section we describe proposed solutions based on Intrusion Detection Systems (IDS). In [1], Abdellaoui and Robert, proposed the SU-OLSR protocol (SU for suspicious) to prevent attacks against OLSR-based routing protocols. In SU-OLSR the MPR selection is based on the trustworthiness of nodes. A malicious node might force its neighbors to choose it as an MPR node. Hence, a node should never select a neighbor as an MPR node if it behaves suspiciously and shows specific characteristics which would influence the MPR selection. Authors also show that to compute optimal paths, the optimality should not depend only on the length of a path but also whether or not it goes through fully or partially trusted MPR nodes. In [3], Adnane et al., proposed a trust based reasoning for OLSR that allows each node to correlate information provided by Hello, TC messages and data packets information so as to validate its local view of the global network topology. In their approach, when an inconsistency is detected between any received messages and its local view, the reasoning node is able to identify the compromised route. Their approach does not require any modification of the bare OLSR, but only the integration of the trust reasoning model on each node. Wu et al. present in [32] an overview of attacks according to the protocol layers, security attributes and mechanisms. Additionally, they present preventive approaches following the order of the layered protocol layers and an overview of reactive approaches based on IDS mechanism for MANET as a second line of defense to thwart attacks.

Vilela et al., present in [29] a feedback reputation mechanism which assesses the integrity of routing control traffic by correlating local routing data with feedback messages sent by the receivers of control traffic. Based on this assessment, misbehaving nodes are shown to be reliably detected and can be adequately punished in terms of their ability to communicate through the network. In [18], Cuppens et al. investigate the use of Aspect-Oriented Programming (AOP) in MANETs to provide availability issues in proactive routing protocols. Their approach is based on a detection-reaction process. Authors formally describe normal and incorrect node behaviors to derive security properties using AOP. The proposed algorithm verifies if those security properties are violated. If they are, then the detector node sends to its neighborhood the detection information to avoid choosing the intruder as part of valid paths to be constructed. A node chooses valid paths based on the reputation of other nodes.

6.2.2 Security Issues in OLSR Networks

In this section, we describe security attacks against the topology map acquisition process in OLSR networks. According to Herberg and Clausen [19], in OLSR networks every node must acquire and maintain a routing table that effectively reflects the network topology. Additionally, the routing tables constructed by every node must converge, i.e., all nodes must have an identical topology map. Therefore, the target of a misbehaving node may be that the nodes in the network (a) build inconsistent routing tables that do not reflect the accurate network topology, or (b) acquire an incomplete topology map. In link state routing protocols, some attacks can be launched even in networks with either cryptographic capabilities or IDS mechanisms implemented, e.g., a misbehaving node refuses to compute a valid MPR set. The exchange of control traffic information and the MPR selection process are important vulnerability targets. In this chapter, we focus on *flooding disruption attacks* [8], Fig. 6.3. In this kind of attacks, the target of an attacker is to disrupt the topology map acquisition process by disturbing the flooding of valid control traffic information. In [8], we presented a taxonomy of these attacks and countermeasures based on the selection of the MPR sets with additional coverage. The taxonomy we presented in [8] divides the attacks in two categories:

- Incorrect MPR Selection: in this category, the malicious node either selects an incomplete MPR set or forces other nodes to compute an incorrect MPR set. To launch the attack, the malicious node may either generate control traffic information with a false identity (i.e., identity spoofing) or report inexistent links

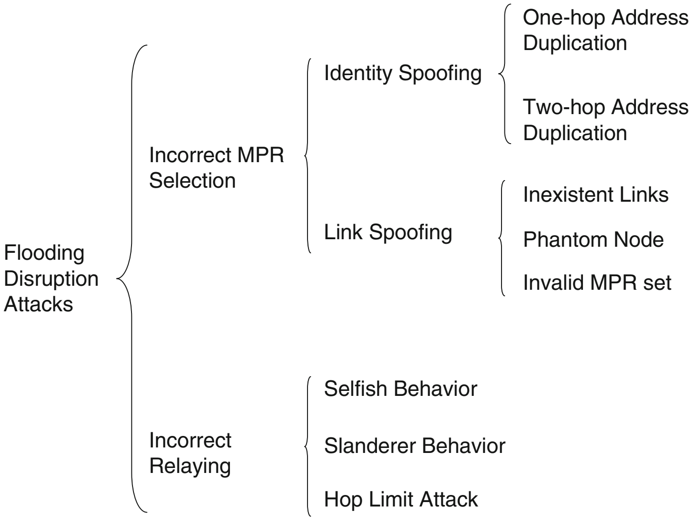


Fig. 6.3 Taxonomy of flooding disruption attacks [8]

to other nodes (i.e., link spoofing). As a consequence, the affected node computes an invalid MPR set, i.e., some of its two-hop neighbors are not covered through at least one node in its MPR set.

- Incorrect Relaying: in this category, the malicious node does not generate control traffic information (i.e., TC, MID or HNA messages) or does not forward valid messages on behalf of other nodes, e.g., selfish attack. In a variation of the attack, a malicious node may report incomplete information or eliminate some information reported by other nodes, e.g., slanderer behavior. Additionally, the misbehaving node can maliciously alter mutable fields in the messages before forwarding them, e.g., hop limit attack.

Figure 6.3 summarizes flooding disruption attacks in OLSR networks and the mechanisms used to perform them. In the sequel, we present these security threats in more detail. In Sect. 6.2.3 we present countermeasures to mitigate the effect of the attacks.

6.2.2.1 Incorrect MPR Selection

In this section, we describe vulnerabilities against the MPR selection process and some techniques to launch the attacks, i.e., link or identity spoofing.

Identity Spoofing. The identity spoofing attack [19] is performed by a malicious node pretending to be a different node in the network. The goal of the attack is to report false information about nodes one or two-hops away in order to maliciously affect the MPR selection process. Figure 6.4a illustrates an example where node *x* spoofs the identity of node *d* and broadcasts Hello message advertising a valid

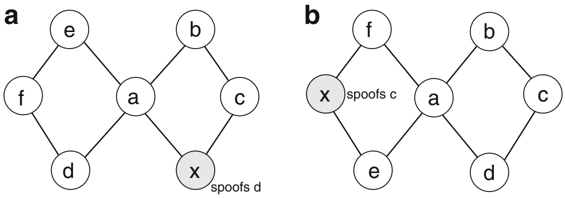


Fig. 6.4 Flooding disruption due to identity spoofing attacks. In Fig. 6.4a node *x* spoofs *d* and reports an incorrect link between nodes *c* and *d* (one-hop address duplication). In Fig. 6.4b, node *x* spoofs *c* and affects node *a*'s MPR selection (two-hop address duplication)

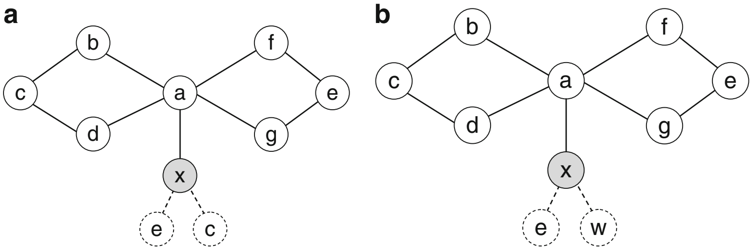


Fig. 6.5 Flooding disruption due to link spoofing attacks. In Fig. 6.5a, node *x* spoofs links to nodes *e* and *c*. In Fig. 6.5b, node *x* spoofs links to nodes *e* and the inexistent node *w*

link with node *c*. Then, node *a* receives Hello messages from node *x* indicating that node *d* has links with nodes *c* and *f*. In this case, node *a* selects incorrectly node *d* as the only element in its MPR set. In consequence, node *c* is unreachable through the MPR set and never receives TC messages. Figure 6.4b, presents an example where the attacker affects the MPR selection of a node at distance two hops. The malicious node *x* spoofs the identity of node *c*, i.e., nodes *f* and *e* generate Hello messages advertising node *c* as a one-hop neighbor. From the point of view of node *a*, nodes *b, e, f* and *d* have node *c* as a one-hop neighbor. As a result of the attack, node *a* can select incorrectly nodes *f* or *e* as an MPR. In this case, nodes *b* and *d* do not forward control traffic information to node *c* because they are not included in the MPR set.

Link Spoofing. The link spoofing attack [19] is performed by a malicious node that reports an inexistent link to other nodes in the network. The objective of the attacker is to manipulate the information about the nodes one or two hops away and be selected as part of the MPR set. Once the malicious node has been selected as an MPR, it neither generates nor forwards control traffic information. The flooding disruption attack due to link spoofing is illustrated in Fig. 6.5a. In this example, node *x* spoofs links to nodes *e* and *c*. Node *x* sends Hello messages and looks like the best option to be selected as an MPR for node *a*. Node *a* receives the Hello messages from node *x* and computes incorrectly its MPR set by selecting node *x* as the only element to reach nodes *e* and *c*. Thus, all routing information do not reach nodes two hops away from node *a*.

A variant of the attack can be performed by a misbehaving node either reporting a link to an inexistent node (i.e. a *phantom* node) or selecting an invalid MPR set.

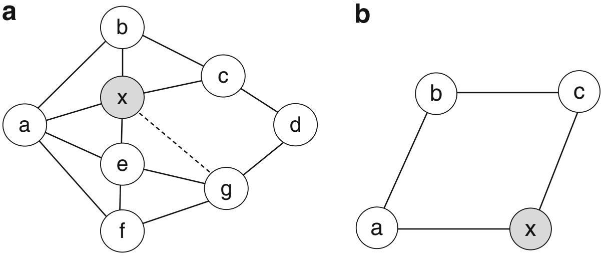


Fig. 6.6 Flooding disruption due to protocol disobedience. In Fig. 6.6a, node *x* never selects a valid MPR set. In Fig. 6.6b, node *x* modifies and forwards incorrectly TC messages

For instance, in Fig. 6.5b, node *a* is forced to select node *x* as an MPR because is the only node to reach the inexistent node *w*. In the second case, a malicious node may disrupt the flooding of topology control information by misbehaving during the MPR selection process. Figure 6.6a illustrates the attack. Node *x* wants to be selected as the only MPR of node *a*. Then, it spoofs a link to node *g* and generates Hello messages announcing node *g* as a one-hop neighbor and its only MPR. From the perspective of node *a*, nodes *c* and *g* can be reached through node *x*. Then, node *x* is the best candidate to be selected as an MPR for node *a*. Thus, node *x* receives and forwards TC messages from node *a*. However, those messages never reach node *d* because any one-hop neighbor of node *x* retransmits the messages. This attack exploits the *source dependent* requirement in OLSR to forward control traffic information. In this case, for nodes *a, b, c* and *e*, node *x* is not included in their selector table and they never forward any message from node *x*.

6.2.2.2 Incorrect Relaying

A misbehaving node can disrupt the integrity of the network by either incorrectly generating or relaying control traffic information on behalf of other nodes. Consider *x* in Fig. 6.6a as a misbehaving node. Node *x* wants to be selected as the only MPR of node *a*. Then, it spoofs a link to node *g* and generates Hello messages announcing node *g* as a one-hop neighbor. From the perspective of node *a*, nodes *c* and *g* can be reached through node *x*. Thus, node *x* is selected by node *a* as its only MPR and might perform the following incorrect behaviors:

- **Selfish behavior**. The attack is performed by a node that misbehaves and neither generates nor forwards TC messages. To increase the effectiveness of the attack, the malicious node might establish false links to other nodes in the network and force its one-hop neighbors to select it as their MPR. Figure 6.6a illustrates an example where node *x* has been selected by node *a* as an MPR but it does not relay control traffic on behalf of other nodes. As a result, node *d* does not receive control traffic information from node *a*. Note that in an OLSR network, the attacker can choose not to forward any particular message, i.e., TC, MID or HNA messages.

- **Slanderer behavior**. Due to message size limitations, an MPR may report only a partial list of elements in its selector set, i.e., an MPR may generate more than one TC message to report its entire selector set. A receiver can not know if an MPR reports its entire selector set in more than one TC message. The information gathered from the TC messages is accumulated in its topology table and is only eliminated when the validity time has expired. Thus, a misbehaving node can always generate TC messages without reporting all nodes in its selector table claiming that the size of the messages is not enough to include all nodes in its selector table. As a result, if node *x* generates TC messages without including node *a*, node *d* is not able to compute a path to node *a*.
- **Hop Limit attack**. A malicious node *x* may drastically decrease the hop limit (TTL value) when forwarding a TC message, e.g., setting the hop limit equal to zero. This reduces the scope of retransmitting the message. The attack can be performed by a malicious node that has not been selected as an MPR. For instance, in Fig. 6.6b, a control message is forwarded by node *a* and received by both nodes *x* and *b*. Previously node *b* was selected by node *a* as its MPR. However node *x* forwards the message without any delay or jitter such that its retransmission is received before the valid message from *b* arrives. Before forwarding, it reduces the hop limit of the message. The affected node, node *c*, processes the message and mark it as already received, ignoring future valid copies from *b*. Thus, the message with a very low hop limit will not reach the whole network.

6.2.3 Countermeasures

In an OLSR network, the MPR selection reduces at minimum the overhead generated by control traffic messages, if every node selects its MPR set with the following conditions:

- The MPR set is kept at minimum,
- An MPR retransmits control traffic messages if and only if the sender node is included in its selector table, and
- Only partial link state information is transmitted, i.e., an MPR reports only links with its selector nodes.

Nevertheless, we can loosen up the previous restriction in order to offer a higher level of security while maintaining a tradeoff between security and performance. In [8], we present strategies based on the selection of MPRs with additional coverage, a non source-dependent forwarding mechanism and redundant information. The selection of MPRs with additional coverage is defined in RFC3626 [10], we named it in [7] the *k*-Covered-MPR selection. In this approach, every node selects its MPR set such that any two-hop neighbor is covered by *k* one-hop neighbors, whenever possible. However, the overhead generated by the excessive number of TC messages reduces the performance of the network. This problem is addressed with the *k*-Robust-MPR selection presented in [7], which balances security and traffic overhead. In the

k-Robust-MPR selection, every node computes an MPR set that is composed of, at most, $k + 1$ disjoint sets, i.e., every two-hop node is covered, when possible, by $k + 1$ disjoint sets of one-hop neighbors. In a k-Robust-MPR selection, it is possible to discard a maximum of k invalid MPR sets and all nodes two hops away are still covered by the remaining elements in the MPR set. In a non source-dependant mechanism the MPRs retransmit all TC messages even if the sender node is not part of their selector set. Redundant information is possible by tunning the TC_redundancy parameter. This parameter is defined in the RFC3626 [10] and has three options:

- MPRs report their selector table when TC_redundancy is equal to zero,
- MPRs report their selector table and MPRs when TC_redundancy is equal to one, and
- MPRs report their one-hop neighbors when TC_redundancy is equal to two.

Advertising redundant information increases the size of the TC messages, but more links are advertised. In [7], we compared both k-Covered-MPR and k-Robust-MPR selections in the presence of misbehaving nodes. We measured the number of nodes with complete routing tables after the execution of the OLSR protocol. Our experiments showed that our k-Robust-MPR selection reduces the amount of traffic generated by the k-Covered-MPR selection, and offered equivalent protection against control traffic attacks. Our k-Robust-MPR selection increased the performance ratio of the number of nodes with complete routing tables over the number of topology control messages.

6.3 Hierarchical OLSR

In this section, we present the Hierarchical OLSR (HOLSR) protocol and its vulnerabilities. By nature, MANETs are formed of heterogeneous nodes that can join the network following an unpredictable pattern. Furthermore, scalability is a problem in MANETs. In [30], Villasenor-Gonzalez et al. define scalability as the capacity of the network to adjust or to maintain its performance even if the number of nodes increases. OLSR is a *flat* routing protocol and its performance degrades when the number of nodes increases due to a higher number of topology control messages propagated through the network. The MPR mechanism is local and therefore very scalable. However, the diffusion of link state information by all the nodes is less scalable. Hence, OLSR's performance decreases in large ad hoc networks. Additionally, OLSR does not differentiate the capabilities of the nodes and, in consequence, does not exploit nodes with higher capabilities. HOLSR is an approach designed to improve the scalability of the OLSR protocol in large-scale heterogeneous networks.

The main improvements are a reduction of topology control traffic and an efficient use of high capacity nodes. HOLSR organizes the network in hierarchical clusters. This architecture reduces the routing complexity, i.e., in case a link is broken only nodes inside the same cluster have to recalculate their routing table while nodes in other clusters are not affected. Nodes are organized according to

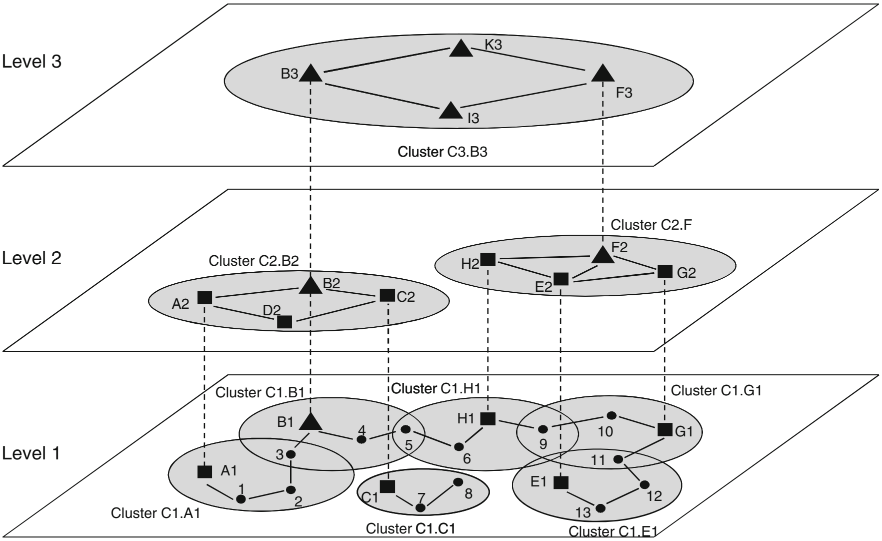


Fig. 6.7 Example of a hierarchical architecture with heterogeneous nodes

their capacities. The network architecture is illustrated in Fig. 6.7. At level 1, we have low-capability nodes with a single interface, represented by circles. Nodes at the topology level 2 are equipped with up to two wireless interfaces, designated by squares. Nodes at level 2 employ one interface to communicate with nodes at level 1. Nodes at level 3, designated by triangles, represent high-capacity nodes with up to three wireless interfaces to communicate with nodes at every level. Thus, in Fig. 6.7, node F3 represents node F's interface at level 3. The only restriction for every node at levels 2 and 3 is that they have at least one interface to communicate with nodes at its levels. For instance, in Fig. 6.7 nodes F2 and F3 represent node F's interfaces at levels 2 and 3 respectively. Nodes A1 and A2 represent node A's interfaces to establishes communication with nodes at levels 1 and 2 respectively. Node D2 has only one interface and can just communicate with nodes at level 2. In the example, the notation used to name the clusters reflects the level of the cluster and cluster head, e.g., C1.A1 means that the cluster is at level 1 and cluster head is node A1, which is node A's interface at level 1. HOLSR allows formation of multiple clusters. Unlike OLSR, HOLSR nodes can exchange Hello and TC messages exclusively within each cluster. This constraint reduces the broadcast traffic.

Across cluster topology control information is exchanged via specialized HOLSR nodes designated as cluster heads. Cluster heads are selected and nodes are classified according to their capabilities at the startup of the HOLSR process. A cluster is formed by a group of same-level mobile nodes that have selected a common cluster head. Nodes can move from one cluster to another and associate with the nearest cluster head. Any node participating in multiple topology levels automatically becomes the cluster head of the lower-level cluster. In HOLSR, a cluster head declares its status and invites other nodes to join in by periodically sending

Table 6.2 Summary of control traffic messages in HOLSR networks

Messages	Generated by	Retransmitted by	Reported information
Hello	Every node	N/A	One-hop neighbors
TC	MPRs	MPRs	MPR selector set
CID	Cluster heads	N/A	Cluster head identification
HTC	Cluster heads	MPRs	Nodes within a cluster

out Cluster ID Announcement (CID) messages. These and Hello messages are transmitted in the same packet using a grouping technique. This reduces the number of packet transmissions. A CID message contains two fields: *cluster head*, that represents the interface address of the originator of the message, and *distance*, which is the distance in hops to the cluster head generating the message. Every time the cluster head generates a CID message, the field *distance* is set to zero. A receiver node joins the cluster head and sends a new CID message. The new CID message increments the value of the distance. It invites other nodes to join the same cluster. The cluster formation process is described in more detail in [30].

The hierarchical architecture must support the exchange of topology control information between clusters without introducing additional overhead. Thus, Hierarchical TC (HTC) messages are generated by the cluster heads and used to transmit the membership information of a cluster to higher level nodes. HTC forwarding is enabled by the MPRs and restricted within a cluster. Nodes at the highest topology level have full knowledge of all nodes in the network. Their routing tables are as large as they would be in an OLSR network. However, in lower levels, the size of the routing table of every node is restricted by the size of the cluster and it is smaller than in OLSR. For instance, in Fig. 6.7 the cluster head A2 generates a HTC message at level 2 announcing that nodes 1, 2 and A1 are members of its cluster at level 1. The message is relayed to all nodes at the same level. Node B3 generates HTC messages at level 3 advertising that nodes 1, 2, 3, 4, 5, 7, 8, A1, B1, C1 (at level 1) and A2, B2, C2, D2 (at level 2) are members of its cluster. Table 6.2 presents a summary of the messages implemented in HOLSR networks.

Control messages are generated and propagated exclusively within each cluster unless a node is located in the overlapping zone of several clusters, i.e., a border node. For example, in Fig. 6.7 node 2 is within the border of cluster C1.A1 and may accept a TC or a HTC message from node 3 located in cluster C1.B1 (i.e., nodes 2 and 3 are border nodes). However, node 2 does not retransmit. Thus, except for the border nodes, knowledge of member nodes is restricted to the cluster itself. Data transfer between nodes in the same cluster is achieved directly using the routing tables. However, when transmitting data to destinations outside the local scope of a cluster, the cluster head is used as a gateway. A different strategy might be used, when transmitting data between border nodes in different clusters at the same level. Border nodes in different clusters at the same topology level can communicate directly without having to follow the strict clustering hierarchy. Therefore, HOLSR offers two main advantages (a) the traffic control reflecting local movements is restricted to each cluster (thus, reducing the routing table computation overhead), and (b) an efficient use of high-capacity nodes without overloading them.

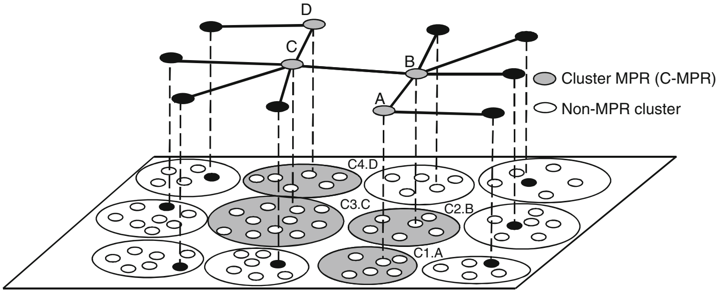


Fig. 6.8 Example of a Cluster OLSR network. Consider *gray* clusters as C-MPRs

6.3.1 Related Work

In this section, we review other hierarchical models based on OLSR to improve scalability in MANETs.

6.3.1.1 Cluster OLSR

In [27], Ros et al. present the Cluster OLSR (C-OLSR) protocol. Unlike HOLSR, C-OLSR does not assume any particular cluster formation algorithm nor existence of higher capacity nodes. C-OLSR implements OLSR inside every cluster and uses the MPR mechanism for distributing control traffic at both inter-cluster and intra-cluster levels. C-OLSR limits the forwarding of TC messages inside every cluster to minimize the number control traffic messages. Every node can compute routes to any other node inside its cluster. To reach nodes in other clusters, nodes create routes to every cluster and not to every node. When a data packet arrives to a destination cluster, every node has enough information to deliver the packet to its final destination. This mechanism reduces the size of the routing tables.

For inter-cluster communications, Cluster Hello (C-Hello) and Cluster Topology Control (C-TC) messages are defined. C-Hello messages are used to sense neighboring clusters and to compute the Cluster MPR (C-MPR) set. C-Hello messages are flooded within the receiver cluster but not retransmitted to neighbor clusters. A C-MPR is a cluster selected to reach other clusters and mitigate the overhead of distributing C-TC messages for inter-cluster communications. C-TC messages advertise the nodes within a cluster to all the network. Figure 6.8, shows an example of a C-OLSR network. At the first level, nodes are organized in clusters. The second level, shows how clusters are linked. Gray clusters are C-MPRs, e.g., C1.A is a C-MPR and node *A* is the cluster head. When a node in a cluster needs to send a data packet to a node inside another cluster, it computes a path through the clusters selected as C-MPRs, i.e., $C1.A$, $C2.B$, $C3.C$ and $C4.D$.

When a C-Hello or C-TC messages arrive to a cluster, they are relayed to every node in the cluster. This allows nodes to learn about clusters topological

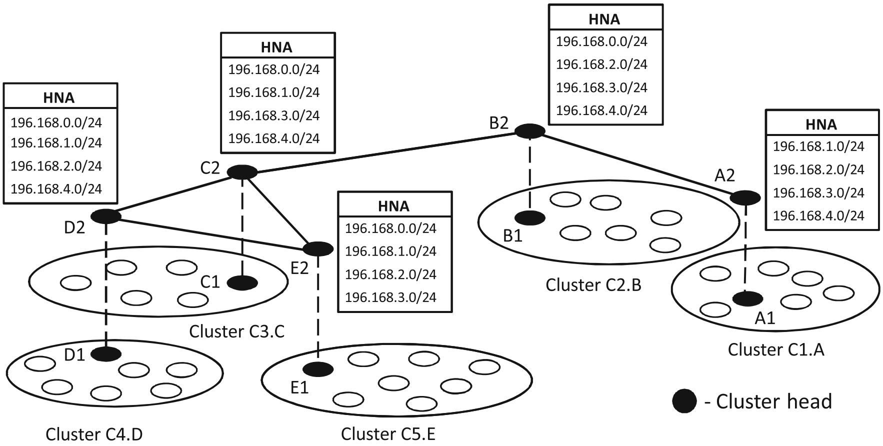


Fig. 6.9 Example of a MORHE network. Consider *black nodes* as backbone nodes

information. C-TC messages must be relayed to adjacent clusters, only if the sender of the message has selected the receiver node as an C-MPR. To support this hierarchical architecture, every C-OLSR node has additional information repositories: one-hop neighbor cluster set, two-hop neighbor cluster set, cluster topology set, cluster MPR set and cluster MPR selector set. The information in these repositories supports inter-cluster communications. In C-OLSR, not every node has to generate inter-cluster information. The generation of C-Hello and C-TC messages can be done according to three different algorithms: a cluster head-based algorithm, a distributed algorithm or a hybrid approach. In the former case, only cluster heads generate control information. In the second algorithm, topology information is generated exclusively by border nodes. Finally, in the hybrid approach, C-Hello messages are generated by border nodes and C-TC messages are generated by the cluster heads. In all cases, the selected C-MPRs are responsible for forwarding C-TC messages.

6.3.1.2 The *Multi-level OLSR Routing Using the HNA Extension*

In [31], Voorhaen et al. present a multi-level routing scheme for ad hoc networks based on OLSR. The *Multi-level OLSR Routing using the HNA Extension* (MORHE) protocol improves scalability by exploiting high capability nodes. Using HNA messages and hierarchical addressing, MORHE constructs an overlay network formed by nodes with higher capabilities. Nodes with higher capabilities are selected as cluster heads. A cluster head is called a *backbone* node. Backbone nodes are chosen before network deployment and have more than one interface. Nodes are organized into clusters around every backbone node. Figure 6.9, shows an example of a two-levels MORHE network. Nodes A, B, C, D and E are backbone nodes. Backbone nodes use one interface to communicate with the nodes inside their cluster and the second interface for inter-cluster communications. For instance, backbone node A, communicates with the nodes at the first level through the interface $A1$ and uses interface $A2$ to communicate with other backbone nodes. OLSR is implemented at each level.

MORHE is similar to HOLSR, nonetheless it only uses HNA messages already defined in the RFC 3626 [10]. Each backbone node periodically sends HNA messages informing other backbone nodes that it can reach all the nodes in the subnet that it is connected to. When a backbone node receives a HNA message, it updates its association database. Every backbone node uses HNA messages to inform all the nodes in its cluster about other clusters that can be reached. HNA messages are distributed using the MPR mechanism as defined in OLSR. Nodes can communicate directly with every node inside its cluster. Backbone nodes enable communication between nodes in different clusters. When a packet arrives at a backbone node, it attempts to find a route to the destination in its cluster. If this fails, then the backbone node retransmits the message to another backbone node. If the receiver finds a route, then it forwards the packet inside its cluster. In a MORHE network, every cluster is identified as a subnetwork. For instance, in Fig. 6.9, the network is divided in five subnetworks. Every backbone node has the IP addresses of every subnetwork in its association table. For example, 192.168.1.0/24 is the prefix of an IPv4 subnetwork, having 24 bits allocated for the network prefix, the remaining 8 bits are reserved for host addressing. If a node inside the subnetwork 192.168.0.0/24 needs to communicate with a node in the subnetwork 192.168.2.0/24, then it sends the packet to its backbone node which retransmits the packet to its final destination.

6.3.1.3 Tree Clustering

In [4, 5], Baccelli proposed a *Tree Clustering* mechanism to enable hierarchical routing within an OLSR network. Each cluster is a *tree*. Their head is the *root*. To organize the network in trees, every node selects as its parent the adjacent node with the maximum number of one-hop neighbors. The parent of a node is called a node's preferred neighbor. A node with maximum degree, i.e., maximum number of neighbors, is selected as the root of the three. The network is then viewed as a *forest*, i.e., a collection of logical trees. To form and maintain trees, OLSR nodes periodically exchange *Branch* messages. These messages are piggy-backed with Hello messages. Branch messages are not retransmitted. Within a Branch message, a node specifies its identity, the tree it belongs to, its parent in the tree and its distance in hops to the root. Roots can choose to limit the size of their three by imposing a *maximum depth* value. The organization in trees is dynamic. A mechanism allows to switch between a traditional flat networking, i.e., flat mode or a hierarchical networking, i.e., tree mode. The mechanism to transit between the flat mode and the tree mode is explained in detail in [4].

Within a tree, OLSR nodes operate as if there was no tree, except that messages originated by a node in a different tree are not considered and not forwarded, the root is responsible for the communication between the tree and the rest of the network, and a node in contact with another tree i.e., a leaf node, must inform its entire tree (specially its root), of the distance to reach other roots. A leaf node must generate a *Leaf* message for each other tree it reaches. In a Leaf message, the node specifies its ID, the root of the neighbor tree and the estimated distance between the roots, i.e., the sum between its depth in its tree, and the distance to the root of the neighbor tree. With this information, every root is able to compute the shortest path to reach its neighbor roots.

This protocol employs Hello and TC messages within every tree, but implements Super-Hello (S-Hello), Super-TC (S-TC) and Super-HNA (S-HNA) messages for inter-cluster communications. Super messages are generated exclusively by the roots. These messages are identical to regular messages except for an additional field that includes the IP address of the next root to reach. Unlike regular messages, Super-messages are routed using the constructed paths instead of being flooded. Super-messages are unicasted using the shortest root-to-root path advertised by Leaf messages. Super-messages are the only messages to be forwarded outside a tree. MPR selection is performed as if there were no trees. When a tree mode is activated, the scope of TC messages is limited to the tree they were generated. However, Super-messages are forwarded between clusters following the MPR flooding mechanism.

To allow hierarchical routing, routes exchange Super-messages in order to identify other roots and construct a Super-topology. S-Hello and S-TC messages allow the roots to construct a super-topology formed by roots. The roots periodically exchange S-Hello messages to learn about other roots in neighbor trees (i.e., one-super-hop neighbors). As in OLSR, every root computes its super-MPR set formed by other roots. A super set of MPRs is used for distributing S-TC messages among clusters. S-Hello messages are not forwarded. S-TC messages are forwarded by the S-MPRs. S-TC messages include the super-selector set, i.e., the roots that have selected the sender as a S-MPR. Finally, every root generates S-HNA messages to inform other roots about the link state information within its cluster. Therefore, every root is aware of the link state information of other threes. Routing among clusters is achieved using the information between S-TC and S-HNA messages. Traffic outside the tree scope is achieved via the root nodes. Figure 6.10 shows an example of a tree clustering hierarchical architecture. Nodes A,B,C,D and E are selected as roots. These nodes have the maximum degree. Root node A selects B as its MPR to reach root trees C, D and E. When a node inside cluster C1.A needs to communicate with a node inside cluster C5.E, it sends the data traffic to its root node A which retransmits the traffic to its final destination trough B and E.

Table 6.3 presents a summary of the features of each hierarchical approach that we reviewed. Unlike MORHE and C-OLSR, HOLSR and the Tree clustering approaches include a cluster formation mechanism. MORHE and HOLSR were designed for heterogeneous networks and multiple hierarchical levels. C-OLSR and Tree clustering were designed for homogeneous networks and two hierarchical levels. Nevertheless, these approaches might be implemented in networks with heterogeneous capabilities. All approaches implement the MPR mechanism for distributing control traffic messages.

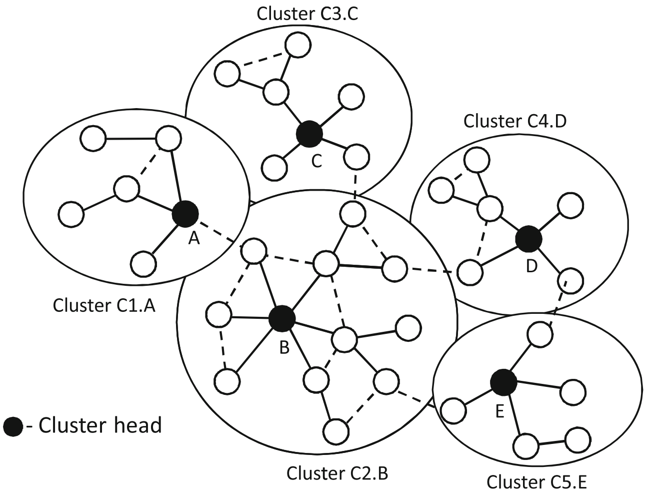


Fig. 6.10 Tree clustering. *Black nodes* represents the roots of the tree. Branches of the trees are shown with *solid lines* between nodes. Links that are not branches are *dashed*

Table 6.3 Comparison of OLSR-based hierarchical approaches. All approaches implement Hello and TC message for intra-cluster communications

Routing protocol	Network	Logical levels	Messages	Cluster formation Alg.
HOLSR	Heterogeneous	n	CID and HTC	Yes
MORHE	Heterogeneous	n	HNA	No
C-OLSR	Homogeneous	2	C-Hello and C-TC	No
Tree	Homogeneous	2	Leaf, Branch, S-Hello, S-TC and S-HNA	Yes

6.3.2 *Security Issues in HOLSR Networks*

Note that in all described approaches, the exchange of control traffic at both intra-cluster and inter-cluster levels is performed by using the MPR mechanism. Security is no addressed. Therefore, they are vulnerable to the flooding disruption attacks described in Sect. 6.2.2. The cluster formation phase is vulnerable to malicious behavior. In [8,9], we describe in detail security threats to both the *cluster formation* and *topology map acquisition* phases.

In HOLSR, the flow of CID messages is an important vulnerability target. The *hop count* has to be updated every time a new message is retransmitted. Thus, a malicious node might alter this field to unsettle the cluster formation process. The attack, has a bigger impact when a malicious node drastically reduces the *hop count* field. Because receivers accept the CID message with the lowest *hop count* value. Thus, when an attacker increases drastically the value, receivers automatically discard the altered message and accept valid messages from other nodes. When a

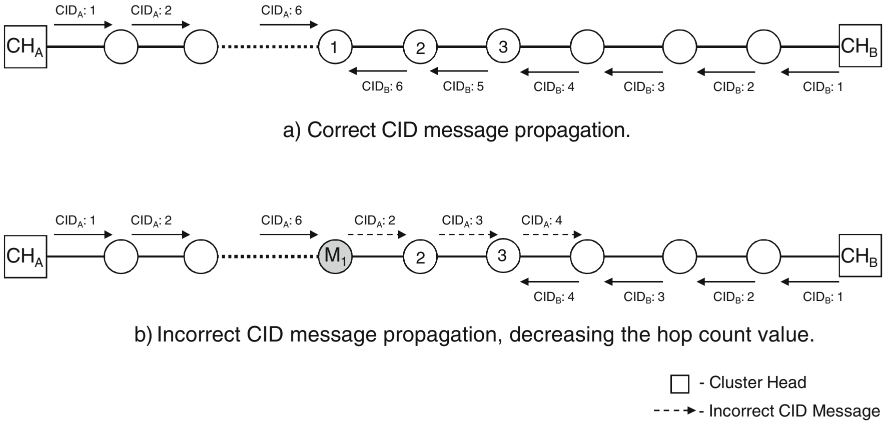


Fig. 6.11 Cluster formation attack in HOLSR networks. (**a**) Correct CID message propagation. (**b**) Incorrect CID message propagation, decreasing the hop count value

node that generates a CID message reinitializes the value of the field *hop count*, the receiver nodes may join a farther cluster head and discard valid CID messages from closer cluster heads. We address the case where the *hop count* field is maliciously reduced. For instance, Fig. 6.11a shows the correct propagation of CID messages. Figure 6.11b shows an example of the attack. In Fig. 6.11b, M_1 is a malicious node at distance six hops from cluster head CH_B. M_1 receives CID messages from CH_B, and generates a new CID message assigning the incorrect value two to the field *hop count*. Thus, all nodes at distance from CH_B, greater or equal than four hops (nodes 2 and 3) process the message and incorrectly join CH_A. Note that the lowest value that can be used to reinitialize the field *hop count* is two because CID messages with a field *hop count* equal to one are generated exclusively by the cluster heads. We assume that the attacker has only one interface. It can not impersonate a cluster head. It only modifies the *hop count* value. This attack can affect other OLSR-based hierarchical approaches. For instance, a misbehaving node may alter the field distance in *Branch* messages in the Tree Clustering approach proposed by Baccelli, reviewed in Sect. 6.3.1.3.

6.3.3 Countermeasures

In [8, 9], we describe in detail security threats in both the *cluster formation* and *topology map acquisition* phases. Countermeasures to mitigate the effect of the attacks are also presented. In the former case, in [8], we analyze the effect of flooding disruption attacks in HOLSR networks to interrupt the propagation of HTC messages. We proposed additional coverage in the selection of MPRs at any hierarchical level. We analyze the effect of flooding disruption attacks. Unlikely flat OLSR networks, when a malicious nodes attempts to interrupt the propagation of

HTC messages the inter-cluster communication is affected. Our proposed solution is based on the selection of MPRs with additional coverage, i.e., k-Covered-MPR and k-Robust-MPR selections. Our results showed that it is possible to mitigate the effect of the attack by adding additional coverage. The k-Covered-MPR selection increased the chances of mitigate the attack but the performance of the network reduces due to an increased number of TC and HTC messages. Our proposed k-Covered-MPR selection offers an equivalent level of protection but reducing the amount of TC and HTC messages flooded in the network.

In [9], we presented a solution based on *hash chains* to protect mutable fields in HOLSR networks. Our algorithm Hash-Chained_CID_Dissemination (HCCD) allows to detect and discard invalid CID messages. A valid cluster head (CH_j) generates a random number s_j, i.e., a nonce that is only known by the originator of the message. After, it initializes the hop count field i equal to one and computes the Max_j value by applying t times the hash function $h(x)$ to the nonce s_j, such that Max_j is equal to $h^t(s_j)$. We assume that Max_j and the value of t are known by all the nodes in the network during the execution of the protocol. Additionally, CH_j applies i times the hash function to s_j, to obtain $h^i(s_j)$. Then, CH_j generates a CID message with the fields: $< Max_j, h^i(s_j), i >$. The receiver node verifies that the CID message is valid by applying $t-i$ times the hash function to $h^i(s_j)$ and comparing the result with Max_j. Therefore, if Max_j is equal to $h^{t-i}(h^i(s_j))$, then the hop count value i has not been altered and the received CID message is valid. Finally, the receiver node joins CH_j until it receives a CID message from a different cluster head with a lower hop count value. In the mean time, the receiver node generates periodically CID messages announcing its cluster head and the hop count distance to reach it, i.e., $< Max_j, h(h^i(s_j)), i+1 >$. Our solution is based on the work presented by Hong et al. in [20]. The authors presented a wormhole detective mechanism and an authentication protocol to strengthen the neighbor relationship establishment in standard OLSR. The authors used digital signatures to ensure the non-mutable fields and hash chains to secure the Hop Count and TTL fields. Their solution is similar to our proposed algorithm, however it is implemented in flat OLSR to protect only standard control traffic messages. We address a different kind of attack in HOLSR networks. Our mechanism protects the integrity of CID messages and enforces the proper distribution of nodes in every cluster. In [9], our experiments showed that the distribution of nodes is less balanced when the hop count in CID messages is maliciously altered. We also showed that we can prevent this kind of attacks by applying our proposed algorithm. Note that our mechanism, can be also applied in other hierarchical routing protocols for MANETs that utilize mutable information to organize the network in clusters.

6.4 Multipath OLSR-Based Routing

In this section, we analyze a multipath routing strategy based on OLSR that takes advantage of the MPR flooding mechanism. In [33–36], Yi et al. proposed the

Multipath OLSR (MP-OLSR) routing protocol aiming to enhance load-balancing, energy-conservation, Quality-of-Service (QoS) and security. MP-OLSR is a hybrid multipath routing protocol. In MP-OLSR, the OLSR proactive behavior is changed for on-demand route computation. MP-OLSR becomes a source routing protocol. There are two phases: *topology discovery* and *routes computation*. During *topology discovery*, nodes obtain a partial topology map just like in OLSR. However, MP-OLSR nodes do not construct routing tables. During *routes computation*, nodes calculate multiple paths to reach any other node in the network following an on-demand scheme. MP-OLSR implements Multiple Description Coding (MDC) for data transfer. MDC adds redundancy to information streams and split them up into several sub-streams to improve the integrity of data. These sub-streams are sent along multiple paths from the source to the destination. MP-OLSR implements source routing with route recovery and loop detection to adapt to the changes in the network topology. Thus, when data transfer is required, route recovery and loop detection allow every node to detect if a path is not valid anymore and to find a new path to reach the final destination. MP-OLSR uses the Dijkstra's algorithm to discover routes. The routes that are obtained can be grouped in two categories:

1. Disjoint: In this category we have two types of disjoint paths: node-disjoint and link-disjoint. Node-disjoint paths type do not share nodes except for the source and destination nodes. Link-disjoint paths can share some nodes but all the links are different.
2. Inter-twisted: In this case, the paths may share several links.

To construct disjoint paths, MP-OLSR defines cost functions to obtain new paths that tend to be node-disjoint or link-disjoint. Once a path is computed, a function f_p is used to increase the costs c of the links that belong to the computed path, e.g., $f_p(c) = 3c$. A function f_e is defined to increase the cost of the links of the nodes included in the path previously obtained. In MP-OLSR, neither nodes nor links used in computed paths are eliminated. This strategy allows MP-OLSR to construct multiple paths in sparse networks where is not always possible to find strictly node-disjoint paths. In addition, to increase the chances of constructing node-disjoint paths, the MPRs report all their one-hop neighbors (i.e., the TC_redundancy parameter is equal to two). Consider f_{id} as the identity function, i.e., $f_{id}(c) = c$. Therefore, to construct disjoint paths, there are three possibilities:

- If $f_{id} = f_e < f_p$, then paths tend to be link-disjoint;
- If $f_{id} < f_e = f_p$, then paths tend to be node-disjoint;
- If $f_{id} < f_e < f_p$, then paths also tend to be node-disjoint, but when not possible they tend to be link-disjoint.

For example, in Fig. 6.12a, node s attempts to construct multiple paths to node d. MP-OLSR implements a Multipath Dijkstra's algorithm to obtain the shortest paths. Consider initial cost c of each link equal to one and $f_p(c) = 3c$ and $f_e(c) = c$, i.e., a penalty is only applied to the used links. The first time the Dijkstra's algorithm is applied, the computed path is $s \to c \to d$. Thus, the cost of the links (s,c) and (c,d) is changed from one to three using f_p, see Fig. 6.12b. The second path we obtain is: $s \to b \to c \to h \to d$. The cost of the links (s,b), (b,c), (c,h) and (h,d) is set

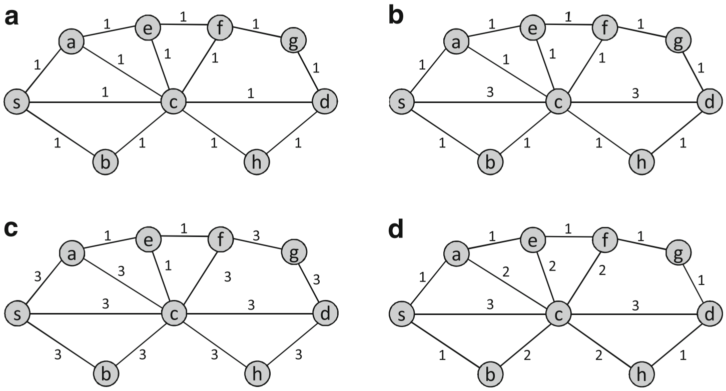


Fig. 6.12 OLSR network. In Fig. 6.12a, consider the cost of all links equal to one

to three. Finally, the third computed path is: $s \rightarrow a \rightarrow c \rightarrow f \rightarrow g \rightarrow d$. The cost of all used links is set to three, see Fig. 6.12c. These three paths are link-disjoint. To obtain paths that tend to be node-disjoint, we define functions $f_p(c) = 3c$ and $f_e(c) = 2c$. In this case, the penalty is also applied to the used nodes. First, the path $s \rightarrow c \rightarrow d$ is computed and the cost of the links is updated. The links that include a node in the computed path -except for the source s and the destination d- are set to two, see Fig. 6.12d. Then, the next path we obtain is: $s \rightarrow a \rightarrow e \rightarrow f \rightarrow g \rightarrow d$. These two paths are node-disjoint. The path: $s \rightarrow a \rightarrow c \rightarrow h \rightarrow d$, is an example of an inter-twisted path.

6.4.1 Related Work

In this section, we present other multipath routing strategies based on OLSR. Several multipath routing approaches take advantage of the proactive behavior and MPR flooding mechanism proposed in OLSR. The strategies proposed, attempt to improve security, QoS, load balancing or energy consumption. However, all strategies proposed are not secure by design. For instance, in [23], Kun et al., proposed a different version of multipath OLSR using IP-source routing. Based on the Dijkstra's algorithm, nodes calculate multiple node-disjoint paths. Additionally, the authors introduce an algorithm of load-assigned to transmit data through the paths based on the congestion information of all intermediate nodes on each path. Badis and Al Agha [6], also proposed a path selection criteria and multi-path calculation based on bandwidth and delay to improve QoS in OLSR networks (QOLSR). The resulting protocol, computes multiple loop-free and node-disjoint paths. The authors implement the shortest-widest path algorithm to guarantee loop-free routes. Additionally, they evaluated and compared QOLSR multipath routing versus a QOLSR single-path routing using a scalable simulation model. In [28], Srinivas and Modiano proposed algorithms for finding minimum energy disjoint paths in wireless networks. Their main contribution is a polynomial time algorithm for the minimum energy k node-disjoint problem. Node-disjoint paths are more resilient to failures. However, the authors showed that link-disjoint paths save more energy. Zhou et al. proposed in [37] the Source Routing based Multi-Path OLSR (SR-MPOLSR) protocol. The protocol implements the Dijkstra's algorithm to calculate multiple disjoint routes. Data transmission at the source is carried out through predetermined multiple paths (i.e., source routing). The loads are distributed in a weighted round-robin fashion. These strategies proposed attempt to construct multiple link-disjoint or node-disjoint paths. However, all approaches are affected by the flooding disruption attacks described in Sect. 6.2.2. Nodes in OLSR-based multipath routing protocols only acquire a partial view of the topology network. These problems are described in the following section.

6.4.2 Security Issues in Multipath OLSR-Based Networks

Multipath OLSR-based approaches are vulnerable to the flooding disruption attacks [8] attacks presented in Sect. 6.2.2 during the *topology discover* and *route computation* phases. An attacker may refuse to retransmit control traffic or may select an invalid MPR set to prevent other nodes from calculating disjoint paths to reach other nodes in the network. MP-OLSR constructs non disjoint multiple paths. The protocol computes several routes, but it is impossible to know how many of them are disjoint. When a node part of several paths misbehaves, all paths are affected. All OLSR-based multipath strategies use the MPR mechanism to flood the network with control traffic. However, only partial topology information is generated by the MPRs. We identify two vulnerabilities in all OLSR-based multipath routing strategies: the nodes in an OLSR network only obtain a partial view of the network topology and they are affected by the security threats presented in Sect. 6.2.2. The MPRs generate and forward TC messages to advertise their selector set to other nodes at more than two hops away. However, with this information nodes only obtain a partial view of the topology. This is because TC messages only report partial link state information. For instance, Fig. 6.13a shows the complete topology of an MP-OLSR network. Gray nodes represent MPRs. Figure 6.13b shows the perspective of node s after the topology discovery phase. The links (g, j), (i, l), (j, d), (l, d), (j, k) and (l, k) are not reported in TC messages. Thus, the link between node g and j is not reported because neither g nor j are MPRs. Node k is an MPR but it does not report links to nodes j and l because they are not included in its selector set. From the perspective of node s, k is the only node that reaches node d. Hence, it is not possible to compute multiple disjoint paths. To increase the chances of finding disjoint paths, the MPRs in an MP-OLSR networks report more information in their TC messages by tunning their TC_redundancy parameter. The TC_redundancy parameter is defined locally by every node. Nodes with different TC_redundancy values can coexist. MP-OLSR nodes set their TC_redundancy

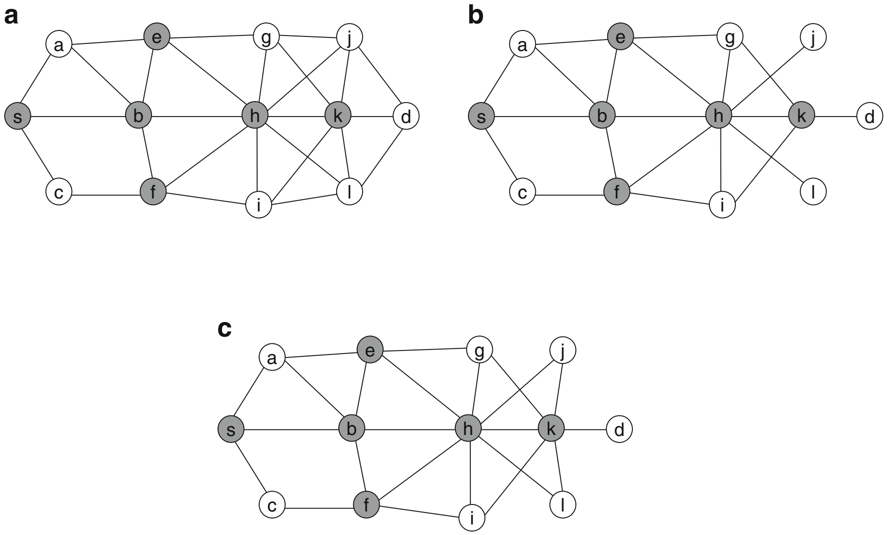


Fig. 6.13 Network topology perspective of node *s*. *Gray nodes* represent MPRs. (**a**) Complete network topology. (**b**) Node s perspective of the network. TC_redundancy equal to 0. (**c**) Node s perspective of the network. TC_redundancy equal to 2

parameter to two. However, the size of the TC messages increases and in some situation it is not enough to report important links. For example, Fig. 6.13c shows the network perspective of node s if the MPRs report their one-hop neighbors, i.e., TC_redundancy parameter equal to two. Hence, node s is aware of the links (j,k) and (l,k). However, the links (g,j), (i,l), (j,d) and (l,d) remain unreported. Figure 6.13c also shows that all the possible routes to reach node d include node k. When node k misbehaves, all the computed paths are compromised.

6.4.3 Countermeasures

The MPR selection with additional coverage (i.e., k-Robust-MPR or k-Covered-MPR) helps to mitigate the attacks against the construction of disjoint paths. Additional coverage helps to advertise more links and construct multiple node-disjoint paths without increasing the size of the messages. In OLSR networks, the MPRs form a Connected Dominating Set (CDS). A CDS is a subset of connected nodes such that if a node in the network is not part of the CDS, then it has a link to a node in the CDS. Every node must be able to construct a CDS of the network with the information gathered during the topology discovery phase. We define an MPRCDS as a CDS such that every node in the CDS has been selected as an MPR. When the nodes select their MPRs following a k-Covered-MPR selection we obtain a k-CCDS. When the nodes compute their MPRs following a k-Robust-MPR selection we obtain a k-RCDS. Therefore, if a node obtains a more complete view of the network (i.e., k-CCDS or k-RCDS), then it is able to find alternative routes to compute disjoint paths.

6.5 Conclusion and Future Work

In link state routing protocols for MANETs, the generation and exchange of control traffic messages are important vulnerability targets. A malicious node may perpetrate an attack by flooding the network with incorrect information or by preventing other nodes from acquiring a complete network topology map. We presented security threats in link state routing protocols based on OLSR. Particularly, we addressed flooding disruption attacks in OLSR networks. This kind of attacks can be carried out in networks with cryptographic capabilities. Additionally, a review of related work and proposed countermeasures is also presented. In addition, we reviewed security threats in other link state routing protocols based on OLSR. We presented vulnerabilities and countermeasures specific to HOLSR and MP-OLSR.

6.5.1 Future Work

The k-Robust-MPR selection may be affected either by a malicious node, that generates false links to avoid the selection of k+1 disjoint MPR sets or due to the network topology. As part of future work, we consider an extended k-Robust-MPR selection to address the cases when is not possible to select multiple disjoint MPR sets. Countermeasures against more complex attacks during the cluster formation phase in hierarchical OLSR-based networks is also part of further research. A mechanism to improve the selection of multiple disjoint routes in OLSR-based networks is required. To improve load balancing, nodes with the smallest number of nodes in their selector set should be privileged to be included in the computed paths. Clearly, in sparse networks is not always possible to compute disjoint paths. Nevertheless, multipath routing takes advantage of large and dense networks. Then, the cases where the construction of multiple node-disjoint paths is affected either by an incomplete view of the network topology or by the presence of a misbehaving node should be addressed.

Acknowledgements The authors graciously acknowledge the financial support received from the following organizations: Natural Sciences and Engineering Research Council of Canada (NSERC), Mathematics of Information Technology and Complex Systems (MITACS), Institut Telecom, Spanish Ministry of Science and Innovation (grants TSI2007-65406-C03-03 E-AEGIS and CONSOLIDER-INGENIO CSD2007-00004 ARES), National Council of Science and Technology (CONACYT), and Ministry of Education of Mexico (SEP, Program for Academic Improvement).

References

1. R. Abdellaoui and J.-M. Robert. SU-OLSR: A new solution to thwart attacks against the OLSR protocol. In *4th Conference on Security in Network Architectures and Information Systems (SAR-SSI)*, pages 239–245, Luchon, France, June 22–26, 2009.
2. C. Adjih, T. Clausen, A. Laouiti, P. Muhlethaler, and D. Raffo. Securing the OLSR routing protocol with or without compromised nodes in the network. Technical Report, INRIA RR-5494, HIPERCOM project, INRIA Rocquencourt, February 2005.
3. A. Adnane, R.T. de Sousa Jr., C. Bidan, and L. Me. Autonomic trust reasoning enables misbehavior detection in OLSR. In *SAC'08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 2006–2013, New York, NY, USA, 2008. ACM.
4. E. Baccelli. OLSR scaling with hierarchical routing and dynamic tree clustering. In *IASTED International Conference on Networks and Communication Systems (NCS)*, Chiang Mai, Thailand, March 2006.
5. E. Baccelli. OLSR trees: A simple clustering mechanism for OLSR. In *Challenges in Ad Hoc Networking, IFIP International Federation for Information Processing*, vol. 197, pages 265–274, 2006.
6. H. Badis and K. Al Agha. QOLSR multi-path routing for mobile ad hoc networks based on multiple metrics: bandwidth and delay. In *Vehicular Technology Conference, 2004*. VTC 2004-Spring. 2004 IEEE 59th, vol. 4, pages 2181–2184, May 2004.
7. G. Cervera, M. Barbeau, J. Garcia-Alfaro, and E. Kranakis. Mitigation of topology control attacks in OLSR networks. In *5th International Conference on Risks and Security of Internet and Systems (CRISIS 2010)*, Jean-Marc Robert, editor, pages 81–88, Montreal, Canada, October 10–13, 2010.
8. G. Cervera, M. Barbeau, J. Garcia-Alfaro, and E. Kranakis. Mitigation of flooding disruption attacks in HOLSR networks. *In 9th Annual Conference on Communication Networks and Services Research Conference (CNSR 2011)*, pages 167–174, 10.1109/CNSR.2011.32. Ottawa, ON, Canada, May 2–5 2011.
9. G. Cervera, M. Barbeau, J. Garcia-Alfaro, and E. Kranakis. Preventing the Cluster Formation Attack Against the Hierarchical OLSR Protocol: Invited Talk. In proceedings of *4th Canada-France MITACS Workshop on Foundations & Practice of Security (FPS 2011*), Paris, France, May 12–13 2011. Springer LNCS.
10. T. Clausen and P. Jacquet. Optimized link state routing protocol (OLSR), RFC3626. IETF Internet Draft, Available via http://www.ietf.org/rfc/rfc3626.txt, October 2003.
11. T. Clausen, C. Dearlove and P. Jacquet. Optimized link state routing protocol version 2 (OLSRv2), RFC3666 ,Work in progress. Project Hipercom, INRIA, Internet Draft, http://bgp.potaroo.net/ietf/all-ids/draft-ietf-manet-olsrv2-13.txt, November 2011.
12. T. Clausen and C. Dearlove. RFC5497: Representing Multi-Value Time in Mobile Ad Hoc Networks (MANETs), std. track, http://www.ietf.org/rfc/rfc5497.txt.
13. T. Clausen, C. Dearlove, and B. Adamson. RFC5148: Jitter Considerations in Mobile Ad Hoc Networks (MANETs), informational, http://www.ietf.org/rfc/rfc5148.txt.
14. T. Clausen, C. Dearlove, and J. Dean. I-D: MANET Neighborhood Discovery Protocol (NHDP), work in progress.
15. T. Clausen, C. Dearlove, J. Dean, and C. Adjih. RFC5444: Generalized mobile ad hoc network (manet) packet/message format, std. track, http://www.ietf.org/rfc/rfc5444.txt.
16. T. Clausen and U. Herberg. Vulnerability analysis of the optimized link state routing protocol version 2 (OLSRv2). In *Wireless Communications, Networking and Information Security (WCNIS)*, 2010 IEEE International Conference on, pages 628–633, 2010.
17. T. Clausen, U. Herberg, and J. Milan. Digital signatures for admittance control in the optimized link state routing protocol version 2. Research Report RR-7216, INRIA, February 2010.

18. F. Cuppens, N. Cuppens-Boulahia, S. Nuon, and T. Ramard. Property based intrusion detection to secure OLSR. In ICWMC '07: Proceedings of the Third International Conference on Wireless and Mobile Communications, pages 52–59, Washington, DC, USA, 2007. IEEE Computer Society.
19. U. Herberg and T. Clausen. Security Issues in the Optimized Link State Routing Protocol version 2 (OLSRv2). *International Journal of Network Security & Its Applications (IJNSA)*, vol. 2(2), 2010.
20. F. Hong, L. Hong, and C. Fu. Secure OLSR. International Conference on Advanced Information Networking and Applications (AINA 2005), vol. 1, pages 713–718, Taipei, Taiwan, March 2005.
21. P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *IEEE International Multi Topic Conference, 2001*. IEEE INMIC 2001. Technology for the 21st Century. Proceedings, pages 62–68. Lahore University of Management Sciences, Pakistan, December 2001.
22. A. R. Khakpour, M. Laurent-Maknavicius, and H. Chaouchi. WATCHMAN: An overlay distributed AAA architecture for mobile ad hoc networks. In *ARES '08: Proceedings of the 2008 Third International Conference on Availability, Reliability and Security*, pages 144–152, Washington, DC, USA, March 2008. IEEE Computer Society.
23. M. Kun, Y. Jingdong, and R. Zhi. The research and simulation of multipath-OLSR for mobile ad hoc network. In *Communications and Information Technology, 2005*. ISCIT 2005. IEEE International Symposium on, vol. 1, pages 540–543, October 2005.
24. J. Moy. Open Shortest Path First (OSPF) version 2, RFC2328. IETF Internet Draft, http://www.ietf.org/rfc/rfc2328.txt, April 1998.
25. D. Raffo. Security schemes for the OLSR protocol for ad hoc networks. PhD thesis, LUniversité Paris 6 - Pierre et Marie Curie, INRIA Roquencourt, September 2005.
26. D. Raffo, C. Adjih, T. Clausen, and P. Muhlethaler. Securing OLSR using node locations. In *Proceedings of 2005 European Wireless (EW 2005)*, pages 437–443, Nicosia, Cyprus, April 10–13 2005.
27. F.J. Ros and P.M. Ruiz. Cluster-based OLSR extensions to reduce control overhead in mobile ad hoc networks. In *Proceedings of the 2007 international conference on Wireless communications and mobile computing*, pages 202–207. ACM, 2007.
28. A. Srinivas and E. Modiano. Minimum energy disjoint path routing in wireless ad-hoc networks. In *Proceedings of the 9th annual international conference on Mobile computing and networking (MobiCom 03)*, pages 122–133, New York, NY, USA, 2003. ACM.
29. J.P. Vilela and J. Barros. A feedback reputation mechanism to secure the optimized link state routing protocol. In *IEEE Communications International Conference on Security and Privacy for Emerging Areas in Communication Networks (Securecomm 2007)*. Los Alamitos (2007), pages 294–303, 2007.
30. L. Villasenor-Gonzalez, Y. Ge, and L. Lamont. HOLSR: A hierarchical proactive routing mechanism for mobile ad hoc networks. *IEEE Communications Magazine*, vol. 43(7), pages 118–125, July 2005.
31. M. Voorhaen, E. Van de Velde, and C. Blondia. MORHE: A transparent multi-level routing scheme for ad hoc networks. In *Challenges in Ad Hoc Networking*, K. Al Agha, I. Guérin Lassous, and G. Pujolle, editors, In *IFIP International Federation for Information Processing*, vol. 197, pages 139–148. Springer Boston, 2006.
32. B. Wu, J. Chen, J. Wu, and M. Cardei. A survey of attacks and countermeasures in mobile ad hoc networks. In *Wireless Network Security, Signals and Communication Technology*, Y. Xiao and X. S. Shen and D. Du, editors, pages 103–135. Springer US, 2007.
33. J. Yi, E. Cizeron, S. Hamma, and B. Parrein. Simulation and performance analysis of MP-OLSR for mobile ad hoc networks. In *IEEE Wireless Communications and Networking Conference*, IEEE WCNC, Las Vegas, March 31-April 3 2008.
34. J. Yi, E. Cizeron, S. Hamma, B. Parrein, and P. Lesage. Implementation of multipath and multiple description coding in OLSR. CoRR, abs/0902.4781, 2009.

35. J. Yi, S. David, H. Adnane, B. Parrein, and X. Lecourtier. Multipath OLSR: Simulation and Testbed. In *5th OLSR Interop/Workshop*, Vienna Autriche, 10 2009.
36. J. Yi, A. Adnane, S. David, and B. Parrein. Multipath optimized link state routing for mobile ad hoc networks. In *Ad Hoc Networks*, vol. 9(1), pages 28–47, 2011.
37. X. Zhou, Y. Lu, and B. Xi. A novel routing protocol for ad hoc sensor networks using multiple disjoint paths. In *Broadband Networks, 2005. BroadNets 2005. 2nd International Conference on*, vol. 2, pages 944–948, Boston, MA, October 2005.

Chapter 7
TCHo: A Code-Based Cryptosystem

Alexandre Duc and Serge Vaudenay

Abstract TCHo is a public-key cryptosystem based on the hardness of finding a multiple polynomial with low weight and on the hardness of distinguishing between the output of an LFSR with noise and some random source. An early version was proposed in 2006 by Finiasz and Vaudenay with non-polynomial (though practical) decryption time. The latest version came in 2007 with more co-authors. It reached competitive (heuristic) polynomial complexity and IND-CPA security. Since then, a key-recovery chosen ciphertext attack was published by Herrmann and Leander in 2009. In this paper we review the state of the art on this cryptosystem, together with some latest improvements regarding implementation and selection of parameters. We provide also more formal results regarding correctness and we update the key generation algorithm.

7.1 Introduction

Public-key cryptography first appeared with the seminal paper of Diffie and Hellman in 1976 [21]. From this work, Rivest, Shamir, and Adleman presented the RSA cryptosystem in 1978 [60], which is still the mostly used one nowadays. Among the popular cryptosystems, there are the Rabin cryptosystem [56], which is very close to RSA, and the ElGamal family of cryptosystems [25].

Every public-key cryptosystem relies on problems that are believed to be computationally infeasible. As far as we know, all cryptosystems which are used in practice rely on two problems: the integer factorization problem [56, 60] and the discrete logarithm problem [25]. However, these two problems can easily be solved

A. Duc (✉) • S. Vaudenay
EPFL, Lausanne, Switzerland
e-mail: alexandre.duc@epfl.ch

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_7,

in polynomial time on a *quantum computer* using Shor's algorithm [62] and its generalizations [32]. Hence, if one can build a quantum computer with sufficiently many qubits to solve these problems, the mostly used public-key cryptographic systems will be broken and will have to be replaced.

To be prepared for this, we need *crypto-diversity*. Then, if one cryptosystem is broken, another ideally well-studied cryptosystem will be available for use. In particular, some of these systems should be secure on quantum computers as well. Such cryptosystems are referred to *post-quantum cryptosystems*. Nowadays, this has become a hot topic and dozens of quantum-resistant schemes have been designed.

Several types of post-quantum cryptosystems have been proposed. Some are based on multivariate equations [22, 36, 45, 52–54], whereas some others are code-based [3, 29, 46, 50] or lattice-based [1, 2, 30, 34, 44, 55, 57, 58]. The former are, to the best of our knowledge, used only to design signature schemes. In the following, we focus on *code-based* and *lattice-based* cryptosystems.

7.1.1 Code-Based Cryptosystems

Code-based cryptosystems rely on error correcting codes and the addition of random noise during the encryption.

The most famous code-based cryptosystem is the McEliece cryptosystem [46]. It was introduced by McEliece in 1978 and is still unbroken. In this scheme, the private key is the generator matrix G of a random $[k,n]$-Goppa code able to correct up to t errors along with two matrices P and S, where P is a random $k \times k$ permutation and S a random $n \times n$ non-singular matrix. The public key is then a scrambled version $\hat{G}$ of G, defined as $\hat{G} := SGP$. A message is encrypted by first encoding it with the code associated to $\hat{G}$ and by adding a random noise with exactly t ones. The security of the system relies on the hardness of decoding a scrambled Goppa code. This problem is also known as the *McEliece problem*. Decryption of a ciphertext can be performed efficiently if one is in possession of P, S, and G, since Goppa codes are efficiently decodable.

Niderreiter described a dual variant of the McEliece cryptosystem [50]. Instead of representing the message as a codeword, the encryption is performed with the parity check matrix H of a Goppa code. The security of the two schemes have been shown to be equivalent [39]. A signature scheme [19] was derived from the Niderreiter cryptosystem by Courtois et al.

A code-based symmetric key cipher, LPN-C [29], was introduced by Gilbert et al. This cipher is based on another problem which is believed to be hard: the *Learning from Parity with Noise* problem (LPN). The LPN problem consists in finding an unknown k-bit vector $\mathbf{x}$ given access to an oracle that returns $(\mathbf{a}, \mathbf{a} \cdot \mathbf{x} + \nu)$, for some biased noise ν and some random vector $\mathbf{a}$.

The main drawback of these schemes is that the key length needed to obtain reasonable security is pretty large. For the McEliece cryptosystem, public keys of size 2^{16} achieve only 84.88-bit security [9]. To obtain 266.94-bit security, 2^{20}-bit keys are needed. Note that, asymptotically, the McEliece cryptosystem has better key sizes than RSA. For a security parameter λ, a McEliece key has size $\lambda^{2+o(1)}$ while RSA keys have size $\lambda^{3+o(1)}$ [8], but this holds only for impractical λ. Another drawback of these schemes is that the ciphertext length has to be bigger than the plaintext. This comes from the use of an error-correcting code and cannot be changed.

TCHo belongs also to the code-based cryptosystem category. However, its security rely on a somewhat different problem: the low weight polynomial multiple problem.

7.1.2 Lattice-Based Cryptosystems

The other category of post-quantum cryptosystems are lattice-based cryptosystems. One of the strengths of lattice-based cryptography is that its security is often based on the *worst-case hardness* of problems instead of average-case hardness.

One of the computationally hard problems on which lattice-based systems rely on is the *Shortest Vector Problem* (SVP). Briefly, this problem consists of finding the shortest non-zero vector in a lattice or an approximation of it within a polynomial factor. Polynomial algorithms like LLL [38] or its improvements [61] can only find subexponential approximations of it.

Lattice-based cryptography was introduced by Ajtai in 1996 [1]. Shortly after, Ajtai and Dwork designed the first lattice-based public-key cryptosystem based on the worst-case hardness of SVP [2]. This scheme was later improved in [30, 57]. However, they suffer from large key sizes and a large expansion factor, and are inefficient in practice. Indeed, for a lattice of dimension n, the keys in this scheme have size $\widetilde{O}\left(n^4\right)$ and the ciphertexts $\widetilde{O}\left(n^2\right)$ [59].[1]

Another class of lattice-based cryptosystems are based on the worst-case complexity of the *Learning With Errors* (LWE) problem [44,55,58]. The LWE problem is the following: given a dimension n, a modulus q and an error distribution χ over $\mathbb{F}_q$, the goal is to find a secret vector $s \in \mathbb{F}_q^n$ using independent LWE-samples:

$$(a, \langle a, s\rangle + \varepsilon) \in \mathbb{F}_q^n \times \mathbb{F}_q, \quad a \overset{U}{\leftarrow} \mathbb{F}_q^n, \quad \varepsilon \overset{\chi}{\leftarrow} \mathbb{F}_q.$$

Reference [44] introduces the *ring-LWE* problem, an algebraic variant of the LWE problem. According to the authors, it is the first truly practical lattice-based cryptosystem based on LWE.

The most famous and efficient lattice-based cryptosystem is NTRU [34] which is based on the work of Goldreich et al. [31]. Its security is based on the hardness

[1] A function is $\widetilde{O}(f(n))$ if it is $O\left(f(n) \cdot \log(f(n))^k\right)$ for some k.

of SVP and the Closest Vector Problem (CVP) in convolution modular lattices, a particular class of lattices. Unlike some schemes we named above, NTRU's security has not been shown equivalent to the hardness of SVP or CVP. Nevertheless, for a security parameter λ, the asymptotic cost for encryption and decryption is $O\left(\lambda^2\right)$ and the key sizes is $O(\lambda)$ which makes of NTRU one of the most efficient public-key cryptosystems.

7.1.3 TCHo

It is often the case in stream cipher cryptanalysis that we need to cancel the effect of a linear feedback shift register with noise by using a low weight multiple of its connection polynomial. This happens in fast correlation attacks [47]. For instance, in the cryptanalysis of Bluetooth E0 [40–43], we need to find such a multiple with low weight for a given polynomial coming from the E0 specifications. Actually, the lower the degree and the weight, the more efficient the attack. This happens to be hard in practice. However, the designer of E0 could have selected the polynomial as a factor of some secret low weight multiple. That is, a trapdoor could have been hidden to break the cipher. Refining this idea, in 2006, Finiasz and Vaudenay [26] came up with the notion of *trapdoor cipher* on which TCHo is based. Indeed, the name "TCHo" stands for "Trapdoor Cipher, Hardware Oriented".[2] This early version of TCHo was using a linear code based on another LFSR.

One drawback of this design was that decryption (using the trapdoor) was not polynomially bounded, although still feasible in practice. Then, Meier suggested using other codes. Finally, in 2007, Aumasson et al. presented the latest version [3] with polynomial complexity using heuristic algorithms. They proved semantic security based on some new complexity assumptions. They further proposed to apply the Fujisaki-Okamoto construction [27] to achieve $\mathsf{IND\text{-}CCA}$ security.

In 2009, Herrmann and Leander [33] have shown that we can mount a key recovery chosen ciphertext attack, which seemingly proves that the key recovery problem and the decryption problem are somewhat equivalent.

Since then, Duc [23] has shown how to generate better parameter vectors, and Bindschaedler [10] implemented it as a new cipher in TLS for a browser and an HTTP server.

In this paper, we survey known results about TCHo. Additionally, we provide more formal (i.e. non-heuristic) results regarding correctness, with an updated key generation algorithm.

[2] The word "tchô" happens to come from some French slang which originated from the famous Swiss cartoonist Zep who created a comics magazine for kids with this name in 1998.

7.1.4 Structure of This Paper

In Sect. 7.2 we describe our notation and give basic definitions used throughout the paper. In Sect. 7.3 we present the problems on which the security of TCHo relies on and we survey algorithms that solve them. The complexity of these algorithm will be needed to find secure parameters for TCHo. In Sect. 7.4 we present the TCHo cipher and prove that it is a cryptosystem with heuristic key generation. In Sect. 7.5 we discuss the security of TCHo, we prove that TCHo is IND-CPA secure and we show how to achieve IND-CCA security. In Sect. 7.6 we give some practical parameters for TCHo and we discuss some implementation results. We conclude in Sect. 7.7.

7.2 Notations and Definitions

We denote by "log" the logarithm in base two and by "ln" the natural logarithm. We write $x \overset{U}{\leftarrow} \mathcal{D}$ if an element x is drawn uniformly at random in a domain $\mathcal{D}$. We write $x \overset{\chi}{\leftarrow} \mathcal{D}$ if x is drawn from domain $\mathcal{D}$ using distribution χ. For TCHo, we consider only binary polynomials, i.e., polynomials with coefficients in $\mathbb{F}_2$. The *degree* of a polynomial $P \in \mathbb{F}_2[x]$ is denoted d_P. We use uppercase characters to represent polynomials and the same letter in lowercase to represent its coefficients. Hence, we write $P = p_0 + p_1X + p_2X^2 + \cdots + p_{d_P}X^{d_P}$. The number of nonzero coefficients of P is called the *weight* of the polynomial and is denoted w_P. In other words, $w_P = \sum_{i=0}^{d_P} p_i$, where p_i's are considered as elements in $\mathbb{Z}$. A polynomial P with $w_P \ll d_P$ is called a *sparse* polynomial or a *low weight* polynomial.

The *bias* γ of a random bit B is the difference between the probability of occurrence of a zero and the probability of occurrence of a one, i.e., $\gamma = \Pr[B = 0] - \Pr[B = 1]$. Hence, a source producing random bits with bias γ outputs a zero with probability $\frac{1}{2}(1+\gamma)$ and a one with probability $\frac{1}{2}(1-\gamma)$. We call a finite sequence of bits x a *bitstring*. We write its *length* $|x|$, which denotes its number of bits. As for polynomials, we call the weight of a bitstring its number of ones. The concatenation of two bitstrings x and y is written $x\|y$. The (possibly infinite) output of a *bitsource* S is called a *bitstream*. If we need to specify the input (e.g. the *seed*) r of a source S, we write $\mathsf{S}(r)$. The bitstring constructed from the first ℓ bits produced by S is denoted S^{ℓ}. We denote by S_{γ} a bitsource producing independent bits with bias γ. Given a bitstring x, we denote by $\mathsf{trunc}_{\ell}(x)$ the substring of x made by its first ℓ bits.

Given some initial parameters Π and a predicate P, we write

$$\Pr\left[P(v_1,\ldots,v_m;r_p) : \begin{array}{l} v_1 \leftarrow f_1(\Pi;r_1) \\ \vdots \\ v_m \leftarrow f_m(\Pi,v_1,\ldots,v_{m-1};r_m) \end{array} \right]$$

to denote

$$\Pr_{\substack{r_1,\dots,r_m\\ r_p}}\left[\bigvee_{v_1,\dots,v_m} P(v_1,\dots,v_m;r_p)\wedge v_1 = f_1(\Pi;r_1)\wedge\cdots\wedge v_m = f_m(\Pi,v_1,\dots,v_m;r_m)\right].$$

A *Linear Feedback Shift Register* (LFSR) can be described by its *feedback polynomial* $P = \sum_{i=0}^{d_P} p_i X^i$. It is then denoted $\mathcal{L}_P$. When given an initial state $s = (s_0, s_1, \dots, s_{d_P-1})$, an LFSR $\mathcal{L}_P$ produces a bitstream denoted $\mathrm{S}_{\mathcal{L}_P}(s)$. Recall that an LFSR with feedback polynomial P and initial state $s = (s_0, s_1, \dots, s_{d_P-1})$ produces the bitstream s_i with $s_{i+d_P} = \sum_{k=0}^{d_P-1} p_k s_{i+k}$ in $\mathbb{F}_2$.

Finally, we define two operations used in TCHo. The *bitwise sum* (in $\mathbb{F}_2$) of two bitstrings x and y of same length is written $x+y$. The *product* of a polynomial $K \in \mathbb{F}_2[X]$ of degree d, $K = \sum_{j=0}^{d} k_j X^j$ and a bitstring $\mathrm{S}^{d+N} = (s_0, \dots, s_{N+d-1})$ is denote $K \otimes \mathrm{S}^{d+N}$ and is defined as

$$K \otimes \mathrm{S}^{d+N} = (s'_0, \dots, s'_{N-1}),$$

with $s'_i := s_i k_0 + s_{i+1} k_1 + \cdots + s_{i+d} k_d$. We can also associate the polynomial K with an $N \times (d+N)$ matrix M_K^N defined as

$$M_K^N := \begin{bmatrix} k_0 & k_1 & \dots & k_d & 0 & 0 & \dots & 0 \\ 0 & k_0 & k_1 & \dots & k_d & 0 & \dots & 0 \\ & & \ddots & & & \ddots & & \\ 0 & 0 & \dots & 0 & k_0 & k_1 & \dots & k_d \end{bmatrix}.$$

Then, we have

$$\left[s'_0 \, \dots \, s'_{N-1}\right]^T = M_K \left[s_0 \, \dots \, s_{N+d-1}\right]^T.$$

Note that $P \otimes \mathrm{S}^{\ell}_{\mathcal{L}_P} = \mathbf{0}$, i.e., when the feedback polynomial is used for the multiplication, we obtain the zero bitstring. This multiplication operator verifies also $(PQ) \otimes \mathrm{S} = P \otimes (Q \otimes \mathrm{S})$. Thus, if P divides K, $K \otimes \mathrm{S}^{\ell}_{\mathcal{L}_P} = \mathbf{0}$.

A function $f(\lambda)$ is *negligible* if for all $d \in \mathbb{R}$ we have $f(\lambda) = O\left(\lambda^{-d}\right)$.

Definition 7.1 (Cryptosystem). A cryptosystem over a given message space $\mathcal{M}$ and random coin space $\mathcal{R}$ consists of three *polynomial-time* algorithms:

- *A probabilistic key-generation algorithm* $\mathsf{Gen}(1^\lambda)$ taking as input some security parameter 1^λ in unary representation, and producing a secret key K_s and a public key K_p;
- *A probabilistic encryption algorithm* $\mathsf{Enc}(K_p, m; r)$ taking as input a public key K_p and a message $m \in \mathcal{M}$ with some random coins $r \in \mathcal{R}$, and producing a ciphertext y in the ciphertext space $\mathcal{C}$;
- *A deterministic decryption algorithm* $\mathsf{Dec}(K_s, c)$ taking as input a secret key K_s and a ciphertext $c \in \mathcal{C}$, and producing a message or an error.

The cryptosystem must satisfy the following correctness property:

$$\max_{m \in \mathcal{M}} \Pr\left[\mathsf{Dec}(K_s, \mathsf{Enc}(K_p, m; \rho)) \neq m : \quad (K_s, K_p) \leftarrow \mathsf{Gen}(1^\lambda; \rho_g)\right]$$

is negligible as λ increases.

We will also use the following security notions and acronyms. Adaptive Chosen Ciphertext Attack is denoted CCA, Chosen Plaintext Attack CPA, Indistinguishability IND, and One-wayness OW.

Definition 7.2 (OW-CCA-security). A cryptosystem is (t,ε)-OW-CCA-secure if no adversary $\mathcal{A}$, with access to a decryption oracle $O_{K_s,c}$ and with running time bounded by t, can recover the plaintext from a given ciphertext with a probability higher than ε. More formally, for all $\mathcal{A}$ bounded by t,

$$\Pr\left[\mathcal{A}^{O_{K_s,c}}(c;\rho) = m \; : \; \begin{array}{r} m \overset{U}{\leftarrow} \mathcal{M};\ r \overset{U}{\leftarrow} \mathcal{R} \\ (K_s, K_p) \leftarrow \mathsf{Gen}(1^\lambda) \\ c \leftarrow \mathsf{Enc}(K_p, m; r) \end{array}\right] \leq \varepsilon,$$

where $O_{K_s,c}(y) = \mathsf{Dec}(K_s, y)$ for $y \neq c$ and $O_{K_s,c}(y) = \perp$ otherwise. Asymptotically, a cryptosystem is OW-CCA-secure if for any polynomial $t(\lambda)$ there exists a negligible function $\varepsilon(\lambda)$ such that it is $(t(\lambda), \varepsilon(\lambda))$-OW-CCA-secure.

Definition 7.3 (IND-CPA-security). A cryptosystem is said (t,ε)-IND-CPA-secure or (t,ε)-*semantically secure* against chosen plaintext attacks if no adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ with running time bounded by t can distinguish the encryption of two different plaintexts m_0 and m_1 with a probability higher than ε. More formally, for all $\mathcal{A}$ bounded by t,

$$\Pr\left[\mathcal{A}_2(K_p, c; \rho) = b \; : \; \begin{array}{l} (K_s, K_p) \leftarrow \mathsf{Gen}(1^\lambda) \\ m_0, m_1 \leftarrow \mathcal{A}_1(K_p; \rho) \\ r \overset{U}{\leftarrow} \mathcal{R};\ b \overset{U}{\leftarrow} \{0,1\} \\ c \leftarrow \mathsf{Enc}(K_p, m_b; r) \end{array}\right] \leq \frac{1}{2} + \varepsilon.$$

Asymptotically, a cryptosystem is IND-CPA-secure if for any polynomial $t(\lambda)$ there exists a negligible function $\varepsilon(\lambda)$ such that it is $(t(\lambda), \varepsilon(\lambda))$-IND-CPA-secure.

IND-CPA-security can also be represented in the real-or-random game model [5,6].

Definition 7.4 (Real-or-random IND-CPA game security). A cryptosystem is (t,ε)-IND-CPA-secure in the real-or-random game model if no adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ with running time bounded by t can distinguish the encryption of a chosen plaintexts m_0 to a random one with a probability higher than ε. More formally, for all $\mathcal{A}$ bounded by t,

$$\Pr\left[\mathcal{A}_2(K_p, c; \rho) = b \ : \ \begin{array}{c} (K_s, K_p) \leftarrow \mathsf{Gen}(1^\lambda) \\ m_0 \leftarrow \mathcal{A}_1(K_p; \rho);\ m_1 \xleftarrow{U} \mathcal{M} \\ r \xleftarrow{U} \mathcal{R};\ b \xleftarrow{U} \{0,1\} \\ c \leftarrow \mathsf{Enc}(K_p, m_b; r) \end{array}\right] \le \frac{1}{2} + \varepsilon.$$

Asymptotically, a cryptosystem is IND-CPA-secure in the real-or-random game model if for any polynomial $t(\lambda)$ there exists a negligible function $\varepsilon(\lambda)$ such that it is $(t(\lambda), \varepsilon(\lambda))$-IND-CPA-secure in the real-or-random game model.

A (t, ε)-IND-CPA-secure system in the real-or-random game model is $(t, 2\varepsilon)$-IND-CPA-secure in the standard model [5]. Conversely, a (t, ε)-IND-CPA-secure system in the standard model is (t, ε)-IND-CPA-secure in the real-or-random game model. Asymptotically, both models are equivalent.

Definition 7.5 (IND-CCA-security). A cryptosystem is said (t, ε)-IND-CCA-secure or (t, ε)-*secure* against adaptive chosen plaintext attacks if no adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, with access to a decryption oracle $O_{K_s,c}$ and with running time bounded by t can distinguish the encryption of two different plaintexts m_0 and m_1 with a probability higher than ε. More formally, for all $\mathcal{A}$ bounded by t,

$$\Pr\left[\mathcal{A}_2^{O_{K_s,c}}(K_p, c; \rho) = b \ : \ \begin{array}{c} (K_s, K_p) \leftarrow \mathsf{Gen}(1^\lambda) \\ m_0, m_1 \leftarrow \mathcal{A}_1^{O_{K_s}}(K_p; \rho) \\ r \xleftarrow{U} \mathcal{R};\ b \xleftarrow{U} \{0,1\} \\ c \leftarrow \mathsf{Enc}(K_p, m_b; r) \end{array}\right] \le \frac{1}{2} + \varepsilon,$$

where $O_{K_s,c}(y) = \mathsf{Dec}(K_s, y)$, and $O_{K_s,c}(y) = \mathsf{Dec}(K_s, y)$ for $y \neq c$ and $O_{K_s,c}(c) = \perp$. Asymptotically, a cryptosystem is IND-CCA-secure if for any polynomial $t(\lambda)$ there exists a negligible function $\varepsilon(\lambda)$ such that it is $(t(\lambda), \varepsilon(\lambda))$-IND-CCA-secure.

Definition 7.6. Given two sources S_0 and S_1, a *distinguisher* between them is an algorithm $\mathcal{D}$ that takes as input one sample x from either S_0 or S_1 and has to decide which source was used. Its *advantage* is

$$\mathrm{Adv}_{\mathcal{D}}(\mathrm{S}_0, \mathrm{S}_1) = \Pr[\mathcal{D}(x) = 1 : x \leftarrow \mathrm{S}_1] - \Pr[\mathcal{D}(x) = 1 : x \leftarrow \mathrm{S}_0].$$

We say that the two sources are (t, ε)-*computationally indistinguishable* if for any distinguisher $\mathcal{D}$ with running time bounded by t,

$$|\mathrm{Adv}_{\mathcal{D}}(\mathrm{S}_0, \mathrm{S}_1)| \le \varepsilon.$$

Asymptotically, two sources are *computationally indistinguishable* if for any polynomial $t(\lambda)$ there exists a negligible function $\varepsilon(\lambda)$ such that, they are $(t(\lambda), \varepsilon(\lambda))$-computationally indistinguishable.

We will be using the following Chernoff bound:

Theorem 7.1 (Chernoff [14]). *For $X_1, \ldots, X_n$ independent identically distributed Bernoulli random variables with $E(X_1) \leq \frac{1}{2}$,*

$$\Pr\left[\frac{1}{n}\sum_{i=1}^{n} X_i \geq \frac{1}{2}\right] \leq \exp\left(-2n\left(E(X_1) - \frac{1}{2}\right)^2\right).$$

7.3 Computational Problems

TCHo 's security is based on computational problems which are believed to be hard. In this section, we survey the existing algorithms used to solve these problems and we focus on their best complexity to find suitable parameters for TCHo.

7.3.1 Low Weight Polynomial Multiple Problem

Unlike the integer factorization, efficient algorithms exist to factor polynomials over a finite field [7, 13]. However, finding a multiple of a given polynomial that has a bounded degree and a bounded weight can be hard. TCHo 's security relies on the *Low Weight Polynomial Multiple problem* (LWPM).

Problem 7.1 (LWPM). Let $w, d, d_P \in \mathbb{N}$ be three parameters such that $0 < d_P < d$ and $w < d$. Given an instance $P \in \mathbb{F}_2[x]$ of degree d_P, find a multiple K of P of degree at most d and weight at most w.

In TCHo, P will be the public key and one will be able to recover some information about the plaintext from a ciphertext using such a low weight polynomial K. Note that in TCHo, the private key will be one of the solutions to this problem. Hence, we are ensured that a solution exists. In fact, the public key P is generated as an irreducible factor of a random polynomial K of weight w_K and degree at most d with a nonzero constant factor. In this case, one can heuristically estimate the number of solutions with nonzero constant term as

$$\mathcal{N}_{\text{sol}} \approx 1 + 2^{-d_P}\binom{d}{w-1}. \tag{7.1}$$

Note that the additional solution comes from the way we generate our polynomial P. We present now algorithms used to find low weight polynomial multiples. In what follows, we review some existing algorithms to solve the LWPM problem in order to derive heuristically some hard domain parameters.

To support the hardness of the LWPM problem, Giesbrecht et al. showed that finding sparse multiple polynomials *with unbounded degree* in a finite field is at least as hard as computing orders in an extension of this field [28], a problem which is believed to be hard. Unfortunately, this result is not directly applicable to TCHo because we consider polynomially bounded degrees and we know that a solution exists within this bound.

7.3.1.1 Exhaustive Search

When d is close to d_P, we can use exhaustive search to find *all* low weight multiples of P. The exhaustive search is performed by simply checking the weight of all multiples of P. The complexity for finding all multiples is $\Theta\left(\mathrm{Poly}(d)2^{d-d_P}\right)$. However, this method is inefficient in TCHo, since $d-d_P$ is very large.

7.3.1.2 Birthday Paradox

Following Meier and Staffelbach [47], we build two lists L_1 and L_2 in which we store respectively polynomials with weight $\lfloor (w-1)/2 \rfloor$, degree smaller than d, and zero constant term and polynomials with weight $\lceil (w-1)/2 \rceil$, degree smaller than d, and constant term equal to one. Once we have these lists, we look for pairs that sum to 0 modulo P. This collision search can be done efficiently using a hash table. Note that when w is odd (as in TCHo), one can use only the list L_1 and search for pairs (in L_1) summing to 1 instead. The list size is $\binom{d}{\lceil (w-1)/2 \rceil}$. Hence, the memory use is

$$2\binom{d}{\lceil (w-1)/2 \rceil}(w-1)\log(d) \approx \Theta\left(d^{\lceil (w-1)/2 \rceil}\log(d)\right)$$

for small weights w. The time complexity is $\Theta\left(\binom{d}{(w-1)/2}\right)$. This strategy is clearly faster than exhaustive search but uses a lot of memory. In the case of TCHo, d is typically greater than 2^{15} and w greater than 68. Hence, the lists contains more than $\Omega\left(2^{388}\right)$ elements, which is way too much. An improvement of this method is proposed by Chose, Joux and Mitton to solve this problem using $\Theta\left(d^{\lceil (w-1)/4 \rceil}\log(d)\right)$ space instead [15]. An alternative solution was also proposed by Didier and Laigle-Chapuy using discrete logarithms [20]. Assuming that in practice the discrete logarithm with base element X has a negligible complexity over $\mathbb{F}_2[X]/\langle P \rangle$, they achieve a time complexity of $\Theta\left(d^{(w-1)/2-1}\right)$ for a memory cost equal to the original birthday paradox method.

7.3.1.3 Generalized Birthday Paradox

When there is a large number of solutions, one can use Wagner's generalized birthday paradox [64] to find more efficiently one solution. The idea is to make use of 2^k lists of polynomial of weight $(w-1)/2^k$ instead of two lists as in the birthday paradox algorithm. Collisions are then found in pairs of lists until one single list remains containing a wanted solution. This algorithm will not return all possible solutions but can find one of them. However, the lists need to have a size greater than $2^{d_P/(k+1)}$. Hence, we need

$$\binom{d}{(w-1)/2^k} \geq 2^{d_P/(k+1)},$$

for a $k > 1$. In this case, a solution can be found with time and memory complexity

$$\Theta\left(2^k 2^{d_P/(k+1)}\right).$$

7.3.1.4 Finding a Low Weight Multiple Using Lattices

El Ailmani and von zur Gathen [24] presented a lattice-based algorithm to solve the LWPM problem. The set of multiples of P with degree lower than d form a lattice L_d. More formally

$$L_d := \{Q \in \mathbb{Z}[X] : Q \in P\mathbb{Z}[X] + 2\mathbb{Z}[X], \deg(Q) < d\}.$$

This lattice has dimension d. Now, note that a low weight polynomial multiple of P is a short vector in L_d. Such a vector can be found by first finding a basis of L_d, then by reducing this basis using for instance the LLL algorithm [38]. The algorithm uses $O\left(d^6\right)$ time and $O\left(d \times d_P\right)$ memory if we use LLL. However, this method is strongly limited by the lattice dimension. When d is too large, the size of the short vector we find using LLL becomes greater than w. This comes from the fact that this vector is only an approximation of the shortest vector in the lattice. Hence, this technique is inefficient to attack TCHo.

7.3.1.5 Syndrome Decoding

Syndrome decoding can also solve this problem. First, we compute the matrix H, whose column i is defined by $X^i \bmod P$, for $1 \leq i \leq d-1$. Once this matrix is computed, we search for a low weight polynomial in the preimages of 1 of this matrix. Following Canteaut and Chabaud [11], one solution can be found in time

$$\Theta\left(\frac{1}{\mathcal{N}_{\text{sol}}}\left(\frac{d-1}{d_P}\right)^{w-1}\right).$$

By using (7.1), this approximates to

$$\Theta\left(\frac{\left(\frac{d-1}{d_P}\right)^{w-1}}{2^{-d_P}\binom{d}{w-1}+1}\right).$$

7.3.1.6 Hardness of the LWPM Problem

Out of this survey, we deduce the following assumption

Assumption 7.2. *Let w, d, P be an instance of the LWPM problem. Let λ be the security parameter. The LWPM problem is believed to require a super-polynomial complexity if*

$$(w-1)\log\frac{d}{d_p} \geq \lambda, \tag{7.2}$$

and

$$\binom{d}{w-1} < 2^{d_P}. \tag{7.3}$$

Indeed, these inequalities give no complexity better than $\Theta\left(2^{\lambda}\right)$ with the previous algorithms. Note that (7.3) implies that we shall expect no more solution than the one which was hidden.

7.3.2 The Noisy LFSR Decoding Problem

TCHo 's security relies also on the *noisy LFSR decoding problem.*

Problem 7.2 (Noisy LFSR Decoding). Let $\ell > 0$ be a length, let P be a polynomial of degree d_P, and let $0 \leq \gamma \leq 1$ be a bias. Recover X, the random seed of an LFSR, given $Y := \mathrm{S}^{\ell}_{L_P}(X) + \mathrm{S}^{\ell}_{\gamma}$, i.e., the bitwise addition between the output of this LFSR with feedback polynomial P and seed X, and some random noise with bias γ.

In TCHo, the plaintext will be hidden by such a Y. Hence, since the noise is strongly biased, if one can easily recover the seed of the LFSR, one can recover the plaintext. We survey now the techniques used to solve the noisy LFSR decoding problem.

7.3.2.1 Information Set Decoding

Information set decoding is performed as follows. We pick d_P random bits out of the ℓ output bits of Y and we solve the linear system induced by the columns of the generator matrix of the LFSR corresponding to these bits (e.g. by performing

Gaussian elimination). X can be recovered if there are no errors among the d_P selected bits. This happens with probability

$$\left(\frac{1}{2}+\frac{\gamma}{2}\right)^{d_P}.$$

Hence, we can recover X with complexity[3]

$$\Theta\left(\frac{1}{2}+\frac{\gamma}{2}\right)^{-d_P}. \tag{7.4}$$

This is $\Omega(2^{\lambda})$ for

$$\gamma \leq 2^{1-\lambda/d_P}-1.$$

7.3.2.2 Maximum Likelihood Decoding

Maximum likelihood (ML) decoding is a bruteforce technique that consists in computing $\mathrm{S}^{\ell}_{\mathcal{L}_P}(X)$ for all possible random seeds X and keep the one with smallest distance to Y. This costs $\Theta\left(2^{d_P}\ell\right)$ and can be improved to $\Theta\left(2^{d_P}d_P\right)$ by using a fast Walsh transform [42]. More subtle ML algorithms exist that decode only a subcode of the full code. In this case, even though we do not recover completely the seed X, we may recover some bits of information which may threaten TCHo. It can be shown [26] that if we take $d_P \geq 2\lambda$, these algorithms yield less than 1 bit of information.

7.3.2.3 Iterative Decoding

Iterative decoding [12] consists in finding low weight multiples of P forming parity check equations. The low-weight parity check code associated to these equations can then be solved. When $d_P \geq 2\lambda$, decoding is not possible using this technique [26].

7.3.2.4 Hardness of the Noisy LFSR Decoding Problem

Out of this survey, we deduce the following assumption.

Assumption 7.3. *Let ℓ, P, γ be an instance of the noisy LFSR decoding problem. Let λ be the security parameter. The problem is believed to require a super-polynomial complexity if*

$$d_P \geq 2\lambda, \tag{7.5}$$

[3] Note that we can neglect the cost of the Gaussian elimination by using improved algorithms [11].

and

$$\gamma \leq 2^{1-\lambda/d_P} - 1. \tag{7.6}$$

Indeed, these inequalities give no complexity better than $\Theta\left(2^{\lambda}\right)$ with the previous algorithms.

7.3.3 The Noisy LFSR Distinguishing Problem

The previous problem has a decisional counterpart.

Problem 7.3 (Noisy LFSR Distinguishing). Let $\ell > 0$ be a length, let P be a polynomial of degree d_P, and let $0 \leq \gamma \leq 1$ be a bias. Given an ℓ-bit string Y, decide whether Y was generated by $Y = \mathrm{S}^{\ell}_{\mathcal{L}_P}(X) + \mathrm{S}^{\ell}_{\gamma}$ or by a uniformly distributed source $Y = \mathrm{S}^{\ell}_{0}$.

7.3.3.1 Linear Noise Cancellation

We can think of two strategies to distinguish $\mathrm{S}^{\ell}_{*} = \mathrm{S}^{\ell}_{\mathcal{L}_P} + \mathrm{S}^{\ell}_{\gamma}$ from $\mathrm{S}^{\ell}_{*} = \mathrm{S}^{\ell}_{0}$. The first one is to apply the solutions we presented to solve the noisy LFSR decoding problem, i.e., to recover the random seed used by $\mathcal{L}_P$. We know that if (7.5) and (7.6) hold, the problem is supposed hard.

Another solution is to multiply S^{ℓ}_{*} by P or any of its multiple Q. If S^{ℓ}_{*} is not S^{ℓ}_{0}, we have

$$Q \otimes \mathrm{S}^{\ell}_{*} \approx \mathrm{S}^{\ell-d_Q}_{\gamma^{w_Q}}.$$

The best advantage [4] one can get to distinguish N bits of bias γ^{w} from random ones is $\mathrm{Adv} \approx \gamma^{w}\sqrt{N/(2\pi)}$. From Sect. 7.3.1, we know that the cost to find a multiple of P with degree bounded by d and weight bounded by w is $\mathrm{Comp} = (d/d_P)^{w-1}$ if we use syndrome decoding. It works with probability bounded by $\mathrm{Success} = 2^{-d_P}\binom{d}{w-1}$.

To identify the range of parameters for which this method does not work, we want

$$\frac{1}{\mathrm{Adv}} + \frac{\mathrm{Comp}}{\mathrm{Success}} \leq 2^{\lambda}.$$

So, N, w, d_P are polynomially bounded. We have

$$\frac{1}{\mathrm{Adv}} + \frac{\mathrm{Comp}}{\mathrm{Success}} = \frac{\gamma^{-w}}{\sqrt{N/2\pi}} + \frac{(d/d_P)^{w-1}}{2^{-d_P}\binom{d}{w-1}} \geq \mathrm{Poly}(\lambda)\left(\frac{\gamma^{-w}}{\sqrt{N/2\pi}} + \left(\frac{we^{-1}}{d_P}\right)^{w} 2^{d_P}\right).$$

Let $f(w)$ be the term under parentheses. The function $w \mapsto \left(we^{-1}/d_P\right)^{w}$ decreases until $w = d_P$ and then increases. Let τ be the root of $\gamma e^{-1} = \tau 2^{-\tau}$. Since we will take $\gamma = 1 - o(1)$, we have $\tau = \tau_0 + o(1)$ where $\tau_0 :\approx 3.053$. Note that $\tau \geq 1$ since $\gamma \leq 1$.

We assume that $d_P \log(1/\gamma) \geq \lambda\tau$. Let $w_0 := d_P/\tau$. If $w \leq w_0$, since $w < d_P$ (because $\tau \geq 1$), we have

$$f(w) \geq \left(\frac{w\mathrm{e}^{-1}}{d_P}\right)^w 2^{d_P} \geq \left(\frac{w_0\mathrm{e}^{-1}}{d_P}\right)^{w_0} 2^{d_P} = \left(2\left(\frac{1}{\tau}\mathrm{e}^{-1}\right)^{1/\tau}\right)^{d_P} = \gamma^{-d_P/\tau} \geq 2^{\lambda}.$$

If $w \geq w_0$, we have

$$f(w) \geq \gamma^{-w} \geq \gamma^{-w_0} = \gamma^{-d_P/\tau} \geq 2^{\lambda}.$$

This leads us to the following assumption:

Assumption 7.4. *Let ℓ, P, γ be an instance of the noisy LFSR distinguishing problem. Let λ be the security parameter. Let τ be the root of $\gamma\mathrm{e}^{-1} = \tau 2^{-\tau}$. The problem is believed to require a super-polynomial complexity if*

$$\gamma \leq 2^{1-\lambda/d_P} - 1,$$

and

$$d_P \log\frac{1}{\gamma} \geq \lambda\tau.$$

Indeed, these inequalities give no complexity better than $\Theta\left(2^{\lambda}\right)$ for advantage $\Omega(2^{-\lambda})$ with the previous algorithms.

Note that if $d_P < 2\lambda$, we have $\gamma > \sqrt{2} - 1$ so $d_P \log(1/\gamma) \leq 1.28 \times d_P$. Furthermore, we have in this case $\tau \geq 5$ which implies that $\lambda\tau \geq 5\lambda$. We deduce from it that $1.28 \times d_P \geq 5\lambda$ which contradicts $d_P < 2\lambda$. Hence, the hypotheses imply $d_P \geq 2\lambda$ which was used in Assumption 7.3.

7.4 Presentation of the TCHo Cryptosystem

In this section, we describe the TCHo cryptosystem and give algorithms for key generation, encryption and decryption. We also prove that TCHo is a cryptosystem.

7.4.1 Parameters

TCHo 's secret key consists in a low weight polynomial K over $\mathbb{F}_2[X]$ of degree d_K bounded by d and of weight w_K. The public key is a polynomial P such that P divides K and whose degree is in a given interval $[d_{\min}, d_{\max}]$. The security of the scheme relies on noise added by an LFSR with the public key as feedback polynomial and some strongly biased random noise. The bias of the noise γ along with the plaintext length k and the ciphertext length $\ell > d$ are the remaining parameters. Hence, for a fixed system with security parameter λ, we can define a parameter vector

$$(k, d_{\min}, d_{\max}, d, w_K, \gamma, \ell)\,.$$

We require that $k, d_{\min}, d_{\max}, d, w_K, \ell$ are positive integers, polynomially bounded, $d_{\max} > d_{\min}$, w_K odd, $3 \leq w_K \leq k$, $d \geq d_{\max}$, $k + d \leq \ell$, and that γ is subject to the following requirement which is needed for correctness.

$$\gamma^{2w_K} \frac{\ell - d}{k} = \Omega(\lambda^{\alpha}) \tag{7.7}$$

for some constant $\alpha > 0$. Later, we shall add the requirements of Assumptions 7.2 and 7.4 for security.

There are two approaches for selecting the parameters: in practice, we select some for which we have good implementation performances and a fair understanding of the security. This will be covered in Sect. 7.6. In theory, we select a family of parameters based on λ so that algorithms are polynomially bounded and whose security relies on complexity assumptions. This will be addressed in Theorems 7.5 and 7.6.

7.4.2 Key Generation

First, a random polynomial K of degree bounded by d and odd weight w_K with constant term 1 is generated. Nonzero coefficients in K shall be selected at positions which are pairwise different modulo k. If $K(X)$ is not coprime with $X^k - 1$ (which would be exceptional), we try again. Then, an irreducible factor P of degree $d_P \in [d_{\min}, d_{\max}]$ is searched. This procedure is repeated with another K until an appropriate P is found:

Generate:

```
repeat
    pick a random subset $I$ of $\{1, \dots, k-1\}$ of cardinality $w_K - 1$
    for each $i \in I$, pick a random $j_i$ such that $j_i \bmod k = i$ and $0 < j_i < d$
    take $K(X) = 1 + \sum_{i \in I} X^{j_i}$
    if $K(X)$ is coprime with $X^k - 1$ then
        factor $K$ as a product of irreducible polynomials over $\mathbb{F}_2$
        pick an irreducible factor $P$ of degree $d_P \in [d_{\min}, d_{\max}]$
    end if
until $P$ found
return $K$ and $P$
```

Note that since K is sparse, it can be stored efficiently using only $\lceil w_K \log(d) \rceil$ bits.

The number of irreducible polynomials of degree d is equivalent to $2^d/d$. So, a random polynomial has an irreducible factor of degree d with probability $1/d$. From that, we deduce that a random polynomial has an irreducible factor of degree in $[d_{\min}, d_{\max}]$ with probability $O((d_{\max} - d_{\min})/d_{\max})$. Hence, $O(d_{\max}/(d_{\max} - d_{\min}))$ factorization attempts are needed in average. Using the Cantor-Zassenhaus [13] factoring algorithm, every attempt costs $O\left(d^2 \log d \log\log d\right)$.

The total complexity of the key generation algorithm is, thus,

$$O\left(\frac{d_{\max}}{d_{\max} - d_{\min}} d^2 \log d \log\log d\right).$$

We make the heuristic assumption that the complexity is the same when the polynomial K is sparse instead of being fully random. This assumption will be discussed in Sect. 7.6. However, we have no formal proof so far that this algorithm is polynomially bounded. This is left open for future work.

7.4.3 Encryption

Let $C(m) : \{0,1\}^k \to \{0,1\}^\ell$ be the repetition code that, given an $m \in \{0,1\}^k$ returns the ℓ bit word

$$m\|m\|\ldots\|\tilde{m},$$

where $\tilde{m}$ is the bitstring m truncated such that $C(m)$ has length ℓ. Given a plaintext m of length k, the ciphertext y of length ℓ is

$$y := \mathsf{Enc}_{\mathrm{TCHo}}(P, m; r_1\|r_2) = C(m) + \mathrm{S}^{\ell}_{\mathcal{L}_P}(r_1) + \mathrm{S}^{\ell}_{\gamma}(r_2),$$

where r_1 and r_2 are random seeds. Care has to be taken about the size of these seeds. The first seed, r_1, consists in a random initial state for the LFSR. Hence, it has to be a random bitstring in $\{0,1\}^{d_P}$. The second seed, r_2, is a random seed for a biased pseudo random bit source. To ensure a proper security, this seed needs to be at least λ-bit long, where λ is the security parameter.

The encryption cost is $O(\ell \times d_P)$ if the random bit generator has not a higher complexity. In the case of a dedicated hardware implementation, the encryption can be done in $O(\ell)$ time with $O(d_P)$ gates.

```
**Encrypt**$(P, m; r_1, r_2)$:
1: compute $y = C(m) + \mathrm{S}^{\ell}_{\mathcal{L}_P}(r_1) + \mathrm{S}^{\ell}_{\gamma}(r_2)$
2: **return** $y$
```

7.4.4 Decryption

Let $y \in \{0,1\}^\ell$ be the ciphertext, i.e., $y = C(m) + \mathrm{S}^{\ell}_{\mathcal{L}_P}(r_1) + \mathrm{S}^{\ell}_{\gamma}(r_2)$. We decrypt y as follows.

First, we remove the contribution of the noise induced by the LFSR $\mathcal{L}_P$. This is done by computing $t := \mathsf{trunc}_{\ell-d}(K \otimes y)$. The resulting t is truncated to $\ell - d$ bits.[4] Since the multiplication operator is distributive and since K is a multiple of P, which is the feedback polynomial of $\mathcal{L}_P$, this operation completely removes the noise generated by the LFSR. However, this operation has also an effect on $C(m)$ and $\mathrm{S}_\gamma(r_2)$. We have $K \otimes S_\gamma^\ell \approx S_{\gamma^{w_K}}^{\ell-d}$ in the sense that every bit of $K \otimes S_\gamma^\ell$ has a bias of γ^{w_K} but they are not perfectly independent. Hence, if the weight of K is low, the noise remains strongly biased. We have also $K \otimes C(m) = C(m')$, where $m' = \psi_K(m)$, for some linear map ψ_K. Thus, $t \approx C(\psi_K(m)) + \mathrm{S}_{\gamma^{w_K}}^{\ell-d}$. Under some conditions, ψ_K is invertible.

Since the noise $\mathrm{S}_{\gamma^{w_K}}^{\ell-d}$ is strongly biased, we can recover $\psi_K(m)$ by performing majority logic decoding (MJD), i.e., by taking for each bit its majority value in the repetition code. For a proper choice of w_K, γ and ℓ, the probability of error will be negligible.

MJD decodes the correct $\psi_K(m)$ if for every $i = 0, \ldots, k-1$, there is a majority of j's such that $(K \otimes \mathrm{S}_\gamma^\ell)_{i+kj} = 0$.

Finally, we invert ψ_K to recover m. Recall that the operation $K \otimes C(m)$ can also be written as $M_K^{\ell-d} C(m)$, with

$$M_K^{\ell-d} := \begin{bmatrix} k_0 & k_1 & \ldots & k_d & 0 & 0 & \ldots & 0 \\ 0 & k_0 & k_1 & \ldots & k_d & 0 & \ldots & 0 \\ & & \ddots & & & \ddots & & \\ 0 & 0 & \ldots & 0 & k_0 & k_1 & \ldots & k_d \end{bmatrix}.$$

Since $C(m)$ is a repetition code, $\psi_K(m)$ can be written as $C_K m$, with

$$C_K = \begin{bmatrix} c_0 & c_1 & \ldots & c_{k-1} \\ c_{k-1} & c_0 & \ldots & c_{k-2} \\ \vdots & & \ddots & \vdots \\ c_1 & c_2 & \ldots & c_0 \end{bmatrix},$$

where

$$c_j = \sum_{\{i \in [0,d] :\, i \equiv j \bmod k\}} k_i.$$

The matrix C_K is invertible if and only if $c_0 + c_1 X + \cdots + c_{k-1} X^{k-1}$ is coprime with $X^k - 1$, which is equivalent to $K(X)$ being coprime with $X^k - 1$, which is a condition in our key generation algorithm. Hence, $m = C_K^{-1} m'$.

[4] Since K may have a degree less than d, $K \otimes y$ may have more than $\ell - d$ bits. To avoid side channels, we only use the first $\ell - d$ bits, as if K had degree d.

The decryption complexity is $O\left(w_K \times \ell + k^3\right)$ since the first operation takes $O\left(w_K \times \ell\right)$, the second $O\left(\ell - d\right)$ and the third $O\left(k^3\right)$ time.

```
Decrypt(K, y):
compute $t = \mathsf{trunc}_{\ell-d}(K \otimes y)$
for $i = 0$ to $k-1$ do
    set $\psi(m)_i = \text{majority}(t_{i+kj}\,;\, 0 \leq i+kj < \ell - d)$
end for
compute $m = C_K^{-1}\psi(m)$
return $m$
```

7.4.5 TCHo Is a Cryptosystem

In this section, we show that TCHo is a cryptosystem as defined in Definition 7.1.

Lemma 7.1. *The probability that a correctly generated ciphertext is badly decrypted satisfies*

$$\mathsf{Pr}[\mathsf{bad\ decoding}] \leq k \times \exp\left(-\frac{1}{2}\left\lfloor\frac{\ell-d}{k}\right\rfloor \gamma^{2w_K}\right).$$

Proof. We note from the requirement on K that nonzero coefficients have indices which are pairwise different modulo k. Hence, for a fixed i, all bits $(K \otimes \mathrm{S}_\gamma^\ell)_{i+kj}$ are independent. So,

$$\mathsf{Pr}[\mathsf{bad\ decoding}] \leq \rho,$$

where

$$\rho := \sum_{i=0}^{k-1} \sum_{w=0}^{\frac{1}{2}\left\lfloor\frac{\ell-d-i}{k}\right\rfloor} \binom{\left\lfloor\frac{\ell-d-i}{k}\right\rfloor}{w} \left(\frac{1+\gamma^{w_K}}{2}\right)^{w} \left(\frac{1-\gamma^{w_K}}{2}\right)^{\left\lfloor\frac{\ell-d-i}{k}\right\rfloor - w}. \tag{7.8}$$

We conclude thanks to the Chernoff bound (Theorem 7.1). □

Theorem 7.5. *There are some parameter selections making TCHo a cryptosystem with heuristic key generation that verifies the inequalities in Assumptions 7.2–7.4.*

Proof. Let λ be the security parameter. We select parameters satisfying

$$w_K = a\lambda \qquad k = w_K + \Theta(\lambda) \qquad d = \Theta\left(\lambda^2 \times k\right)$$

$$d_{\min} = \Theta\left(\lambda^2\right) \qquad d_{\max} = d_{\min} + \Theta\left(\lambda^2\right) \qquad \ell = d + \Theta\left(\lambda^2 \times k\right) \qquad \gamma = \lambda^{-c/\lambda},$$

such that these are positive integers, w_K is odd, $a, c > 0$ and with the following condition:

$$0 < ac < 1.$$

With these parameters, key generation takes $O\left(\lambda^4 \times k^2 \times \log\lambda \times \log\log\lambda\right)$ (heuristically), encryption takes $O\left(\lambda^4 \times k\right)$, decryption $O\left(\lambda^3 \times k\right)$ and (7.7) is verified. Furthermore, these parameters satisfy the inequalities in Assumptions 7.2 and 7.4, which will be needed to show the security of our scheme.

Since the parameters satisfy (7.7), there exists a constant $f \geq 0$ such that

$$\mathsf{Pr}[\mathsf{bad\ decoding}] \leq k \times \exp(-f\lambda^{\alpha})$$

when λ is large enough. So, this probability is negligible. Hence, the cryptosystem satisfies also the correctness property. □

7.5 Security of TCHo

In this section, we show results on the security of TCHo. In particular, we show that TCHo is IND-CPA-secure and not OW-CCA-secure. We show also how to achieve IND-CCA security.

7.5.1 TCHo Is IND-CPA-Secure

Theorem 7.6 (Aumasson et al. [3]). *Let* S_0 *be an unbiased source of random bits. There exists a constant* μ *such that, for every TCHo parameters, if* $\mathrm{S}^{\ell}_{\mathcal{L}P} + \mathrm{S}^{\ell}_{\gamma}$ *cannot be distinguished from* S^{ℓ}_0 *in time* t *with an advantage larger than* ε*, then TCHo is* $(t-\mu\ell, 2\varepsilon)$-IND-CPA-*secure.*

Proof. Instead of proving the semantic security directly, we prove it in the real-or-random game model. Recall that in this model, the adversary submits first a chosen plaintext x following an algorithm $\mathcal{A}^{\mathrm{ror}}_1(K_p;\rho)$. Then, given a ciphertext z, the adversary has to decide using an algorithm $\mathcal{A}^{\mathrm{ror}}_2(K_p,z;\rho)$ whether z is a ciphertext corresponding to x or to another random plaintext.

We show that using this adversary $\mathcal{A}^{\mathrm{ror}} = (\mathcal{A}^{\mathrm{ror}}_1, \mathcal{A}^{\mathrm{ror}}_2)$, we can build a distinguisher between $\mathrm{S}^{\ell}_{\mathcal{L}P} + \mathrm{S}^{\ell}_{\gamma}$ and S^{ℓ}_0. Let S^{ℓ}_* be the unknown instance we have to distinguish. First we recover a plaintext $x = \mathcal{A}^{\mathrm{ror}}_1(P)$. Let $z = C(x) + \mathrm{S}^{\ell}_*$. If S^{ℓ}_* is random, then z is also a totally random bitstring. Note that this z corresponds also to a valid ciphertext, since every bitstring in $\{0,1\}^{\ell}$ is valid. On the other hand, if S^{ℓ}_* is $\mathrm{S}^{\ell}_{\mathcal{L}P} + \mathrm{S}^{\ell}_{\gamma}$, then z is a valid ciphertext of x. Hence, using $\mathcal{A}^{\mathrm{ror}}_2(P,z)$, we can decide, whether z is a ciphertext corresponding to x or not. The cost of this simulation is $\mu\ell$, for $\mu > 0$ large enough. Thus, since we know by assumption that we cannot distinguish $\mathrm{S}^{\ell}_{\mathcal{L}P} + \mathrm{S}^{\ell}_{\gamma}$ from S^{ℓ}_0 in time t with an advantage larger than ε, $\mathcal{A}^{\mathrm{ror}}$ has complexity at least $t - \mu\ell$. Hence, TCHo is $(t-\mu\ell,\varepsilon)-$IND-CPA secure in the real-or-random game model. This implies that TCHo is $(t-\mu\ell,2\varepsilon)-$IND-CPA secure. □

Now, we just have to find suitable parameters such that the Noisy LFSR Distinguishing Problem is hard to obtain an IND-CPA-secure scheme.

7.5.2 TCHo Is Not OW-CCA Secure

We recall two negative results from [3] regarding the security of TCHo.

Lemma 7.2. *TCHo is malleable.*

Proof. Given a ciphertext y corresponding to a plaintext x, $y+C(x')$ is a valid ciphertext of $x+x'$ with correct distribution. □

This result implies also that TCHo is not IND-CCA secure as we can just use the malleability of ciphertexts and call the decryption oracle on the modified ciphertext to play the IND-CCA game.

Lemma 7.3. *TCHo is* not OW-CCA *secure.*

Proof. Given a ciphertext y to invert, modify the first bit an submit it to the decryption oracle. With high probability, the obtained plaintext will correspond to the original one. □

7.5.3 The Herrmann-Leander Attack

In PKC 2009, Herrmann and Leander presented a chosen ciphertext key recovery attack on TCHo [33]. They were able to recover the secret key in about $d^{3/2}$ queries to a decryption oracle. As shown in Lemma 7.3, TCHo is not OW-CCA secure. However, their attack is by far worse than the traditional OW-CCA message recovery attack since it reveals the private key. It is important to notice that their attack does not solve the LWPM problem but extracts this low weight polynomial by querying the decryption oracle in a clever way.

This proves that the key recovery problem is easy with a Dec_K oracle. This does not prove that decryption and key recovery are equivalent because we are using the Dec_K oracle with some inputs which do not follow the distribution of genuine ciphertexts. So, a decryption oracle able to decrypt ciphertexts may not fully emulate the Dec_K oracle for our purpose. We briefly present Herrmann and Leander's attack here.

The attack is an adaptive differential attack, i.e., pairs of ciphertexts with a well-chosen difference are submitted to the decryption oracle. Let the private key be $K=\sum_{j=0}^{d} k_j X^j$. Let also $N := \ell - d$. For simplicity, we assume that $\lfloor \frac{N}{k} \rfloor$ is odd so that there is always a non-ambiguous majority among $\lfloor \frac{N}{k} \rfloor$ bits. The idea is to recover every bit one after the other starting from k_1 to k_{d_K}. (Note that k_0 is fixed to 1.) To recover the key bit k_i knowing $k_0,\ldots,k_{i-1}$, two ciphertexts y and y' are submitted to the decryption oracle such that the difference between them before the MJD step during the decryption process is $t \oplus t' = \Delta := (1 \oplus k_i, 0, \ldots, 0, 1, 0, \ldots, 0)$, where the 1 is at index i. Let the two obtained plaintexts be respectively m and m'. If m and m' differ, this means that the MJD algorithm made different decisions for the two ciphertexts. With a clever choice of the ciphertexts y and y' and using our knowledge of the previous bits of K, we can ensure that $t = (b, 0, r)$, with $b = \mathsf{trunc}_i(1, 0^{(2k-1)}, 1, 0^{(2k-1)}, \ldots)$ and r an uniformly distributed random bitstring of length $\ell - i - d - 1$. The b part ensures that the first sum in the majority decoding algorithm is as much balanced as possible.

Let

$$M' := \begin{bmatrix} k_0 & k_1 & \ldots & k_{i-1} \\ 0 & k_0 & \ldots & k_{i-2} \\ \vdots & & \ddots & \vdots \\ 0 & \ldots & 0 & k_0 \end{bmatrix}.$$

To construct y, we take $y := (\hat{y}, 0^{(d+1)}, r)$, where $\hat{y}$ is the solution of $M'\hat{y} = b$ and r is a random bit string of size $N - i - 1$. For y', let δ be defined as

$$\delta_0 = \sum_{j=0}^{i-2} \delta'_j k_{j+1} \oplus 1,$$

$$\delta_j = \delta'_{j-1} \qquad \text{for } 1 \leq j \leq i,$$

$$\delta_j = 0 \qquad \text{for } i+1 \leq j < \ell,$$

where δ' is the solution of $M'\delta' = (0, \ldots, 0, 1)^t$. Then, $y' := y + \delta$. We refer the reader to [33] for a proof of correctness of this construction. These two ciphertexts produce the required t and t'.

To recover k_i, we distinguish two cases:

- When $i \equiv 0 \pmod{k}$, both the difference $1 \oplus k_i$ and 1 in Δ contribute to the same bit during MJD. Since $t_0 = 1$ and $t_i = 0$, we have $t'_0 = k_i$ and $t'_i = 1$. Hence, $m \neq m'$ implies that $k_i = 1$. Thus, the resulting plaintext will be different only when $\sum_{j=0}^{\lfloor N/k \rfloor} t_{kj} = \left\lfloor \frac{1}{2} \left\lfloor \frac{N}{k} \right\rfloor \right\rfloor$ and $k_i = 1$. However, in the case $k_i = 0$, the majority cannot change. By repeating this attack with a sufficient number of ciphertext pairs we recover k_i with negligible probability of error by making statistics.
- When $i \not\equiv 0 \pmod{k}$, the difference $1 \oplus k_i$ and the difference 1 does not contribute to the same bit during MJD. If $k_i = 0$, t and t' differ in their coordinates at index 0 and i and the majority at index 0 changes if $\sum_{j=0}^{\lfloor N/k \rfloor} t_{kj} = \left\lfloor \frac{1}{2} \left\lfloor \frac{N}{k} \right\rfloor \right\rfloor$ and $k_i = 1$. When $k_i = 1$, this majority cannot change. The one at index i may change depending on $\sum_{j=0}^{\lfloor (N-i)/k \rfloor} t_{kj+i}$. However, the difference $m \oplus m'$ will not be the same as when the majority changes at index 0, so it can be filtered out. Like in the previous case, we recover k_i by submitting sufficiently many ciphertext pairs.

We refer the reader to [33] for further details.

This attack implies that TCHo cannot be used in its original form. We show in the next section how TCHo can be transformed into an IND-CCA secure scheme.

7.5.4 *Achieving IND-CCA Security*

Following the Fujisaki-Okamoto construction [27], we can obtain an IND-CCA secure scheme using TCHo. For this, we need first to define Γ-uniformity.

Definition 7.7 (Γ-uniformity). Let Asym be an asymmetric encryption scheme, with key generation algorithm $\mathsf{Gen}(1^\lambda)$ and encryption algorithm $\mathsf{Enc}_{\mathsf{Asym}}(K_p, m; r)$ over the message space $\mathcal{M}$ and the random coins space $\mathcal{R}$. Asym is Γ-*uniform* if for any plaintext $m \in \mathcal{M}$, for any keys drawn by Gen, and for any $y \in \{0,1\}^*$, we have

$$\Pr\left[h \overset{U}{\leftarrow} \mathcal{R} : y = \mathsf{Enc}_{\mathsf{Asym}}(K_p, m; h)\right] \leq \Gamma,$$

i.e., the probability that a plaintext and a ciphertext match is bounded.

Now, we recall the Fujisaki-Okamoto paradigm: Given an OW-CPA and Γ-uniform asymmetric cipher Asym with public (resp. private) key K_p (resp. K_s), a one-time secure symmetric cipher Sym, and two random oracles G and H, the following construction is IND-CCA secure in the random oracle model:

Encryption: Given a message m:

```
1: Let $\sigma \overset{U}{\leftarrow} \{0,1\}^k$.
2: Let $\psi \leftarrow G(\sigma)$ be the symmetric key.
3: Encrypt the message using the symmetric cipher: $y = \mathsf{Enc}_{\mathsf{Sym}}(\psi, m)$.
4: Encapsulate the key with the asymmetric cipher: $\chi \leftarrow \mathsf{Enc}_{\mathsf{Asym}}(K_p, \sigma; H(\sigma \| m))$.
5: return $(\chi, y)$.
```

Decryption: Given a ciphertext (χ, y):

```
1: Decrypt the asymmetric part: $\hat{\sigma} = \mathsf{Dec}_{\mathsf{Asym}}(K_s, \chi)$.
2: Recover the symmetric key: $\hat{\psi} = G(\hat{\sigma})$.
3: Recover the message: $\hat{m} = \mathsf{Dec}_{\mathsf{Sym}}(\hat{\psi}, y)$.
4: if $\chi = \mathsf{Enc}_{\mathsf{Asym}}(K_p, \hat{\sigma}; H(\hat{\sigma} \| \hat{m}))$ then
5:     return $\hat{m}$.
6: else
7:     return $\perp$.
8: end if
```

Note that the check done at Step 4 during the decryption is necessary to obtain an IND-CCA secure scheme.

To use this construction with TCHo, we need to show that TCHo is Γ-uniform.

Lemma 7.4. *TCHo is* $((1+\gamma)/2)^\ell$*-uniform.*

Proof. Recall that the TCHo encryption of x is $y = C(x) + \mathrm{S}^{\ell}_{\mathcal{L}P}(r_1) + \mathrm{S}^{\ell}_{\gamma}(r_2)$, for random coins r_1 and r_2. We need to bound the probability (taken over r_1 and r_2) that a given plaintext x and ciphertext y match. Since in TCHo we consider only positive bias, the most probable ciphertext corresponds to $y = C(x) + \mathrm{S}^{\ell}_{\mathcal{L}P}$, i.e., when $\mathrm{S}^{\ell}_{\gamma}$ is the zero bitstring. This happens with probability $((1+\gamma)/2)^{\ell}$. When we take the average on the possible r_1, this probability can only decrease. Hence, TCHo is $((1+\gamma)/2)^{\ell}$-uniform. □

Since TCHo is $((1+\gamma)/2)^{\ell}$-uniform and since IND-CPA security implies OW-CPA security, we can use the Fujisaki-Okamoto paradigm to obtain a IND-CCA secure scheme. An example of a simple one-time secure symmetric cipher one could use is $\mathsf{Enc}_{\mathsf{Sym}}(\psi, m) = m + F(\psi)$ for a random oracle F. We obtain the following cryptosystem:

Encryption: Given a message m:

```
Let $\sigma \overset{U}{\leftarrow} \{0,1\}^k$.
Let $\psi \leftarrow G(\sigma)$ be the symmetric key.
Encrypt the message using the symmetric cipher: $y = m + F(\psi)$.
Encapsulate the key with the asymmetric cipher: $\chi \leftarrow \mathsf{Enc}_{\mathrm{TCHo}}(P, \sigma; H(\sigma \| m))$.
return $(\chi, y)$.
```

Decryption: Given a ciphertext (χ, y):

```
Decrypt the asymmetric part: $\hat{\sigma} = \mathsf{Dec}_{\mathrm{TCHo}}(K, \chi)$.
Recover the symmetric key: $\hat{\psi} = G(\hat{\sigma})$.
Recover the message: $\hat{m} = y + F(\hat{\psi})$.
if $\chi = \mathsf{Enc}_{\mathrm{TCHo}}(P, \hat{\sigma}; H(\hat{\sigma} \| \hat{m}))$ then
    return $\hat{m}$.
else
    return $\perp$.
end if
```

7.6 Implementation Results

In this section, we discuss various topics regarding the implementation of TCHo. First we describe a way to find good parameters and give some examples. Next we discuss our heuristic assumption used in the key generation algorithm. Finally, we comment on TCHo software and hardware implementation.

7.6.1 Parameter Selection

We show now how to find good parameters vectors that can be used for TCHo in practice. Recall from Sect. 7.4.4 that the probability that a message is incorrectly

Table 7.1 Example of parameter vectors for TCHo

ID	k	λ	$d_{\min}$	$d_{\max}$	d	w_K	γ	ℓ	$\rho_{\max}$
I	128	80	15,000	16,000	37,069	69	0.98862	55,000	2^{-20}
II	128	128	23,740	24,740	67,805	91	0.98853	100,233	2^{-20}
III	256	256	63,500	64,500	221,169	147	0.99141	326,100	2^{-20}
IV	384	384	145,000	146,000	455,356	237	0.99433	644,900	2^{-20}
V	512	512	155,000	156,000	845,405	213	0.99243	1,291,800	2^{-20}

decoded is bounded by ρ, defined by (7.8). We call this value the *unreliability* of the system. This value has an huge impact on the ciphertext length and the maximum unreliability accepted $\rho_{\max}$ has to be selected carefully depending on the application we consider. Recall that the parameters have to verify

$$(w_K - 1)\log\frac{d}{d_{\max}} \geq \lambda \qquad \text{and} \qquad \binom{d}{w_K - 1} < 2^{d_{\min}}$$

from Assumption 7.2 and

$$d_{\min}\log\frac{1}{\gamma} \geq \lambda\tau \qquad \text{and} \qquad \gamma \leq 2^{1-\lambda/d_{\min}} - 1,$$

where τ is the root of $\gamma e^{-1} = \tau 2^{-\tau}$, from Assumption 7.4. We need also to verify

$$\rho \leq \rho_{\max}.$$

Inequalities in Assumption 7.3 are consequences of the ones in Assumption 7.4 as already observed.

To find the best possible parameters, we used the following approach. We fix first the plaintext length k, the security parameter λ, and ρ_{max}, the maximum unreliability accepted, since all these three values will clearly depend on the application we consider and will drastically modify the ciphertext length. Then, we search for the best $d_{\min}$ that minimizes the ciphertext length. Indeed, d_{min} has a huge impact on the ciphertext length: a too small d_{min} implies that semantic security is harder to achieve, which leads to a smaller γ and finally to a larger ℓ too keep a tolerable unreliability. Similarly, a too large d_{min} implies a larger d or w_K, which leads also to a larger ciphertext length. Table 7.1 shows possible parameter vectors for $k = 128, 256, 384$ and 512 bits. We also set $\lambda = k$ for 4 of the 5 vectors, since TCHo will mostly be used to encrypt symmetric keys and, hence, should provide at least as much security as the key length.

7.6.2 Heuristic Assumption for the Key Generation Algorithm

In Sect. 7.4.2, we made the heuristic assumption that the probability for a random sparse polynomial to have an irreducible polynomial factor of degree in $[d_{\min}, d_{\max}]$

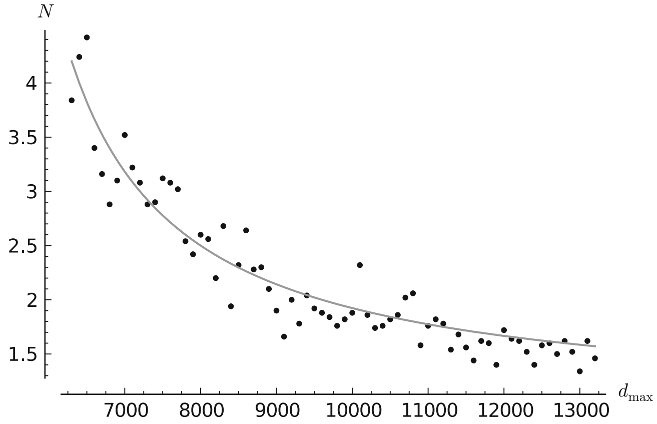


Fig. 7.1 Comparison between $d_{\max}/(d_{\max}-d_{\min})$ (*continuous line*) and the average number of iterations N for the key generation algorithm with parameters $k=128$, $w_K=69$, $d=32,069$ and $d_{\min}=4,800$ and using 40 samples (*points*)

is the same than when the polynomial is fully random, i.e., this probability is $O((d_{\max}-d_{\min})/d_{\max})$. To verify this heuristic assumption, we made some simulations in which we compared the average number of iterations N needed to generate the public key P with $d_{\max}/(d_{\max}-d_{\min})$. The result is depicted in Fig. 7.1. We can see that both distributions are quite similar and, hence, that our heuristic assumption seems reasonable.

7.6.3 Software Implementation

TCHo was implemented in software [10] as an extension of Network Security Services (NSS) [48] so that it can be used as a TLS/SSL cipher suite in browsers making use of NSS. One implementation issue concerning the LFSR was raised there. LFSRs are a very efficient component in hardware since a bit can be produced every clock cycle. However, in software, the complexity necessary to compute a bit is proportional to the weight of the feedback polynomial used to describe the LFSR. Hence, for TCHo, this complexity is proportional to w_P, the weight of the public key. Note that $w_P = d_P/2$ in average.

The trivial software implementation can be improved by considering blocks of outputs instead of a single bit. Typically, the size of the block b will be a small multiple of the machine word size. Then, using methods described in [16, 17], it is possible to obtain a cost of $O(w_P/b)$ operations per bit. However, the cost needed to compute the output of the LFSR is still large and dominates the cost of encryption.

This issue shows that TCHo is meant to be a hardware-oriented cipher and is, hence, less efficient in software.

Table 7.2 Performances of TCHo

ID	Key generation (s)	Encryption (ms)	Decryption (ms)	Secret key (bit)	Public key (bit)
I	882	54	11	1,048	16,000
II	7,033	158	42	1,461	24,740
III	46,994	253	120	2,610	64,500

Table 7.2 shows performances for the first three parameter vectors of TCHo presented in Table 7.1. These results were performed on a 2.66 GHz Core 2 Duo with a C++ implementation using the NTL library [63].

7.6.4 *Hardware Implementation*

We discuss in this section how to implement the encryption and the decryption in hardware.

For encryption, the implementation of the repetition code is straightforward. The LFSR can be efficiently implemented using integrated circuits. However, the length of the LFSRs we are dealing with is unusually big and we assume that this length does not alter the performances too much. Encryption requires also a biased source of noise. As mentioned in [3], this can be implemented using a precomputed binary tree where each leaf corresponds to a biased sequence of bits. Then, using a uniform random bitstream fed with physical entropy, one can decide which branch to take in the binary tree and output the biased bits.

Decryption is harder to implement. However, the product $K \otimes y$ consists of a sequence of dot products and can, hence, be implemented using some library able to do linear algebra computations for FPGA devices—for instance [65]. Similarly, such a library can be used to compute $C_K^{-1}\psi(m)$. Majority logic decoding is easy to implement but requires some additional memory—$k \times \log((\ell - d)/k))$—to store the number of occurrences of each bit.

Reference [3] estimates that a 128 bit key with $\lambda = 80$ can be encrypted with a circuit of about $10,000$ gates. Hence, TCHo is well suited for RFID.

7.7 Conclusion

TCHo is still a young cryptosystem and has a large margin of progression. We indicate directions for further work.

Complexity. We are still missing a rigorous complexity analysis about the key generation algorithm, although we have a heuristic complexity matching well experimental results.

A Shorter Ciphertext Length. The use of a repetition code during the encryption process leads to a very simple decryption algorithm. However, the length of the ciphertexts is quite long. A possible solution would be to replace the repetition code with a code with a smaller overhead size.

A More Efficient IND-CCA Scheme. One drawback of the Fujisaki-Okamoto construction is that a TCHo encryption has to be performed during the decryption process. If we neglect the cost of the hash functions, this increases the cost of decryption to $O\left((w_K + d_P) \times \ell + k^3 + \rho\right)$, where ρ is the complexity of decryption of the symmetric scheme. This complexity can be reduced by using other hybrid constructions like, for instance, REACT [51] or GEM [18]. For this, we need the asymmetric scheme to be OW-PCA-secure, i.e., one-way against an adversary with an plaintext-checking oracle. Hence, we wonder whether TCHo is OW-PCA-secure.

Generalization. Another further work is to generalize the TCHo construction by replacing the LFSR with a random linear code. We wonder also if we can link TCHo to lattice-based cryptography.

In conclusion, TCHo is asymptotically an efficient encryption scheme based on a new hard problem, the low weight polynomial multiple problem. It is IND-CPA secure and can be used to obtain an IND-CCA scheme. However, it still suffers from two drawbacks: the key generation algorithm is expensive and the expansion factor is huge. In this paper, we reviewed the existing previous work on TCHo. We also provided new non-heuristic proofs of correctness and new parameters for different plaintext sizes.

References

1. Ajtai, M.: Generating Hard Instances of Lattice Problems (Extended Abstract). In: STOC, pp. 99–108 (1996)
2. Ajtai, M., Dwork, C.: A Public-Key Cryptosystem with Worst-Case/Average-Case Equivalence. In: STOC, pp. 284–293 (1997)
3. Aumasson, J.P., Finiasz, M., Meier, W., Vaudenay, S.: TCHo: A Hardware-Oriented Trapdoor Cipher. In: J. Pieprzyk, H. Ghodosi, E. Dawson (eds.) ACISP, *Lecture Notes in Computer Science*, vol. 4586, pp. 184–199. Springer (2007)
4. Baignères, T., Junod, P., Vaudenay, S.: How Far Can We Go Beyond Linear Cryptanalysis? In: Lee [37], pp. 432–450
5. Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A Concrete Security Treatment of Symmetric Encryption: Analysis of the DES Modes of Operation (Full Version) (1997). Available at http://cseweb.ucsd.edu/users/mihir
6. Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A Concrete Security Treatment of Symmetric Encryption (Extended Abstract). In: FOCS, pp. 394–403 (1997)
7. Berlekamp, E.: Factoring polynomials over large finite fields. Mathematics of Computation **24**(111), 713–735 (1970)
8. Bernstein, D.J.: Introduction to post-quantum cryptography. In: D.J. Bernstein, J. Buchmann, E. Dahmen (eds.) Post-Quantum Cryptography, pp. 1–14. Springer (2009)
9. Bernstein, D.J., Lange, T., Peters, C.: Attacking and Defending the McEliece Cryptosystem. In: J. Buchmann, J. Ding (eds.) PQCrypto, *Lecture Notes in Computer Science*, vol. 5299, pp. 31–46. Springer (2008)

10. Bindschaedler, V.: TCHo Software Implementation: Extending Firefox's Security Services Library. EPFL Bachelor Thesis (unpublished) (2010)
11. Canteaut, A., Chabaud, F.: A New Algorithm for Finding Minimum-Weight Words in a Linear Code: Application to McEliece's Cryptosystem and to Narrow-Sense BCH Codes of Length 511. IEEE Transactions on Information Theory **44**(1), 367–378 (1998)
12. Canteaut, A., Trabbia, M.: Improved Fast Correlation Attacks Using Parity-Check Equations of Weight 4 and 5. In: B. Preneel (ed.) EUROCRYPT, *Lecture Notes in Computer Science*, vol. 1807, pp. 573–588. Springer (2000)
13. Cantor, D., Zassenhaus, H.: A new algorithm for factoring polynomials over finite fields. Mathematics of Computation **36**(154), 587–592 (1981)
14. Chernoff, H.: A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. The Annals of Mathematical Statistics **23**(4), 493–507 (1952)
15. Chose, P., Joux, A., Mitton, M.: Fast Correlation Attacks: An Algorithmic Point of View. In: L.R. Knudsen (ed.) EUROCRYPT, *Lecture Notes in Computer Science*, vol. 2332, pp. 209–221. Springer (2002)
16. Chowdhury, S., Maitra, S.: Efficient Software Implementation of Linear Feedback Shift Registers. In: C.P. Rangan, C. Ding (eds.) INDOCRYPT, *Lecture Notes in Computer Science*, vol. 2247, pp. 297–307. Springer (2001)
17. Chowdhury, S., Maitra, S.: Efficient Software Implementation of LFSR and Boolean Function and Its Application in Nonlinear Combiner Model. In: J. Zhou, M. Yung, Y. Han (eds.) ACNS, *Lecture Notes in Computer Science*, vol. 2846, pp. 387–402. Springer (2003)
18. Coron, J.S., Handschuh, H., Joye, M., Paillier, P., Pointcheval, D., Tymen, C.: GEM: A Generic Chosen-Ciphertext Secure Encryption Method. In: B. Preneel (ed.) CT-RSA, *Lecture Notes in Computer Science*, vol. 2271, pp. 263–276. Springer (2002)
19. Courtois, N., Finiasz, M., Sendrier, N.: How to Achieve a McEliece-Based Digital Signature Scheme. In: C. Boyd (ed.) ASIACRYPT, *Lecture Notes in Computer Science*, vol. 2248, pp. 157–174. Springer (2001)
20. Didier, F., Laigle-Chapuy, Y.: Finding low-weight polynomial multiples using discrete logarithm. In: IEEE International Symposium on Information Theory, 2007 (ISIT 2007), pp. 1036–1040 (2007)
21. Diffie, W., Hellman, M.: New directions in cryptography. Information Theory, IEEE Transactions on **22**(6), 644–654 (1976)
22. Ding, J., Schmidt, D.: Rainbow, a New Multivariable Polynomial Signature Scheme. In: J. Ioannidis, A.D. Keromytis, M. Yung (eds.) ACNS, *Lecture Notes in Computer Science*, vol. 3531, pp. 164–175 (2005)
23. Duc, A.: TCHo: a Postquantum Public-Key Cryptography Toolkit. Unpublished Report (2010)
24. El Aimani, L., von zur Gathen, J.: Finding low weight polynomial multiples using lattices. Cryptology ePrint Archive, Report 2007/423 (2007). http://eprint.iacr.org
25. El Gamal, T.: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In: CRYPTO, pp. 10–18 (1984)
26. Finiasz, M., Vaudenay, S.: When Stream Cipher Analysis Meets Public-Key Cryptography. In: E. Biham, A.M. Youssef (eds.) Selected Areas in Cryptography, *Lecture Notes in Computer Science*, vol. 4356, pp. 266–284. Springer (2006)
27. Fujisaki, E., Okamoto, T.: Secure Integration of Asymmetric and Symmetric Encryption Schemes. In: M.J. Wiener (ed.) CRYPTO, *Lecture Notes in Computer Science*, vol. 1666, pp. 537–554. Springer (1999)
28. Giesbrecht, M., Roche, D.S., Tilak, H.: Computing Sparse Multiples of Polynomials. In: O. Cheong, K.Y. Chwa, K. Park (eds.) ISAAC (1), *Lecture Notes in Computer Science*, vol. 6506, pp. 266–278. Springer (2010)
29. Gilbert, H., Robshaw, M.J.B., Seurin, Y.: How to Encrypt with the LPN Problem. In: L. Aceto, I. Damgård, L.A. Goldberg, M.M. Halldórsson, A. Ingólfsdóttir, I. Walukiewicz (eds.) ICALP (2), *Lecture Notes in Computer Science*, vol. 5126, pp. 679–690. Springer (2008)
30. Goldreich, O., Goldwasser, S., Halevi, S.: Eliminating Decryption Errors in the Ajtai-Dwork Cryptosystem. In: Kaliski Jr. [35], pp. 105–111

31. Goldreich, O., Goldwasser, S., Halevi, S.: Public-Key Cryptosystems from Lattice Reduction Problems. In: Kaliski Jr. [35], pp. 112–131
32. Hallgren, S., Vollmer, U.: Quantum computing. In: D.J. Bernstein, J. Buchmann, E. Dahmen (eds.) Post-Quantum Cryptography, pp. 15–34. Springer (2009)
33. Herrmann, M., Leander, G.: A Practical Key Recovery Attack on Basic TCHo. In: S. Jarecki, G. Tsudik (eds.) Public Key Cryptography, *Lecture Notes in Computer Science*, vol. 5443, pp. 411–424. Springer (2009)
34. Hoffstein, J., Pipher, J., Silverman, J.H.: NTRU: A Ring-Based Public Key Cryptosystem. In: J. Buhler (ed.) ANTS, *Lecture Notes in Computer Science*, vol. 1423, pp. 267–288. Springer (1998)
35. Kaliski Jr., B.S. (ed.): Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17–21, 1997, Proceedings, *Lecture Notes in Computer Science*, vol. 1294. Springer (1997)
36. Kipnis, A., Patarin, J., Goubin, L.: Unbalanced Oil and Vinegar Signature Schemes. In: J. Stern (ed.) EUROCRYPT, *Lecture Notes in Computer Science*, vol. 1592, pp. 206–222. Springer (1999)
37. Lee, P.J. (ed.): Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5–9, 2004, Proceedings, *Lecture Notes in Computer Science*, vol. 3329. Springer (2004)
38. Lenstra, A., Lenstra, H., Lovász, L.: Factoring polynomials with rational coefficients. Mathematische Annalen **261**(4), 515–534 (1982)
39. Li, Y.X., Deng, R.H., Wang, X.M.: On the equivalence of McEliece's and Niederreiter's public-key cryptosystems. IEEE Transactions on Information Theory **40**(1), 271 (1994)
40. Lu, Y., Meier, W., Vaudenay, S.: The Conditional Correlation Attack: A Practical Attack on Bluetooth Encryption. In: V. Shoup (ed.) CRYPTO, *Lecture Notes in Computer Science*, vol. 3621, pp. 97–117. Springer (2005)
41. Lu, Y., Vaudenay, S.: Cryptanalysis of Bluetooth Keystream Generator Two-Level E0. In: Lee [37], pp. 483–499
42. Lu, Y., Vaudenay, S.: Faster Correlation Attack on Bluetooth Keystream Generator E0. In: M.K. Franklin (ed.) CRYPTO, *Lecture Notes in Computer Science*, vol. 3152, pp. 407–425. Springer (2004)
43. Lu, Y., Vaudenay, S.: Cryptanalysis of an E0-like Combiner with Memory. Journal of Cryptology **21**(3), 430–457 (2008)
44. Lyubashevsky, V., Peikert, C., Regev, O.: On Ideal Lattices and Learning with Errors over Rings. In: H. Gilbert (ed.) EUROCRYPT, *Lecture Notes in Computer Science*, vol. 6110, pp. 1–23. Springer (2010)
45. Matsumoto, T., Imai, H.: Public Quadratic Polynominal-Tuples for Efficient Signature-Verification and Message-Encryption. In: C.G. Günther (ed.) EUROCRYPT, *Lecture Notes in Computer Science*, vol. 330, pp. 419–453. Springer (1988)
46. McEliece, R.: A public-key cryptosystem based on algebraic coding theory. DSN progress report **42**(44), 114–116 (1978)
47. Meier, W., Staffelbach, O.: Fast correlation attacks on certain stream ciphers. Journal of Cryptology **1**(3), 159–176 (1989)
48. Mozilla Corporation: Network Security Services (NSS) (2009). http://www.mozilla.org/projects/security/pki/nss/
49. Naccache, D. (ed.): Topics in Cryptology - CT-RSA 2001, The Cryptographer's Track at RSA Conference 2001, San Francisco, CA, USA, April 8–12, 2001, Proceedings, *Lecture Notes in Computer Science*, vol. 2020. Springer (2001)
50. Niederreiter, H.: Knapsack-type cryptosystems and algebraic coding theory. Problems of Control and Information Theory **15**(2), 159–166 (1986)
51. Okamoto, T., Pointcheval, D.: REACT: Rapid Enhanced-Security Asymmetric Cryptosystem Transform. In: Naccache [49], pp. 159–175

52. Patarin, J.: Asymmetric Cryptography with a Hidden Monomial. In: N. Koblitz (ed.) CRYPTO, *Lecture Notes in Computer Science*, vol. 1109, pp. 45–60. Springer (1996)
53. Patarin, J., Courtois, N., Goubin, L.: FLASH, a Fast Multivariate Signature Algorithm. In: Naccache [49], pp. 298–307
54. Patarin, J., Courtois, N., Goubin, L.: QUARTZ, 128-Bit Long Digital Signatures. In: Naccache [49], pp. 282–297
55. Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: M. Mitzenmacher (ed.) STOC, pp. 333–342. ACM (2009)
56. Rabin, M.: Digitalized signatures and public-key functions as intractable as factorization (1979)
57. Regev, O.: New lattice-based cryptographic constructions. J. ACM **51**(6), 899–942 (2004)
58. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: H.N. Gabow, R. Fagin (eds.) STOC, pp. 84–93. ACM (2005)
59. Regev, O.: Lattice-Based Cryptography. In: C. Dwork (ed.) CRYPTO, *Lecture Notes in Computer Science*, vol. 4117, pp. 131–141. Springer (2006)
60. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM **21**(2), 120–126 (1978)
61. Schnorr, C.P.: A Hierarchy of Polynomial Time Lattice Basis Reduction Algorithms. Theoretical Computer Science **53**, 201–224 (1987)
62. Shor, P.W.: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM J. Comput. **26**(5), 1484–1509 (1997)
63. Shoup, V.: NTL: A Library for doing Number Theory. http://www.shoup.net/ntl/
64. Wagner, D.: A Generalized Birthday Problem. In: M. Yung (ed.) CRYPTO, *Lecture Notes in Computer Science*, vol. 2442, pp. 288–303. Springer (2002)
65. Zhuo, L., Prasanna, V.K.: High Performance Linear Algebra Operations on Reconfigurable Systems. In: SC, p. 2. IEEE Computer Society (2005)

Chapter 8
Formal Model for (k)-Neighborhood Discovery Protocols

Raphaël Jamet and Pascal Lafourcade

Abstract Neighborhood discovery is a critical part of wireless sensor networks, yet little work has been done on formal verification of the protocols in presence of both intruder nodes and mobility. We present a formal trace-based model to verify protocols doing neighborhood discovery, and we provide a formal definition of (1)-neighborhood and (k)-neighborhood. We also analyze a protocol from the literature, and show some conditions needed for its correctness. Finally, we present the groundwork for a protocol which discovers (k)-neighborhood based on (1)-neighborhood data under some assumptions, and prove that it remains secure even if an intruder interferes.

8.1 Introduction

The number of wireless networks is ever increasing. Cellular phones are now more common than wired ones, the number of mobile connections to Wireless Local Area Networks (WLANs) is increasing, and wireless devices are now used everyday at homes, companies and administrations. Wireless devices have become so small and cheap that they can be used in sensor networking applications such as environmental or building monitoring. These devices communicate by relaying packets of other devices across multiple wireless links (hops). Since the devices are often mobile the topology of the network can change over time. Even if the nodes are static, a node can disappear from the network due to battery shortage, temporary interference or physical damage, which will also alter the network configuration. As a result one of the main issues for these networks is that each node must discover or rediscover which nodes are within its communication range: a process called *neighborhood discovery*.

R. Jamet (✉) • P. Lafourcade
Université Grenoble 1, CNRS, Verimag, Grenoble, France
e-mail: raphael.jamet@imag.fr; pascal.lafourcade@imag.fr

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_8,

Neighborhood discovery protocols are basic components in mobile wireless systems. Knowledge of the neighbors of a node is for instance essential for routing protocols. The goal of a neighborhood discovery protocol is to identify as neighbors only those devices that are really neighbors, even in presence of intruders. Designing a secure neighborhood discovery protocol is not an easy task, as illustrated in [1] with the famous *"MIG-in-the-middle"* attack mounted in the late 1980s. In this attack, Angolan MIG airplanes were able to be impersonate a South African unit by relaying challenge messages from South African defense to another South Africa units, which then answered the challenge for them. The Angolan MIGs could therefore bomb their targets without being attacked by the South African air defense. A secure neighborhood discovery system would have prevented such a problem, by detecting that the MIG is not actually in range for the authentication protocol, and then preventing further impersonation by enemy forces. In [20], the authors survey the question of secure neighborhood discovery, providing definitions of neighborhood types and neighbor discovery protocol properties. They also describe different attacks against wireless networks.

We consider that two nodes are neighbors if they can communicate directly together, without the help of another node. In this, they are closely related to distance bounding protocols, which are designed to determine a upper bound on the distance separating two nodes. Our aim is to define formally this neighborhood property. The ability to verify that two nodes are neighbors can be used to prevent some attacks like wormhole attacks [14]. In this context, we consider that an attack is any situation in which two nodes communicate together believing that they are neighbors when in fact, other possibly malicious nodes are relaying the communication, whether nodes are MIGs, smart-cards or wireless sensors.

One aim of modern security is to provide rigorous guaranties that the designed system satisfies the required security properties. It is about giving rigorous models for systems, formal definitions of security properties and rigorous proofs under precisely identified assumptions. Proving that a protocol is secure is not an easy task, even for simple and short protocols. In 1996 Lowe [15] found an attack on the famous Needham-Schroeder protocol [18] 17 years after his publication. He found this flaw using his automatic verification tool Casper [16] based on the CSP (Communicating Sequential Processes) model checker FDR [13, 26, 28]. The protocol proposed by Needham and Schroeder was proven secure by using a formalism called BAN logic under different assumptions on the intruder model: only for one single execution of the protocol. The flaw proposed uses two parallel executions of the same protocol with different participants and assumes the so-called *perfect encryption hypothesis*. More precisely it means that the only way to uncrypt a cipher text is to know the inverse key. This hypothesis abstracts the cryptography in order to detect "logical flaw" due to all possible interleavings of different execution of the protocol. In this formalism:

- Messages are represented by terms build over a signature
- Intruder controls the network
- Perfect encryption assumption is done
- Intruder has a limited abilities.

The intruder capabilities are usually represented by the so-called *Dolev-Yao* intruder model [10]. This intruder model captures the perfect encryption hypothesis. This approach is called *symbolic* by contrast to the *computational* approach proposed by the cryptographer. The discover of the "logical" attack by G. Lowe shows that even experts can miss some flaws even on small protocol (only three message exchanges) and certainly underestimate the complexity of the security analysis of such protocols. It also indicates that automatic analysis is critical for assessing the security of cryptographic protocols, because humans cannot reasonably verify their correctness. Hence symbolic and automatic verification of cryptographic protocols became a main and active topic in security. In [3] one can find a survey of formal approaches for proving security protocols.

8.1.1 Motivational Scenario

Let us consider a wireless network, intended to assist firefighters during an operation inside disaster sites. This kind of WSNs are being investigated since a few years. For instance in [32], Wilson et al. present the SmokeNet WSN, used to provide firefighters critical data about the building currently on fire, and also monitor personnel health and position. SmokeNet itself is a pre-existing network consisting of small sensors (motes) scattered through the building. All the motes regularly emit localization data. Additionally, some motes also monitor smoke, temperature, carbon monoxide levels, which combined give the possibility to follow the progress of an hypothetical fire. The applications are designed to resist node failures, and thus empty batteries, physical damage, and other causes of disruption are not critical. Firefighters usually are organized in a command post, outside of the building, and agents inside the building. The command post has access to the SmokeNet data, and can thus monitor the progress of the fire. Agents inside do not have the time to sort through all of this data, and instead must be relayed orders from the command post. This relaying happens through a network which is built with the cooperation of both the pre-existing motes and the equipment worn by each firefighter. They also consider the possibility of relay motes dropped by firefighters as they progress into the building.

Taking all of this into account, we propose a few changes in the SmokeNet system and firefighters equipment which would enable new features. In certain kinds of incidents (chemical, for instance), firefighters are required to stay within a certain range of each other (rescue distance). We can use SmokeNet for this, but the current localization systems in place in the network requires pre-existing calibration of the motes, maps, and mote placement information. This may not be available in all buildings, and to work around this, our proposition does not need this data.

Let us assume that the command center wants firefighters to stay at 40 m of each other, but sensors can only communicate with another sensor 20 m away at most. How could we guarantee that if the distance between two firefighters gets higher than 40 m, then the sensors detect it and alert their wearers? By ensuring that two worn nodes can always communicate directly, or through one other node. Alerts when the distance is less than 40 m are not much of an issue, but if the distance is dangerous, the sensors must react. This is where neighborhood detection and distance bounding gets useful. If we can use a proven protocol which detects the distance between nodes, we can keep track of the firefighter rescue radius. If we know the communication range, using the mere possibility of communication is similar. All of this does not require any localization data. And since we do not want to be restricted by the range of the sensors, it should be possible to use more than two nodes in the distance detection.

To worsen things, we would like to be able to guarantee that within some known limits, these protocols still provably work in presence of serious malfunctions. In order to model this, we choose a worst-case assumption, malicious sensors which actively try to disrupt the protocol. This is a more demanding requirement than simple faults in sensors.

8.1.2 Contributions

One of our main goals is to formally define the concept of neighborhood. In order to achieve it and based on the observation that the neighborhood is a physical property,[1] we distinguish two layers: the physical layer representing physical characteristics of the communications (radio for instance) and the abstract layer modelling node behavior. Then we formalize the communication between nodes by a trace-based model inspired by the symbolic approach proposed by Paulson in [21]. For this, we introduce *send* and *receive* events for the two layers for modelling the communications between nodes. We propose a system of rules for modeling exchanges between nodes given a topology which represents the environment and position between nodes. Moreover, we notice that most of the neighborhood discovery protocols use time measurements and can work even for mobile nodes, so we add timestamps in all our events. We also explain how it is possible to model a protocol in our framework, by generating new rules according to the specification of a given protocol. Finally we give a deduction system which models the abilities of an intruder. Using all these ingredients we can define the property of neighborhood by generating a special event called *END* when a nodes conclude that he is neighbor of another node. Consequently, an attack is the situation where a nodes concludes that another node is his neighbor according to the protocol but indeed they are not able to communicate directly. A protocol is secure for this property if for any execution in presence of intruder, in a given topology, there exists no attack. Then we extend it to (k)-neighborhood, i.e. we propose a formal definition for modelling that two nodes can communicate with k hops. We give examples illustrating how our framework

[1] We remark that is not the case for the authentication which is a property based on exchanged messages.

works to discover attack on a protocol given an intruder and a topology or to prove the security of a protocol under some assumptions. Finally we propose a protocol, called *Sharek* protocol, for discovering the (k)-neighborhood of a node based on a secure (1)-neighborhood secure protocol. This protocol can be used to securely discover the set of neighbors of neighbors, and further. It can help to determine the dominating sets of a wireless network, as explained in [4, 8, 33]. The main goal of this protocol is to present, as far as we know, the first secure protocol which can be used to securely discover the (k)-neighborhood of a node. Of course using our formalism, we then provide a formal proof of security for this protocol.

8.1.3 Related Works

Neighborhood discovery and analysis of protocols are active research topics in wireless networks security. One of the first neighborhood discovery protocol was given by Brands and Chaum in [5]. Later other works have approached the problem differently using for instance directional antennas [11, 31], probabilistic protocols and challenge-response in RFID context [12], or specific protections against some attacks by analyzing network topology based on time of flight of messages in [29].

One of the first formal verification of neighbor discovery protocol is the work done by Meadows et al. in [17]. In this paper they developed a formal methodology to prove properties of distance bounding protocols [17]. They extend the authentication logic to reason about distance bounding property and they extend the protocol of S. Brands and D. Chaum. However, they do not consider (k)-neighborhoods, and deal only with static nodes.

In [23], the authors investigate the possibility of neighborhood detection. They consider several transmission speeds, directional emissions, localization and clock synchronization, and conclude by proving that time-based protocols cannot securely detect neighborhoods if and only if intruders can forward faster than legitimate nodes. They also consider protocols which use location data.

Another formal approach for distance bounding protocols is introduced by Basin et al. in [27]. Their approach also uses notions of distances, time and events-trace. They use the Isabelle/HOL proof assistant [19] to check results from their model. They apply their model on three security protocols, the authentication ranging protocol [6], the distance bounding protocol of [5], which calculates distance between two nodes using communication time and finally the TESLA protocol of [22]. In their work they only consider static nodes and do not use (k)-neighborhoods.

In [30] the authors present a systematic technique for verifying that location discovery protocols satisfy locale authentication whereby an entity can authenticate the physical location of a device, even in the presence of malicious adversaries. They extend the strand space theory with a metric that captures the geometric properties of time and space. They prove that several prominent location discovery protocols including GPS do not satisfy the local authentication goal and analyze a location discovery protocol that does satisfy the goal under some reasonable assumptions.

In some recent works, Cremers et al. [7] propose distance bounding protocols resistant to distance hijacking attacks. The authors in [25] and [24] build a formal model based on traces. They apply their model on a protocol they introduced and on the temporal packet leash protocol [14]. However their method does not take the mobility of the nodes into account.

More recently, in [2] the authors propose a framework for analyzing distance bounding protocols in RFID, using a computational approach and proposed a white-box and black-box point of view. Our work differs from those by using a symbolic approach which abstracts a way most of the cryptographic considerations.

8.1.4 Outline

In Sect. 8.2, we present our model. We first give the network representation, the events and traces definitions, and show how to specify a protocol and an intruder. We also illustrate by a running toy example all our definitions. Then in Sect. 8.3, we define the neighborhood property. We extend the usual notion of (1)-neighborhood to (k)-neighborhood. In Sect. 8.4, we illustrate our approach by analyzing a protocol proposed in [23]. We prove that it is secure against a certain class of intruders, but is vulnerable against stronger intruders. In Sect. 8.5, we propose a secure protocol to build the (k)-neighborhoods of a node, based on any secure (1)-neighborhood protocol. Finally we prove it is secure against a class of intruders, before concluding in the last section.

8.2 Timed Model

Let us consider the topology depicted in Fig. 8.1. In this case, if communication times are proportional to the distances, then, due to signal reflection on the wall and to the presence of a wall between A and B, we have that the transfer time between A and I plus the transfer time between I and B is strictly smaller than

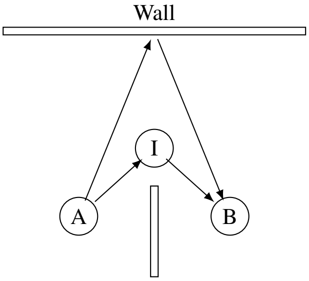


Fig. 8.1 Triangle inequality counter-example

$$A \xrightarrow{Ping} B$$

$$A \xleftarrow{\langle B{,}Pong\rangle} B$$

Fig. 8.2 Ping-Pong protocol communications diagram

the transfer time between A and B. This simple situation shows that in some setting triangular inequality may not always hold in wireless connections. It is why we consider transfer time and not distances in our approach.

Moreover our model is only relying on the possibility of communication between two nodes. We consider the time needed to transmit a message between two nodes. By consequence two nodes are neighbors if they can communicate directly, i.e. their transmission time is finite. Moreover, in order to analyze the security of the protocols we consider either honest nodes, which follow the protocol, or malicious nodes. We illustrate our model by using a flawed toy example: Ping-Pong protocol. This protocol finds neighbors of a node by simply sending them a solicitation (the Ping), to which neighbors will answer with their name and a Pong, denoted by the tuple $\langle B, Pong\rangle$. Once the Pong is received, the protocol claims that the initiator and the node whose identity is in the Pong message are neighbors. It is obviously not secure, but will help us to present our framework. A high-level description of this protocol is given in Fig. 8.2.

In this section we start by formally defining characteristics of a network and nodes. Then we model behaviors of nodes using two family of events with timestamps. We explain using rules describing a given protocol how to build traces, which are sequences of events. We model by a set of rules the intruder capabilities. Finally, we explain two realistic assumptions, which allow us to consider mobility in our approach.

8.2.1 Networks and Nodes

Wireless Sensors Networks are composed of several nodes, that are small devices usually equipped with sensors, a battery and a radio. They use their radio for sending their measurements through the network. To keep each device cheap, they have a small memory and limited computing power. Each node has an identity and can also possess pre-shared cryptographic keys depending on the application. A network is composed of a set of nodes (identified by an unique number) and a topology, which represents communications between nodes.

Definition 8.1 (Network). A network $\mathbf{N}$ is defined by two components $(V, \mathbf{T})$ where:

- V denotes the set of all node identities, which is partitioned into two sets: $V_{\mathfrak{P}}$ denotes all the honest nodes who are following the protocol and V_I represents the intruder nodes which are malicious.

- $\mathbf{T}$ represents the matrix containing the time of communication between two nodes. More precisely, $\mathbf{T}_{A,B}$ denotes the time that a message takes starting from node A to reach node B. If A cannot reach directly B then $\mathbf{T}_{A,B}$ is equal to $+\infty$. We also require that the communication time between a node and itself is null: $\forall A \in V, \mathbf{T}_{A,A} = 0$.

We denote honest node identities by $A,B,C,\ldots$, and $I,J,..$ for intruders. By extension, we will often refer to nodes by their identity (for example, node A).

This definition models only static networks. In order to express the possible movements of nodes, we need to take into account the changes in topology over time, and so we extend the previous definition by making the communication time depend on time. $\mathbf{N} = (V, \mathbf{T})$ becomes $\mathbf{N} = (V, \mathbf{T}(t))$, where the element of the matrix are functions with parameter t. Each element $\mathbf{T}_{A,B}(t)$ models how much time a message **emitted** by the node A at time t takes to reach node B. If needed, we use $\mathbf{T}_{A,B}(t)$ instead of $\mathbf{T}_{A,B}$.

We denote $\mathcal{N}$ the set of all possible networks, which takes into account any number of nodes, any number of intruders, and any possible evolution of the connections over time. Here are a few examples of useful network families:

- $\mathcal{N}_0$ is the family of networks where no intruder is present at all. This allows us to show that a protocol is insecure even when no intruder is present.

$$\mathcal{N}_0 = \left\{\mathbf{N} = (V, \mathbf{T}(t)) \in \mathcal{N} \mid V_I = \emptyset\right\}.$$

- $\mathcal{N}_{sym}$ is the family of networks where message transfer times are symmetrical.

$$\mathcal{N}_{sym} = \left\{\mathbf{N} = (V, \mathbf{T}(t)) \in \mathcal{N} \mid \forall t, \forall A,B \in V, \mathbf{T}_{A,B}(t) = \mathbf{T}_{B,A}(t)\right\}.$$

- $\mathcal{N}_{fully\_connected}$ is the family of networks where each node can communicate with any other node at any time.

$$\mathcal{N}_{fully\_connected} = \left\{\mathbf{N} = (V, \mathbf{T}(t)) \in \mathcal{N} \mid \forall t, \forall A,B \in V, \mathbf{T}_{A,B}(t) \neq +\infty\right\}.$$

- $\mathcal{N}_{still}$ is the family of networks where there is no change in transfer times depending on time, or in other words a static network.

$$\mathcal{N}_{still} = \left\{\mathbf{N} = (V, \mathbf{T}(t)) \in \mathcal{N} \mid \forall t_1, t_2, \forall A,B \in V, \mathbf{T}_{A,B}(t_1) = \mathbf{T}_{A,B}(t_2)\right\}.$$

8.2.2 Events

Neighborhood is a physical property, which is linked to communication channels. In order to isolate physical communications and abstract commands, we consider two distinct layers, one of which is strictly restricted to model physical behavior, and the other corresponding to the abstract behavior of the nodes, which correspond to computations performed by a node. In order to model communication between nodes and behaviors of a node on these two layers we define the following events:

- $send_{\phi}(A,m,t)$: A transmits m at time t using the physical layer.
- $recv_{\phi}(A,m,t)$: A receives m at time t using the physical layer.
- $send_{\alpha}(A,m,t)$: A orders the transmission of m at time t using the logical layer.
- $recv_{\alpha}(A,m,t)$: A received and processed m at time t using the logical layer.

Physical events are annotated by ϕ and abstract events are annotated by α. In order to construct meaningful sequences of events, nodes need to be able to know when a message was sent or received at the physical layer. It is why each event has a timestamp. In the Ping-Pong protocol the first action performed by A is modeled by the event $send_{\alpha}(A,Ping,t_0)$ and $send_{\phi}(A,Ping,t_1)$, where t_0 is the time when the node A transmit to the radio the message $Ping$ and t_1 is the time when the radio of the node A emits the message.

We also introduce an extra event $END(N_1,\ldots,N_k,t_{prop},t)$, which models the fact that a protocol ended well at time t, and concludes something about the nodes $N_1,\ldots,N_k$ at time t_{prop}. For example, if we keep our objective of neighborhood discovery protocols, that conclusion could be the possibility of direct communication between the specified nodes at the time of the property. In our running example, A would like to conclude after the exchange of messages that he has B as neighbor. We model it by the event $END(A,B,t_2,t_4)$, where t_4 is the time when the protocol ended and t_2 is the time when the node A concludes that he has B as neighbor. We decide to store in the event $END(A,B,t_{prop},t)$ both the time t_{prop} when the message was sent by A to B, because if B receives it they are neighbors and the time t when A receives a reply from B, because here also they are neighbors.

8.2.3 Rules and Traces

We now explain how to construct possible communications between different nodes. We build traces which are sequence of events. In order to generate these sequence we use three sets of rules which model the protocol, the communications between nodes and also the intruder. We first present how to use a set of rules for building a trace. We illustrate how to model a protocol with our running example, before showing set of communication rules which are always present in our model. Finally we give the description of intruder rules in Sect. 8.2.4.

8.2.3.1 Building Traces from Rules

Our approach is based on the trace-based modeling presented originally by Paulson for cryptographic protocols in [21]. The rules represents a step in the construction of a trace and has the following form:

$$(R)\frac{H_1 \ldots H_n}{C}$$

where R is the name of the rule, $H_1 \ldots, H_n$ are the hypothesis that have to be satisfied in order to produce the conclusion C.

A trace is a set of events which models the communication between nodes during the execution of a protocol in a given network and a given intruder behavior. Each node can emit and receive messages on the abstract layer, messages can be transferred from a layer to another, and lastly the communication between two nodes's physical layers allow messages to go from a node to another. Those three processes are modeled with rules which are common to all networks (details are given in the next paragraphs). We remark that the traces are sets of events, not sequences as in Paulson's model: since we have timestamps for each event, the order is implicit.

Intruders are modeled by $\mathcal{I}$, a set of rules that describes their abilities and behaviors. A precise description of the intruder model is given in Sect. 8.2.4. The behavior of a node which respects a protocol is modeled by a set of rules denoted $\mathcal{P}$. Usually protocols are specified at the logical layer, and generate an *END* event when the protocol finishes to model what the protocol claims. We denote by $S_{\mathbf{N},\mathcal{I},\mathcal{P}}$ the set of all traces, i.e. possible executions, built by successive applications of rules of the system.

We now write the three rules of $\mathcal{P}_{PingPong}$ needed to model this protocol.

$$(PingPong\_1)\frac{tr \in S_{\mathbf{N},\mathcal{I},\mathcal{P}}}{(tr \cup \{send_\alpha(A, Ping, t)\}) \in S_{\mathbf{N},\mathcal{I},\mathcal{P}}}$$

$$(PingPong\_2)\frac{tr \in S_{\mathbf{N},\mathcal{I},\mathcal{P}} \qquad recv_\alpha(A, Ping, t_1) \in tr \qquad t_1 \leq t_2}{(tr \cup \{send_\alpha(A, \langle A, Pong\rangle, t_2)\}) \in S_{\mathbf{N},\mathcal{I},\mathcal{P}}}$$

$$(PingPong\_3)\frac{\begin{array}{c} tr \in S_{\mathbf{N},\mathcal{I},\mathcal{P}} \\ send_\alpha(A, Ping, t_1) \in tr \qquad send_\phi(A, Ping, t_2) \in tr \\ recv_\alpha(A, \langle B, Pong\rangle, t_3) \in tr \qquad t_1 \leq t_2 \leq t_3 \leq t_4 \end{array}}{(tr \cup \{END(A, B, t_2, t_4)\}) \in S_{\mathbf{N},\mathcal{I},\mathcal{P}}}$$

The two first rules correspond to the two rules of the protocol, properly specified with timestamps on the abstract layer. The last rule raises the *END* flag, which models the fact that the protocol reached a conclusion about A and B.

8.2.3.2 Communication Rules

In Fig. 8.3, we present the set of rules inherent to all networks.

The first rule is the initialization of a trace which is made by axiom (*Begin*). It means that all trace starts empty. Application of other rules adds events in order to obtain a complete execution trace. To generate events from one layer to the other,

$$(Begin)\frac{}{\emptyset \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(Con0)\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad send_{\alpha}(A,m,t_1) \in tr \qquad t_1 + \delta/2 \leq t_2}{(tr \cup \{send_{\phi}(A,m,t_2)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(Con1)\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad recv_{\phi}(A,m,t_1) \in tr \qquad t_1 + \delta/2 \leq t_2}{(tr \cup \{recv_{\alpha}(A,m,t_2)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(Phy)\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad send_{\phi}(A,m,t_1) \in tr \qquad \mathbf{T}_{A,B}(t_1) \leq (t_2 - t_1)}{(tr \cup \{recv_{\phi}(B,m,t_2)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

Fig. 8.3 Basic intruder knowledge building rules

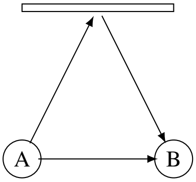


Fig. 8.4 Network where the same message hits the same node twice

there are two rules $(Con0)$ and $(Con1)$, where δ denotes the delay a node needs to relay a message. Finally we have the rule (Phy) which allows messages to go from a node's physical layer to another node, in a time greater than or equal to $\mathbf{T}_{A,B}(t_1)$. This rule may be applied multiple times for a single send event, to model signal reflections which may cause a message to be received twice or more, as shown in Fig. 8.4. It is why there is no upper bounds on t_2. We also assume that all the nodes have access to a $New\_Nonce()$ function, which returns a fresh random value at each new call.

We consider our running example and give the trace corresponding to the following scenario. A ordered the broadcast of message "*Ping*", which was transferred to the physical layer. Then B receives the message in their physical layer, which later on got shifted onto the logical layer. Finally B replies to A, his name and "*Pong*", and A concludes that he is neighbor with B. We obtain the following trace assuming that $\delta = 1$ and the communication between A and B is symmetric and for all t, $\mathbf{T}_{A,B}(t) = 10$. For the *END* event, see rule $(PingPong_3)$ for the choice of $t_{prop} = 1$.

- $send_{\alpha}(A, Ping, 0)$
- $send_{\phi}(A, Ping, 1)$
- $recv_{\phi}(B, Ping, 11)$
- $recv_{\alpha}(B, Ping, 12)$
- $send_{\alpha}(B, \langle B, Pong \rangle, 12)$

- $send_{\phi}(B, \langle B, Pong\rangle, 13)$
- $recv_{\phi}(A, \langle B, Pong\rangle, 23)$
- $recv_{\alpha}(A, \langle B, Pong\rangle, 24)$
- $END(A, B, 1, 24)$

The first time stored in the event $END(A, B, 1, 24)$ corresponds to the time when A sent a message to B and the second one is the time when A receives an answer from B. With these times we catch all the times where the nodes are neighbors. It helps us to cover for instance the following situation: node B receives a message from A and moves out to the range of A during a while, then he answers when he is again at communication distance of A.

8.2.4 Intruder Rules

Intruder capabilities are described by the set of rules denoted I, common to all the intruder nodes. I describes both what an intruder node can know, and what it can do.

8.2.4.1 Intruder Knowledge

Each of the intruder nodes has some knowledge, which is built with rules, alongside the execution of the protocol. That knowledge depends on both what the intruder node was able to hear, and what it knew before the protocol executes. This knowledge is specific to a given node: what a node learns does not instantly propagates to other intruders' memories.

We call that knowledge IK_I for the node I (and IK_J for J, and so on). More precisely, $IK_I(tr,t)$ represents what I knows at time t assuming the events from trace tr happened, and $\widehat{IK_I}(tr,t)$ represents what I can deduce from $IK_I(tr,t)$. We denote $IK_I(\emptyset, 0)$ the initial knowledge of the intruder I. For instance, in a classical outsider attack, all of the $IK_I(\emptyset, 0)$ would contain the identifiers of the nodes in the network and some fictive ones.

On the abstract level, we adapt the classical Dolev-Yao intruder models [9] for building the intruder knowledge. The adapted Dolev-Yao deduction system is given by the rules in Fig. 8.5 (containing the basic knowledge-building rules) and Fig. 8.6, which describes the rules for symmetric and asymmetric cryptography, and nonce generation. To avoid confusion, all symmetric keys are labelled k, and asymmetric keys are couples (sk, pk) and encryptions are represented by $\{m\}_k$. Notice that an intruder I can increase his knowledge using a message m at time t_2 only if there is a $recv_{\alpha}(I, m, t_1) \in tr$ with $t_1 \leq t_2$.

$$(IK\_init)\,\frac{m \in IK_I(\emptyset,0)}{m \in IK_I(tr,t)}$$

$$(IK\_hat\_promotion)\,\frac{m \in IK_I(tr,t)}{m \in \widehat{IK_I(tr,t)}}$$

$$(IK\_left)\,\frac{\langle m,n\rangle \in \widehat{IK_I(tr,t)}}{m \in \widehat{IK_I(tr,t)}}$$

$$(IK\_right)\,\frac{\langle m,n\rangle \in \widehat{IK_I(tr,t)}}{n \in \widehat{IK_I(tr,t)}}$$

$$(IK\_pair)\,\frac{m \in \widehat{IK_I(tr,t)} \qquad n \in \widehat{IK_I(tr,t)}}{\langle m,n\rangle \in \widehat{IK_I(tr,t)}}$$

Fig. 8.5 Basic intruder knowledge building rules

$$(IK\_decrypt\_sym)\,\frac{\{m\}_k \in \widehat{IK_I(tr,t)} \qquad k \in \widehat{IK_I(tr,t)}}{m \in \widehat{IK_I(tr,t)}}$$

$$(IK\_encrypt\_sym)\,\frac{m \in \widehat{IK_I(tr,t)} \qquad k \in \widehat{IK_I(tr,t)}}{\{m\}_k \in \widehat{IK_I(tr,t)}}$$

$$(IK\_decrypt\_asym\_sk)\,\frac{\{m\}_{sk} \in \widehat{IK_I(tr,t)} \qquad pk \in \widehat{IK_I(tr,t)}}{m \in \widehat{IK_I(tr,t)}}$$

$$(IK\_encrypt\_asym\_sk)\,\frac{m \in \widehat{IK_I(tr,t)} \qquad sk \in \widehat{IK_I(tr,t)}}{\{m\}_{sk} \in \widehat{IK_I(tr,t)}}$$

$$(IK\_encrypt\_asym\_pk)\,\frac{m \in \widehat{IK_I(tr,t)} \qquad pk \in \widehat{IK_I(tr,t)}}{\{m\}_{pk} \in \widehat{IK_I(tr,t)}}$$

$$(IK\_nonce)\,\frac{N_I = New\_Nonce()}{N_I \in \widehat{IK_I(tr,t)}}$$

Fig. 8.6 Cryptography-related intruder knowledge building rules

$$(IK\_recv)\frac{recv_{\alpha}(I,m,t_1) \in tr \qquad t_1 \leq t_2}{m \in IK_I(tr,t_2)}$$

$$(Intrude\_replay)\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad m \in IK_I(tr,t_1) \qquad t_1 \leq t_2}{(tr \cup \{send_{\alpha}(I,m,t_2)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(Intrude\_forge)\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad m \in \widehat{IK_I}(tr,t_1) \qquad t_1 \leq t_2}{(tr \cup \{send_{\alpha}(I,m,t_2)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

Fig. 8.7 Intruder knowledge input and usage rules

8.2.4.2 Intruder Actions

The rules to link the transmission model and the intruder knowledge are given in Fig. 8.7. (IK_recv) is used so that intruders can hear communications, and learn from them. The rule $(Intrude\_replay)$ models how the intruder can replay messages it heard. The rule $(Intrude\_forge)$ represents how an intruder can deduce and build new messages from his knowledge and send them. The distinction between those two rules allows simple intruders which would only be able to relay messages, formally described by not including $(Intrude\_forge)$ in I. Usually, I can forge the same messages as the protocol $\mathcal{P}$, in the case of an intruder with a large enough initial knowledge $IK_I(\emptyset,0)$.

8.2.5 Mobility

Mobility of nodes is already included in the model, since the values of $\mathbf{T}(t)$ varies with time. However, we need to add two realistic assumptions in our model.

Assumption 8.1. *A node moves much slower than a message.*

This allows us to assume that values in $\mathbf{T}(t)$ takes into account the (negligible) movements of the nodes during the subsequent transfer time (since $\mathbf{T}(t)$ stores a fixed configuration of nodes and their movement depending on time). This is straightforward: radio messages travel near the speed of light, while wireless sensors are usually slower.

Assumption 8.2. δ *is in the same order of magnitude of value as the message transfer time.*

This assumption is related to rules $(Con0)$ and $(Con1)$. Remember that δ is the time needed to forward a message.

Taking into account the previous assumption and this one, we can deduce that sending and relaying messages does not take a significant time with regard to node movement. To restate these two rules in a less formal way, we assume that it is possible that during message transfers and relayings, the values of $t_{transfer}(A,B,t)$ do not change significatively, no matter which $\mathbf{N}$ is used.

On the other hand, it is always possible to add arbitrary delays to operations: see rules (Phy), $(Con0)$, $(Con1)$ and $(Intrude\_replay)$. For instance, these delays may be used in an attack by allowing synchronization of multiple sessions of a protocol.

8.3 Neighborhood and (k)-Neighborhoods

8.3.1 Neighborhood

A node A is *neighbor or* (1)*-neighbor to the node B at time t* if $\mathbf{T}_{A,B}(t)$ is finite which means that A is in communication range of B.

Definition 8.2 (Neighborhood). Let $\mathbf{N} = (V, \mathbf{T}(t))$ be a network, the *neighborhood of a node at time t*, denoted by $\mathbf{Ng}^1_A(t)$, is the set of nodes that A can reach with a message sent at time t. It is formally defined by:

$$\mathbf{Ng}^1_A(t) = \{X \mid X \in V \wedge \mathbf{T}_{A,X}(t) < +\infty\}.$$

If $tr \in S_{\mathbf{N},I,\mathcal{P}}$ is a trace, we denote by $\widehat{tr}$ the set of all possible traces that can be inferred from tr using rules defined previously. It allows us to define $t_{transfer}(A,B,)$, the smallest time possible needed to send a fresh message m from A to B at time t using only communication rules.

We also define a protocol $\mathcal{P}_{Forward}$ and an intruder $I_{Forward}$ which consist in the following unique rule which makes nodes forward message on abstract layer:

$$(Forward)\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad recv_{\alpha}(A,m,t_1) \in tr \qquad t_1 \leq t_2}{(tr \cup \{send_{\alpha}(A,m,t_2)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

Definition 8.3 ($t_{transfer}(A,B,t)$). Let $\mathbf{N} = (V, \mathbf{T}(t))$ be a network, and $tr = \{send_{\phi}(A, m,t)\}$ be a trace in $S_{\mathbf{N},I_{Forward},\mathcal{P}_{Forward}}$. We define $t_{transfer}(A,B,t)$ by:

- $t_{transfer}(A,B,t) = \min\{x - t \mid recv_{\phi}(B,m,x) \in \widehat{tr} \subseteq S_{\mathbf{N},I_{Forward},\mathcal{P}_{Forward}}\}$.
- If the event $recv_{\phi}(B,m,t)$ does not belong to $S_{\mathbf{N},I_{Forward},\mathcal{P}_{Forward}}$ then we state that $t_{transfer}(A,B,t) = +\infty$.

We use $I_{Forward}$ and $\mathcal{P}_{Forward}$ to consider the simplest situation, where node can only forward messages. The send event introduced in all generated trace allows us to build all possible routes taken by this message. Note that the $(Forward)$ rule does not impose to forward the message immediately.

$t_{transfer}(A,B,t)$ is different than $\mathbf{T}_{A,B}(t)$, because it can take into account multiple hops, and the relay times needed in the intervals. Also, there may be routes which are faster than the direct ones (with some conditions on δ and $\mathbf{N}$, as in the first example in Fig. 8.1).

We denote by $t_{max\_emitter}$ the maximum positive finite time in $\mathbf{T}(t)$, i.e. the maximal communication time of two connected nodes. If we assume that

$\delta > t_{max\_emitter}$ then we are able to characterize the definition of $\mathbf{Ng}^1_A(t)$ by relations between $t_{max\_emitter}$ and $t_{transfer}(A,B,t)$.

Lemma 8.1. *Let $\mathbf{N} \in \mathcal{N}$ be a network containing at least the two nodes A and B. If we assume that $\delta > t_{max\_emitter}$, the following properties are equivalent:*

1. $B \in \mathbf{Ng}^1_A(t)$
2. $t_{transfer}(A,B,t) \leq t_{max\_emitter}$

Proof. We prove a double implication.

- 1 ⇒ 2: Let us assume that $B \in \mathbf{Ng}^1_A(t)$. We know there exists a sequence of events in $S_{\mathbf{N},I_{Forward},\mathcal{P}_{Forward}}$ containing only the event $send_\phi(A,m,t)$. Since $B \in \mathbf{Ng}^1_A(t)$, we can generate a valid trace $tr' = \{send_\phi(A,m,t), recv_\phi(B,m,t+\mathbf{T}_{A,B}(t))\}$, using (Phy). Therefore, $t_{transfer}(A,B,t) \leq \mathbf{T}_{A,B}(t)$. We now prove that this time is the smallest value for $t_{transfer}(A,B,t)$. We assume the opposite $t_{transfer}(A,B,t) < \mathbf{T}_{A,B}(t)$ meaning that $t'-t < \mathbf{T}_{A,B}(t)$ where $recv_\phi(B,m,t')$. The only other way to generate a trace containing $recv_\phi(B,m,t')$ would be to consider that there exists a node X who has sent m to B (using rule (Phy) again). In other words there is a trace containing $send_\phi(X,m,t_{s_X})$ with $t' > t_{s_X}$. Since m is fresh, this event can only be generated from a trace containing a $recv_\phi(X,m,t_{r_X})$ event (using rules $(Con0)$ and $(Con1)$), and we know that $t_{s_X} - t_{r_X} \geq \delta > t_{max\_emitter}$ (by hypothesis). Moreover, also due to the freshness of m, we have $t_{r_X} > t$. We deduce that $t'-t > t_{s_X} - t_{r_X} > t_{max\_emitter}$. This leads to a contradiction with $t'-t < \mathbf{T}_{A,B}(t)$. We conclude that $t_{transfer}(A,B,t) = \mathbf{T}_{A,B}(t)$ then by definition of $t_{max\_emitter}$ we obtain that $t_{transfer}(A,B,t) \leq t_{max\_emitter}$.
- 2 ⇒ 1: We make a proof by contradiction. We assume that $B \notin \mathbf{Ng}^1_A(t)$, our aim is now to prove that $t_{transfer}(A,B,t) > t_{max\_emitter}$. We distinguish two cases:
  - $t_{transfer}(A,B,t) = +\infty$. In this case, by definition of $t_{max\_emitter}$, we immediately conclude.
  - $t_{transfer}(A,B,t)$ is finite. It means that there exists a way to send a fresh message m between A and B, using a forwarder node which take at least δ units of time. Since $\delta > t_{max\_emitter}$, we conclude that $t_{transfer}(A,B,t) > t_{max\_emitter}$.

□

As we have already explained, the description of a protocol makes claims (by the mean of an *END* event). In the modeling of the protocol this rule can only be used if the protocol claims that a node is a (1)-neighbor of another node. Our aim is to give a definition which is satisfied only if the protocol is secure: meaning that the protocol reach the flag *END* and the two considered nodes are really neighbor. Our idea for (1)-neighborhood is to prove that if a protocol claims to satisfy (1)-neighborhood then there exists a direct way of communication between the nodes. It is captured by the following definition.

Definition 8.4. Let $\mathcal{N}$ be a family of networks and I an intruder. A protocol $\mathcal{P}$ verifies the (1)-neighborhood relation in presence of an intruder I over $\mathcal{N}$ if and only if $\forall \mathbf{N} = (V, \mathbf{T}(t)) \in \mathcal{N}, \forall A,B \in V$, $\nexists tr \in S_{\mathbf{N},I,\mathcal{P}}$ such that $END(A,B,t,t_x) \in tr \wedge B \notin \mathbf{Ng}^1_A(t)$.

To state it otherwise, a protocol does not verify a neighborhood property in presence of an intruder I over a set of networks $\mathcal{N}$ if and only if there is at least a trace tr in the networks in $\mathcal{N}$, built by the given protocol and intruder rules, such that $END(A,B,t_{prop},t) \in tr$ and there is no direct communication possibility between those nodes at time t_{prop}. This trace contains an example attack.

Let us illustrate the neighborhoods, the intruder's rules and the formal definition of verifying a property by building an attack on our running example. We define $\mathbf{N}$ by $\forall t, \mathbf{Ng}_A^1(t) = \mathbf{Ng}_I^1(t) = \{A,I\}$ which is simply two nodes in range of each other, one being honest (A) and one malicious (I). We assume for simplicity that nodes do not move and can communicate in finite time.

Regarding the intruder, we consider a basic I who can forward and forge messages as previously described. We also choose $IK_I(\emptyset,0) = \{A,I,Z,Ping,Pong\}$, the initial knowledge of I at time 0, with Z an nonexistent node identity.

The scenario of the attack is the following: A starts the protocol by broadcasting $Ping$ using rules ($Begin$) and ($PingPong\_1$). The intruder I receives the $Ping$ using (Con_0), (Phy) and (Con_1). Now, I forges $\langle Z, Pong\rangle$, which is known since both parts are in IK_0, and sends that message to A (rule ($Intrude\_forge$)). Then using ($PingPong\_1$) it is possible to infer $END(A,Z,t_2,t_9) \in tr$ with the appropriate time values. The final trace we obtain corresponds to a possible attack, and contains the following set of events:

$\{send_\alpha(A,Ping,t_1), send_\phi(A,Ping,t_2), recv_\phi(I,Ping,t_3), recv_\alpha(I,Ping,t_4),$
$send_\alpha(I,\langle Z,Pong\rangle,t_5), send_\phi(I,\langle Z,Pong\rangle,t_6), recv_\phi(A,\langle Z,Pong\rangle,t_7),$
$recv_\alpha(A,\langle Z,Pong\rangle,t_8), END(A,Z,t_2,t_9)\}$, where $t_1 < t_2 < t_3 < t_4 < t_5 < t_6 < t_7 < t_8 < t_9$.

Now let us go back to Definition 8.4. Here, we have a $\mathbf{N}$, where $Z \notin \mathbf{Ng}_A^1(t_2)$, because there is no Z in the network. We just built a possible trace in this setting including $END(A,Z,t_2,t_9)$, which means the protocol has detected a neighborhood relation between A and Z at time t_2. Therefore the protocol does not verify the neighborhood property with respect to the previously defined intruder I, with his knowledge $IK_I(\emptyset,0)$, on all the families of networks $\mathcal{N}$ which contain this example $\mathbf{N}$.

8.3.2 (k)-Neighborhood

We extend our definition in order to determine if a node is neighbor to another one after k hops. For simplicity's sake, we consider that all nodes have the same relaying time δ. We can easily generalize our results with a different time for each node in the network.

Definition 8.5 ((k)-or-less-neighborhood). Let $\mathbf{N} = (V, \mathbf{T}(t))$ be a network, $\mathbf{Ng}_A^{\leq k}$ (the (k)-or-less-neighborhood of A) is the set of all the nodes B for which there is a path starting at A at time t and ending at B with k hops or less, taking into account the minimal flight time of messages and the relaying time δ. We define it recursively by:

$$\mathbf{Ng}_A^{\leq k}(t) = \left\{ B \mid \left(X \in \mathbf{Ng}_A^1(t) \right) \wedge \left(B \in \mathbf{Ng}_X^{\leq (k-1)}(t + \mathbf{T}_{A,X}(t) + \delta) \right) \right\}.$$

This recursive definition means that the (k)-or-less neighborhood of A at time t is the union, for all the neighbors X reachable by A at time t, of the $(k-1)$-or-less-neighborhoods of the different X at time $t + \mathbf{T}_{A,X}(t) + \delta$. Colloquially, a (k)-neighbor of A is a node which can be reached in k or less hops by a message sent from A at time t, and relayed through any other nodes. As expected, the (1)-or-less-neighborhood is the same thing as the neighborhood given in Definition 8.2. According to our definition, (k)-or-less-neighborhood has the following straightforward property.

Property 8.1. *Let* $\mathbf{N} = (V, \mathbf{T}(t))$ *be a network, then we have:*

$$\mathbf{Ng}_A^{\leq k}(t) \subseteq \mathbf{Ng}_A^{\leq (k+1)}(t).$$

We now define the (k)-neighborhood, which is the set of nodes for which there is a path (in the sense of the previous definition) of length exactly k. It is the set of all the nodes B for which there is a path starting at A at time t and ending at B with k hops, but there is no such path with $k-1$ hops or less.

Definition 8.6 ((k)-neighborhood). Let $\mathbf{N} = (V, \mathbf{T}(t))$ be a network, $\mathbf{Ng}_A^k$ ((k)-neighborhood of A) is defined by:

$$\mathbf{Ng}_A^k(t) = \left(\mathbf{Ng}_A^{\leq k}(t) \right) \setminus \left(\mathbf{Ng}_A^{\leq k-1}(t) \right).$$

Now, as in the previous subsection, we can formally define what is a protocol which verifies the (k)-or-less-neighborhood relation in our framework.

Definition 8.7. Let $\mathcal{N}$ be a network family. A protocol $\mathcal{P}$ verifies the (k)-or-less-neighborhood relation in presence of an intruder I if and only if $\forall \mathbf{N} = (V, \mathbf{T}(t)) \in \mathcal{N}, \forall A, B \in V$, $\nexists tr \in S_{\mathbf{N},I,\mathcal{P}}$ such that $END(A, B, t, t_x) \in tr \wedge B \notin \mathbf{Ng}_A^{\leq k}(t)$.

8.4 Example: Authenticated Ranging Protocol

We consider the protocol proposed in [23]. This protocol aims to verify (1)-neighborhood between two nodes, by using an upper bound on message transfer time between neighbors. We first describe the protocol, then we give a modelling of this protocol, and after that we prove its correctness in our setting.

$$A \xrightarrow{N_A} B$$

$$A \xleftarrow{\{B,N_A\}_{sk_B}} B$$

Fig. 8.8 Authenticated protocol communications diagram, where N_A denotes the nonce generated by A and $\{m\}_{sk_B}$ denotes the message m signed by B

8.4.1 *Description of the Protocol*

The idea of the protocol is simple: a message is sent with a nonce, and anyone hearing it replies with that nonce, signed. An Alice-Bob representation of this protocol is given in Fig. 8.8.

8.4.2 *Protocol Modelling*

This protocol is in one way a secured version of the Ping Pong protocol described above. If the resulting message is received before $t_{send} + 2 * t_{max\_emitter} + \delta$, then the emitter is a neighbor (since it is both closer than $t_{max\_emitter}$ and able to communicate with us). This way, the protocol can successfully determine if the node is in communication range or not based on physical property induced by the properties of the nodes. We now give the rules modelling this protocol:

$$(AuthRanging\_1)\,\frac{tr \in S_{\mathbf{N},I,\mathcal{P}}}{(tr \cup \{send_\alpha(A, New\_Nonce(), t)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(AuthRanging\_2)\,\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad recv_\alpha(A, N_B, t_1) \qquad t_1 \leq t_2}{(tr \cup \{send_\alpha(A, \langle A, \{N_B\}_{sk_A}\rangle, t_2)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(AuthRanging\_3)\,\frac{\begin{array}{c} tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad send_\alpha(A, N_A, t_1) \qquad send_\phi(A, N_A, t_{s_A}) \\ recv_\phi(A, \langle B, \{N_A\}_{sk_B}, t_{r_A}) \qquad recv_\alpha(A, \langle B, \{N_A\}_{sk_B}, t_4) \\ t_1 \leq t_{s_A} \leq t_{r_A} \leq t_4 \leq t_5 \qquad t_{r_A} - t_{s_A} \leq (2 * t_{max\_emitter} + \delta) \end{array}}{(tr \cup \{END(A, B, t_{s_A}, t_5)\}) \in S_{\mathbf{N},I,\mathcal{P}}}$$

8.4.3 *Authenticated Ranging Protocol Satisfies the Neighborhood Property*

In order to keep our proof concise, we assume symmetry in the time needed to send and receive a message, i.e. $\mathbf{T}_{A,B} = \mathbf{T}_{B,A}$ (the $\mathcal{N}_{sym}$ family of networks). We first precise the intruder model we are using, then we show a proof of the protocol.

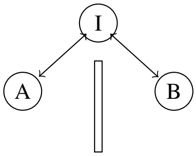


Fig. 8.9 Network of this counter-example

We consider that there is at most one compromised secret key and one compromised node in the whole network. Otherwise, if an intruder has access to a secret key that is not his own (whether it's shared between different nodes, or the intruder has access to a set of them), it is easy to find a trace where an intruder I masquerades as the other intruder J and leads to a $END(A,J)$ event, which by Definitions 8.4 and 8.7 makes the protocol insecure. Therefore we assume intruders can only use their own secret key.

We show that if $\delta < t_{max\_emitter}$ then we can find an attack, but we also prove that if $\delta \geq t_{max\_emitter}$ then the protocol is correct. Our analysis allows us to deduce sufficient conditions of application for the protocol.

8.4.3.1 Existence of an Attack if $\delta < t_{max\_emitter}$

We consider a network containing three nodes A, B and an intruder I, with constant time of transfer ε. The topology of this network is given in Fig. 8.9, where there is no link between nodes A and B, and I can communicate with A and B. Applying the rules of the authenticated ranging protocol, we can construct the following valid tr in $S_{\mathbf{N},I,\mathcal{P}}$ which contains:

- $send_\alpha(A,N_A,0)$
- $send_\phi(A,N_A,0.5\delta)$ (Emission time for A)
- $recv_\phi(I,N_A,0.5\delta+\varepsilon)$
- $recv_\alpha(I,N_A,1\delta+\varepsilon)$
- $send_\alpha(I,N_A,1\delta+\varepsilon)$
- $send_\phi(I,N_A,1.5\delta+\varepsilon)$
- $recv_\phi(B,N_A,1.5\delta+2\varepsilon)$
- $recv_\alpha(B,N_A,2\delta+2\varepsilon)$
- $send_\alpha(B,\{B,N_A\}_{sk_B},2\delta+2\varepsilon)$
- $send_\phi(B,\{B,N_A\}_{sk_B},2.5\delta+2\varepsilon)$
- $recv_\phi(I,\{B,N_A\}_{sk_B},2.5\delta+3\varepsilon)$
- $recv_\alpha(I,\{B,N_A\}_{sk_B},3\delta+3\varepsilon)$
- $send_\alpha(I,\{B,N_A\}_{sk_B},3\delta+3\varepsilon)$
- $send_\phi(I,\{B,N_A\}_{sk_B},3.5\delta+3\varepsilon)$
- $recv_\phi(A,\{B,N_A\}_{sk_B},3.5\delta+4\varepsilon)$ (Reception time for A)
- $recv_\alpha(A,\{B,N_A\}_{sk_B},4\delta+4\varepsilon)$
- $END(A,B,0.5\delta,4\delta+4\varepsilon)$

Now, the *END* event could only be generated if $(3.5\delta + 4\varepsilon) - (0.5\delta) < 2 * t_{max\_emitter} + \delta$, according to rule (*AuthRanging*$_3$). It can be simplified to $\delta + 2 * \varepsilon < t_{max\_emitter}$. Since the *END* event claims neighborhood between A and B at time 0.5δ while there is no neighborhood relation between them, the protocol is flawed if $\delta < t_{max\_emitter}$.

8.4.3.2 The Protocol Is Secure if $\delta \geq t_{max\_emitter}$

In other words we prove that the protocol will never claim there is a neighborhood property between two nodes, when there is actually none. In order to do this, we proceed by contradiction, assuming the protocol is broken and showing a contradiction. We assume the protocol does not satisfy the definition, this means that:

$$\exists tr \in S_{\mathbf{N},I,\mathcal{P}} \ s.t. \ END(A,B,t_{s_A},t) \in tr \wedge B \notin \mathbf{Ng}^1_A(t_{s_A}).$$

Due to the cryptographic primitives, we are sure that the replied message is forged by B and cannot be modified, since an intruder who is not B would not have access to sk_B. Hence we are sure that the message went through B. Then we compute a lower bound for the running time of the protocol by decomposing it in three phases: from A to B, during B's computations and from B to A.

- From A to B: Following the rules of the protocol, in order to generate $END(A,B,t_{s_A},t) \in tr$, the following properties must be true:

  - $send_\phi(A,N_A,t_{s_A}) \in S_{\mathbf{N},I,\mathcal{P}}$
  - $recv_\phi(A,\langle B,\{N_A\}_{sk_B},t_{r_A}) \in S_{\mathbf{N},I,\mathcal{P}}$
  - $t_{r_A} - t_{s_A} \leq (2 * t_{max\_emitter} + \delta)$

  N_A has been sent by A at t_{s_A}, the time when B receives it is denoted by t_{r_B}. Hence we have that $t_{transfer}(A,B,t_{s_A}) \leq t_{r_B} - t_{s_A}$. Applying Lemma 8.1 we obtain:

  $$B \notin \mathbf{Ng}^1_A(t_{s_A}) \Leftrightarrow t_{transfer}(A,B,t_{s_A}) > t_{max\_emitter}.$$

  We deduce that:

  $$t_{r_B} - t_{s_A} > t_{max\_emitter}$$

  B also has access to N_A at least at time t_{r_B} (and not before).
- During B's computations: Applying the rules, we know that the forward time $(t_{s_B} - t_{r_B})$ is greater or equal to δ.
- From B to A: We have $t_{transfer}(B,A,t_{s_B}) \leq t_{r_A} - t_{s_B}$. Because we are in a symmetrical network and due to Assumptions 8.1 and 8.2, we also have $A \notin \mathbf{Ng}^1_B(t_{s_B})$ then we apply Lemma 8.1 once again regarding this message, and therefore A receives the signed answer at best at time t_{r_A} such that:

  $$A \notin \mathbf{Ng}^1_B(t_{s_B}) \Leftrightarrow t_{r_A} - t_{s_B} > t_{max\_emitter}$$

By summing all these bounds we get:

$$t_{r_A} - t_{s_A} = (t_{r_A} - t_{s_B}) + (t_{s_B} - t_{r_B}) + (t_{r_B} - t_{s_A}) > 2 * t_{max\_emitter} + \delta$$

According to the hypothesis related to the *END* event, we have $t_{r_A} - t_{s_A} \leq 2 * t_{max\_emitter} + \delta$ which leads us to a contradiction.

8.5 (k)-or-Less-Neighbors Discovery Protocol

Assuming that we have a secure (1)-neighbor discovery protocol, we propose a (k)-or-less-neighbor discovery protocol based on it. This protocols aims to construct the (k)-or-less-neighborhood, by using the knowledge each node has about its neighborhood. For this protocol, we need to consider several *END* events, each being a stepping stone towards a final conclusion. For instance $END^k(A, B, t_{prop}, t)$ expresses the (k)-or-less neighborhood property between A and B at time t_{prop}. We notice that definitions of how a protocol verifies a property follows the same idea as before: there should not exist a trace where an *END* contradicts the physical property it claims. We call this protocol the *Sharek* protocol.

8.5.1 Description of the Protocol

In the first phase of the protocol, each node floods the network with its (1)-neighborhood. After this, no more communications happen.

Then, all the nodes try to map their (2)-or-less-neighborhoods based on **two or more of their (1)-neighbors claiming neighborhood to another node**. After that, each node continues by computing its upper neighborhood based on its previous deductions. We do not require that each node knows its whole (1)-neighborhood, but only a subset of it. We model this with the *View* function, which is used in the rules of the protocol.

This protocol is sensitive to the number of intruders, this simple version works only when there is only one intruder in the network. This way, the single intruder cannot create false conclusions by lying, since there will not be the same claim from another (necessarily legitimate) node unless it was true in the first place. If we increase the number of approving nodes in the deduction, then we can tolerate more intruders.

There is a small improvement we stacked on top of this: if two nodes claim neighborhood with a third one, but are not at the same level of neighborhood with the center, then the third one will be accepted at the highest level. Since the $(k-1)$-or-less neighborhood is included in the (k)-or-less neighborhood, we can choose the most conservative option and still satisfy the conditions for the verification of neighborhood properties.

$$\forall X \xRightarrow{\{View(\mathbf{Ng}^1_X(t_{prop}))\}_{sk_X}} *$$

Fig. 8.10 Sharek protocol communications diagram

The security proof shows that if the protocol ends by reaching a conclusion, the results are secure. We also assume one of the two following assumptions:

- Nodes have synchronized clocks. This is required by the protocol, which makes nodes use the timestamped data sent by their neighbors. If honest nodes send erroneous times, then the intruder can easily leverage this and trick the protocol into a false conclusion. In the rest, we consider this assumption.
- Nodes are static. That is $\mathcal{N}_{still}$ as defined above. Then, the neighborhoods will not change over time, and the timestamps become trivially useless. It is easy to infer a proof with this assumption from the previous case.

We model the knowledge of a part of the (1)-neighbors by the View function, which returns the part of the (1)-neighbors that the node knows. A high level description of Sharek Protocol is given in Fig. 8.10, where the double arrow symbolize the flooding by repeated broadcasts.

8.5.2 Rules of Sharek Protocol

We give the few rules modeling our protocol:

$$(Sharek\_begin)\,\frac{tr \in S_{\mathbf{N},I,\mathcal{P}}}{tr.send_{\alpha}(A,\{\langle t_{prop},A,View(A,\mathbf{Ng}^1_A)\rangle\}_{sk_A},t) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(Sharek\_basis)\,\frac{tr \in S_{\mathbf{N},I,\mathcal{P}} \qquad B \in View(A,\mathbf{Ng}^1_A(t_{prop})) \qquad t_{prop} \le t}{tr.END^1(A,B,t_{prop},t) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(Sharek\_forward)\,\frac{\begin{array}{c} tr \in S_{\mathbf{N},I,\mathcal{P}} \\ recv_{\alpha}(A,\{\langle t_{prop},B,\{N_1,\ldots,N_k\}\rangle\}_{sk_B},t_1) \in tr \\ t_1 \le t_2 \end{array}}{tr.send_{\alpha}(A,\{\langle t_{prop},B,\{N_1,\ldots,N_k\}\rangle\}_{sk_B},t_2) \in S_{\mathbf{N},I,\mathcal{P}}}$$

$$(Sharek\_makestep)\,\frac{\begin{array}{c} tr \in S_{\mathbf{N},I,\mathcal{P}} \\ recv_{\alpha}(A,\{\langle t_{prop},B,\{D,\ldots\}\rangle\}_{sk_B},t_1) \in tr \\ recv_{\alpha}(A,\{\langle t_{prop},C,\{D,\ldots\}\rangle\}_{sk_C},t_2) \in tr \\ END^b(A,B,t_{prop},t_b) \in tr \\ END^c(A,C,t_{prop},t_c) \in tr \\ \forall t_x \in \{t_1,t_2,t_b,t_c\}, t_x \le t_d \\ (max(b,c)+1) \le k \end{array}}{tr.END^{max(b,c)+1}(A,D,t_{prop},t_d) \in S_{\mathbf{N},I,\mathcal{P}}}$$

8.5.3 Security of Sharek

In order to prove that the protocol verifies the (k)-or-less-neighborhood property, we proceed with a proof by induction. First we prove that our protocol verifies (1)-neighborhood property in Lemma 8.2, then we assume Sharek protocol verifies all (i)-neighborhood property for $i \leq k$ and we show that it verifies $(k+1)$-neighborhood property in Lemma 8.3.

We define $\mathcal{N}_{1,sym}$ as the family of networks where there is only a single intruder, and where all the connections are symmetric.

Lemma 8.2. *The protocol verifies the* (1)*-neighborhood property over* $\mathcal{N}_{1,sym}$*:*

$$\forall \mathbf{N} \in \mathcal{N}_{1,sym}, \nexists tr \in S_{\mathbf{N},I,\mathcal{P}} \ s.t.\ END^1(A,B,t_{prop},t) \in tr \wedge B \notin \mathbf{Ng}_A^1(t_{prop})$$

Proof. We assume that $\exists tr \in S_{\mathbf{N},I,\mathcal{P}}\ s.t.\ END^1(A,B,t_{prop},t) \in tr \wedge B \notin \mathbf{Ng}_A^1(t_{prop})$. The only way we can generate a $END^1(A,B,t_{prop},t)$ event in a trace $tr \in S_{\mathbf{N},I,\mathcal{P}}$ is using $(Sharek\_basis)$. This rule requires $B \in View(A, \mathbf{Ng}_A^1(t_{prop}))$. The $View$ function, by definition, returns a subset of $\mathbf{Ng}_A^1(t_{prop})$: therefore, there is a contradiction. □

Lemma 8.3. *If the protocol verifies all the i-or-less neighborhood properties,* $0 < i \leq k$ *over* $\mathcal{N}_{1,sym}$*, then it verifies the* $(k+1)$*-neighborhood property over* $\mathcal{N}_{1,sym}$*. More formally it means that:*

$$\forall \mathbf{N} \in \mathcal{N}_{1,sym}, \forall i \leq k, \nexists tr \in S_{\mathbf{N},I,\mathcal{P}}\ s.t.\ END^i(A,D,t_{prop},t) \in tr \wedge D \notin \mathbf{Ng}_A^{\leq i}(t_{prop})$$

$$\Rightarrow$$

$$\forall \mathbf{N} \in \mathcal{N}_{1,sym}, \nexists tr \in S_{\mathbf{N},I,\mathcal{P}}\ s.t.\ END^{k+1}(A,D,t_{prop},t) \in tr \wedge D \notin \mathbf{Ng}_A^{\leq k+1}(t_{prop})$$

Proof. We do a proof by contradiction. We assume that there is a $\mathbf{N} \in \mathcal{N}_{1,sym}$ and a $tr \in S_{\mathbf{N},I,\mathcal{P}}$ such that $END^{k+1}(A,D,t_{prop},t) \in tr \wedge D \notin \mathbf{Ng}_A^{\leq k+1}(t_{prop})$. Since $END^{k+1}(A,D,t_{prop},t) \in tr$ then it can only be build from $(Sharek\_makestep)$, meaning that we must have all the hypothesis true, in particular:

- $recv_\alpha(A, \{\langle t_{prop}, B, \{D,\ldots\}\rangle\}_{sk_B}, t_1) \in tr$
- $recv_\alpha(A, \{\langle t_{prop}, C, \{D,\ldots\}\rangle\}_{sk_C}, t_2) \in tr$
- $END^k(A,B,t_{prop},t_x) \in tr$
- $END^i(A,C,t_{prop},t_y) \in tr, i \leq k$

We apply the induction hypothesis on the two last items. We obtain that $B \in \mathbf{Ng}_A^{\leq k}(t_{prop})$ and $C \in \mathbf{Ng}_A^{\leq i}(t_{prop}) \subseteq \mathbf{Ng}_A^{\leq k}(t_{prop})$ (and due to Property 8.1). By hypothesis we know that $D \notin \mathbf{Ng}_A^{\leq k+1}(t_{prop})$, which is equivalent to

$$\left(\nexists X \in V\ s.t.\ X \in \mathbf{Ng}_A^{\leq k}(t_{prop}) \wedge D \in \mathbf{Ng}_X^1(t_{prop}+t_X)\right).$$

We deduce that $D \notin \mathbf{Ng}_X^1(t_{prop}+t_X)$ for $X \in \{B,C\}$ with t_X the minimal time needed to forward a message from A to X at t_{prop}. Both messages contained in the receive events are signed by their pretended emitters, respectively B and C. We can then conclude that both B and C crafted their respective claims of neighborhood with D. But C and B claims in these messages that D is in their (1)-neighborhood. Since nodes cannot lie about their neighborhoods using rules in $\mathcal{P}$, this means that both their messages were built by $(Intrude\_forge)$, and as both messages are forged by their pretended emitter, we can conclude that both B and C are intruders. This is in contradiction with our initial hypothesis that the network consists of only one intruder. □

8.6 Conclusion

We have proposed a way of modeling physical properties in a wireless network in order to verify protocols in the presence of intruders, taking into account time and movement of nodes. We focused on the protocols which discover the distance between nodes, from a single hop (usually called neighborhood property) to an arbitrary number of them ((k)-neighborhood). After introducing the model, we have applied the model to two protocols and proved that one is correct under some assumptions about the network, and the intruder. Finally we propose the Sharek protocol, which securely discover (k)-neighbors based on the knowledge of the (1)-neighborhood, in presence of one intruder. We can generalize this protocol in order to be resistant to several intruders. We also provide a formal proof of the security of the Sharek protocol as a third example of an application using our formal model. This formal modelling of the (k)-neighborhood is the first step toward an automatic tool for verifying neighborhood discovery protocols.

Acknowledgements This work was supported by Projects TERRA and ARESA2.

References

1. Ross J. Anderson. *Security Engineering: A Guide to Building Dependable Distributed Systems*. John Wiley & Sons, Inc., New York, NY, USA, 2001.
2. Gildas Avoine, Muhammed Ali Bingöl, Süleyman Kardaş, Cédric Lauradoux, and Benjamin Martin. A framework for analyzing rfid distance bounding protocols. *J. Comput. Secur.*, 19:289–317, April 2011.
3. David Basin, Cas Cremers, and Catherine Meadows. *Model Checking Security Protocols*, chapter 24. Springer, 2011. To appear.
4. Prosenjit Bose, Pat Morin, Ivan Stojmenovi, and Jorge Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks*, 7:609–616, 2001. 10.1023/A:1012319418150.
5. S. Brands and D. Chaum. Distance-bounding protocols. In *Advances in Cryptology (EUROCRYPT'93)*, pages 344–359. Springer, 1994.
6. Srdjan Capkun and Jean-Pierre Hubaux. Secure positioning of wireless devices with application to sensor networks. In *IEEE INFOCOM*. IEEE Journal on Selected Areas in Communications: Special Issue on Security in Wireless Ad Hoc Networks, 2005.

7. Cas Cremers, Kasper Bonne Rasmussen, and Srdjan Capkun. Distance hijacking attacks on distance bounding protocols. Cryptology ePrint Archive, Report 2011/129, 2011. http://eprint.iacr.org/.
8. B. Das and V. Bharghavan. Routing in ad-hoc networks using minimum connected dominating sets. In *Communications, 1997. ICC 97 Montreal, 'Towards the Knowledge Millennium'. 1997 IEEE International Conference on*, volume 1, pages 376 –380 vol.1, jun 1997.
9. D. Dolev and A. Yao. On the security of public key protocols. *Information Theory, IEEE Transactions on*, 29(2):198–208, 1983.
10. D. Dolev and A.C. Yao. On the security of public key protocols. In *Proc. of the 22nd Symp. on Foundations of ComputerScience*, pages 350–357. IEEE Computer Society Press, 1981.
11. J. Du, E. Kranakis, O. Morales Ponce, and S. Rajsbaum. Neighbor discovery in a sensor network with directional antennae. In *Algosensors*, Saarbruecken, Germany, September 2011.
12. G.P. Hancke and M.G. Kuhn. An rfid distance bounding protocol. In *Security and Privacy for Emerging Areas in Communications Networks, 2005. SecureComm 2005. First International Conference on*, pages 67–73. IEEE, 2005.
13. C.A.R. Hoare. *Communicating Sequential Processes*. Prentice Hall, 1985.
14. Y. C. Hu, A. Perrig, and D. B. Johnson. Packet leashes: a defense against wormhole attacks in wireless networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, volume 3, pages 1976–1986 vol.3, 2003.
15. G. Lowe. Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In T. Margaria and B. Steffen, editors, *Tools and Algorithms for the Construction and Analysis of Systems (TACAS'96)*, volume 1055, pages 147–166. Springer-Verlag, Berlin Germany, march 1996. Also in Software Concepts and Tools, 17:93–102, 1996.
16. G. Lowe. Casper: A compiler for the analysis of security protocols. In *Proc. of 10th Computer Security Foundations Workshop (CSFW'97)*. IEEE Computer Society Press, 1997. Also in Journal of Computer Security, Volume 6, pages 53–84, 1998.
17. Catherine Meadows, Radha Poovendran, Dusko Pavlovic, LiWu Chang, and Paul Syverson. Distance bounding protocols: Authentication logic analysis and collusion attacks. In *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks, edited volume, Springer, Nov. 2006*, Nov 2006.
18. R. Needham and M. Schroeder. Using encryption for authentication in large networks of computers. *Communication of the ACM*, 21(12):993–999, 1978.
19. Tobias Nipkow, Lawrence C. Paulson, and Markus Wenzel. *Isabelle/HOL — A Proof Assistant for Higher-Order Logic*, volume 2283 of *LNCS*. Springer, 2002.
20. Panagiotis (Panos) Papadimitratos, Marcin Poturalski, Patrick Schaller, Pascal Lafourcade, David Basin, Srdjan Capkun, and Jean-Pierre Hubaux. Secure Neighborhood Discovery: A Fundamental Element for Mobile Ad Hoc Networking. *IEEE Communications Magazine*, 46(2), 2008.
21. L.C. Paulson. The inductive approach to verifying cryptographic protocols. *Journal of computer security*, 6(1–2):85–128, 1998.
22. Adrian Perrig, Ran Canetti, J. D. Tygar, and Dawn Song. The tesla broadcast authentication protocol. *RSA CryptoBytes*, 5, 2002.
23. M. Poturalski, P. Papadimitratos, and J.P. Hubaux. Secure neighbor discovery in wireless networks: formal investigation of possibility. In *Proceedings of the 2008 ACM symposium on Information, computer and communications security*, pages 189–200. ACM, 2008.
24. Marcin Poturalski, Panos Papadimitratos, and Jean-Pierre Hubaux. Secure Neighbor Discovery in Wireless Networks: Formal Investigation of Possiblity. Technical report, EPFL, 2007.
25. Marcin Poturalski, Panos Papadimitratos, and Jean-Pierre Hubaux. Towards Provable Secure Neighbor Discovery in Wireless Networks. In *The 6th ACM Workshop on Formal Methods in Security Engineering*, pages 31–42, Alexandria, VA, 2008. ACM.
26. P.Y.A. Ryan, S.A. Schneider, M.H. Goldsmith, G. Lowe, and A.W. Roscoe. *The modelling and analysis of security protocols: the CSP approach*. Addison-Wesley, 2000.

27. Patrick Schaller, Benedikt Schmidt, David Basin, and Srdjan Capkun. Modeling and verifying physical properties of security protocols for wireless networks. In *Proceedings of the IEEE Computer Security Foundations Symposium (CSF)*, pages 109–123. IEEE, 2009.
28. S.A. Schneider. Security properties and CSP. In *Proc. of the Symposium on Security and Privacy*, pages 174–187. IEEE Computer Society Press, 1996.
29. R. Shokri, M. Poturalski, G. Ravot, P. Papadimitratos, and J.P. Hubaux. A practical secure neighbor verification protocol for wireless sensor networks. In *Proceedings of the second ACM conference on Wireless network security*, pages 193–200. ACM, 2009.
30. F. Javier Thayer, Vipin Swarup, and Joshua D. Guttman. Metric strand spaces for locale authentication protocols. In Masakatsu Nishigaki, Audun Jøsang, Yuko Murayama, and Stephen Marsh, editors, *Trust Management IV - 4th IFIP WG 11.11 International Conference, IFIPTM 2010, Morioka, Japan, June 16–18, 2010. Proceedings*, volume 321 of *IFIP Conference Proceedings*, pages 79–94. Springer, 2010.
31. S. Vasudevan, J. Kurose, and D. Towsley. On neighbor discovery in wireless networks with directional antennas. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2502–2512. IEEE, 2005.
32. J. Wilson, V. Bhargava, A. Redfern, and P. Wright. A wireless sensor network and incident command interface for urban firefighting. In *Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, pages 1–7. IEEE, 2007.
33. Jie Wu and Hailan Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, DIALM ’99, pages 7–14, New York, NY, USA, 1999. ACM.

Chapter 9
A Tutorial on White-Box AES

James A. Muir

Abstract White-box cryptography concerns the design and analysis of implementations of cryptographic algorithms engineered to execute on untrusted platforms. Such implementations are said to operate in a *white-box attack context*. This is an attack model where all details of the implementation are completely visible to an attacker: not only do they see input and output, they see every intermediate computation that happens along the way. The goal of a white-box attacker when targeting an implementation of a cipher is typically to extract the cryptographic key; thus, white-box implementations have been designed to thwart this goal (i.e., to make key extraction difficult/infeasible). The academic study of white-box cryptography was initiated in 2002 in the seminal work of Chow et al. (White-box cryptography and an AES implementation. In: Selected areas in cryptography: 9th annual international workshop, SAC 2002. Lecture notes in computer science, vol 2595, pp 250–270, 2003). Here, we review the first white-box AES implementation proposed by Chow et al. and give detailed information on how to construct it. We provide a number of diagrams that summarize the flow of data through the various look-up tables in the implementation, which helps clarify the overall design. We then briefly review the impressive 2004 cryptanalysis by Billet et al. (Cryptanalysis of a white box AES implementation. In: Selected areas in cryptography: 11th international workshop, SAC 2004. Lecture notes in computer science, vol 3357, pp 227–240, 2005). The BGE attack can used to extract an AES key from Chow et al.'s original white-box AES implementation with a work factor of about 2^{30}, and this fact has motivated subsequent work on improved AES implementations.

J.A. Muir (✉)
Irdeto Canada, Ottawa, ON, Canada
e-mail: james.muir@irdeto.com

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_9,

9.1 Introduction

Suppose cryptographic software is deployed on a host that is not fully trusted. Examples of this include software distributed to end-users as part of some digital rights management (DRM) system, or client software running in the cloud, or even a cryptographic operation being executed on a smart-card [3,11,12]. Continuing with the DRM example, suppose that after installing the software on a PC, laptop, tablet or mobile phone, the end-user is then able to purchase some type of premium content (e.g., a television show, sports feed, video game or e-book). The content arrives at the user's device encrypted, and is decrypted by the software as it is viewed.

A malicious end-user may attempt to extract cryptographic keys from the software and then use them to redistribute content outside the DRM system. An attacker such as this is much more powerful than a traditional cryptographic attacker who sees only the inputs and outputs of a cryptographic operation (i.e., an attacker who treats the implementation as a black-box). This attacker is targeting *software* running on their own device. They are able to examine its inputs, outputs, and, with the help of a disassembler/debugger (e.g., IDA Pro, OllyDbg), the result of every intermediate computation it carries out. Essentially, this attacker has total visibility into the cryptographic operation.

The study of cryptographic implementations in this type of attack context was introduced in the academic literature in 2002 by Chow et al. [4]. In their seminal work, they motivated and defined the *white-box attack context* and presented some generic techniques that can be used to help create cryptographic implementations that resist key-extraction. They also applied those techniques to produce example implementations of AES [4] and (in another work) DES [5].

The terms *white-box AES* and *white-box DES* have become synonymous with the first implementations disclosed by Chow et al., but these terms are actually more general. Any AES implementation engineered to resist key extraction in the white-box attack context could be called white-box AES. And note that there are a number of ways that the techniques proposed by Chow et al. could be applied to AES and DES to create protected implementations.

With the 10 year anniversary of the papers by Chow et al. upcoming, it seems an appropriate time to give them another look. Here, we review their original AES implementation and give detailed information on how to construct it. A fair criticism of Chow et al.'s AES paper is that it is quite dense, and extracting the complete details of their protected AES implementation from it can be challenging. Our goal here is to make that information more accessible; this may be of particular benefit to new researchers and software engineers who are beginning to learn about white-box cryptography.

We also give a brief review of the 2004 algebraic cryptanalysis by Billet et al. [2] that shows how a white-box attacker can extract the key from Chow et al.'s original AES implementation using 2^{30} work-steps in the worst case. This impressive cryptanalysis has motivated the design of new white-box AES implementations more resistant to key extraction and a number of subsequent works in the open literature have appeared on this topic (e.g., [10,13,14,18]).

Outline. We begin by discussing the definitional results on program obfuscation by Barak et al. in Sect. 9.2. We then start our review of Chow et al.'s public AES implementation in Sect. 9.3 by describing a table-based implementation that does not include any protections against white-box attacks. In Sect. 9.4, we explain how encodings and mixing bijections are applied to the implementation with the goal of making it more resistant to key extraction attacks. Then we review the cryptanalysis by Billet et al. in Sect. 9.5, and end with some remarks in Sect. 9.6.

Preliminary Facts and Notation. Let x and y be bit-strings of equal length. We denote the bit-wise *exclusive-or* of x and y by $x \oplus y$. When we say that a transformation, L, from bit-strings to bit-strings is *linear*, we mean that the identity $L(x \oplus y) = L(x) \oplus L(y)$ holds for all inputs x, y. In particular, any transformation that permutes the bits of x is linear. If L is linear, then it can be represented using matrix-vector multiplication over GF(2); that is, there exists a matrix representation of L. The composition of two functions f and g is denoted by $f \circ g$, where $f \circ g(x) = f(g(x))$. If v is a column vector, then we use v^{T} to denote its transpose. We sometimes abuse functional notation and apply it to matrices; for example, if M and N are matrices that can be multiplied together, then $M \circ N$ denotes the transformation $v \mapsto MNv$. If c is a constant bit-string, then $\oplus_c$ denotes the function $x \mapsto x \oplus c$.

9.2 Barak et al.'s Impossibility Theorem

In 2001, Barak et al. [1] published foundational results on *program obfuscation*. They defined a *program obfuscator* as an algorithm that takes a program description as input (e.g., C code) and transforms it into a functionally equivalent obfuscated program description that satisfies the *virtual black-box property*; that is, any information that can be efficiently learned from the obfuscated program description can also be efficiently learned by studying only inputs and outputs of the original program. Their main result is that *generic* program obfuscators cannot exist – they show that there must always be some class of programs that when run through the obfuscator leak information that is not available through black-box interaction with the original programs.

The results of Barak et al. are sometimes incorrectly cited to refute the possibility of designing cryptographic implementations that resist white-box attacks.[1] However, there is no evidence that common block ciphers and their component operations belong to the special family of programs that cannot be securely obfuscated; see the statements to this effect by Billet et al. [2, p. 239] and by Wyseur [16, p. 91]. The successful white-box cryptanalysis of Chow et al.'s published AES [2] and DES [8,17] implementations do not point to any fundamental flaw that must be present in all white-box implementations of block ciphers; parts of

[1] More generally, the results are also cited incorrectly in anti-DRM commentaries. Barak has published a non-technical summary of their results in an attempt to dispel some of the confusion (see http://www.cs.princeton.edu/~boaz/Papers/obf_informal.html).

those attacks exploit details particular to the AES and DES algorithms. It is possible that some block ciphers can be securely obfuscated (in a strict definitional strict); and it is also possible that, with the introduction of new techniques, strong white-box implementations of AES and DES could be created.

Barak et al. suggest that the virtual black-box property used in their definition of secure obfuscation may be too strong (i.e., perhaps obfuscated programs necessarily leak some non-black-box information, which may or may not be useful to an attacker). Other definitions of secure obfuscation have been proposed, and using those definitions a number of positive results have been derived (cf. [9]).

9.3 Table-Based Implementation

One completely impractical way to create a white-box implementation of a block cipher that does not leak any more information than a black-box implementation is to create a massive look-up table that maps, say, plaintext to ciphertext under some fixed key. If the block length of the cipher is ℓ bits, then the look-up table consists of 2^{ℓ} entries with each entry being an ℓ-bit string. Since ℓ is typically 64 or 128, the amount of memory required to store this table is beyond the capabilities of any real-world device. However, using a number of smaller look-up tables can lead to practical solution.

We begin our description of Chow et al.'s white-box AES implementation by first presenting an implementation that does not offer any resistance to white-box attacks. This implementation makes extensive use of look-up tables, and the cipher key can be easily recovered from some of them. Techniques for resisting key extraction are covered in Sect. 9.4.

9.3.1 AES-128

AES-128 is specified in FIPS 197 [7]. It is an iterated block cipher that maps a 16-byte input to a 16-byte output using a 16-byte key. It has ten rounds. Each round updates a 16-byte state variable, which we treat as a one-dimensional array,[2] by applying a combination of four basic transformations:

- `AddRoundKey` takes a 16-byte round key, k_r, and uses exclusive-or to add it into the 16-byte state (i.e., $\texttt{state}[i] \leftarrow \texttt{state}[i] \oplus k_r[i]$ for $i = 0 \dots 15$).
- `SubBytes` utilizes a substitution table, S, that maps bytes to bytes. Each byte of the state is updated by applying S to it (i.e., $\texttt{state}[i] \leftarrow S(\texttt{state}[i])$ for $i = 0 \dots 15$).

[2] The state variable is usually described as a two-dimensional array of bytes (i.e., a 4×4 array). However, the four columns can be concatenated end-to-end to form a one-dimensional array. Using a one-dimensional array simplifies some of our notation and diagrams.

- `ShiftRows` rearranges the bytes of the state using the following permutation:

0 5 10 15	4 9 14 3	8 13 2 7	12 1 6 11

  that is, $\texttt{state}[0], \texttt{state}[5], \texttt{state}[10], \texttt{state}[15]$ form the first 4 bytes of the updated state, and so on.
- `MixColumns` updates the state 4 bytes at a time. An invertible 4×4 matrix, MC, with entries from GF(2^8), is multiplied by a 4×1 column vector formed from four state bytes. The state bytes are interpreted as elements of GF(2^8). More precisely, the transformation is

$$\begin{bmatrix} \texttt{state}[i] \\ \texttt{state}[i+1] \\ \texttt{state}[i+2] \\ \texttt{state}[i+3] \end{bmatrix} \leftarrow MC \cdot \begin{bmatrix} \texttt{state}[i] \\ \texttt{state}[i+1] \\ \texttt{state}[i+2] \\ \texttt{state}[i+3] \end{bmatrix}$$

  for $i = 0, 4, 8, 12$. The matrix MC is defined to be

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}.$$

Let k denote an AES-128 key. The AES specification explains how to expand k into 11 round keys $k_0, k_1, \ldots, k_{10}$ (one additional round key, k_0, is required for an initial `AddRoundKey` operation that takes place before round 1). We do not require the exact details of key expansion here; note, however, that k_0 is equal to k.

The conventional way to describe AES-128 encryption is as follows:

```
state ← plaintext
AddRoundKey(state, $k_0$)
for $r = 1 \ldots 9$
    SubBytes(state)
    ShiftRows(state)
    MixColumns(state)
    AddRoundKey(state, $k_r$)
SubBytes(state)
ShiftRows(state)
AddRoundKey(state, $k_{10}$)
ciphertext ← state
```

However, there are many other valid descriptions. Consider the following two observations:

1. The for-loop can be redefined to bring the transformation `AddRoundKey(state,` k_0`)` inside it while pushing `AddRoundKey(state,` k_9`)` out.

2. Since SubBytes applies the same S-box to each byte of the state, SubBytes followed by ShiftRows gives the same result as ShiftRows followed by SubBytes.

From these observations, we can generate the following description:

```
state ← $plaintext$
for $r = 1 \ldots 9$
    AddRoundKey(state, $k_{r-1}$)
    ShiftRows(state)
    SubBytes(state)
    MixColumns(state)
AddRoundKey(state, $k_9$)
ShiftRows(state)
SubBytes(state)
AddRoundKey(state, $k_{10}$)
$ciphertext$ ← state
```

Here is another observation:

3. Since ShiftRows is a linear transformation (recall that is a permutation), AddRoundKey(state, k_{r-1}) followed by ShiftRows(state) gives the same result as ShiftRows(state) followed by AddRoundKey(state, $\widehat{k}_{r-1}$); here, $\widehat{k}_{r-1}$ is the result of applying ShiftRows to the round key k_{r-1}.

This gives us

```
state ← $plaintext$
for $r = 1 \ldots 9$
    ShiftRows(state)
    AddRoundKey(state, $\widehat{k}_{r-1}$)
    SubBytes(state)
    MixColumns(state)
ShiftRows(state)
AddRoundKey(state, $\widehat{k}_9$)
SubBytes(state)
AddRoundKey(state, $k_{10}$)
$ciphertext$ ← state
```

With AES written in this way, we are able to combine AddRoundKey, SubBytes, and part of MixColumns into a series of table look-ups. This technique is similar to one used by Daemen and Rijmen in their AES proposal document [6, see Sect. 5.2.1]. However, in the implementation we are about to review below, we will see that bytes of round keys are embedded into some of the tables, and a number of redundant tables are included; this differs from the implementation by Daeman and Rijmen. Essentially, the for-loop above is unrolled and a collection of tables is created for each of the ten rounds with no regard as to whether or not an identical table might exist elsewhere in the implementation.

9.3.2 T-Boxes

In each round, the `AddRoundKey` and `SubBytes` transformations can be combined into a series of 16 look-up tables that map bytes to bytes (i.e., 8- to 8-bits). These so-called *T-boxes* are defined as follows:

$$\begin{aligned} T_i^r(x) &= S(x \oplus \widehat{k}_{r-1}[i]), && \text{for } i = 0\ldots15 \text{ and } r = 1\ldots9, \\ T_i^{10}(x) &= S(x \oplus \widehat{k}_9[i]) \oplus k_{10}[i], && \text{for } i = 0\ldots15. \end{aligned}$$

Note that the T-boxes for round 10 incorporate the bytes of two round keys ($\widehat{k}_9$ and k_{10}). There are 160 T-boxes in total.

9.3.3 Ty_i Tables

In rounds 1–9, after a byte is mapped through a T-box, it is then input into a `MixColumns` transformation. In particular, in round 1, the outputs of $T_0^1, T_1^1, T_2^1, T_3^1$ are interpreted as a column vector and then multiplied with the matrix MC. This computation can also be implemented using tables.

Let x_0, x_1, x_2, x_3 be 4 bytes that are to be multiplied with MC. The multiplication can be decomposed into an exclusive-or of four 32-bit values like so:

$$\begin{bmatrix} \texttt{02} & \texttt{03} & \texttt{01} & \texttt{01} \\ \texttt{01} & \texttt{02} & \texttt{03} & \texttt{01} \\ \texttt{01} & \texttt{01} & \texttt{02} & \texttt{03} \\ \texttt{03} & \texttt{01} & \texttt{01} & \texttt{02} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_0 \begin{bmatrix} \texttt{02} \\ \texttt{01} \\ \texttt{01} \\ \texttt{03} \end{bmatrix} \oplus x_1 \begin{bmatrix} \texttt{03} \\ \texttt{02} \\ \texttt{01} \\ \texttt{01} \end{bmatrix} \oplus x_2 \begin{bmatrix} \texttt{01} \\ \texttt{03} \\ \texttt{02} \\ \texttt{01} \end{bmatrix} \oplus x_3 \begin{bmatrix} \texttt{01} \\ \texttt{01} \\ \texttt{03} \\ \texttt{02} \end{bmatrix}.$$

The terms of the sum on the right (denote them by y_0, y_1, y_2, y_3) are each a function of 1 byte of input. Thus, each y_i can take on only 256 possible values.

The so-called *Ty_i tables* map 8- to 32-bits and are defined as follows:

$$\begin{aligned} Ty_0(x) &= x \cdot [\texttt{02 01 01 03}]^{\mathsf{T}} \\ Ty_1(x) &= x \cdot [\texttt{03 02 01 01}]^{\mathsf{T}} \\ Ty_2(x) &= x \cdot [\texttt{01 03 02 01}]^{\mathsf{T}} \\ Ty_3(x) &= x \cdot [\texttt{01 01 03 02}]^{\mathsf{T}}. \end{aligned}$$

Using these tables, we see that the 32-bits that result from applying `MixColumns` to the 4 bytes x_0, x_1, x_2, x_3 can be computed via four table look-ups and three exclusive-ors:

$$Ty_0(x_0) \oplus Ty_1(x_1) \oplus Ty_2(x_2) \oplus Ty_3(x_3).$$

We create 144 Ty_i tables (36 copies of each of Ty_0, Ty_1, Ty_2, Ty_3) to accept the outputs of the T-boxes in rounds 1–9 (recall that `MixColumns` is not applied in round 10).

9.3.4 XOR Tables

The exclusive-or operations, which combine 32-bit values from the Ty_i tables, can also be implemented using tables. Define a look-up table, XOR, that takes two nibbles (i.e., two 4-bit values) as input and maps them to their exclusive-or:

$$\mathrm{XOR}(x,y) = x \oplus y.$$

Note that XOR maps 8- to 4-bits. The exclusive-or of two 32-bit values can be computed using eight copies of the XOR look-up table.

In each of rounds 1–9, twelve 32-bit exclusive-ors are required to determine the result of `MixColumns`. To carry out this computation, we create 96 copies of the XOR table in each of these rounds (i.e., 864 copies of the XOR table in total). Although it seems that we could make do with only one XOR table, the protections introduced in Sect. 9.4 do not permit this.

9.3.5 Table Composition

Wherever a T-box feeds directly into a Ty_i table (i.e., in rounds 1–9), we can replace the two separate tables with their composition. For example, in round 1, T_0^1 and Ty_0 could be replaced with the new look-up table $Ty_0 \circ T_0^1$ where

$$Ty_0 \circ T_0^1(x) = Ty_0(T_0^1(x)).$$

Composing look-up tables reduces the number of individual table accesses required to carry out an encryption. Throughout rounds 1–9, the T-boxes and Ty_i tables are composed.

9.3.6 Summary

We now have all the tables (144 composed T-boxes/Ty_i tables, 864 XOR tables, 16 T-boxes) we need for our implementation, which can be summarized as follows:

```
state ← plaintext
for $r = 1 \dots 9$
    ShiftRows(state)
    TBoxesTyiTables(state)
    XORTables(state)
ShiftRows(state)
TBoxes(state, 10)
ciphertext ← state
```

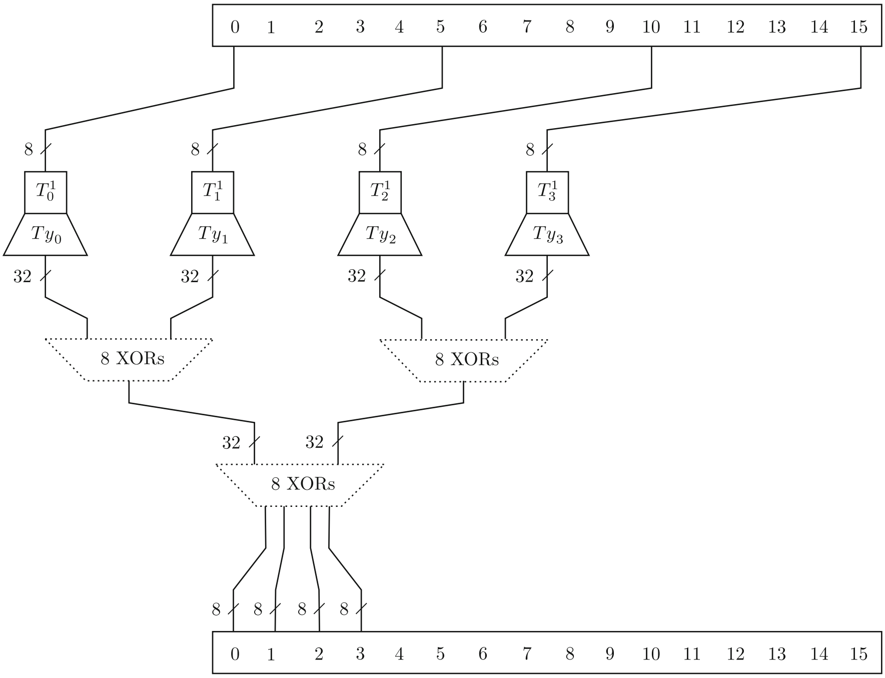


Fig. 9.1 The data flow for round 1 of AES with respect to bytes 0, 5, 10, 15 of the input state (i.e., the plaintext). The data flow for the other bytes is similar. Note that the input state is at the *top* of the diagram and the output state is at the *bottom*

The flow of 4 bytes of state through round 1 is illustrated in Fig. 9.1. Note that there are a number of different ways that the XOR tables could be utilized to determine the value of the state variable at the end of rounds 1–9; the flow in Fig. 9.1 is only an example. A zoomed in look at the XOR computation is given in Fig. 9.2.

9.4 Protected Implementation

We consider now how to protect the table-based implementation of the previous section in the white-box attack context. Recall that this means that the software implementing AES-128 encryption for a particular key executes in an environment that is under the control of an attacker. By using a disassembler/debugger, it is easy for the attacker to learn the contents of the various look-up tables, including the composed T-boxes/Ty_i tables that incorporate bytes of round keys.

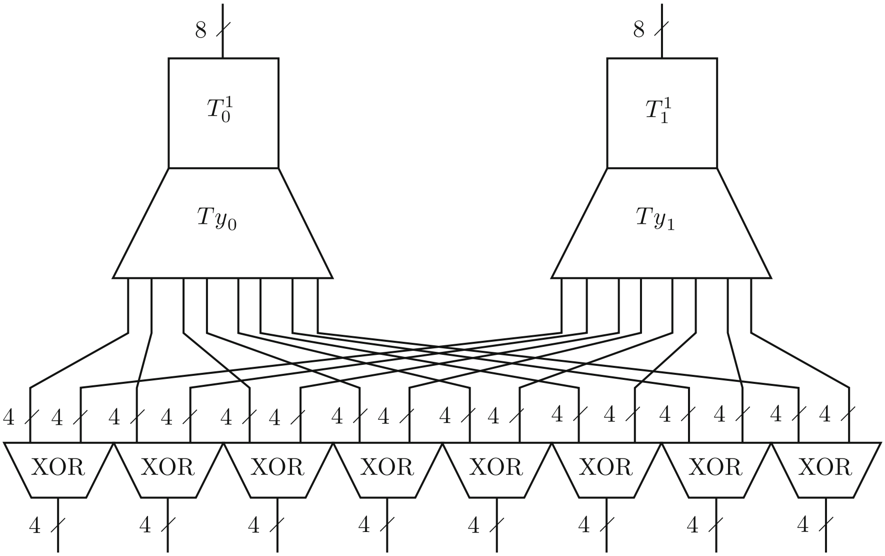


Fig. 9.2 Computing an exclusive-xor of two 32-bit values utilizes eight XOR tables. The inputs enter at the *top* of the diagram and the outputs appear at the *bottom*

9.4.1 Encodings

If the composed T-box/Ty_i tables from round 1 are known to an attacker, then they can easily recover the AES key. Consider table $Ty_0 \circ T_0^1$, and let a denote the byte of round key k_0 used to build T_0^1. There are only 256 different constructions of $Ty_0 \circ T_0^1$, and thus the attacker can enumerate them and simply look-up the value of a from that list.[3]

We must do something to protect the contents of the composed T-box/Ty_i tables and the T-boxes in round 10 if our implementation is going to resist key extraction. The technique proposed by Chow et al. [4] is to use *input and output encodings*.

An encoding is simply a bijection. To protect a table, T, we choose bijections f and g and form the new table T' where

$$T' = g \circ T \circ f^{-1}.$$

f is called the *input encoding* and g is called the *output encoding*. This new table maps encoded inputs to encoded outputs and can still be used to compute $T(x)$. To retrieve the value of $T(x)$, we map $f(x)$ through T' and then apply g^{-1} to the result.

If the output of table T feeds into another table R, then encodings are applied to those two tables in a so-called *networked* fashion; that is, the output encoding of

[3] The attacker can also compute the key byte directly: $a = S^{-1} \circ Ty_0^{-1} \circ (Ty_0 \circ T_0^1)(0)$.

T and the input encoding of R are chosen so that they cancel each other out. For example, T and R would be protected as follows,

$$T' = g \circ T \circ f^{-1} \quad \text{and} \quad R' = h \circ R \circ g^{-1},$$

from which we see that

$$R' \circ T' = (h \circ R \circ g^{-1}) \circ (g \circ T \circ f^{-1}) = h \circ (R \circ T) \circ f^{-1}.$$

Encodings are used to obfuscate the contents of all look-up tables in Chow et al.'s AES implementation. They are selected uniformly at random and independently wherever possible. Because of the design of the XOR tables, which consume 4 bits from one look-up table and 4 bits from another, almost all the encodings used in the protected implementation are *concatenated encodings*. These are bijections formed from smaller bijections. For example, we can build an 8-bit encoding, f, from two 4-bit encodings, f_0 and f_1, like so:

$$f(x_0 \| x_1) = f_0(x_0) \| f_1(x_1).$$

Here, the symbol $\|$ denotes the concatenation of bit-strings, and x_0, x_1 are 4-bit strings. Similarly, we can build a 32-bit encoding, g, using eight 4-bit encodings:

$$g(x_0 \| x_1 \| \cdots \| x_7) = g_0(x_0) \| g_1(x_1) \| \cdots \| g_7(x_7).$$

Concatenated 4-bit input and output encodings are individually selected and applied to all look-up tables, with the following exceptions:

- The output encodings applied to the XOR tables (these are just 4-bit encodings, not concatenated encodings),
- The input encodings applied to the composed T-box/Ty_i tables in round 1,
- The output encodings applied to the T-boxes in round 10.

The encodings mentioned in the latter two items do not have to be networked with XOR tables, so we have more freedom in their selection. We will discuss this further when we consider external encodings in Sect. 9.4.3.

Local security. When the composed T-box/Ty_i tables are protected with encodings, there are now too many table constructions for an attacker to enumerate. Consider a composed T-box/Ty_i table from round 2. There are $(16!)^2 \cdot (16!)^8$ ways of choosing input and output encodings for this table. If the input encoding is fixed, then it can be shown that all of the $(16!)^8$ possible output encodings produce distinct look-up tables. Thus, the number of table constructions is at least $(16!)^8 \approx 2^{354}$ (and is at most $(16!)^{10} \approx 2^{442}$).

An attacker might hope to deduce the key byte from an encoded T-box/Ty_i table by studying the lists of table constructions (i.e., the table constructions for each possible value of the key byte). However, this approach will not yield any information about the key byte. It can be shown, by manipulating input encodings, that all 256 lists of table constructions are the same.

From the previous fact, we can conclude that the protected tables are information theoretically secure. It is not possible for the attacker to extract the key byte from the encoded version of, say, $Ty_0 \circ T_0^2$, if he or she studies only that table. Chow et al. [5] refer to this property as *local security*. However, even though no information about the key leaks from this protected table, other information may. For example, the definition of the output encodings may leak, which could be useful for extracting key bytes from T-box/Ty_i tables in the next round.

Although the use of encodings is actually what motivates the table-based implementation of AES in the first place, encodings are the very last form of protection applied. Note that encodings are selected uniformly at random and will be non-linear with very high probability.

9.4.2 Mixing Bijections

The look-up tables that incorporate bytes of round keys can be considered miniature block ciphers. The application of concatenated input and output encodings help these components achieve *confusion*, as defined by Shannon [15]. To help them achieve *diffusion*, linear transformations are also composed at their input and output (these compositions are done before the application of the non-linear concatenated input and output encodings). An invertible linear transformation is referred to as a *mixing bijection*.

Mixing bijections are applied to all the key-dependent look-up tables in the implementation. In general, each mixing bijection is selected uniformly at random. Their usage in each of the interior rounds (i.e., rounds 2–9) is the same; for the exterior rounds (i.e., round 1 and 10), there are a few differences, which we will explain.

We begin by selecting all the required mixing bijections:

- For each of rounds 2–10, select sixteen 8- to 8-bit mixing bijections (i.e., 144 mixing bijections in total). These will be composed at the input of each T-box in rounds 2–10.
- For each of rounds 1–9, select four 32- to 32-bit mixing bijections (i.e., one mixing bijection for each of the four matrix multiplication steps in each of those rounds). These will be composed at the output of each Ty_i table in rounds 1–9.

Note that mixing bijections can be selected uniformly at random by constructing invertible matrices over GF(2). Now consider, for example, the first four key-dependent look-up tables in round 2:

$$Ty_0 \circ T_0^2,$$

$$Ty_1 \circ T_1^2,$$

$$Ty_2 \circ T_2^2,$$

$$Ty_3 \circ T_3^2.$$

Let $L_0^2, L_1^2, L_2^2, L_3^2$ be the four 8- to 8-bit mixing bijections selected for these tables. The inverses of these transformations are composed at their input. Let *MB* be the 32- to 32-bit mixing bijection chosen for these four tables. *MB* is composed at the output of each table. This produces the following tables:

$$MB \circ Ty_0 \circ T_0^2 \circ L_0^{2^{-1}},$$
$$MB \circ Ty_1 \circ T_1^2 \circ L_1^{2^{-1}},$$
$$MB \circ Ty_2 \circ T_2^2 \circ L_2^{2^{-1}},$$
$$MB \circ Ty_3 \circ T_3^2 \circ L_3^{2^{-1}}.$$

The 4 bytes entering these tables are computed in round 1 and will have the appropriate mixing bijections applied to them there (i.e., the four inputs will have the form $L_0^2(x_0), L_1^2(x_1), L_2^2(x_2), L_3^2(x_3)$).

The outputs of these tables feed into the XOR tables, as in Fig. 9.1. However, at the end of the third stage of XOR tables, the resulting 32-bit value is now

$$MB \circ MC \left[z_0\ z_1\ z_2\ z_3\right]^{\mathsf{T}}.$$

We need to remove the transformation *MB*, and also apply the 8-bit mixing bijections required for the next round.

Four new 8- to 32-bit tables are introduced to remove the effect of *MB*. These tables are generated using the familiar technique of decomposing a matrix multiplication into an exclusive-xor of four 32-bit vectors:

$$MB^{-1}\begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \end{bmatrix} = MB^{-1}\begin{bmatrix} z_0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \oplus MB^{-1}\begin{bmatrix} 0 \\ z_1 \\ 0 \\ 0 \end{bmatrix} \oplus MB^{-1}\begin{bmatrix} 0 \\ 0 \\ z_2 \\ 0 \end{bmatrix} \oplus MB^{-1}\begin{bmatrix} 0 \\ 0 \\ 0 \\ z_3 \end{bmatrix}.$$

Let $MB_0^{-1}, MB_1^{-1}, MB_2^{-1}, MB_3^{-1}$ denote the four 8- to 32-bit tables corresponding to the terms of the sum on the right. Let L^3 denote a 32- to 32-bit mixing bijection constructed by concatenating four 8- to 8-bit mixing bijections from round 3. L^3 is used to put the proper encodings on bytes 0, 1, 2, 3 of the state array that enters round 3. Accounting for the `ShiftRows` transformation at the beginning of round 3, L^3 is defined as follows:

$$L^3 = L_0^3 || L_{13}^3 || L_{10}^3 || L_7^3.$$

L^3 is composed at the output of each of $MB_0^{-1}, MB_1^{-1}, MB_2^{-1}, MB_3^{-1}$ resulting in the tables

$$L^3 \circ MB_0^{-1},$$
$$L^3 \circ MB_1^{-1},$$
$$L^3 \circ MB_2^{-1},$$
$$L^3 \circ MB_3^{-1}.$$

The outputs of these tables are combined using three new stages of XOR tables.

A summary of this process is presented Fig. 9.3. In comparison with Fig. 9.1, we see that the number of tables has doubled.

The application of mixing bijections in round 1 is very similar to Fig. 9.3. The only difference is that there are no input mixing bijections applied to the T-boxes. In round 10, there are input mixing bijections applied to the T-boxes, but output mixing bijections are not applied. The reason for these differences is related to the external encodings, which we discuss next.

9.4.3 External Encodings

One question that often arises when considering the white-box attack context is this: why would an attacker want to extract the cipher key when they already have software that will decrypt ciphertext for them? The answer, given by Chow et al., is to design the implementation so that it does not map raw ciphertext to raw plaintext, but rather *encoded ciphertext* to *encoded plaintext*. Encodings that affect the input and output of the cipher are referred to as *external encodings*.

Denote an AES-128 decryption by D_k. Select 128- to 128-bit bijections F and G. Chow et al. recommend that the implementation computes

$$D'_k = G \circ D_k \circ F^{-1}.$$

As with the encodings discussed in Sect. 9.4.1, a ciphertext, x, must be encoded with F, which gives $F(x)$, before it is passed as input to the implementation. In our DRM example, this external encoding could be applied on the server that supplies premium content for downloading (i.e., on a host separate from the one where the client software runs). After the transformation D'_k is applied by the user's software, the result is $G(D_k(x))$. The remaining encoding could be removed by the content viewer, perhaps as the content is played.

Use of external encodings is mainly to ensure that there is no point during the execution of the client software where raw ciphertext and raw plaintext appear. Chow et al. suggest that the external encodings be 128- to 128-bit mixing bijections. Using external encodings such as these requires that a number of new tables be added to the protected implementation: sixteen 8- to 128-bit tables along with supporting XOR tables (480) to compute the matrix multiplication for F^{-1} prior to round 1, and similarly for the matrix multiplication for G after round 10 (note that the 8- to 128-bit tables for G can be composed with the round 10 T-boxes).

Here, to simplify our exposition, we will not use mixing bijections for the external encodings. Instead, we will use concatenated 8-bit encodings; that is,

$$F = F_0 \| F_1 \| \cdots \| F_{15} \quad \text{and} \quad G = G_0 \| G_1 \| \cdots \| G_{15},$$

where each F_i and G_i is an 8- to 8-bit bijection (selected uniformly at random). For each T-box in round 1, the corresponding F_i^{-1} is composed at its input.

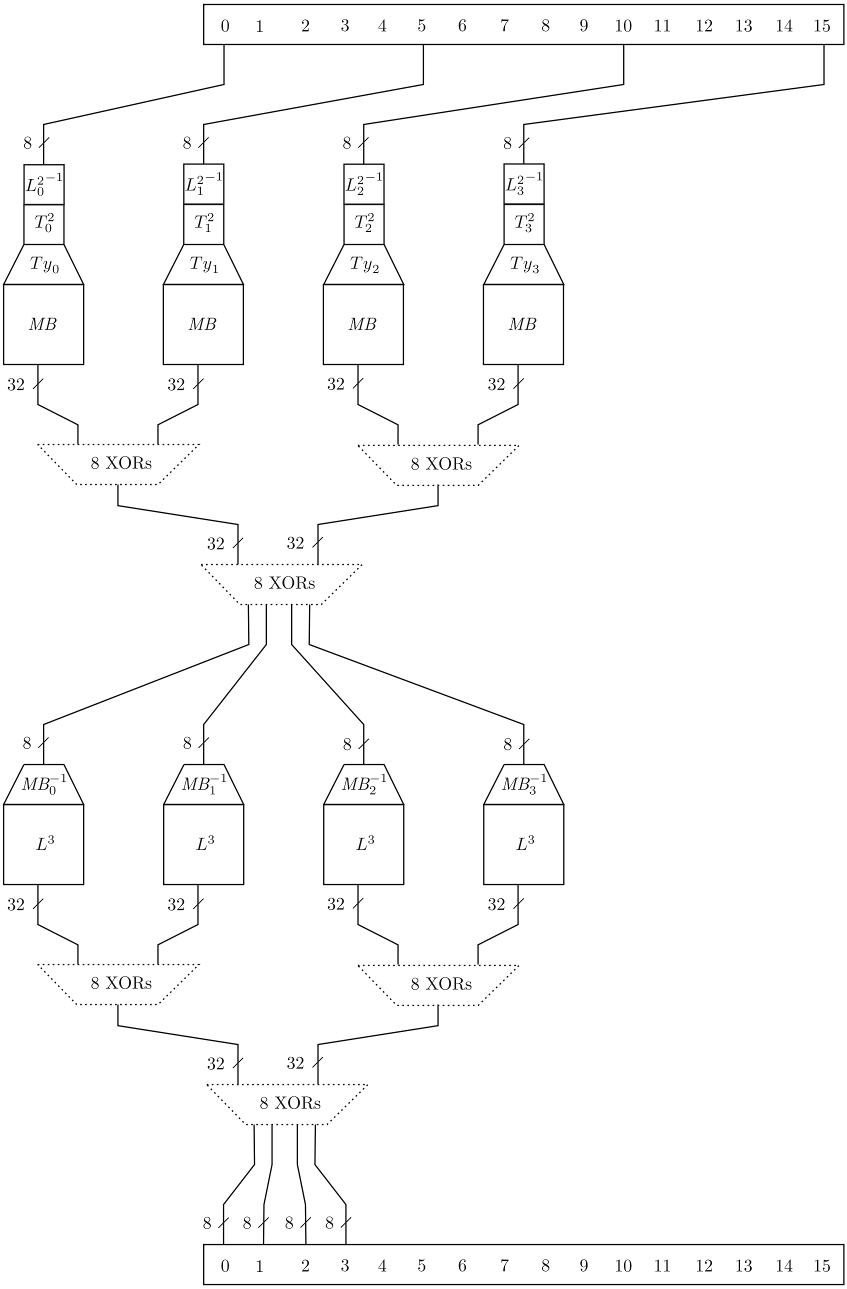


Fig. 9.3 The application of mixing bijections in round 2. The picture for rounds 3–9 is the same. For round 1, the only difference is the absence of the input mixing bijections on the T-boxes. For round 10, mixing bijections are applied at the input of the T-boxes, but not the output. At the bottom of the diagram, note that bytes 0, 1, 2, 3 of the state array that enters round 3 feed T-boxes $T_0^3, T_{13}^3, T_{10}^3, T_7^3$, respectively

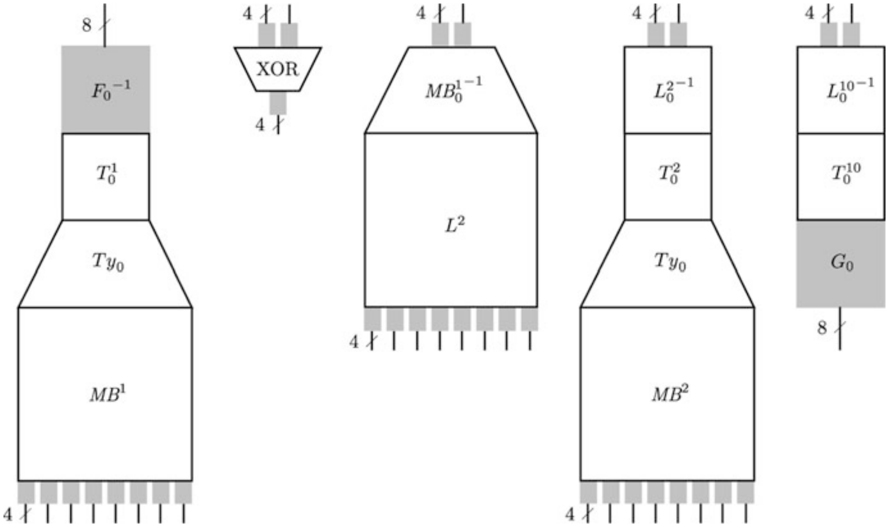


Fig. 9.4 Representatives from the five classes of protected look-up tables. *Grey boxes* are used to denote input and output encodings (*big grey boxes* are external encodings). For each representative (going from *left* to *right*), we list the rounds where tables like it can be found: round 1 only, rounds 1–9, rounds 1–9, rounds 2–9, round 10 only

And for each T-box in round 10, the corresponding G_i is composed at its output. External encodings of this type do not require the addition of any new tables to the implementation.

9.4.4 Summary

To protect the tabled-based AES implementation from Sect. 9.3, we apply mixing bijections, then (internal) encodings, and then external encodings. After the application of mixing bijections, the number of look-up tables in rounds 1–9 doubles. The table counts are as follows: The total storage requirement for the tables is 508 KB.

288	8- to 32-bit tables (1,024 bytes each),
1,728	8- to 4-bit tables (128 bytes each),
16	8- to 8-bit tables (256 bytes each).

A summary of the various tables, with all protections applied to them, is given in Fig. 9.4.

9.5 Cryptanalysis

Chow et al. [4] present some interesting attacks on weakened variants of their protected implementation, which justify some of their design choices. In particular, they show that if external encodings are not used, then a linear relation amongst round key bytes can be found that makes a key search feasible (the search space is reduced from 2^{128} to 2^{32}). They also show that if mixing bijections are not applied (recall that their absence roughly halves the number of look-up tables), then it is possible to deduce the output encodings for any key-dependent look-up table in rounds 1–9. Knowledge of those output encodings in, say, round 1 leads to discovery of the input encodings in round 2 because the encodings on the XOR tables at the end of round 1 can be deduced. Now, with knowledge of the input and output encodings in round 2, round key bytes can be easily extracted.

In 2004, Billet et al. [2] published an algebraic attack against Chow et al.'s first AES implementation. They showed that the cipher key can be extracted using at most 2^{30} work-steps and negligible memory. We give a brief review of their method here.

9.5.1 *The BGE Attack*

As is illustrated in Fig. 9.3, an AES round can be interpreted as the parallel application of four 32- to 32-bit transformations to the state array. Although the mixing bijection *MB* is present in the protected implementation, it has no influence on the 4 bytes output at the bottom of Fig. 9.3. The effect of *MB* and any other internal encodings are canceled out (this is by design), and the 32- to 32-bit transformation has the form displayed in Fig. 9.5.

Figure 9.5 introduces the notation used by Billet et al. when they examine one of the 32- to 32-bit transforms. The P_i's are the combination of input encodings and mixing bijections; the Q_i's are the combination of mixing bijections and output encodings. Let x_0, x_1, x_2, x_3 denote four input bytes and let y_0, y_1, y_2, y_3 denote the resulting output. From the definition of the matrix *MC*, the relation between the inputs and outputs can be summarized like so:

$$y_0 = Q_0\big(\mathtt{02}\cdot T_0'(x_0) \oplus \mathtt{03}\cdot T_1'(x_1) \oplus \mathtt{01}\cdot T_2'(x_2) \oplus \mathtt{01}\cdot T_3'(x_3)\big), \qquad (9.1)$$

$$y_1 = Q_1\left(\mathtt{01}\cdot T_0'(x_0) \oplus \mathtt{02}\cdot T_1'(x_1) \oplus \mathtt{03}\cdot T_2'(x_2) \oplus \mathtt{01}\cdot T_3'(x_3)\right), \qquad (9.2)$$

$$y_2 = Q_2\left(\mathtt{01}\cdot T_0'(x_0) \oplus \mathtt{01}\cdot T_1'(x_1) \oplus \mathtt{02}\cdot T_2'(x_2) \oplus \mathtt{03}\cdot T_3'(x_3)\right), \qquad (9.3)$$

$$y_3 = Q_3\left(\mathtt{03}\cdot T_0'(x_0) \oplus \mathtt{01}\cdot T_1'(x_1) \oplus \mathtt{01}\cdot T_2'(x_2) \oplus \mathtt{02}\cdot T_3'(x_3)\right); \qquad (9.4)$$

here, T_i' is a shorthand for $T_i \circ P_i$. Note that each y_i is a function of x_0, x_1, x_2, x_3.

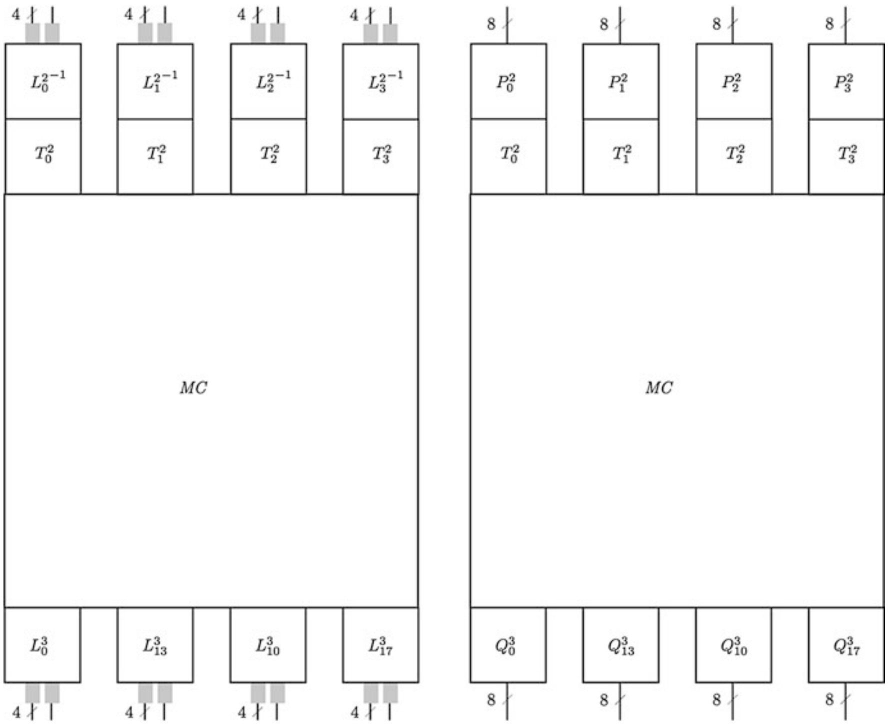


Fig. 9.5 One of the 32- to 32-bit transforms applied in round 2. *MC* is the `MixColumns` matrix. The *left diagram* matches the notation used in the previous figures. The *right diagram* introduces the equivalent notation used by Billet et al. [2] where the mixing bijections and concatenated encodings are combined. Note that the inverses of the output encodings become input encodings in the subsequent round

Billet et al. found that information about the output encodings (i.e., the Q_i's) leak from the four identities above. For each Q_i, they showed that it is possible to build an approximation, $\widetilde{Q}_i$, that differs from it by an unknown affine transformation; that is,

$$\widetilde{Q}_i = Q_i \circ A_i,$$

where A_i consists of an invertible linear transformation followed by an exclusive-or with a constant.

The approximations are built by analyzing a new set of look-up tables derived from Fig. 9.5. As noted previously, y_0 is a function of x_0, x_1, x_2, x_3; that is, $y_0 = f(x_0, x_1, x_2, x_3)$. However, if x_2, x_3 are kept constant, then y_0 can be considered a function of only x_0 and x_1. Fix x_2, x_3 to $\texttt{00}, \texttt{00}$ and let $f_{x_1}(x_0)$ denote the function $f(x_0, x_1, \texttt{00}, \texttt{00})$. Build the look-up table for $y_0 = f_{\texttt{00}}(x_0)$ and for $y_0 = f_{\texttt{01}}(x_0)$. From Eq. (9.1), we see that

$$f_{00}(x_0) = Q_0(02 \cdot T_0'(x_0) \oplus \beta_{00}),$$

$$f_{01}(x_0) = Q_0(02 \cdot T_0'(x_0) \oplus \beta_{01}),$$

where β_{00} and β_{01} are unknown 8-bit strings.

From the look-up tables for f_{00} and f_{01}, we can construct the look-up table for $f_{01} \circ {f_{00}}^{-1}$, which has a very simple description. Using functional notation, we can write

$$f_{00} = Q_0 \circ \oplus_{\beta_{00}} \circ 02 \cdot T_0',$$

$$f_{01} = Q_0 \circ \oplus_{\beta_{01}} \circ 02 \cdot T_0'.$$

Thus, we see that

$$\begin{aligned} f_{01} \circ {f_{00}}^{-1} &= (Q_0 \circ \oplus_{\beta_{01}} \circ 02 \cdot T_0') \circ ((02 \cdot T_0')^{-1} \circ \oplus_{\beta_{00}} \circ {Q_0}^{-1}) \\ &= Q_0 \circ \oplus_{\beta} \circ {Q_0}^{-1}, \end{aligned}$$

where $\beta = \beta_{01} \oplus \beta_{00}$.

There are exactly 256 bijections of the form $Q_0 \circ \oplus_\delta \circ {Q_0}^{-1}$ where δ is an 8-bit string. This set of bijections forms a commutative group under composition, denoted by $(G, \circ)$. All the group elements (i.e., look-up tables) are generated by computing the following compositions

$$f_{00} \circ {f_{00}}^{-1}, f_{01} \circ {f_{00}}^{-1}, \ldots, f_{\mathtt{ff}} \circ {f_{00}}^{-1}.$$

It is not difficult to find eight group elements, $g_1, g_2, \ldots, g_8$, such that a subset of them can be composed to generate any group element. These elements act like a vector-space basis for $(G, \circ)$ and can be used to build an isomorphism $\psi : (G, \circ) \to (\mathrm{GF}(2)^8, \oplus)$. Without going into further detail (see [2, Theorem 1]), it is this isomorphism that is used to construct the approximation to Q_0: for any $g \in G$, we set

$$\widetilde{Q_0}(\psi(g)) = g(00).$$

Thus, $\widetilde{Q_0}$ is constructed as a look-up table. An analogous process builds approximation for Q_1, Q_2, Q_3.

Returning to Fig. 9.5, with the approximations at hand, the output encodings in the diagram can be simplified. The simplification is done by composing new output encodings with the existing ones. After each Q_i, the bijection ${\widetilde{Q_i}}^{-1}$ is applied. These two bijections compose to give

$${\widetilde{Q_i}}^{-1} \circ Q_i = {A_i}^{-1} \circ {Q_i}^{-1} \circ Q_i = {A_i}^{-1}.$$

The inverse of an affine bijection is an affine bijection, and so the new output encodings are much simpler than the original (very likely non-linear) ones. And since the output encodings in a given round correspond to input encodings in the subsequent round, the output encoding approximations also lead to input encoding approximations. Thus, the input encodings in Fig. 9.5 can also be simplified.

From Billet et al.'s approximations, we can continue under the assumption that the P_i's and Q_i's in Fig. 9.5 are affine bijections. Each Q_i has the form $M_i(x) \oplus q_i$, where M_i is a linear bijection and q_i is an 8-bit string. By setting x_1, x_2, x_3 all to 00 in Eq. (9.1)–(9.4), an explicit linear relation between M_0 and each of M_1, M_2, M_3 can be derived. Thus, if M_0 is recovered, then so too will M_1, M_2, M_3.

The next step in the attack recovers M_0 and q_0 (see the original paper for details). Thus, the value of the output encodings can be determined completely. With complete knowledge of Q_0, Q_1, Q_2, Q_3, then from Fig. 9.5 we see that it is possible to compute the outputs of the T-boxes. From the complete knowledge of the output encodings in the previous round, we also learn P_0, P_1, P_2, P_3 completely. Now, it is easy to extract the key bytes, as discussed in Sect. 9.4.1.

The time complexity of Billet et al.'s key extraction attack is dominated by the work required to build the approximation to each output encoding. They estimate this to be 2^{24} work-steps. Thus, computing approximations for an entire AES round is $16 \cdot 2^{24} = 2^{28}$ work-steps. They recommend computing approximations for three consecutive AES rounds ($3 \cdot 2^{28} < 2^{30}$ work-steps), which leads to the recovery of two complete round keys, so that any ambiguity in the order of the key bytes recovered from the tables can be eliminated.

9.6 Remarks

White-box cryptography was introduced in the academic literature by Chow, Eisen, Johnson and van Oorschot 10 years ago and is still a relatively new area of research, with plenty of real-world applications and room for new contributions. For those interested in working in this area, a good understanding of the original white-box AES implementation [4] and the BGE attack [2] are essential, and hopefully this tutorial can help provide that. Although the BGE attack permits the key to be extracted from Chow et al.'s original white-box AES implementation, the attack has served mainly as motivation for work on stronger white-box implementations, and this line of research has been particularly active in the last few years (e.g., [13, 18]).

Acknowledgements The author thanks Phil Eisen who, over a number of conversations and presentations at Irdeto, motivated the style of exposition on AES in Sect. 9.3. Thanks are also extended to Michael Wiener who provided valuable comments on a preliminary draft of this work (especially with regards to the local security of the composed T-box/Ty_i tables). Also, conversations on white-box cryptography with Jeremy Clark, Alfred Menezes and Anil Somayaji were helpful in directing some of our commentary. Thanks also go to Elif Bilge Kavun who pointed out a notational error in a previous version of Sect. 9.4.2.

References

1. B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S. Vadhan, and K. Yang. On the (Im)possibility of Obfuscating Programs (Extended Abstract). In "Advances in Cryptology – CRYPTO 2001: 21st Annual International Cryptology Conference", *Lecture Notes in Computer Science* **2139** (2001), 1–18. Full version available from http://eccc.hpi-web.de/report/2001/057/.
2. O. Billet, H. Gilbert, and C. Ech-Chatbi. Cryptanalysis of a White Box AES Implementation. In "Selected Areas in Cryptography: 11th International Workshop, SAC 2004", *Lecture Notes in Computer Science* **3357** (2005), 227–240.
3. D. Boneh, R. DeMillo, and R. Lipton. On the importance of checking cryptographic protocols for faults. *Journal of Cryptology* **14** (2001), 101–119.
4. S. Chow, P. Eisen, H. Johnson, and P.C. van Oorschot. White-Box Cryptography and an AES Implementation. In "Selected Areas in Cryptography: 9th Annual International Workshop, SAC 2002", *Lecture Notes in Computer Science* **2595** (2003), 250–270.
5. S. Chow, P. Eisen, H. Johnson, and P.C. van Oorschot. A White-box DES Implementation for DRM Applications. In "Digital Rights Management: ACM CCS-9 Workshop, DRM 2002", *Lecture Notes in Computer Science* **2696** (2003), 1–15.
6. J. Daemen and V. Rijmen. AES submission document on Rijndael, Version 2, September 1999. Available from http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf
7. FIPS 197. *Advanced Encryption Standard*. Federal Information Processing Standards Publication 197, U.S. Department Of Commerce / National Institute of Standards and Technology, 2001. Available from http://www.csrc.nist.gov/publications/fips/
8. L. Goubin, J.-M. Masereel, and M. Quisquater. Cryptanalysis of White-Box DES Implementations. In "Selected Areas in Cryptography: 14th International Workshop, SAC 2007", *Lecture Notes in Computer Science* **4876** (2007), 278–295.
9. S. Hohenberger, G. Rothblum, A. Shelat, and V. Vaikuntanathan. Securely Obfuscating Re-Encryption. In "Theory of Cryptography: 4th Theory of Cryptography Conference, TCC 2007", *Lecture Notes in Computer Science* **4392** (2007), 233–252.
10. M. Karroumi. Protecting White-Box AES with Dual Ciphers. In "Information Security and Cryptology – ICISC 2010", *Lecture Notes in Computer Science* **6829** (2010), 278–291.
11. P. Kocher. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In "Advances in Cryptology – CRYPTO '96", *Lecture Notes in Computer Science* **1109** (1996), 104–113.
12. P. Kocher, J. Jaffe, and B. Jun. Differential Power Analysis. In "Advances in Cryptology – CRYPTO '99", *Lecture Notes in Computer Science* **1666** (1999), 388–397.
13. W. Michiels and P. Gorissen. "Cryptographic Method for a White-Box Implementation". U.S. Patent Application 2010/0080395 A1, filed November 9, 2007.
14. W. Michiels and P. Gorissen. "Cryptographic System". U.S. Patent Application 2011/0116625 A1, filed March 2, 2009.
15. C. E. Shannon. Communication Theory of Secrecy Systems. *Bell System Technical Journal* **28** (1949), 656–715.
16. B. Wyseur. "White-Box Cryptography", PhD thesis, Katholieke Universiteit Leuven, 2009.
17. B. Wyseur, W. Michiels, P. Gorissen, and B. Preneel. Cryptanalysis of White-Box DES Implementations with Arbitrary External Encodings. In "Selected Areas in Cryptography: 14th International Workshop, SAC 2007", *Lecture Notes in Computer Science* **4876** (2007), 264–277.
18. Y. Xiao and X. Lai. A Secure Implementation of White-Box AES. In "2009 2nd International Conference on Computer Science and its Applications: CSA 2009", IEEE (2009), 6 pages.

Chapter 10
Efficient 1-Round Almost-Perfect Secure Message Transmission Protocols with Flexible Connectivity

Reihaneh Safavi-Naini and Mohammed Ashraful Alam Tuhin

Abstract In the **S**ecure **M**essage **T**ransmission (**SMT**) problem, a sender $\mathcal{S}$ wants to send a message m to a receiver $\mathcal{R}$ in a *private* and *reliable* way. $\mathcal{S}$ and $\mathcal{R}$ are connected by n *wires*, t of which controlled by the adversary. The n wires represent n node disjoint communication paths between the sender and the receiver. The adversary is assumed to have *unlimited computational power*. An Almost Perfectly Secure Message Transmission (**APSMT**, for short) provides perfect privacy for the transmitted message, and the probability that the received message is different from the sent one is bounded by δ and, $\delta = 0$ corresponds to perfect SMT. It has been shown that APSMT is possible if $n \geq 2t+1$ and for 1-round perfect SMT, $n \geq 3t+1$. SMT protocols and techniques have found applications in practice, including key distribution and key strengthening in wireless sensor networks.

In this paper we show two general methods of constructing 1-round APSMT protocols for different levels of network connectivity. We consider two cases: $n = (2+c)t, c > \frac{1}{t}$ where a fraction of wires are corrupted, and $n = 2t+k, k \geq 1$ where a constant number of extra wires (over the required minimum) exists. The proposed methods use the whole, or part of, the previously constructed protocols to construct new protocols with flexible connectivity, whose privacy, reliability and efficiency can be derived from the component parts. The new protocols are efficient and in some cases have optimal transmission rates. The flexibility that is provided by these constructions facilitate application of APSMT in practical applications.

R. Safavi-Naini (✉) • M.A.A. Tuhin
Department of Computer Science, University of Calgary, Calgary, AB, Canada
e-mail: rei@ucalgary.ca; maatuhin@ucalgary.ca

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_10,

10.1 Introduction

The **P**erfectly **S**ecure **M**essage **T**ransmission (**PSMT**) problem was introduced by Dolev et al. [9] to address the problem of secure communication between two nodes in an incomplete network. In the PSMT problem, the sender $\mathcal{S}$ and the receiver $\mathcal{R}$ do not share a key but are connected by n 'wires' where *at most t* of which are controlled by an adversary $\mathcal{A}$, having unlimited computational power. Wires are abstractions of node-disjoint paths between $\mathcal{S}$ and $\mathcal{R}$. Security means $\mathcal{R}$ receives the message m sent by $\mathcal{S}$ in a *private* and *reliable* way. 'Private' means that the adversary does not learn any information about m and 'reliable' means that $\mathcal{R}$ receives the same message m that $\mathcal{S}$ has sent.

The initial motivation of this model has been to reduce connectivity requirements in secure multi-party protocols [4,6,20] and allow secure links between parties to be simulated in incomplete networks. Multiparty computation (MPC) allows a group of users, each holding a private input, to participate in a computation that depends on their input, and obtain the output of the computation, without revealing their inputs. MPC protocols, in information theoretic setting (i.e. when the adversary has unlimited computational power), require secure and reliable links between every two nodes. That is, the secure network connecting the users forms a complete graph. This assumption is very strong in practice. SMT protocols are proposed to simulate secure links between nodes using multiplicity of routes in the networks: that is, as long as there are sufficient number of paths in the network, the nodes can establish direct secure communication among them.

In recent years, however, algorithms and techniques that are developed in the study of SMT, and in particular one round protocols, have found other applications including key distribution and key strengthening in sensor networks [7, 26, 27]. In sensor networks, sensor nodes are employed in large geographical area and cryptographic keys are installed in the sensor nodes randomly before the nodes are deployed [13]. Key predistribution schemes in sensor networks do not always ensure that each pair of nodes share a predistributed key. Due to the limited communication range of the sensor nodes, two nodes, who do not share a key, can establish a path key using a secure multi-hop path between them. Different authors have considered the multi-path key establishment schemes in sensor networks, but most of them only considered a passive adversary [7, 11, 16, 28]. But in application like battle field surveillance, sensor nodes deployed in a hostile environment can be captured and compromised by an adversary with unlimited computational power. For this kind of applications, SMT protocols can be used as pointed out in [26, 27]. The 1-round (0, δ)-SMT protocol of [12] was employed for sensor networks in [26], whereas [27] used the 1-round PSMT protocol of [14].

It was shown by Dolev et al. that 1-round protocol with perfect privacy and perfect reliability requires $n \geq 3t+1$ [9]. Franklin and Wright relaxed the original security requirements of PSMT and defined (ε, δ)-SMT $(0 \leq \varepsilon, \delta \leq 1)$ [15] where the loss of privacy and reliability is bounded by ε and δ, respectively, with $\varepsilon = \delta = 0$ being a PSMT, i.e., (0, 0)-SMT. Relaxing security requirements allows construction

Table 10.1 Possibility of a 1-round (0, 0)-SMT and 1-round (0, δ)-SMT; 'Y' refers to the possibility of a secure protocol, whereas 'N' refers to the impossibility of a secure protocol

n	$t=3$	$t=3$	$t=4$	$t=4$	$t=5$	$t=5$
	PSMT	(0, δ)-SMT	PSMT	(0, δ)-SMT	PSMT	(0, δ)-SMT
10	Y	Y	N	Y	N	N
15	Y	Y	Y	Y	N	Y
20	Y	Y	Y	Y	Y	Y

of protocols with better communication efficiency. Of particular interests are $(0,\delta)$-SMT protocols that provide perfect privacy and guarantee that the probability of failure in receiving the correct message is at most δ. 1-round $(0,\delta)$-SMT protocols exist for $n=2t+1$ which is substantially less than $n=3t+1$ connectivity that is required for PSMT. A perfectly private ($\varepsilon=0$) and δ-reliable secure message transmission, denoted by (0, δ)-SMT, is called an **A**lmost **P**erfectly **S**ecure **M**essage **T**ransmission (**APSMT**, for short).

Communication efficiency of SMT protocols is measured in terms of the number of communication *rounds* and the *transmission rate*. The number of rounds is the number of required interactions between $\mathcal{S}$ and $\mathcal{R}$, and transmission rate is the number of bits that needs to be communicated for sending a message of size one bit. A lower bound on transmission rate of 1-round APSMT protocol is $\Omega(\frac{n}{n-2t})$ [19]. When $n=2t+k, k\geq 1$, the bound becomes $\Omega(n)$ and when $n=(2+c)t$, where $c>\frac{1}{t}$ is a constant, it becomes *constant*. SMT protocols that asymptotically achieve the above bounds, for respective connectivities, are called *rate-optimal* (or *optimal*, for short) with respect to the corresponding bound.

An Example. Depending on the fraction of corrupted wires, one can achieve a 1-round (0, 0)-SMT or a 1-round (0, δ)-SMT. The table below will illustrate this (Table 10.1).

10.1.1 Motivation of Our Work

The main focus of research on SMT has been on the construction of optimal protocols for minimum connectivity ($n=3t+1$ for one round PSMT and $n=2t+1$ for other cases). An SMT protocol, in general, sends a message block that consists of one or more elements of an alphabet. Optimal constructions of SMT require *messages blocks* of specific sizes. For example, the 2-round optimal PSMT protocol in [18] requires a message block that consists of t^2 (i.e. $O(n^2)$) field elements.

In practice, for a given t, the network connectivity may be above the required minimum, and the number of messages that needs to be sent may be different from the block length required by the protocols for optimal performance. With higher connectivity it is unclear how to efficiently use the extra connectivity for transmission of more messages. Also, the minimum message block size requirement of the protocol, ℓ_{min}, that is usually a function of the number of wires, may not be satisfiable for a particular application. For example when $n = 10$, the optimal performance of the protocol in [18] can be achieved if a message block of size 16 be sent. In applications of SMT such as [26] the protocol is used to securely transmit a cryptographic key from one node to another. Using a PSMT protocol over a field of size larger than 7 means that to transmit a 128 bit key, only seven field elements are required to send the key and the remaining elements of the message block will be un-used for transmission of information. These are important considerations when using optimal SMT protocols in practice and in particular in wireless sensor networks, where sensors have limited computational power and battery power is a scarce resource.

In this paper we consider the 1-round APSMT problem for different levels of network connectivity, starting from the minimum requirement of $n = 2t + 1$ up to higher connectivity of the form $n = (2 + c)t$, where a constant fraction of wires are corrupted, or $n = 2t + k$, where a constant number of additional wires exists. We will show two methods of constructing 1-round APSMT protocols for connectivity $n = 2t + k, k \geq 1$ and $n = (2 + c)t, c > \frac{1}{t}$, where k and c are constants. Our first method is by composing protocols. This can be used for a wide range of connectivities and results in less restriction on the message block size. The second method is a modular construction for designing SMT protocols. We use this method to construct a computationally efficient 1-round APSMT protocol for $n = 2t + 1$, and a 1-round *optimal* APSMT protocol for connectivity of the form $n = (2 + c)t, c > \frac{1}{t}$. Both methods can be applied to construct other protocols.

10.1.2 Our Results

The following is an outline of our results in this paper. We assume the sender and the receiver are connected by at least $2t + 1$ wires. We consider two types of extra connectivity. In the first case, n and t are fixed and $n = 2t + k$, where $k \geq 1$ is a constant. In the second case, $n = (2 + c)t$, where c is a constant satisfying $c > \frac{1}{t}$ and so, a *constant fraction* of wires are corrupted. We present the following results.

1. In [14] a *wire virtualization* method was proposed to construct an optimal protocol by using two existing component protocols, each with known security properties, such that the security of the resulting protocol could be derived from the component protocols. We employed wire virtualization to construct an *optimal* 1-round $(0, \delta)$-SMT protocol for connectivity $n = (2 + c)t, c > \frac{1}{t}$. Our constructed protocol gives the first general construction for connectivity less than $3t + 1$ (of the form $n = (2 + c)t$). The protocol has polynomial (in n) computation and so is computationally efficient.

2. We present a modular construction for 1-round APSMT protocol for $n = 2t + k$ which consists of two modules. The first module (n, t, δ)-*Send*, is a protocol that is used to deliver with δ-reliability, an information matrix of size $(n - 2t) \times n$ random elements chosen from an alphabet, e.g., a finite field, to the receiver such that the adversary can learn *at most* a sub-matrix of size $(n - 2t) \times t$. At the end of the protocol, sender and receiver share a sub-matrix of size $(n - 2t) \times (n - t)$ which remains completely unknown to the adversary. However, the sender and the receiver cannot identify the sub-matrix.
   The second module is a privacy amplification (PA) protocol that extracts $(n - t)$ elements that are completely unknown to the adversary, from a shared vector of size n which has at most t elements known by the adversary. We propose a new construction for the first module. For the second module, we adapt an existing PA technique which is computationally more efficient than the one used by Patra et al. [19]. We show a construction that uses these two modules and has linear (in n) transmission rate, which is optimal for this connectivity.
3. We present a direct construction for a 1-round APSMT protocol for $n = (2 + c)t$, where c is a constant and $c > \frac{1}{t}$ using our modular approach. This protocol has *constant* transmission rate and is *optimal*. The protocol uses the two modules (n, t, δ)-*Send* and $PA(n, n - t)$ used in designing the second protocol above. We also adapt an existing protocol as the third module, to send the ciphertexts with *constant* transmission rate. The ciphertexts are obtained by encrypting the message block using the one-time pads produced by using $PA(n, n - t)$.

All the protocols presented in this paper have *optimal* transmission rate. They are either the most efficient protocol known in the literature (the second protocol), or the first in the literature for the respective connectivity (the first and the third). The modular construction of SMT protocols is a new approach to constructing SMT protocols and could result in the construction of other efficient protocols using existing modules derived from other SMT protocols.

We note here that the first protocol is based on the result published in [24], whereas the second and the third protocols are based on the results published in [25].

Related Work. An optimal and efficient 1-round APSMT protocol for $n = 2t + 1$ is given in [19]. All other known 1-round APSMT protocols are either *not optimal* [10, 12] or *computationally inefficient* [17]. There are also efficient APSMT protocols for $n = 3t + 1$ [1, 19], but the one in [1] is *not optimal*. We give the first construction for 1-round APSMT problem for connectivity $n = (2 + c)t$, where $c > \frac{1}{t}$.

Organization. Section 10.2 gives definitions and notations that will be needed throughout this paper. Section 10.3 presents our 1-round optimal and efficient $(0, \delta)$-SMT protocol for connectivity $n = (2 + c)t, c > \frac{1}{t}$ using the wire virtualization method. In Sect. 10.4 we present the modular construction of our 1-round APSMT protocol for $n = 2t + k, k \geq 1$ together with security and efficiency analysis and comparison with related work. In Sect. 10.5, we give our 1-round APSMT protocol for $n = (2 + c)t, c > \frac{1}{t}$ and analyze its security and efficiency. In Sect. 10.6 we conclude our work with possible future work.

10.2 Background

Communication Model. We consider a *synchronous*, *incomplete* network. The sender $\mathcal{S}$ and the receiver $\mathcal{R}$ are connected by n vertex-disjoint paths, also known as wires or channels. Both $\mathcal{S}$ and $\mathcal{R}$ are honest. The goal is for $\mathcal{S}$ to send a message m, drawn from the message space $\mathcal{M}$, to $\mathcal{R}$ such that $\mathcal{R}$ receives it correctly and privately. The wires are undirected and two-way. The protocol can have one or more rounds. In a round, a message is sent by either $\mathcal{S}$ or $\mathcal{R}$ to the other party over the wires. Messages are delivered to the recipient of the round before the next round starts.

Adversary Model. The adversary $\mathcal{A}$ has *unlimited* computational power and corrupts a subset of nodes in the network. Honest nodes forward the received messages to the next nodes on the path. A path (wire) that includes a corrupted node is controlled by $\mathcal{A}$. Corrupted nodes can arbitrarily *eavesdrop, modify or block* messages sent over the corrupted wires. $\mathcal{A}$ is *adaptive* and uses all the information obtained from the corrupted wires to choose and corrupt any new wire. $\mathcal{A}$ is also *rushing*, i.e., in each round it sees the messages sent by $\mathcal{S}$ and $\mathcal{R}$ over the corrupted wires before deciding on the messages to be sent over those wires in that round. $\mathcal{A}$ can corrupt at most t out of the n wires. $\mathcal{S}$ and $\mathcal{R}$ *do not know which wires are corrupted.*

Notation. $\mathcal{M}$ is the message space from which messages are chosen according to a probability distribution $\Pr(m)$. Let $M_{\mathcal{S}}$ be the message randomly selected by $\mathcal{S}$. We assume $\mathcal{M}$ and $\Pr(m)$ are known in advance to all parties including the adversary. Let $R_{\mathcal{A}}$ be the random coins used by $\mathcal{A}$ to choose t out of total n wires to corrupt.

In an execution of an SMT protocol Π, $\mathcal{S}$ draws $M_{\mathcal{S}}$ from $\mathcal{M}$ using the distribution $\Pr(m)$, and aims to send it to $\mathcal{R}$ privately and reliably. We assume that at the end of the protocol, $\mathcal{R}$ outputs a message $M_{\mathcal{R}} \in \mathcal{M}$ or 'NULL'. An execution is completely determined by the random coins of the sender and the adversary and the message selected by the sender.

Let $V_{\mathcal{A}}(M_{\mathcal{S}}, r_{\mathcal{A}})$ denotes the view of the adversary $\mathcal{A}$ when $\mathcal{S}$ has chosen $M_{\mathcal{S}}$ and $R_{\mathcal{A}} = r_{\mathcal{A}}$. The view $V_{\mathcal{A}}(M_{\mathcal{S}}, R_{\mathcal{A}})$ is a random variable that depends on the random coins of $\mathcal{S}$ and the choice of $M_{\mathcal{S}}$. We use $V_{\mathcal{A}}(M_{\mathcal{S}} = m, r_{\mathcal{A}} = r)$ or $V_{\mathcal{A}}(m, r)$, for short, when $r_{\mathcal{A}} = r$ and $M_{\mathcal{S}} = m$.

The *statistical distance* of two random variables X, Y over a set $\mathcal{U}$ is defined as

$$\Delta(X,Y) = \frac{1}{2} \sum_{u \in \mathcal{U}} |\Pr[X = u] - \Pr[Y = u]|.$$

Definition 10.1 ([15]). An SMT protocol is called an (ε, δ)-Secure Message Transmission ((ε, δ)-SMT) protocol if the following two conditions are satisfied:

- ***Privacy:*** For every two messages $m_0, m_1 \in \mathcal{M}$ and every $r \in \{0,1\}^*$,

$$\Delta(V_{\mathcal{A}}(m_0, r), V_{\mathcal{A}}(m_1, r)) \leq \varepsilon,$$

where the probability is over the randomness of $\mathcal{S}$ and $\mathcal{R}$.

- ***Reliability:*** $\mathcal{R}$ receives the message $M_{\mathcal{S}}$ with probability $\geq 1-\delta$. That is,

$$\Pr[M_{\mathcal{R}} \neq M_{\mathcal{S}}] \leq \delta,$$

where the probability is over the randomness of all the players and the choice of $M_{\mathcal{S}}$.

When $\varepsilon = 0$, the protocol is said to achieve *perfect privacy* and when δ = 0, the protocol is said to achieve *perfect reliability*.

Parameters of SMT. The security parameters of SMT protocols are ε and δ, which bound the success of adversary in breaking privacy and reliability, respectively, of the protocol. The efficiency measures of SMT protocols are: (1) number of rounds, (2) communication complexity and (3) computation complexity.
The number of ***rounds*** of a protocol is the number of interactions between $\mathcal{S}$ and $\mathcal{R}$. We consider a synchronous network where time is divided into clock ticks and in each clock tick the sender or the receiver sends a message and the message is received by the other party before the next clock tick.
Communication complexity is the total number of bits transmitted between $\mathcal{S}$ and $\mathcal{R}$ for communicating the message. Communication efficiency is often measured in terms of *transmission rate*, which is the ratio of the total communicated bits to the size of the message block m (in bits). That is,

$$\mathit{Transmission\ Rate} = \frac{\text{total number of bits transmitted}}{\text{size of the message block in bits}(|m|)}$$

Computation complexity is the amount of computation performed by $\mathcal{S}$ and $\mathcal{R}$ throughout the protocol. A protocol which needs *exponential* (in n) computation is called *inefficient*. Efficient protocols need *polynomial* (in n) computation.
The relationship between n and t required for the existence of an SMT protocol is referred to as the ***connectivity requirements*** of the SMT protocol.

Bounds. Dolev et al. showed (1, 0)-SMT (**P**erfectly **R**eliable **M**essage **T**ransmission-**PRMT**) protocols are possible if $n \geq 2t+1$ [9]. It was proved that the transmission rate for a 1-round PSMT is lower bounded by $\Omega(\frac{n}{n-3t})$ [14]. When $n=(3+c)t$, where $c > \frac{1}{t}$, the lower bound becomes $\Omega(1)$, i.e. *constant*.

Virtualization. Wire virtualization (termed as 'player virtualization' in [14]) in SMT protocols was inspired by the *Bracha assignments* [3], which can be used to amplify the resilience of a distributed computation protocol.

In wire virtualization, we assume $n=(2+c)t, c > \frac{1}{t}$, physical wires are available and N *virtual wires* are created using subsets of size $\nu < n$ of these physical wires. Two protocols π_1 and π_2 are used. Protocol π_2 runs over a ν subset of physical wires, and protocol π_1 runs over the N virtual wires. For π_1 we use a 1-round PSMT (although it can also be a $(0,\delta)$-SMT) and π_2 is a PSMT or $(0,\delta)$-SMT (1-round or 2-round). In virtualization method for a given c, a minimum ν is calculated such that the number of corrupted virtual wires is less than $N/3$. A virtual wire is considered *uncorrupted* if less than $\lceil \frac{\nu}{2} \rceil$ of the physical wires associated with it are corrupted. Otherwise, the virtual wire is considered *corrupted*.

1-round PRMT Protocol for $n = (2+c)t$. We now present a 1-round PRMT protocol Π_1 for $n = (2+c)t, c > \frac{1}{t}$. The main idea of this protocol is to use codewords of Reed-Solomon codes to send a message block of size ct (each consisting of one field element) with perfect reliability by sending n field elements. The sender constructs a polynomial $f(x)$ of degree at most $(ct-1)$ such that the ct coefficients of $f(x)$ is the message block to be sent with perfectly reliability. The sender then sends evaluations of $f(x)$ on distinct points, each associated with a wire, through the corresponding wire. The receiver can correct the t possible errors, reconstruct the polynomial, and recover the message block of size ct. This protocol can be seen as an adaptation of the protocol REL-SEND of [23]. For details see the Appendix.

10.3 1-Round $(0,\delta)$-SMT Protocol Using Wire Virtualization

In this section, we give a construction for an optimal 1-round $(0,\delta)$-SMT protocol Π_2 for $n = (2+c)t, c > \frac{1}{t}$. Π_2 uses the 1-round PSMT protocol for $n = 3t+1$ of [14] as π_1, and the 1-round $(0,\delta)$-SMT protocol for $n = 2t+1$ of [22] as π_2. The protocol will send a message block $m_{\mathcal{S}}$ containing $O(n^{\nu})$ field elements by communicating $O(n^{\nu})$ field elements. So, the transmission rate of the protocol is *constant* and thus optimal.

The sender $\mathcal{S}$ uses π_1 to send the secret message $m_{\mathcal{S}}$ over the N virtual wires, each constructed (simulated) from a group of ν physical wires. Messages over the virtual wires are referred to as *shares* of $m_{\mathcal{S}}$. Virtual wires are simulated by running an instance of the protocol π_2 over ν physical wires and allows a share of $m_{\mathcal{S}}$ to be transmitted to $\mathcal{R}$. $\mathcal{R}$ uses her knowledge of the protocol specification including the simulation relationship between physical and virtual wires to recover the message block by first running the decoding algorithm of π_2 to recover the shares over the virtual wires, and then recovering the secret message block by running the decoding algorithm of π_1 using the recovered shares as input.

10.3.1 Virtualization Method

Let $C = \{C_1, \cdots, C_n\}$ be the set of physical wires. We use all ν-combinations of n physical wires as virtual wires. This results in $N = |\hat{C}| = \binom{n}{\nu}$ such wires. We use the following convention to label and order the virtual wires. Let $\hat{C} = \{\hat{C}_1, \ldots, \hat{C}_N\}$ be the set of virtual wires. Let $\hat{C}_i$ be simulated by $(C_{i_1}, \ldots, C_{i_\nu})$ where $C_{i_j} \in C$ and $i_j < i_k$ for $j < k$. This is an ordered list of physical wires and the order induces an order on $\hat{C}$: for a pair of virtual wires, $\hat{C}_i$ and $\hat{C}_j$, we have $i < j$, if there is a $0 \le k \le \nu - 1$ such that $i_l = j_l$ for $l \le k$, but $i_l < j_l$ for $l > k$. We note that each physical wire is in $\binom{n-1}{\nu-1}$ different virtual wires. The virtualization process is known to the adversary. Let $\mathsf{Cor} \overset{\text{def}}{=} \{i \mid C_i \text{ is a corrupted physical wire}\}$ be the set of corrupted physical wires. Then $t = |\mathsf{Cor}|$. Note that if more than $\lfloor \frac{\nu}{2} \rfloor$ of the physical wires that constitute a

Table 10.2 Minimum values of ν for different values of n and c

n	c	ν
105	0.1	83
110	0.2	23
115	0.3	11
120	0.4	7
125	0.5	5
130	0.6	3
150	1.0	3
$\vdots$	$\vdots$	3
280	4.3	3

virtual wire are corrupted, the share that is sent over that wire becomes accessible to the adversary. This is because π_2 becomes insecure in this case. We define the set of *corrupted* virtual wires as $\mathsf{vCor} \overset{\text{def}}{=} \{i : |\{i_1, \dots, i_\nu\} \cap \mathsf{Cor}| > \lfloor \frac{\nu}{2} \rfloor\}$. Let $T \overset{\text{def}}{=} |\mathsf{vCor}|$. For the security of π_1, we need $T \le \frac{N}{3+\lambda}$.

Finding **ν.** We count the number of virtual wires that contains at least $\lceil \frac{\nu}{2} \rceil$ corrupted physical wires. This is the number of virtual wires that are corrupted. We require this number to be less than one-third of the total number of virtual wires. We use the smallest ν that satisfies this condition. We also note here that ν should satisfy the condition that $\nu < n$.

The total number of virtual wires is $N = ((2+c)t)^\nu$. Suppose, $\nu = 2\nu' + 1$. If more than ν' physical wires contained in a virtual wire are corrupted, that virtual wire will be considered as corrupted. To find the number of corrupted virtual wires, T, we do the following.

The number of virtual wires that contains exactly $\nu' + i$ corrupted physical wires is = $\binom{\nu}{\nu'+i} t^{\nu'+i}((1+c)t)^{\nu'+1-i}, 1 \le i \le \nu' + 1$.

And so,

$$\begin{aligned} T &= \sum_{i=1}^{\nu'+1} \binom{\nu}{\nu'+i} t^{\nu'+i}((1+c)t)^{\nu'+1-i} \\ &= t^\nu \sum_{i=1}^{\nu'+1} \binom{\nu}{\nu'+i} (1+c)^{\nu'+1-i} \end{aligned}$$

Since we require $\frac{T}{N} \le \frac{1}{3+\lambda}$, this means,

$$\frac{\sum_{i=1}^{\nu'+1} \binom{\nu}{\nu'+i} (1+c)^{\nu'+1-i}}{(2+c)^\nu} \le \frac{1}{3+\lambda}. \tag{10.1}$$

Direct computation shows that $c = 0.1$ yields $\nu = 83$; $c = 0.3$ yields $\nu = 11$; and $c \ge 0.6$ yields $\nu = 3$. Note that $c = \frac{n}{t} - 2$ and ν is related to the ratio $\frac{n}{t}$ and not the concrete value of n.

Table 10.2 shows the values of ν for different values of c, assuming $t = 50$.

Input: a message block, $c, t, \nu, n = (2+c)t, \pi_1, \pi_2$ (π_1: a 1-round PSMT for $n = 3t+k, k \geq 1$, π_2: a 1-round $(0,\delta)$-SMT for $n = 2t+1$).
$\mathcal{S}$ constructs $N = \binom{n}{\nu}$ virtual wires. Each virtual wire is simulated by ν physical wires using an instance of protocol π_2.

R1 ($\mathcal{S} \longrightarrow \mathcal{R}$):

1. $\mathcal{S}$ runs the protocol π_1 to generate N shares $(S_1, \ldots, S_N)$ of the secret message block. The share S_i will be transmitted over the virtual wire $\hat{C}_i$, for $i \in \{1, \cdots, N\}$.
2. For $1 \leq i \leq N$, $\mathcal{S}$ runs the protocol π_2 independently to generate ν sub-shares of S_i, denoted by $(s_{i_1}, \cdots, s_{i_\nu})$.
3. Each physical wire is in the simulation of $L = \binom{n-1}{\nu-1}$ virtual wires and so each physical wire transmits L sub-shares. For physical wire C_i, $i \in \{1, \cdots, n\}$, denote the virtual wires that it is involved in by $(\hat{C}_{i_1}, \cdots, \hat{C}_{i_L})$, with order as $i_j < i_k$ for $j < k$. $\mathcal{S}$ organizes the sub-shares over wire C_i as $(s_{i_1}, \cdots, s_{i_L})$, where s_{i_j} is a sub-share of S_{i_j}. Finally $\mathcal{S}$ transmits the ordered sub-shares over C_i.

$\mathcal{R}$ recovers message block:

1. For $1 \leq i \leq n$, if $\mathcal{R}$ receives less than $\binom{n-1}{\nu-1}$ elements over C_i, then he marks C_i as *corrupted* and will neglect it from the following computation.
2. $\mathcal{R}$ organizes the received transmission according to the wire virtualization plan. Let the arranged results be $\{(s'_{i_1}, \ldots, s'_{i_\nu})\}_{i=1}^{N}$. If a virtual wire includes more than or equal to $\lfloor \frac{\nu}{2} \rfloor$ corrupted physical wires, it is marked as *corrupted*, and will be ignored from the following computation.
3. For $1 \leq i \leq N$, if the virtual wire $\hat{C}_i$ is not corrupted, $\mathcal{R}$ recovers a share S'_i by invoking the decoding algorithm of π_2 with input $(s'_{i_1}, \ldots, s'_{i_\nu})$; otherwise S'_i is set to be blank.
4. After recovering all shares $(S'_1, \ldots, S'_N)$, $\mathcal{R}$ recovers the secret message block by invoking the decoding algorithm of π_1 on the shares $(S'_1, \ldots, S'_N)$ and outputs it.

Fig. 10.1 The 1-round $(0,\delta)$-SMT protocol Π_2 for $n = (2+c)t, c > \frac{1}{t}$

Figure 10.1 describes the protocol Π_2. We note here that in Π_2 the receiver $\mathcal{R}$ can accept an incorrect message block and the probability of that event happening is bounded by δ. This is due to the protocol π_2 we use in protocol Π_2. In π_2, $\mathcal{R}$ can output an incorrect message block. If, on the other hand, we use a 1-round $(0,\delta)$-SMT protocol, in which $\mathcal{R}$ never outputs an incorrect message block (like the protocol USMT_Single_Phase in [19] or the one in [10]), then in the resulting protocol Π_2, $\mathcal{R}$ will either output the correct message block or 'NULL'.

Theorem 10.1. *Assume π_1 is a PSMT protocol for $n = 3t+k, k \geq 1$, π_2 is a $(0,\delta')$-SMT protocol for $n = 2t+1$, and $n = (2+c)t$ physical wires connect $\mathcal{S}$ and $\mathcal{R}$. If c and ν satisfy the condition (10.1), then the 1-round protocol Π_2 is a $(0,\delta)$-SMT protocol for $n = (2+c)t$ with reliability parameter $\delta \leq \frac{n^\nu}{\delta'}$, where δ' is the reliability parameter for the 1-round $(0,\delta')$-SMT protocol π_2.*

Proof. **Reliability**: In order to bound the probability of unreliable executions, let c and ν be selected according to the condition (10.1) such that $N = (3+\lambda)T$ for $\lambda > \frac{1}{T}$. For these values, the protocol π_1 provides perfect reliability if at most T virtual wires recover shares incorrectly. In this case the protocol Π_2 will also be reliable. Thus,

$$\begin{aligned}\delta &\le \Pr\left[\text{at least } T+1 \text{ virtual wires recover shares } \textit{incorrectly}\right] \text{ (I)}\\ &= \Pr\left[\text{at least one infected virtual wire recovers shares } \textit{incorrectly}\right] \text{ (II)}\\ &\le (N-T)\delta'\\ &< n^{\nu}\delta',\end{aligned}$$

where (II) is obtained from (I) because at most T corrupted virtual wires can be tolerated by π_1. Now if one extra virtual wire fails to recover its share, the message block recovered by $\mathcal{R}$ would not be equal to the message block sent by $\mathcal{S}$. δ' is the reliability parameter of the protocol π_2. $\binom{n}{\nu} \le \frac{n^{\nu}}{\nu!} \le n^{\nu}$ and $\frac{2+\lambda}{3+\lambda} < 1$.

The proof of perfect privacy follows from the fact that both sub-protocols provide perfect privacy. □

10.3.2 The Implementation of Protocols π_1 and π_2

We show how to use some existing protocols to instantiate π_1 and π_2 such that the resulting protocol Π_2 has constant transmission rate.

Implementation of π_1. We use the 1-round PSMT protocol with $n \ge 3t+1$ in [14] as π_1. The protocol is described in Fig. 10.7 of the Appendix "1-Round PSMT Protocol for $n > 3t$".

Lemma 10.1 ([14]). *Let $\mathbb{F}$ be a finite field with size q. The 1-round protocol π_1 is a PSMT protocol for $n \ge 3t+1$ and is polynomial time (in q,n) computable. In each round, π_1 can transmit $(n-3t)$ field elements by transmitting n field elements.*

Implementation of π_2. We use the 1-round $(0,\delta)$-SMT protocol for $n \ge 2t+1$ in [12] as π_2. The protocol is described in Fig. 10.8 of the Appendix "The 1-Round (0, δ)-SMT Protocol for $n > 2t$".

Lemma 10.2 ([12]). *Let $\mathbb{F}$ be a finite field with size q. The protocol π_2 is a $(0,\delta')$-almost SMT protocol for $n \ge 2t+1$. The protocol is polynomial time (in q,n) computable. In each execution, π_2 can transmit 1 field element by sending $O(n^2)$ field elements.*

10.3.3 Security and Efficiency Analysis

Theorem 10.2. *Let κ be the security parameter. Assume there are $n = (2+c)t$ physical wires between $\mathcal{S}$ and $\mathcal{R}$. By the wire virtualization method, if c, ν satisfy the condition (10.1), then the protocol Π_2, constructed by combining the protocols π_1 and π_2 specified above, is a 1-round $(0,\delta)$-SMT protocol for $n = (2+c)t$ with $c > \frac{1}{t}$ with reliability parameter $\delta \leq \frac{n^{\nu}\delta'}{q}$. Moreover, the protocol is polynomial time (in q,n) computable, and its transmission rate is constant when the length of the message is $m = \Omega(n^{\nu}\kappa)$.*

Proof. The privacy and computation complexity are straightforward from Theorem 10.1.

For the transmission rate, let N, T be defined as in the virtualization method. Note that by Lemma 10.1, π_1 transmits $(N-3T)$ field elements (in $\mathbb{F}$) by sending N field elements (shares). From Lemma 10.2, for each share, protocol π_2 needs to transmit ν^3 field elements. Therefore, when the secret message block has size $(N-3T)$ field elements, the protocol Π_2 needs to transmit $N\nu^3$ field elements in total. The transmission rate is $\frac{N\nu^3}{N-3T} = \frac{(\delta+3)}{\delta}\nu^3 = O(\nu^3) = O(1)$, since ν is selected to be a small constant which is independent of n.

Now $\delta = \frac{n^{\nu+1}}{q} = 2^{-\kappa}$. Then $\log q = \Omega(\kappa + \log n)$. Thus when the message block is $m = (N-3T)\log q = \frac{\lambda}{3+\lambda}\binom{n}{\nu}\log q = \Omega(n^{\nu}\kappa)$ bits length, the transmission rate of the protocol is constant. □

Comparison with Related Work. Araki's protocol works for $n = 3t+1$ but it is not optimal [1]. Patra et al. presented an optimal 1-round $(0,\delta)$-SMT protocol for $n = 2t+1$ ([19], Sect. 4.2). They showed that their protocol can be adapted for $n = 3t+1$ to achieve the optimal transmission rate (i.e. *constant*). To the best of our knowledge, our protocol Π_2 is the first optimal 1-round $(0,\delta)$-SMT protocol for connectivity less than $3t+1$ (i.e. of the form $n = (2+c)t, c > \frac{1}{t}$).

10.4 Modular Construction of a 1-Round Optimal APSMT Protocol for $n = 2t + k$

10.4.1 Main Idea

Our construction is modular and consists of two sub-protocols that will be used as black-boxes in the final protocol. The first sub-protocol is called (n,t,δ)-*Send* and is a 1-round protocol that sends a matrix of $(n-2t) \times n$ random elements from an alphabet set to the receiver, such that the communication is δ reliable and that at most a sub-matrix of size $(n-2t) \times t$ will be leaked to the adversary. This means that at the end of the protocol the sender and the receiver will share $(n-2t) \times n$ elements with probability at least $1-\delta$ ($\delta < \frac{1}{2}$) such that a sub-matrix of size $(n-2t) \times (n-t)$ is completely unknown to the adversary. The sender and the receiver, however, do not know exactly which is this sub-matrix. The second subprotocol is a non-interactive *privacy amplification (PA)* protocol, $PA(n,n-t)$ that allows two users sharing a string of n elements, where *at most* t of which is known by the adversary, to construct a sequence of $(n-t)$ elements, which is completely unknown to the adversary.

Non-interactive privacy amplification was defined and constructed in the context of SMT protocols (See Sect. 3.3 [2]). We use an existing simple construction that is computationally more efficient than the PA technique used in [19].

The 1-round APSMT protocol calls the above two subprotocols. The APSMT protocol works as follows.

The sender selects a matrix of $(n-2t)\times n$ random elements $r_{ij}, i=1,\cdots,n-2t, j=1,\cdots,n$, from a finite field $\mathbb{F}$, and sends them to the receiver by a call to (n,t,δ)-*Send* protocol. The sender also calls $PA(n,n-t)$, using the $(n-2t)n$-sequence as input, $(n-2t)$ times to generate $(n-2t)(n-t)$ elements that will remain completely unknown to the adversary. These are used as one-time pads $R_{ij}, i=1,\cdots,n-2t, j=1,\cdots,(n-t)$ to construct $(n-2t)(n-t)$ ciphertexts (add one pad to a message). The sender broadcasts the ciphertexts (sends on all wires) to the receiver.

The receiver, obtains the $(n-2t)n$ random elements as the output of (n,t,δ)-*Send*, uses $PA(n,n-t)$ to reconstruct the $(n-2t)(n-t)$ pads and finally recovers the $(n-2t)(n-t)$ secrets. Sending the ciphertexts over all wires provides a reliable broadcast by taking the values received on the majority of wires.

Below we will first describe the subprotocols: (n,t,δ)-*Send* and $PA(n,n-t)$ and their possible instantiations using existing protocols, and then present our instantiation of the subprotocols that results in an optimal 1-round APSMT protocol. We note here that there exists an *optimal* 1-round APSMT protocol for $n=2t+1$ by Patra et al. [19]. But the instantiations using our proposed and adapted subprotocols will result in simpler and computationally efficient 1-round APSMT protocol. This is because both of our used subprotocols are simpler and computationally efficient.

10.4.2 (n,t,δ)-*Send*

This protocol constructs an input matrix R of size $n\times n$, $R=(r_{11},\ldots,r_{1n},\ldots,r_{n,1},\ldots,r_{n,n})$ consisting of $(n-2t)n$ randomly chosen elements (that the adversary has no knowledge about) together with $2tn$ elements that are computed from them, and delivers it to the receiver as R' such that (1) $\Pr(R=R')\geq 1-\delta, \delta<\frac{1}{2}$, and (2) at most a sub-matrix of size $(n-2t)\times t$ of R will become known to the adversary $\mathcal{A}$. Therefore, a sub-matrix of size $(n-2t)\times(n-t)$ will remain unknown to $\mathcal{A}$.

Informally the protocol works as follows.

The sender generates n polynomials in such a way that any t polynomials are linearly dependent on the other $(n-t)$ random polynomials. Each polynomial will be sent through one distinct wire, but the evaluations of each polynomials on n

Transmission: Consider a sequence of $(n-2t)n$ random elements $R = (r_{11}, \ldots, r_{1n}, \ldots, r_{n-2t,1}, \ldots, r_{n-2t,n})$ as a matrix of size $(n-2t) \times n$. The sender performs two steps as follows:

- Step 1. For each $j, 1 \leq j \leq n$:
  Constructs a polynomial $q_j(x)$ of degree $\leq (n-t-1)$ such that $q_j(x) = \sum_{i=1}^{n-t} a_{i,j} x^{i-1}$. Here $a_{ij} = r_{ij}$ when $i \leq n-2t$, otherwise a_{ij}'s are random elements from $\mathbb{F}$.
- For each $i, 1 \leq i \leq n$:
  Constructs a polynomial $p_i(x)$ of degree $\leq (n-1)$ such that $p_i(x) = \sum_{j=1}^{n} q_j(i) x^{j-1}$.
- Step 2. Randomly selects n^2 field elements $s_{ij}, 1 \leq i, j \leq n$ and constructs pairs $(s_{ij}, p_i(s_{ij})), 1 \leq i, j \leq n$.
- Sends $p_i(x)$ through wire i and $(s_{ij}, p_i(s_{ij}))$ through wire j, for $1 \leq i, j \leq n$.

Recovery: The receiver does the following.

- Step 1. For each $i, 1 \leq i \leq n$:
  Receive $p'_i(x)$ over wire i, and $(s'_{ij}, v_{i,s_{ij}})$ through wire j, for $1 \leq j \leq n$. Suppose a'_{ij} are the coefficients of the received polynomials.
- Compute $k = |\{j : p'_i(s'_{ij}) \neq v_{i,s_{ij}}|$.
- If $k \geq t+1$, then decide wire i as corrupted and adds i to a list $FAULTY$.
- Step 2. Suppose $i_1, i_2, \ldots, i_{n'}, n' \geq n-t$ are the indices of the wires $\notin FAULTY$. For each $j, 1 \leq j \leq n$, do the following:

  * Form a polynomial $q'_j(x)$ of degree $\leq n-t-1$ using $a'_{i_1 j}, \ldots, a'_{i_{n-t} j}$ and verify whether $q'_j(i_l) \neq p'_{i_\ell}(i_\ell), \ell > n-t$. If there exists one such ℓ, then output 'NULL' and terminate the protocol.

- Reconstruct the polynomials $q_j(x), 1 \leq j \leq n$ by considering any $(n-t)$ of its coefficients carried by wires not in the list $FAULTY$.
- Recover the $(n-2t) \times n$ random elements from the polynomials $q_j(x), 1 \leq j \leq n$.

Fig. 10.2 (n,t,δ)-*Send*

random points are sent through different wires. Thus, if the adversary changes a polynomial, with very high probability, it will be detected by the receiver.
Our proposed (n,t,δ)-*Send* module is shown in Fig. 10.2.

As seen in Fig. 10.2 recovery has two steps. In step 1, the receiver $\mathcal{R}$ runs a test on each polynomial (carried by a wire) that determines all uncorrupted wires together with some that are corrupted but pass the test. At the end of this step receiver has $n' \geq n-t$ wires, some possibly (undetectably) corrupted. In Step 2, $\mathcal{R}$ runs a second test to determine if any of the n' wires that have passed the test of Step 1 is corrupted. If there is one such corrupted wire, the protocol is terminated. This test is done by constructing a polynomial of degree at most $(n-t-1)$ considering the first $(n-t)$ shares corresponding to the n' wires not in $FAULTY$. Then if the constructed polynomial satisfies all the remaining $n' - (n-t)$ shares, then the test continues, otherwise the test fails. Note that in Step 2, $\mathcal{R}$ only detects the existence of a corrupted wire but cannot identify it.

Theorem 10.3. *Module (n,t,δ)-Send sends $(n-2t)\times n$ random elements so that all the $(n-2t)\times n$ will be received with probability $1-\delta$, and the adversary can learn at most $(n-2t)\times t$ elements, while $(n-2t)\times(n-t)$ elements are completely unknown to the adversary. The total required communication is $O(n^2\log|\mathbb{F}|)$.*

Proof. We need to prove that $(n-2t)\times(n-t)$ out of $(n-2t)\times n$ elements will be perfectly private w.r.t. to the adversary and the probability that $p'_i(x)=p_i(x), 1\le i\le n$, is bounded by δ.

Perfect Privacy The polynomials $p_i(x)$ are generated using the evaluations of he polynomials $q_j(x)$ as coefficients. Since the degree of the polynomials $q_j(x)$ are at most $(n-t-1)$, any of the $(n-t)$ polynomials $p_i(x)$ determine the remaining t polynomials. Thus there are $(n-t)$ independently generated random polynomials $p_i(x)$. The adversary sees *at most* any t polynomials and t points of any other polynomials. Since polynomials are of degree $(n-1)$ then all n coefficients are independent and so in total $(n-2t)\times(n-t)$ elements remain unknown to the adversary. More specifically, there are total $n(n-t)$ random elements in the polynomials $q_j(x), 1\le j\le n$ (the polynomials are of degree at most $(n-t-1)$). The adversary knows nt shares of the t polynomials $p_i(x)$ he controls ($p_i(x)$ are of degree at most $n-1$). For of the remaining independent $(n-2t)$ polynomials $p_i(x)$, the adversary sees t shares of each polynomials. Thus he knows total $nt+(n-2t)t$ shares. Thus the number of remaining information symbols is $n(n-t)-nt-(n-2t)t=n(n-2t)-(n-2t)t=(n-2t)(n-t)$. These information symbols will remain perfectly private.

δ-reliability The objective of the adversary is to change the polynomials sent through the wires that he corrupts such that at least one of the changed polynomials is not detected as corrupted by the receiver. This could happen, if the changed polynomial goes through at least $(t+1)$ points of the original polynomial. In that case, the receiver will output 'NULL'. To fulfill this objective, the adversary randomly changes all the t polynomials, sent through the wires that he controls, to polynomials different than the original polynomials, with the hope that at least one of the changed polynomials will have at least one point common with the corresponding original polynomial.

If the adversary $\mathcal{A}$ succeeds in Step 1, the protocol will be aborted in Step 2 of Recovery. This means $\mathcal{A}$ has to construct a polynomial with at least $n-t$ points matching the points associated with the original polynomial. Below we calculate δ for $n=2t+1$. Similar calculations can be made for $n=2t+k$. The probability of success of the adversary, in other words, δ for $n=2t+1$ is calculated as follows:
A random polynomial $p_i(x)$ has n points $(s_{i,1},p_i(s_{i,1})),(s_{i,2},p_i(s_{i,2})),\ldots,$ $(s_{i,n},p_i(s_{i,n}))$. The adversary succeeds if $p_i(x)$ is changed to $p'_i(x)$ and the new polynomial $p'_i(x)$ passes through at least $(n-t)$ of the n points of $p_i(x)$.

The probability that $p'_i(x)$ goes through a random point is $\frac{1}{q}$. The probability that $p'_i(x)$ does not go through a random point is $1-\frac{1}{q}$. The probability that $p'_i(x)$

does not go through $(n-t)$ random points is $(1-\frac{1}{q})^{n-t}$. The probability that $p_i'(x)$ goes through at least $(n-t)$ random points is $1-(1-\frac{1}{q})^{n-t} < \frac{n-t}{q}$. This is the probability that the polynomial is not detected as corrupted. For all the t polynomials this probability is less than $\frac{t(n-t)}{q}$. Thus, the probability that the protocol will output 'NULL' is at most $\frac{t(n-t)}{q}$. That is, $\delta < \frac{t(n-t)}{q}$.

By using a larger size field, we can decrease the upper bound of δ. For example, when we choose $q = ct(n-t)$, the reliability of the protocol becomes $1-(1-\frac{t}{ct(n-t)})^{n-t} = 1-e^{-c}$. Therefore, with very high probability, the receiver will receive all the polynomials $p_i(x)$ as was sent by the sender.

Efficiency. The sender sends n polynomials, each of degree at most $(n-1)$ through the n wires. This incurs a communication of $n^2 \log|\mathbb{F}|$. He also sends each of the n pair of values (evaluation points) through each wire, for all the polynomials. This needs a communication of $2n^2 \log|\mathbb{F}|$. Therefore, the total communication of this protocol is $n^2 \log|\mathbb{F}| + 2n^2 \log|\mathbb{F}| = O(n^2 \log|\mathbb{F}|)$. □

10.4.3 Non-interactive Privacy Amplification for SMT

Privacy amplification allows the sender and the receiver to non-interactively generate a random elements which will be completely unknown to the adversary, from $b > a$ random elements, where the adversary knows *at most* $(b-a)$ elements.

This primitive had been used as part of the protocol in Sect. 10.4 by Srinathan et al. [23] and later defined by Agarwal et al. [2]. Srinathan et al. [23] referred to this primitive as 'extracting randomness' and gave the algorithm EXTRAND with two parameters n and f, where f out of n random elements are completely unknown to the adversary. In their algorithm, they used a public $n \times f$ Vandermonde matrix V. The algorithm extracts f randomness by multiplying the n random numbers with V, and the result is f random numbers.

Agarwal et al. used another non-interactive privacy amplification technique, which is a generalization of Shamir's Secret Sharing Scheme [21]. They considered a $(b-1, a+b)$ secret sharing, where $(b-a)$ elements are completely known to the adversary. Eventually, $a < b$ elements will be obtained from the secret sharing, all of which will be information-theoretically random. The proof of perfect privacy of the a random elements follows from the security of Shamir's $(t, n+t)$-secret sharing scheme [21].

More specifically, the sender $\mathcal{S}$ generates a polynomial $f(x)$ of degree at most $b-1$ such that $f(i) = r_i, 1 \leq i \leq b$. $\mathcal{S}$ then uses $f(b+1), \ldots, f(b+a)$ as the a information-theoretic one-time pads.

We use the protocol in [2] given in Fig. 10.3.

$PA(b, b-a)$; $a < b$: input $(x_1, \ldots, x_b) \in \mathbb{F}^b$; output: $(X_1, X_2, \ldots, X_a) \in \mathbb{F}^a$

1. Forms a polynomial $f(x)$ of degree $\leq (b-1)$ such that $f(i) = x_i, 1 \leq i \leq b$.
2. Outputs $(f(b+1), \ldots, f(b+a))$.

Fig. 10.3 The non-interactive privacy amplification technique $PA(b, b-a)$

The sender $\mathcal{S}$ wishes to send a message block of size $n-t = (t+1)$ $[m_0, m_1, \ldots, m_t] \in \mathbb{F}^{t+1}$ to the receiver $\mathcal{R}$. Since $n = 2t+1$, here $(n-2t)n = n$.

- Step 1. The sender $\mathcal{S}$ does the following:

  1. Calls (n,t,δ)-*Send* to send n random elements $r_i, 1 \leq i \leq n$.
  2. Calls $PA(n, n-t)$ with $(r_1, r_2, \ldots, r_n)$ as input and get $(R_1, R_2, \ldots, R_{t+1})$.
  3. Forms $t+1$ ciphertexts as $c_i = m_i \oplus R_i$, and broadcasts $c_i, 1 \leq i \leq t+1$.

- Step 2. The receiver does the following.

  1. Receives the n random elements $r_1, r_2, \ldots, r_n$.
  2. Calls $PA(n, n-t)$ with $(r_1, r_2, \ldots, r_n)$ as input and get $(R_1, R_2, \ldots, R_{t+1})$
  3. Recovers the message block of size $t+1$ as $m_i = c_i \oplus R_i, 1 \leq i \leq t+1$.

Fig. 10.4 The 1-round APSMT protocol Π_3 for $n = 2t+1$

10.4.4 Description of the Protocol

For simplicity we will describe the protocol for $k = 1$, but it can be easily used for larger k. Our 1-round APSMT protocol Π_3 for $n = 2t+1$ is given in Fig. 10.4. The receiver in this protocol will never output incorrect message block. He will either output the correct message block or output 'NULL'.

Using the above two modules, the main protocol is straightforward. The sender $\mathcal{S}$ first randomly selects n random numbers and communicate them to the receiver $\mathcal{R}$ using (n,t,δ)-*Send*. The sender also uses PA to generate $(n-t)$ one-time pads and uses them to encrypt the message block of size $(n-t)$. The resultant ciphertexts are broadcasted. The receiver, on receiving the n random numbers, similar to the sender $\mathcal{S}$, uses PA to regenerate the $(n-t)$ one-time pads and use them to decrypt the $(n-t)$ ciphertexts to recover the secret message block of size $(n-t)$.

Security and Efficiency Analysis. The protocol uses (n,t,δ)-*Send* and $PA(n, n-t)$, followed by a step in which $(t+1)$ ciphertexts are broadcasted. The broadcast can be seen as a third module with perfect reliability. The broadcast is also perfectly secure for the message block because of the one-time-pad encryption used for the message block. Reliability of the overall protocol directly follows from the reliability of (n,t,δ)-*Send* and the final broadcast. Perfect privacy of the protocol follows from the perfect security of the one-time-pads that are generated through the application of $PA(n, n-t)$ and Theorem 10.1.

Table 10.3 Comparison with 1-round APSMT protocols for $n = 2t + 1$ (here Comp. refers to computation complexity and q is the field size)

Author	Comp.	δ	Optimality
Kurosawa and Suzuki [17]	*Exp.*	$\leq (\binom{n}{t+1} - \binom{n-t}{t+1})\lambda$[1]	Yes
Patra et al. [19]	*Poly.*	$\leq \frac{n^3}{q}$	Yes
Desmedt et al. [10]	*Poly.*	$\leq \frac{t(t+1)}{q}$	No
This work	*Poly.*	$\leq \frac{n^2}{q}$	Yes

The transmission rate of Π_3 is $O(n)$. This is true because (n,t,δ)-*Send* has communication cost of $O(n^2 \log|\mathbb{F}|)$. The protocol Π_3 also broadcasts $(t+1)$ ciphertexts with a communication cost of $n(t+1)\log|\mathbb{F}|$. Therefore, the total communication of this protocol is $O(n^2 \log|\mathbb{F}|)$.The protocol sends a message block of size $(t+1)$ of total size $(t+1)\log|\mathbb{F}| = O(n \log|\mathbb{F}|)$ and so the transmission rate is $O(n)$ which is optimal for a 1-round (0, δ)-SMT protocol for $n = 2t + 1$.

Comparison. To the best of our knowledge, our protocol Π_3 is the most efficient and most reliable (lower δ than any *optimal* protocol) *optimal* 1-round APSMT protocol. The comparison with related work is outlined in Table 10.3. Our protocol also needs less amount of computation than that of [19] due to the second module used for generating the one-time pads. We adapted Agarwal et al.'s privacy amplification (PA) technique based on Shamir's secret sharing. On the other hand, Patra et al.'s protocol uses the PA technique of [23]. That technique involves a matrix multiplication for generating the one-time pads. We know the most efficient matrix multiplication algorithm by Coppersmith and Winograd [8] needs $O(n^{2.376})$ computation, therefore, their protocol will need a computation cost of $O(n^{4.752})$ as they need to multiply a $(1 \times n^2)$ matrix with a $n^2 \times f$ matrix $(f > \frac{n}{2})$. On the other hand, our protocol will need $O(n^4)$ computation (requires using the decoding algorithm of [5] n times). Thus our protocol is computationally much efficient than their approach.

10.5 Modular Construction of a 1-Round Optimal APSMT Protocol for $n = (2+c)t$

We present a 1-round APSMT protocol for $n = (2+c)t$, where c is a constant satisfying $c > \frac{1}{t}$. The protocol has *optimal* transmission rate. The protocol is designed by extending the protocol Π_3 and using the 1-round PRMT protocol Π_1 for $n = (2+c)t, c > \frac{1}{t}$ showed in Sect. 10.2.

[1] Here λ is the probability that the cheater win in a secret sharing scheme with a cheater.

Table 10.4 Possibility of the protocol Π_2. Here '*' refers to the case when a SMT protocol is not possible using wire virtualization method

n/t	3	5	7	9	11
10	Y	N	N	N	N
15	Y	Y	N*	N	N
20	Y	Y	Y	Y	N

The sender wishes to send a message block of size $(n-t)(n-2t)$ $[m_{11},\ldots,m_{1,n-t},m_{21},\ldots,m_{2,n-t},\ldots,m_{n-2t,1},\ldots,m_{n-2t,n-t}] \in \mathbb{F}^{(n-2t)(n-t)}$ to the receiver $\mathcal{R}$.

- Step 1. The sender $\mathcal{S}$ does the following:
  1. Calls (n,t,δ)-*Send* (communicate $(n-2t)n$ random elements $r_{ij}, 1 \le i \le n-2t, 1 \le j \le n$).
  2. Calls $PA(n,n-t)$, $(n-2t)$ times, for $1 \le i \le n-2t$ (use $(r_{ij}, 1 \le j \le n)$) as input to obtain $(n-t)$ random-elements $(R_{i1},\ldots,R_{i,n-t})$).
  3. Generates $(n-2t)(n-t)$ ciphertexts, $c_{ij} = m_{ij} \oplus R_{ij}, 1 \le i \le n-2t, 1 \le j \le n-t$ and send them by calling Π_1, in parallel, $(n-t)$ times.
- Step 2. The receiver does the following.
  1. Receives the $(n-2t)n$ random elements $r_{ij}, 1 \le i \le n-2t, 1 \le j \le n$..
  2. Calls $PA(n,n-t)$, $(n-2t)$ times with $(r_{ij}, 1 \le j \le n))$ as input to get $(n-t)$ random-elements $(R_{i1},\ldots,R_{i,n-t})$, for $1 \le i \le n-2t$.
  3. Receives the $(n-2t)(n-t)$ ciphertexts perfectly reliably and recovers the message block of size $(n-2t)(n-t)$ using $(R_{11},\ldots,R_{1,n-t},\ldots,R_{n-2t,1},\ldots,R_{n-2t,n-t})$ as $m_{ij} = c_{ij} \oplus R_{ij}, 1 \le i \le n-2t, 1 \le j \le n-t$.

Fig. 10.5 The 1-round APSMT protocol Π_4 for $n = (2+c)t$

Our protocol Π_2 of Sect. 10.3, although works for $n = (2+c)t$, for some values of c and n, cannot work as illustrated in the following table (Table 10.4).

In the above for $n = 15, t = 7$, protocol Π_2 does not work, as in this case $c = 0.15$, and for this c, the minimum values of ν is more than 23. But the total number of wires is only 15. This shows one of the limitation of protocol Π_2.

To the best of our knowledge, this is the first direct construction of 1-round APSMT protocol for $n = (2+c)t$, with optimal transmission rate which works for all values of n and c. There is a 1-round APSMT protocol for $n = 3t+1$ in [1] and [19]. But no one has considered an optimal 1-round APSMT protocol for connectivity less than $3t+1$.

We use the following three modules as blackboxes in designing our protocol: (1) module (n,t,δ)-*Send* (Sect. 3.1), (2) module $PA(n,n-t)$ (Sect. 3.2), and (3) 1-round PRMT protocol Π_1 (showed in Sect. 10.2).

Description of the protocol. The protocol is given in Fig. 10.5. The receiver will either output the correct message block or output 'NULL'.

Perfect Privacy and δ-Reliability. Perfect privacy of Π_4 follows from the perfect privacy of the first two modules and independent executions of each invocation of the module $PA(n,n-t)$. The reliability of Π_4 follows directly from the reliability of (n,t,δ)-*Send* and that of Π_1. Therefore, Π_4 is unreliable with probability *at most* $\frac{n^2}{|\mathbb{F}|}$.

Efficiency. (n,t,δ)-*Send* needs a communication of $O(n^2 \log|\mathbb{F}|)$. The communication for using Π_1 is $n(n-2t)\log|\mathbb{F}| = O(n^2 \log|\mathbb{F}|)$ bits. Therefore, the total communication of the protocol Π_4 is $O(n^2 \log|\mathbb{F}|)$. The protocol sends a message block of size $(n-2t)(n-t) = ct(t+ct)$ of total size $ct(t+ct)\log|\mathbb{F}| = O(n^2 \log|\mathbb{F}|)$, resulting in $O(1)$ transmission rate and is thus *optimal*.

10.6 Conclusion and Future Work

We gave two methods of designing 1-round (0, δ)-SMT protocols for flexible connectivities. The first method uses wire virtualization to design a 1-round (0, δ)-SMT protocol for $n = (2+c)t, c > \frac{1}{t}$. We then present a modular approach of designing SMT protocols. We gave two 1-round *optimal* APSMT protocols for connectivities $n = 2t+k$, where $k \geq 1$ is a constant, and $n = (2+c)t, c > \frac{1}{t}$, respectively. The first modular construction gives a protocol with the highest reliability compared to all existing optimal 1-round APSMT protocol. The second modular protocol is the first for this kind of connectivity. It remains an interesting open problem to construct optimal 1-round APSMT protocols with more reliability than the protocols presented in this paper. We proposed modular construction for designing optimal SMT protocols. It is interesting to extend this approach to protocols with more than one round.

Acknowledgements Financial support for this research was provided in part by Alberta Innovates Technology Future in the Province of Alberta, as well as NSERC (Natural Sciences and Engineering Research Council) in Canada.

Appendix

A 1-Round PRMT Protocol for $n = (2+c)t$

The protocol is given in Fig. 10.6.

Theorem 10.4. Π_1 *is an efficient and optimal 1-round PRMT Protocol which sends a message block of size ct by communicating n field elements.*

Proof. **Reliability.** Since $n = 2t+ct$ and the degree of the polynomial $f(x)$ is at most $ct-1$, the minimum hamming distance between the sent and the received codeword is $n-ct+1 = 2t+ct-ct+1 = 2t+1$. Therefore, the receiver can correct *at most t* possible errors and recover the original codeword. These recovered codeword is the original shares corresponding to the polynomial $f(x)$. Therefore, by taking any ct of these shares the receiver $\mathcal{R}$ can reconstruct $f(x)$ and thus recover correctly the message block of size ct (the ct coefficients of the polynomial).

1. (**S** $\longrightarrow$ **R:** Given the message block $[m_0, m_1, \cdots, m_{ct-1}]$ which is taken randomly from $\mathbb{F}^{ct}$, $\mathcal{S}$ randomly forms a polynomial $f(x) \in \mathbb{F}[x]$ with degree $\leq ct - 1$ as

$$f(x) = m_0 + m_1 x + \cdots + m_{ct-1} x^{ct-1}$$

   $\mathcal{S}$ then sends $f(\alpha^i)$ to $\mathcal{R}$ over wire i, $1 \leq i \leq n$, where α is a generator of the multiplicative group of $\mathbb{F}$.
2. **R recovers message block:** $\mathcal{R}$ uses the Welch-Berlekamp decoding algorithm [5] on the received values in order to recover the messageblock $[m_0, m_1, \ldots, m_{ct-1}]$.

Fig. 10.6 The 1-round optimal PRMT protocol Π_1 for $n = (2+c)t, c > \frac{1}{t}$

- Round 1: ($\mathcal{S} \longrightarrow \mathcal{R}$): Given a message block $m_{\mathcal{S}} = [m_0 m_1 \ldots m_{k-1}] \in \mathbb{F}^k \, (k = n - 3t)$, $\mathcal{S}$ randomly forms a polynomial $p(x) \in \mathbb{F}[x]$ with degree $d = n - 2t - 1$ as

$$p(x) = c_d x^d + \ldots + c_k x^k + m_{k-1} x^{k-1} + \ldots + m_1 x + m_0,$$

  where c_i is uniformly selected from $\mathbb{F}$ ($k \leq i \leq d$).
  $\mathcal{S}$ then sends $p(\alpha^i)$ to $\mathcal{R}$ over wire i, $1 \leq i \leq n$, where α is a generator of the multiplicative group of $\mathbb{F}$.
- $\mathcal{R}$ recovers message block: $\mathcal{R}$ uses the Welch-Berlekamp decoding algorithm [5] on the received values in order to obtain the message block.

Fig. 10.7 The 1-round PSMT protocol π_1 for $n \geq 3t + 1$

Efficiency. It is easy to see that the computation done by both the sender and the receiver are polynomial in n. Therefore the protocol Π_1 is *efficient*. To determine the transmission rate, we see that the sender sends n field elements for reliably sending the message block of size ct (i.e., ct field elements). Therefore, the transmission rate of π_1 is $\frac{n}{ct} = \frac{n}{O(n)} = O(1)$. Thus, the protocol Π_1 is *optimal* in transmission rate. ■

1-Round PSMT Protocol for $n > 3t$

The protocol is given in Fig. 10.7 [14].

The 1-Round (0, δ*)-SMT Protocol for* $n > 2t$

The authentication function **auth** in the protocol is defined as $\mathbf{auth}(M, a, b) = aM + b$ with $a, b, M \in \mathbb{F}$ [12]. It is known the one-time MAC function can be used to authenticate one message M without leaking any information about the key (a, b), R1 ($\mathcal{S} \longrightarrow \mathcal{R}$):

1. Suppose $m_{\mathcal{S}}$ be the secret that $\mathcal{S}$ want to send to $\mathcal{R}$. $\mathcal{S}$ constructs shares $(s_1,\ldots,s_{2t+1})$ of $m_{\mathcal{S}}$ by invoking a $(t+1)$-out-of-$(2t+1)$ secret sharing scheme.
2. $\mathcal{S}$ runs the following steps in parallel for $i \in \{1,\ldots,n\}$.

   a. $\mathcal{S}$ chooses $\{(a_{i,j},b_{i,j}) \in \mathbb{F}^2\}$ for $1 \le j \le 2t+1$.
   b. $\mathcal{S}$ sends $(s_i, \mathbf{auth}(s_i;a_{i,1},b_{i,1}),\ldots,$ $\mathbf{auth}(s_i;a_{i,2t+1},b_{i,2t+1}))$ over channel i, and sends $(a_{i,j},b_{i,j})$ over channel j for $1 \le j \le 2t+1$.

$\mathcal{R}$ recovers message:

1. For each share, $\mathcal{R}$ decides whether it is valid as follows.

   a. $\mathcal{R}$ receives $(s'_i,c'_{i,1},\ldots,c'_{i,2t+1})$ via channel i, and receives $(a'_{i,j},b'_{i,j})$ via channel j for each $1 \le j \le 2t+1$.
   b. $\mathcal{R}$ computes $k = |\{j : c'_{i,j} = \mathbf{auth}(s'_i;a'_{i,j},b'_{i,j})\}|$. If $k \ge t+1$, then $\mathcal{R}$ decides that s'_i is a valid share, and discards it otherwise.

2. Since the number of valid shares is $\ge t+1$, $\mathcal{R}$ randomly choose $t+1$ of them and recover the secret using the reconstruction method.

Fig. 10.8 The 1-round $(0,\delta)$-SMT protocol π_3 for $n \ge 2t+1$

and the substitution probability of the function is equal to $\frac{1}{q}$. A $(t+1)$-out-of-n threshold secret sharing scheme is a probabilistic function from $\mathbb{F} \to \mathbb{F}^n$ and satisfies that $t < n$ shares cannot get any information about the secret, but $t+1 \le n$ shares can recover the secret.

The protocol is given in Fig. 10.8.

References

1. T. Araki. Almost Secure 1-Round Message Transmission Scheme with Polynomial-time Message Decryption. In Proc. of ICITS, pages 2–13 (2008).
2. S. Agarwal, R. Cramer, and R. de Haan. Asymptotically Optimal Two-round Perfectly Secure Message Transmission. In CRYPTO, volume 4117 of LNCS, pages 394–408 (2006).
3. G. Bracha. An $O(\log n)$ Expected Rounds Randomized Byzantine Generals Protocol. In Journal of the ACM, 34(4): 910–920 (1987).
4. M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness Theorems for Non-cryptographic Fault-tolerant Distributed Computation (extended abstract). In Proc. of STOC, pages 1–10 (1988).
5. E. Berlekamp and L. Welch. Error correction of algebraic block codes.
6. D. Chaum, C. Crepeau, and I. Damgard. Multiparty Unconditionally Secure Protocols (Extended Abstract). In Proc. of FOCS, pages 11–19 (1988).
7. H. Chan, A. Perrig, and D. Song. Random Key Predistribution for Sensor Networks. In Proc. of IEEE Conference on Security and Privacy, pages 197–213 (2003).

8. D. Coppersmith and S. Winograd. Matrix Multiplication via Arithmetic Progressions. In Journal of Symbolic Computation 9(3): 251–280 (1990).
9. D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly Secure Message Transmission. In Journal of the ACM, 40(1):17–47 (1993).
10. Y. Desmedt, S. Erotokritou, and R. Safavi-Naini. Simple and Communication Complexity Efficient Almost Secure and Perfectly Secure Message Transmission Schemes. In Proc. of AFRICACRYPT, pages 166–183 (2010).
11. J. Deng and Y. Han. Multipath Key Establishment for Wireless Sensor Networks Using Just Enough Redundancy Transmission. In IEEE Transactions on Dependable and Secure Computing, Vol. 5, pages 177–190 (2008).
12. Y. Desmedt and Y. Wang. Perfectly Secure Message Transmission Revisited. In Proc. of EUROCRYPT, LNCS 2332, pages 502–517 (2002).
13. L. Eschenauer and V. Gligor. A Key-management Scheme for Distributed Sensor Networks. In Proceedings of the 9th ACM Conference on Computer and Communication Security, pages 41–47 (2002).
14. M. Fitzi, M. Franklin, J. Garay, and S. H. Vardhan. Towards Optimal and Efficient Perfectly Secure Message Transmission. In Proc. of TCC, LNCS 4392, pages 311–322 (2007).
15. M. K. Franklin and R. N. Wright. Secure Communication in Minimal Connectivity Models. In Journal of Cryptology, 13(1):9–30 (2000).
16. D. Huang and D. Medhi. A Byzantine Resilient Multi-path Key Establishment Scheme and its Robustness Analysis for Sensor Networks. Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 12 - Volume 13, pages 240.2 (2005).
17. K. Kurosawa and K. Suzuki. Almost Secure (1-round, n-channel) Message Transmission Scheme. IEICE Transactions 92-A(1): 105–112 (2009).
18. K. Kurosawa and K. Suzuki. Truly Efficient 2-round Perfectly Secure Message Transmission Scheme. In Proc. of EUROCRYPT, volume 4965 of LNCS, pages 324–340 (2008).
19. A. Patra, A. Choudhary, K. Srinathan, and C. Rangan. Unconditionally Reliable and Secure Message Transmission in Undirected Synchronous Networks: Possibility, Feasibility and Optimality. In IJACT 2(2): pages 159–197 (2010).
20. T. Rabin and M. Ben-Or. Verifiable Secret Sharing and Multiparty Protocols with Honest Majority (Extended Abstract). In Proc. of STOC, pages 73–85 (1989).
21. A. Shamir. How to Share a Secret. Commun. ACM, 22(11):612–613 (1979).
22. H. M. Sayeed and H. Abu-Amara. Efficient Perfectly Secure Message Transmission in Synchronous Networks. In *Inf. Comput.*, 126(1):53–61 (1996).
23. K. Srinathan, A. Narayanan, and C. P. Rangan. Optimal Perfectly Secure Message Transmission. In Proc. of CRYPTO, volume 3152 of LNCS, Springer, pages 545–561 (2004).
24. R. Safavi-Naini, M. A. A. Tuhin, and H. Shi. Optimal Message Transmission Protocols with Flexible Parameters. In Proceedings of ACM Symposium on Information, Computer and Communications Security, pages 453–458 (2011).
25. M. A. A. Tuhin and R. Safavi-Naini. Optimal One Round Almost Perfectly Secure Message Transmission. In Proceedings of Financial Cryptography and Data Security, pages 173–181 (2011).
26. Y. Wang. Robust Key Establishment in Sensor Networks. In SIGMOD Record 33(1): 14–19 (2004).
27. J. Wu and D. R. Stinson. Three Improved Algorithms for Multi-path Key Establishment in Sensor Networks Using Protocols for Secure Message Transmission. In IEEE Trans. Dependable Sec. Comput. 8(6): 929–937 (2011).
28. S. Zhu, S. Xu, S. Setia, and S. Jajodia. Establishing Pairwise Keys for Secure Communication in Ad hoc Networks: A Probabilistic Approach. In ICNP, IEEE Computer Society, pages 326–335 (2003).

Part III
Social Networks

Chapter 11
Mathematical Modelling to Evaluate Measures and Control the Spread of Illicit Drug Use

Afsaneh Bakhtiari and Alexander Rutherford

Abstract Millions of street-involved-youth worldwide are vulnerable to using and trading illicit drugs, which also place this group at high risk of drug-related criminality and health problems. It is often the case that drug users begin trafficking under the social influences within the drug culture to generate income for supporting their drug habits. The relative merits of behavioural (primary) or law enforcement (secondary) interventions for controlling the spread of drug use are widely debated. In this paper, we develop a network model to evaluate the effectiveness of modelling strategies. A network model with traffickers, current drug users and potential users is constructed. Traffickers exert social influence on current users to deal drugs and on potential users to initiate drug use. Primary intervention prevents potential users from initiating drug use while secondary intervention acts to reduce initiation into trafficking. To accomplish this, we vary the hypothetical social influence parameters in the model. Next, we analyze the properties of this system using dynamical system methods including mean field approximation (MFA), fixed point theory and bifurcation analysis. Furthermore, to evaluate the relative effectiveness of the two interventions, we study the properties of the phase transition between a drug-free and a drug-endemic state at equilibrium mathematically. Drug-free and drug-endemic states are separated by a curved phase transition. Via the shape of the phase transition curve we obtain the optimal intervention. Our findings confirm that a combination of primary and secondary interventions is the optimal intervention strategy. The optimal mixture of the two strategies depends on the relative numbers of drug users and traffickers.

A. Bakhtiari (✉)
Dalla Lana School of Public Health, University of Toronto, 155 College Street, Toronto, ON, M5T 3M7, Canada
e-mail: afsaneh.bakhtiari@utoronto.ca

A. Rutherford
Complex System Modeling Group, IRMACS, Simon Fraser University, 8888 University Drive, Burnaby, BC, V5A 1S6, Canada
e-mail: sandyr@irmacs.sfu.ca

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_11,

11.1 Introduction

Drug dependency is known as the main motivation for drug users to participate in the illicit drug trade and become drug dealers [10, 16]. Illicit drug use cause harm indirectly as well as directly. In fact, indirect negative effects of drug use are amplified disproportionally and far outweigh direct negative effects [11, 12]. There have been a number of studies to explore the risk of a future rise of drug related criminality, loss of productivity, infectious diseases (HIV/AIDS and hepatitis) and mortality among drug users [5, 8, 15].

The need for a proper intervention and a follow-up strategy as well as evaluating the results of such strategy seem remarkably clear. In Sect. 11.2, we construct a mathematical model to understand the impact of social influence among drug users and traffickers within a community. We consider three types of individuals and describe them as *susceptibles, light drug users* and *dealers*. Dealers are assumed to constantly employ drug users, who are in need to generate income to pay for their own drug use. This model is motivated by Werb et al. [16]. Furthermore, Rossi's compartmental Mover-Stayer model [13] for the epidemic of problematic drug use can be used to point out the importance of social influences in spreading drug use behaviour.

The interactions between the three types of individuals in our model can be viewed as a many-body system [14]. A many-body system with interactions is generally difficult to solve. The Mean Field Approximation (MFA) replaces an n-body problem with a chosen external field, ignoring space dependence and neglects local correlations. The external field replaces the interaction of all the other particles to an arbitrary particle.

In Sect. 11.4, we develop a social network model to study the impact of social influence among drug users and perform MFA to compare the analytical and simulation results for the network model.

MFA provides a good approximation near the bifurcation for network models. This is because the nodes are highly correlated near the bifurcation and the global effects dominate over the local effects. Thus, MFA can provide a good approximation for the behaviour of the system. MFA estimates the density and other properties of network. One way to get an estimation is to make the assumption that the type of a given node (individual) at each time step is completely random. With this assumption, if the overall density of light users at the particular step is p, then each node at that step should independently have probability p of being a light user. Using the same method, the probabilities for all possible configuration of n neighbourhood can be calculated [7, 17]. The more expanded version of this research is described in [1].

MFA is used as a point of comparison between the analytical results and simulations of the network model. Many researchers [2, 3, 6] have used MFA as a point of comparison with the network models they develop. Kleczkowski et al. in [9] proposed an epidemic model using MFA examining the spread of childhood measles. The comparison shows how accurately the MFA estimates the behaviour of the network model near the bifurcation. However, differences between MFA and exact limiting densities can be expected because of the presence of correlations in actual network. The mean field approximation typically estimates the exact limiting density within 10–20 % error [7].

Our model described in Sect. 11.2 allows us to compare the potential effectiveness of different types of responses to the drug epidemic using both simulation and MFA. In this model, we mostly refer to the "Primary Prevention Intervention" as advertising, e.g. health promotions towards susceptibles. A "Secondary Prevention Intervention" includes law enforcement towards drug users/dealers, e.g. police involvement. Contrary to Rossi's conclusion in [13] that primary prevention intervention is the most effective strategy to control drug use in Italy, we find that both prevention interventions are effective on their own, but combined interventions proved to be most effective when taking into account the sensitivity analysis of the parameters in the model.

11.2 Model

Figure 11.1 describes the main features of the proposed model. In this model, the population is divided into three subgroups: susceptibles, light drug users individuals, and dealers. Susceptibles are individuals who have a potential to become a drug user by a contact with dealers. Susceptibles can stay susceptible or make a transition and become drug users. Light drug users individuals have already initiated some form of drug use. Light users are at the risk of becoming dealers through contacts with a pusher operating in the black market. A light user can either stop using drugs or start a drug dealing career. Dealers are assumed to constantly employ drug users, who are in need to generate income to pay for their own drug use.

The constant-population restriction is in place, in other words individuals who die are reborn as susceptibles. The model assumes no drug using discouragement, However drug using and dealing encouragement only comes from dealer individuals in the model.

This model is motivated by a cohort study by Werb et al. [16] at the BC Centre for Excellence in HIV/AIDS, 2008. This is a scale-free network model which can be used to point out the importance of social influences in spreading drug use behaviour. To address individuals variation in behaviour we study the phase diagram of the network model using both simulation and MFA.

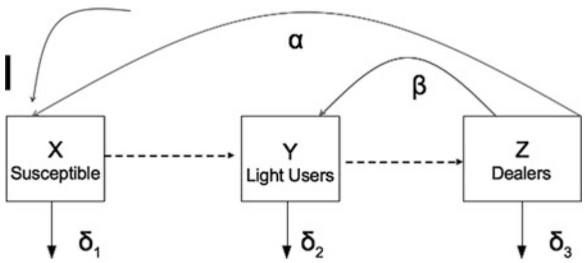


Fig. 11.1 The network model of the social influence among drug users and traffickers

11.3 Degree Distribution in the Scale Free Network Model

In a scale-free network, the distribution is approximation of the number of links that each node has (degree) is approximately given by $p_n \equiv \Pr\{\text{links} = n\} \sim p_1 n^{-\gamma}$, where p_1 is the normalizing constant $p_1 \approx (\sum_{n=1}^{\infty} n^{-\gamma})^{-1} = \frac{1}{\zeta(\gamma)}$, $\zeta(\gamma)$ is the Riemann Zeta function and γ is a parameter whose value is typically in the range $2 < \gamma < 3$, although occasionally it may lie outside these bounds, see [4]. The distribution is an approximation in the sense that it holds true for large values of n. The mean number of links $(E(n))$ in such a network is approximately

$$E(n) = \sum_{n=1}^{\infty} n p_n \approx p_1 \sum_{n=1}^{\infty} n \cdot n^{-\gamma} = p_1 \sum_{n=1}^{\infty} n^{-(\gamma-1)} = \frac{\zeta(\gamma-1)}{\zeta(\gamma)}.$$

Assuming a drug user community network can be modeled with a scale-free network with decay coefficient $\gamma = 2.1$, then on average each drug dealer interacts with approximately seven other individuals.

11.4 Mean Field Approximation of the Model

First, we list the main assumptions that we have made in this model. We consider no eligibility criteria for light drug users to become a dealer. In other words, any light drug user individual has the same probability of becoming a dealer as any other. Transition probability is a constant for all time and individuals. The population size is constant. This is achieved by assuming that all individuals who die are reborn as a susceptibles. In addition, it is assumed that the new individual inherits the network of relationships (links) that the dead individual had. The network of individuals has a scale-free structure which does not change with time. All individuals have the same behavioral influence on one another in the model and life expectancy is exponentially distributed. Dealer individuals are the only group that exert behavioural influence on their peers; i.e. both susceptible and light drug user individuals receive encouragement to either use or deal drugs.

Here we represent the MFA for the network model corresponding to the model depicted in Fig. 11.1. The transition probabilities of states for each node has n links (peers) with probability p_n are given by:

$$\begin{aligned}
&X \longrightarrow X \;\; \delta_1 X \\
&Z \longrightarrow X \;\; \delta_3 Z \\
&Y \longrightarrow X \;\; \delta_2 Y \\
&Y \longrightarrow Z \;\; (1-\delta_2)Y \sum_{n=1}^{\infty}\sum_{k=1}^{n} p_n \binom{n}{k} \left[1-(1-\beta)^k\right] Z^k (X+Y)^{n-k} \\
&X \longrightarrow Y \;\; (1-\delta_1)X \sum_{n=1}^{\infty}\sum_{k=1}^{n} p_n \binom{n}{k} \left[1-(1-\alpha)^k\right] Z^k (X+Y)^{n-k}
\end{aligned}$$

The transition probabilities for the first three cases are easy to obtain as they do not involve interaction with neighbours. The transition probability of $Y \longrightarrow Z$ can be obtained as follows: assume that the node under consideration has n neighbours which k neighbours are of type Z. The probability of this happening is $\binom{n}{k} Z^k (X + Y)^{n-k}$. The probability that at least one of the interactions between the current Y node and one of the k neighbouring Z nodes leads to a type conversion is $1-(1-\beta)^k$. However, this transition can only occur if the node is alive at the end of the period corresponding to the given iteration; the probability of this being true is $1-\delta_2$. Summing over all the possibilities for k and conditioning on the probability that the node has n neighbours gives the expression above. Similarly, the transition probability for the $X \longrightarrow Y$ conversion can be obtained.

Note that $p_n \sim \zeta(\gamma)^{-1} n^{-\gamma}$, where γ is the decay coefficient characterizing the scale-free network. We now obtain the approximation to the MFA by dropping all terms of higher than second order in X, Y, or Z. This gives

$$Z \longrightarrow X \ \delta_3 Z$$

$$Y \longrightarrow X \ \delta_2 Y$$

$$Y \longrightarrow Z \ (1-\delta_2) \sum_{n=1}^{\infty} \zeta(\gamma)^{-1} n^{-\gamma} \cdot (n\beta YZ) = \beta YZ \frac{\zeta(\gamma-1)}{\zeta(\gamma)}$$

$$X \longrightarrow Y \ (1-\delta_1) \sum_{n=1}^{\infty} \zeta(\gamma)^{-1} n^{-\gamma} \cdot (n\alpha XZ) = \alpha XZ \frac{\zeta(\gamma-1)}{\zeta(\gamma)}$$

Let us define $\eta \equiv \frac{\zeta(\gamma-1)}{\zeta(\gamma)}$ as the mean node degree. Also, introduce the rescaling $\eta\alpha \to \alpha$ and $\eta\beta \to \beta$. Thus, the ODE system corresponding to the approximated MFA becomes

$$\begin{aligned} X' &= \delta_2 Y + \delta_3 Z - \alpha(1-\delta_1) XZ \\ Y' &= \alpha(1-\delta_1) XZ - \delta_2 Y - \beta(1-\delta_2) YZ \\ Z' &= \beta(1-\delta_2) YZ - \delta_3 Z \end{aligned}$$

The same set of equations is derived from the compartmental model. Thus, the approximated MFA of the network model is the same as the compartmental model if we use the rescaling $\eta(1-\delta_1)\alpha \to \alpha$ and $\eta(1-\delta_2)\beta \to \beta$. Using constant population restriction $Z = 1 - X - Y$, and letting $\delta_1 = \delta_2 = \delta$ the ODE system can be reduced to:

$$\begin{aligned} X' &= \delta Y + \delta(1-X-Y) - \alpha X(1-X-Y) \\ Y' &= \alpha X(1-X-Y) - \delta Y - \beta Y(1-X-Y) \end{aligned} \tag{11.1}$$

Assuming the parameters in the above ODE system are rates rather than probabilities, the system can be solved for a wide range of parameters. We are mainly interested in the effect of α and β (behavioural influence). After simplification we have the following system of equations:

$$\begin{aligned} X' &= \delta - (\delta+\alpha)X + \alpha XY + \alpha X^2 \\ Y' &= \alpha X - (\beta+\delta)Y - \alpha X^2 + \beta Y^2 + (\beta-\alpha)XY \end{aligned} \tag{11.2}$$

11.4.1 Fixed Points of the System of Equations

Fixed points of the system of Eqs. (11.2) are as follow.

Clearly, $[x_1 = 1, y_1 = 0]$ is a trivial solution of the above system. Let $a := \beta\alpha - \delta\alpha + \beta\delta$. Then the two other nontrivial solutions are as follows:

$$[x_2 = \frac{a+\sqrt{a^2-4\beta^2\alpha\delta}}{2\beta\alpha}, y_2 = \frac{\delta}{\beta}], \quad [x_3 = \frac{a-\sqrt{a^2-4\beta^2\alpha\delta}}{2\beta\alpha}, y_3 = \frac{\delta}{\beta}].$$

Note that the second and third solutions exist if and only if:

$$\alpha_1 = \frac{(\delta+\beta+2\sqrt{\beta\delta})\delta\beta}{\beta^2-2\beta\delta+\delta^2}, \text{and} \quad \alpha_2 = -\frac{(-\delta-\beta+2\sqrt{\beta\delta})\delta\beta}{\beta^2-2\beta\delta+\delta^2}.$$

In order to approximate the *phase portrait* near the fixed points we linearize the system using the Jacobian around the fixed points. Let Δ and τ denote the determinant and trace of the *Jacobian* evaluated at a fixed point, respectively.

11.4.1.1 First Fixed Point

We linearize the system around $[x_1 = 1, y_1 = 0]$ and find that if $\alpha < \alpha_1$ then $[x_1 = 1, y_1 = 0]$ is a stable node and if $\alpha > \alpha_2$ then this fixed point is a stable spiral.

11.4.1.2 Second Fixed Point

Now we linearise the system around $[x_2, y_2]$. Let $b = \beta\alpha - \delta\alpha - \beta\delta$. The determinant of the Jacobin matrix is: $\Delta = \frac{-1}{2\alpha\beta}\left(\sqrt{b^2-4\delta^2\alpha\beta}(b-\sqrt{b^2-4\delta^2\alpha\beta})\right)$.

- If $b > 0$ then $b - \sqrt{b^2-4\delta^2\alpha\beta} > 0$. So, $\Delta < 0$. This suggests that the fixed point is Saddle.

- If $b < 0$ then $b - \sqrt{b^2 - 4\delta^2\alpha\beta} < 0$. So, $\Delta > 0$. In this case we need to further investigate τ. We further find that $\tau < 0$ and so $\tau^2 - 4\Delta > 0$. Thus the fixed point is a Stable Node.

11.4.1.3 Third Fixed Point

The analysis of this case is very similar to the previous case. We linearize the system using Jacobian around the third fixed point.

- If $b > 0$ then $b - \sqrt{b^2 - 4\delta^2\alpha\beta} > 0$. So, $\Delta > 0$. Now, we need to look at the trace of the matrix evaluated at this fixed point. Also, we find $\tau < 0$ and $\tau^2 - 4\Delta > 0$. Thus, the fixed point is a Stable Node.
- If $b < 0$ then $b - \sqrt{b^2 - 4\delta^2\alpha\beta} < 0$. Hence, $\Delta < 0$. This suggests that the fixed point is a Saddle. So there must be a bifurcation happening at $b = 0$.

11.5 Bifurcation When b Varies

In the previous section we investigated the stability of the fixed points. We learned that the stability of the two nontrivial fixed points $[x_2, y_2]$ and $[x_3, y_3]$ depends on positivity or negativity of expression $b = \beta\alpha - \delta\alpha - \beta\delta$.

- If $b > 0$, that is $\alpha(\beta - \delta) > \beta\delta$, then the second fixed point is Saddle and a Stable Node, otherwise. So this situation occurs if $\beta > \delta$.
- On the other hand, if $b < 0$, that is $\alpha(\beta - \delta) < \beta\delta$, then the third fixed point is Saddle and a Stable Node, So this can happen only if $\beta \leq \delta$.

We solve for β in $b = 0$ and obtain the bifurcation curve where $\beta = \frac{\alpha\delta}{-\delta+\alpha}$. Note that bifurcation happens only at the points (α_0, β_0) and the points (α_0, β_0) below the curve correspond to case when $b < 0$. Similarly the points above the curve correspond to $b > 0$. This means for those (α_0, β_0) points below the phase transition curve, drug problem can be eliminated. Given that the trivial fixed point $[x_1, y_1]$ and $[x_3, y_3]$ become stable and unstable fixed point, respectively.

11.6 Simulation Result

The scale free network model in Fig. 11.1 mimics the spread of drug use in a community. Each node can be in one of a finite number of states 0, 1 or 2 which correspond to the three defined types susceptible, light drug users and dealers. Each node represents an individual interacting with the network and its status gets updated according to the interaction rules. Dealers are assumed to exert influence on susceptibles and light users with the probability of success α and β, respectively.

Table 11.1 Description of experiment and parameters

Model type	Scale free network with average of four nodes	
Parameters		
δ_1	0.17 % per month	Mortality rate of susceptibles compartment
δ_2	0.28 % per month	Mortality rate in drug users compartment
δ_3	0.42 % per month	Mortality rate in dealers compartment
α	[0,0.01] per month	Probability of becoming drug users
β	[0,0.01] per month	Probability of becoming dealers
Initial proportions		
Susceptibles	40 %	
Light drug users	40 %	
Dealers	20 %	
Simulation conditions		
Population size	400	
Stopping condition	3,000 iterations	
Repetitions	10	

Note that, transitions are defined as independent probabilistic events. Dealers are assumed to act independently with certain probability to convince individual to use or deal drugs. For instance, If $\#R_2(t)$ is the number of type 2 (dealer) nodes at time t then the change in states is given by a binomial random variable for $\#R_2(t)$ trials with a probability of success either α or β. For example, the transition from type $0 \rightarrow 1$ are given by X for $\#R_2(t)$ trials with a probability of successes α. The model is run for a certain number of time steps, with the state of nodes being updated at every time step.

This model employs a novel approach to represent the effect of prolonged social relationships between members of a community where dealing drug is prevalent. In such an environment, a person is more likely to experiment drug use if in a lengthly relationship with at least one dealer. In the model, dealers in the neighbourhood can act independently with certain probability to convince a susceptible or a light user to use or deal drugs. Table 11.1 specifies valued used for the parameters in the simulation, unless they are varied as in the case of α and β. Figure 11.2 shows the results with the network model simulation.

11.7 Comparison Between Analytical Result and Simulation Result

In this section we investigate whether the simulation produces similar results to the ODE system derived from the MFA. As we explained in Sect. 11.4, we can obtain the same set of equation from the compartmental model if we make the following identification, $\alpha = \eta\alpha(1-\delta)$ and $\beta = \eta\beta(1-\delta)$. Therefore, we substitute $\alpha = \frac{\alpha}{\eta(1-\delta)}$

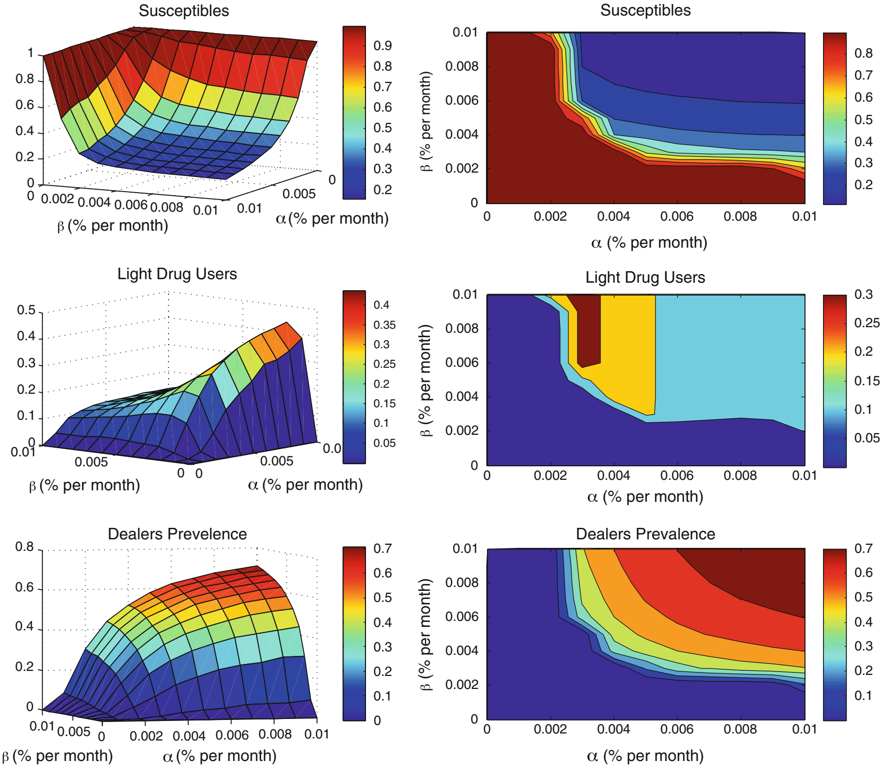


Fig. 11.2 *Left*: Network model results. *Right*: Compartmental model results

and $\beta = \frac{\beta}{\eta(1-\delta)}$ in the bifurcation curve equation $\beta = \frac{\alpha\delta}{-\delta+\alpha}$, where the mean node degree is $\eta = 4$. We then sketch the bifurcation curve obtained from analyzing the ODE system against the steady state prevalences of drug dealers in the network model. The goal is to investigate whether the simulation produces similar results to the ODE system derived from the MFA.

We are interested in how well the ODE system approximates the simulation result for the network model. Figure 11.3 shows that the MFA generally provides a good approximation for the network model near the bifurcation.

11.8 Conclusions

Our observations in this study show that the social influence can dramatically affect the spread of drug use in the community. We should point out here that our results not only come from the network model simulation, but also from a deterministic system which provides a good approximations for predicting drug use epidemic.

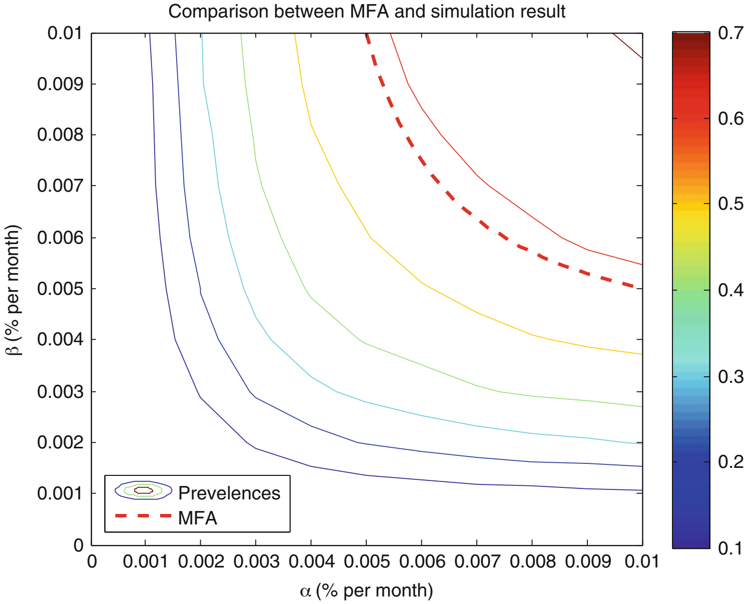


Fig. 11.3 Bifurcation curve

By modelling the interaction among individuals, we are able to understand the role of social influence in this dynamics without the need for complex mathematical methods. Whether an individual can successfully be convinced to begin or persist drug use in the long term depends on the interactions that exist in the neighbourhood. In the following, we identify the main result of this study.

To summarize, from simulating the interaction among individuals who are randomly distributed in the network, we identify different effects of the parameters in the model. Although the short term behaviour of the model may depend on the network size and details of the degree distribution, the long term behaviour mainly depends on three parameters: the social influences of dealer on susceptibles (α), on drug users (β), and the life expectancies (δ). We change parameters α and β, which we believe can affect the spread of drug use in the community. Reduction for α can be viewed as the indirect strategy (health promotion) and reducing β implies the direct strategy (police involvement). In Fig. 11.4 drug-free and drug-endemic states were separated by a curved phase transition (bifurcation curve). The phase transition curve demonstrates that reduction for α and β is effective on their own but reducing both simultaneously results in faster transitions to the drug free state. In other words the combination of the indirect and direct strategies are shown to be most effective. Note that the green and red arrows in Fig. 11.4 demonstrates the reduction of α and β, towards the phase transition curve, respectively.

Due to the complexity of the interactions among drug users, it is difficult to analyze the precise impact of social influences on drug use spread in the network model. So an approximate solution is obtained by performing the mean

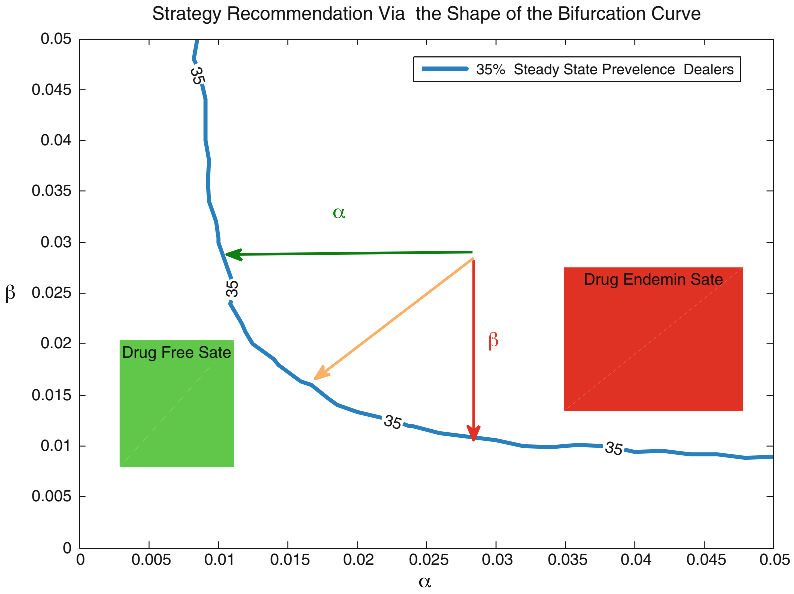


Fig. 11.4 The effect of combined strategies

field approximation for the network model. MFA results are used as a point of comparison for the network model. The comparison is between the analytical results and simulation results of the network model. We have demonstrated in Fig. 11.3 that MFA up to second degree in densities provides a good picture of network model near the bifurcation.

In this study we have only considered three states in the network. There are other states that need to be taken in to account in order to determine the prevalences such as clients of the health care services, recidivist drug users and no users (temporary). In addition to include these states, the transitions between them also need to be considered. Moreover the constant population assumption in the model simplifies the complexity dramatically. In the next phase of this study, we would like to consider a non-constant population to make the model more realistic. Finally, it is of interest to investigate the policy by which the drug use interventions are applied to the communities and suggest policies based on our findings to authorities.

References

1. A. Bakhtiari, Social Influences Among Drug users and Mean Field Approximation of Cellular Automata, Masters Thesis, (Simon Fraser University, 2009).
2. N. Boccara, and K. Cheong, Critical behaviour of a probabilistic automata network SIS model for the spread of an infectious disease in a population of moving individuals. *Journal of Physics A: Mathematical and General*. 26, 5 (1993), 3707–3717.

3. N. Boccara, K. Cheong, and M. Oram, A probabilistic automata network epidemic model with births and deaths exhibiting cyclic behaviour. *Journal of Physics A: Mathematical and General*. 27 (1994), 1585–1597.
4. A. Clauset, C.R. Shalizi, M. E. J. Newman, Power-law distributions in empirical data, *SIAM Review* 51, 661–703 (2009).
5. D.C. Des Jarlais, Preventing HIV infection among injection drug users: Intuitive and counter-intuitive findings.*Applied & Preventive Psychology*, (1999), 8:63–70.
6. M. Duryea, T. Caraco, G. Gardner, W. Maniatty, and B.K. Szymanski, Population dispersion and equilibrium infection frequency in a spatial epidemic.*Physica D* 132, 4 (1999), 511–519.
7. A. Ilachinski, Cellullar Automata, Discrete Universe *World Scientific Publishing co., Inc*; (2001).
8. M. Kertzschmar and L.G. Wiessing, Modeling the spread of HIV in social networks of injection drug users, *AIDS*, (1998), 12:801–811.
9. A. Kleczkowski, and B.T. Grenfell, Mean-field-type equations for spread of epidemics: The small world model, *Physica A*. 274, 1–2 (1999), 355–360.
10. B. Maas, N. Fairbairn, T. Kerr, K. Li, J. Montaner and E. Wood, Neighbourhood and HIV infection among IDU: place of residence independently predicts HIV infection among a cohort of injection drug users, *Health & Place* (2007), 13, 432–439.
11. B.M. Mathers, L. Degenhardt, B. Phillips, L. Wiessing, M. Hickman, S.A. Strathdee, A. Wodak, S. Panda, M. Tyndall, A. Touk, R.P. Mattick, Global epidemiology of injecting drug use and HIV among people who inject drugs: a systematic review. *for the 2007 Reference Group to the UN on HIV and Injecting Drug Use: www.thelancet.com*. (2008); vol 372, November 15.
12. D. Riley, Drugs and Drug Policy in Canada: A Brief Review & Commentary *Canadian Foundation for Drug Policy & International Harm Reduction Association*.
13. C. Rossi, Operational models for epidemics of problematic drug use: the Mover-Stayer approach to heterogeneity, *Socio-Economic Planning Sciences* 38 (2004) 73–90.
14. S. Jenkins, Many Body Theory of Solids (Plenum, New York, 1984).
15. A. Taylor, M. Frischer, and R. Covell, Reduction in needle sharing among injection drug users, *International conference AIDS* (1992), July 19–24.
16. D. Werb, T. Kerr, K. Li, J. Montaner, and E. Wood, surrounding drug trade involvement among street-involved youth, *American Journal of Drug and Alcohol Abuse*, (2008); 34(6): 810–1820.
17. S. Wolfram, A New Kind of Science, *Wolfram Media*, (2002).

Chapter 12
Complex Networks and Social Networks

Anthony Bonato and Yanhua Tian

Abstract We give an overview of the properties and models for complex networks, with particular emphasis on models of on-line social networks.

12.1 Introduction

Complex networks arise in many diverse contexts, ranging from web pages and their links, protein-protein interaction networks, and social networks. The modelling and mining of these large-scale, self-organizing systems is a broad effort spanning many disciplines. A number of common properties have been observed in complex networks, such as power law degree distributions and the small world property (see Sect. 12.2 for further background on these properties).

While classical binomial random graphs form a well studied field in their own right, in the last decade we have seen a wealth of new random graph models modelling complex networks. These stochastic graph models simulate properties of complex networks, but also expanding our theoretical understanding of random graphs. Models for complex networks also give insight into the underlying generative properties of complex networks, and can serve as a predictive tool in their evolution.

With the current popularity of *on-line social networks* (or *OSNs*) such as Facebook, LinkedIn, and Twitter, there is an increasing interest in their measurement and modelling. In addition to other complex networks properties, OSNs exhibit shrinking distances over time, increasing average degree, and bad spectral expansion. Unlike other complex networks such as the web graph, models for OSNs are relatively new and lesser known. In on-line social networks, models may help detect and classify communities, and better clarify how news and gossip is spread in social networks.

A. Bonato (✉) • Y. Tian
Department of Mathematics, Ryerson University, Toronto, ON, M5B 2K3, Canada
e-mail: abonato@ryerson.ca; yanhua.tian@ryerson.ca

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_12,

We will survey the properties of complex networks and their models, focusing on the case of OSNs. A more detailed survey of complex networks (without the focus on OSNs) may be found in the book [3]. In Sect. 12.2 we give a brief overview of some the main observed properties of OSNs. These are the key properties that network models attempt to simulate. We study various complex network models in Sect. 12.3. Our focus is on five models, each rigorously formulated and analyzed: Kronecker graphs [22, 23], the MAG model [17], the $G(Q,E)$ affiliation graph model [21], the ILT model [4], and the GEO-P model [6]. These models focus primarily on simulating properties of social networks, and are relatively recent. We finish with a list of open problems surrounding the modelling of OSNs.

12.2 Properties of Complex Networks

Researchers are now in the enviable position of observing how OSNs evolve over time, and as such, network analysis and models of OSNs typically incorporate time as a parameter. Unlike in traditional social network analysis, we can now mine the social interactions of millions of people from across the globe. While by no means exhaustive, some of the main observed properties of OSNs include the following. For definitions of the terms used below (such as diameter, clustering coefficient, etc), see [3].

(i) *Large-scale.* OSNs are examples of complex networks with number of nodes (which we write as n) often in the millions; further, some users have disproportionately high degrees. About half a billion users are registered on Facebook [11]. Some of the nodes of Twitter corresponding to well-known celebrities including Lady Gaga and Justin Bieber have degree over ten million [32].

(ii) *Small world property and shrinking distances.* The small world property, introduced by Watts and Strogatz [33], is a central notion in the study of complex networks (see also [18]). The small world property demands a low diameter of $O(\log n)$, and a higher clustering coefficient than found in a binomial random graph with the same number of nodes and same average degree. Adamic et al. [1] provided an early study of an OSN at Stanford University, and found that the network has the small world property. Similar results were found in [2] which studied Cyworld, MySpace, and Orkut, and in [28] which examined data collected from Flickr, YouTube, LiveJournal, and Orkut. Low diameter (of 6) and high clustering coefficient were reported in the Twitter by both Java et al. [16] and Kwak et al. [20]. Kumar et al. [19] reported that in Flickr and Yahoo!360 the diameter actually decreases over time. Similar results were reported for Cyworld in [2]. Well-known models for complex networks such as preferential attachment or copying models have logarithmically growing diameters with time.

(iii) *Power law degree distributions.* In a graph G of order n, let $N_k = N_k(n)$ be the number of nodes of degree k. The degree distribution of G follows a *power law* if N_k is proportional to k^{-b}, for a fixed exponent $b > 2$ and some range of k. Power laws were observed over a decade ago in subgraphs sampled from the web graph, and are ubiquitous properties of complex networks (see Chap. 2 of [3]). Kumar et al. [19] studied the evolution of Flickr and Yahoo!360, and found that these networks exhibit power-law degree distributions. Power law degree distributions for both the in- and out-degree distributions were documented in Flickr, YouTube, LiveJournal, and Orkut [28], as well as in Twitter [16, 20].

(iv) *Bad spectral expansion.* Social networks often organize into separate clusters in which the intra-cluster links are significantly higher than the number of inter-cluster links. In particular, social networks contain communities (characteristic of social organization), where tightly knit groups correspond to the clusters [29]. As a result, it is reported in [10] that social networks, unlike other complex networks, possess bad spectral expansion properties realized by small gaps between the first and second eigenvalues of their adjacency matrices.

(v) *Bad compressibility.* A recent study of [7] contrasts the compressibility of OSNs with the web graph. Assume that the vertex set of the digraph G is given by $[n] = \{1, 2, \ldots, n\}$. The so-called *minimum logarithmic arrangement* or *MLOGA problem*, is to find a permutation $\pi : V(G) \to [n]$ such that the term

$$\sum_{(u,v)\in E} \log|\pi(u) - \pi(v)| \tag{12.1}$$

is minimized. The motivating idea is minimize the sum of the edge lengths according to the ordering of vertices. The cost (12.1) represents the compression size in an encoding that is nearly informational-theoretically optimal. While MLOGA is **NP**-hard [7], the authors of [7] introduce heuristics for its computation. Using data from LiveJournal and Flickr, it was found in [7] that the compression performance with different orderings were worse than that found in web graph samples. The lack of a natural ordering of social networks when compared say to the URL ordering of web pages may be the cause of the poor incompressibility. Nevertheless, bad compressibility appears to be another feature peculiar to OSNs when, say, contrasted with the web graph.

(vi) *Densification power law.* Let $(G_t : t \geq 0)$ be sequence of graphs such that G_t is an induced subgraph of G_{t+1} for all $t \geq 0$, and suppose that G_t has e_t edges and n_t nodes. The graph sequence satisfies a *densification power law* if there is a constant $a \in (1, 2)$ such that for sufficiently large t, e_t is proportional to n_t^a. We call a the *exponent* of the densification power law. In particular, the average degree of the network grows to infinity with the order of the network. In [22], densification power laws were reported in several real-world networks such as a physics citation graph and the internet graph at the level of autonomous systems. Densification power laws were found in Flickr and Yahoo!360 in [19].

12.3 Models of Complex Networks

In this chapter, we do not give an exhaustive overview of models for complex networks. A survey of such models may be found in Chap. 4 of the book [3]. We focus, rather, on the relatively new models for OSNs introduced over the last few years. We are content to survey the results here, pointing the reader to further details and proofs in the papers cited. Many properties of the models hold with probability tending to 1 as time (or the order of the graphs considered) tends to infinity; we say such properties hold *asymptotically almost surely*, or *aas*.

12.3.1 Kronecker Graphs

Kronecker graphs [22, 23] were one of the early successful models for complex networks with densification. Their definition relies on a certain well known graph product. Given graphs G and H, form the *categorical* (or *Kronecker*) *product* $G \times H$ by setting vertices to be pairs (a,b) with $a \in V(G)$ and $b \in V(H)$, and (a,b) joined to (c,d) if and only a is joined to c in G, and b is joined to d in H. See Fig. 12.1.

Let A and B be two real matrices, with sizes $n \times m$ and $n' \times m'$, respectively. The *Kronecker* (or *tensor*) *product* of A and B, is the matrix $A \otimes B$ with size $nn' \times mm'$ given by

$$\begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1m}B \\ a_{21}B & a_{22}B & \cdots & a_{2m}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}B & a_{n2}B & \cdots & a_{nm}B \end{pmatrix}.$$

If $A(G)$ is the adjacency matrix of G, then note that

$$A(G \times H) = A(G) \otimes A(H).$$

The Kronecker graphs are formed by forming the kth power G^k of G with respect to this product; we call G here the *initiator graph*. The motivation behind this definition is that to produce G^k from G^{k-1}, nodes of a community expand to copies of the community. Note that the Kronecker model is a deterministic one.

The authors prove the following theorem, which leads to power law graphs by the choice of the initiator graph.

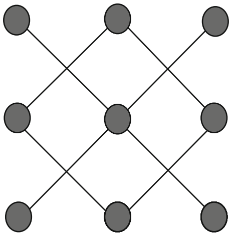


Fig. 12.1 The Kronecker product of the path with three vertices with itself

Theorem 12.1 ([22,23]). *Kronecker graphs have multinomial degree distributions.*

Further, Kronecker graphs satisfy a densification power law and constant diameter.

Theorem 12.2 ([22, 23]). *Kronecker graphs G^k satisfy a densification power law with exponent*

$$\log(|E(G)|)/\log(|V(G)|).$$

Further, if G is reflexive, then the diameter of G^k is the diameter of G.

The model is made tuneable by allowing the initial adjacency matrix $A(G)$ to have real entries in $[0,1]$. Hence, we may think of the initiator graph G as a probability space, where the probability there is an edge between i and j is the ij entry of $A(G)$ (although this is not exactly the case as the row (or column) sums may add up to a quantity greater than 1). Such *stochastic Kronecker graphs* with certain initiator graphs of order 2, were studied by Mahdian and Xu [27]. They studied the giant component of graphs generated by the model, and proved it *aas* has a constant diameter beyond the connectivity threshold.

It is shown in [24] that the Kronecker graph model with certain 2×2 initiator matrices is useful in simulating complex networks. Their work shows that certain stochastic Kronecker matrices fit samples of the web graph, the internet AS graph, Flickr, and certain biological networks. A fast, scalable algorithm KRONFIT was introduced to fit real network data to Kronecker graphs.

12.3.2 The ILT Model

The *Iterated Local Transitivity* (ILT) model [4], simulates OSNs and other complex networks. The central idea behind the ILT model is what sociologists call *transitivity*: if u is a friend of v, and v is a friend of w, then u is a friend of w (see, for example, [13,30,34]). In its simplest form, transitivity gives rise to the notion of *cloning*, where u is joined to all of the neighbours of v. In the ILT model, given some initial graph as a starting point, nodes are repeatedly added over time which clone *each* node, so that the new nodes form an independent set. The only parameter of the model is the initial graph G_0, which is any fixed finite connected graph. Assume that for a fixed $t \geq 0$, the graph G_t has been constructed. To form G_{t+1}, for each node $x \in V(G_t)$, add its *clone* x', such that x' is joined to x and all of its neighbours at time t. Note that the set of new nodes at time $t+1$ form an independent set of cardinality $|V(G_t)|$. As with Kronecker model, the ILT model is deterministic.

We write $\deg_t(x)$ for the degree of a node at time t, n_t for the order of G_t, and e_t for its number of edges. Define the *volume* of G_t by

$$\text{vol}(G_t) = \sum_{x \in V(G_t)} \deg_t(x) = 2e_t.$$

Theorem 12.3 ([4]). *For $t > 0$, the average degree of G_t equals*

$$\left(\frac{3}{2}\right)^t \left(\frac{\mathrm{vol}(G_0)}{n_0} + 2\right) - 2.$$

Note that Theorem 12.3 supplies a densification power law with exponent $a = \frac{\log 3}{\log 2} \approx 1.58$.

Define the *Wiener index* of G_t as

$$W(G_t) = \frac{1}{2} \sum_{x,y \in V(G_t)} d(x,y).$$

The Wiener index may be used to define the *average distance* of G_t as

$$L(G_t) = \frac{W(G_t)}{\binom{n_t}{2}}.$$

Theorem 12.4 ([4]). *For $t > 0$,*

$$L(G_t) = \frac{4^t \left(W(G_0) + (e_0 + n_0)\left(1 - \left(\frac{3}{4}\right)^t\right)\right)}{4^t n_0^2 - 2^t n_0}.$$

Note that the average distance of G_t is bounded above by $\mathrm{diam}(G_0) + 1$ (in fact, by $\mathrm{diam}(G_0)$ in all cases except cliques). Further, for many initial graphs G_0 (such as large cycles) the average distance decreases.

The clustering coefficient of the graph at time t generated by the ILT model is estimated as follows.

Theorem 12.5 ([4]).

$$\Omega\left(\left(\frac{7}{8}\right)^t t^{-2}\right) = C(G_t) = O\left(\left(\frac{7}{8}\right)^t t^2\right).$$

Observe that $C(G_t)$ tends to 0 as $t \to \infty$. If we let $n_t = n$ (so $t \sim \log_2 n$), then this gives that

$$C(G_t) = n^{\log_2(7/8) + o(1)}.$$

In contrast, for a random graph $G(n,p)$ with comparable average degree

$$pn = \Theta((3/2)^{\log_2 n}) = \Theta(n^{\log_2(3/2)})$$

as G_t, the clustering coefficient is $p = \Theta(n^{\log_2(3/4)})$ which tends to zero much faster than $C(G_t)$. (For a discussion of the clustering coefficient of $G(n,p)$, see Chap. 2 of [3].)

Let A denote the adjacency matrix and D denote the diagonal adjacency matrix of a graph G of order n. Then the normalized Laplacian of G is

$$\mathcal{L} = I - D^{-1/2} A D^{-1/2},$$

where I is the $n \times n$ identity matrix. Let $0 = \lambda_0 \leq \lambda_1 \cdots \leq \lambda_{n-1} \leq 2$ denote the eigenvalues of $\mathcal{L}$. The *spectral gap* of the normalized Laplacian is

$$\lambda = \max\{|\lambda_1 - 1|, |\lambda_{n-1} - 1|\}.$$

The following theorem suggests a significant spectral difference between graphs generated by the ILT model and random graphs. Define $\lambda(G_t)$ to be the spectral gap of the normalized Laplacian of G_t.

Theorem 12.6 ([4]). *For $t \geq 1$, $\lambda(G_t) > \frac{1}{2}$.*

Theorem 12.6 represents a drastic departure from the good expansion found in random graphs, where $\lambda = o(1)$ [8].

Let $\rho_0(t) \geq |\rho_1(t)| \geq \ldots$ denote the eigenvalues of the adjacency matrix of G_t. If A is the adjacency matrix of G_t, then the adjacency matrix of G_{t+1} is

$$M = \begin{pmatrix} A & A+I \\ A+I & 0 \end{pmatrix},$$

where I is the identity matrix of order n_t. We note the following recurrence for the eigenvalues of the adjacency matrix of G_t. As in the Laplacian case, there is a small spectral gap of the adjacency matrix.

Theorem 12.7 ([4]). *Let $\rho_0(t) \geq |\rho_1(t)| \geq \cdots \geq |\rho_{n-1}(t)|$ denote the eigenvalues of the adjacency matrix of G_t. Then*

$$\frac{\rho_0(t)}{|\rho_1(t)|} = \Theta(1).$$

That is, $\rho_1(t) \geq c|\rho_0(t)|$ for some constant $c > 0$. Theorem 12.7 is in contrast to the fact that in $G(n,p)$ random graphs, $|\rho_1| = o(\rho_0)$ (see [8]).

As shown in Theorem 12.3, the ILT model has a fixed densification exponent equalling $\log 3/\log 2$. A randomized version of the model, where edges are randomly added between new nodes, is presented in [4]. In the randomized model, the densification exponent is tuneable, and with high probability it generates graphs with the small world property and bad spectral expansion.

12.3.3 Affiliation Networks

In [21], a model for social networks was given by first introducing a bipartite model called *affiliation graphs*. Paths of length two in the affiliation graphs are *folded* onto edges to derive a model for social networks. The central thesis behind using a folded affiliation network is that friendships between members of social networks arise from common shared affiliations, such as sharing the same hobby or profession.

More precisely, the model evolves in two stages. First, a bipartite random graph model $B(Q,U)$ is introduced, with colours Q and U. For instance, Q represents a set of users, while U represents a set of groups of users. The parameters of the models are positive integers c_q and c_u, along with a probability $p \in (0,1)$. The model evolves over discrete time-steps. At time $t = 0$, the graph $B_0(Q,U)$ is a (deterministic) bipartite graph with at least $c_q c_u$ edges, so that each node in Q has degree at least c_q, while each node of U has degree at least c_u. At time $t > 0$, a new node q is added to Q. A node q' from Q is chosen proportional to its degree, and c_q neighbours of q' chosen uniformly at random (without replacement) become neighbours of q. Similarly, a node u is added to U with a similar copying process.

Now to define the (multi)graph $G(Q,U)$, the parameters of the models are positive integers c_q, c_u, and s along with a probability $p \in (0,1)$. At $t = 0$, $G_0(Q,U)$ is the set Q in $B_0(Q,U)$, and two nodes of Q have an edge between them for each common neighbour they share in U. At time $t > 0$ we do the following. With probability p a new node q is added to Q. The edges of q are determined in $B(Q,U)$, and edges are added between q and other nodes if they share common neighbours in U. With probability $1-p$, an edge is added between existing nodes q_1 and q_2 if they share as a common neighbour the new vertex u in U. A set of s nodes are chosen independently of each of other, proportionally by degree, and are joined to q.

It is proved in [21] that *aas* the degree distributions of the graphs generated by $G(Q,U)$ follows a power law. Further, if $c_u < \frac{p}{1-p}c_q$, then *aas* the graph $G(Q,U)$ is dense with $\omega(|Q|)$ many edges.

For a graph G, let R be the set of node pairs which are connected by a path. For $0 < q < 1$, define the *q-effective diameter of* G to be the minimum d such that, for at least $q|R|$ of node pairs in R, their distance is at most d. The $G(Q,U)$ exhibits low distances between nodes as made precise by the following theorem.

Theorem 12.8 ([21]). *For constants $m,q \in (0,1)$, if $c_u < \frac{p}{1-p}c_q$, then* aas *the q-effective diameter of the graph $G(Q,U)$ is non-increasing.*

12.3.4 The MAG Model

In the *Multiplicative Attribute Graph* (or *MAG*) model [17], nodes are assigned a set of attributes represented by a binary vectors. These could be viewed as answer to yes or no questions about the users interests or background. The MAG model accounts for *heterophily* (that is, love of the same) and *homophily* (that is, love of the different). More precisely, the MAG model $M(n,r,\mu,\Theta)$ has parameters equalling n the number of nodes, r the number of attributes of each node, μ the probability that an attribute takes the value of 1, and Θ the attribute affinity matrix

$$\begin{pmatrix} \alpha & \beta \\ \beta & \gamma \end{pmatrix},$$

where $\alpha > \beta > \gamma$ are fixed probabilities in $(0,1)$. We use the notation $\Theta_{00} = \alpha$, $\Theta_{01} = \Theta_{10} = \beta$, and $\Theta_{11} = \gamma$. Each node u is assigned a binary *attribute vector* $a(u)$ of length r; we denote the ith entry of $a(u)$ by $a_i(u)$. An independent and identically distributed Bernouilli distribution parameterized by μ is used to model the attribute vectors, where the probability that the ith attribute of a node is 1 is given by μ. The probability that nodes u and v are joined is given, independently, by

$$\prod_{i=1}^{r} \Theta_{a_i(u)a_i(v)}.$$

In particular, the ith entry of attribute vectors $a_i(u)$ and $a_i(v)$ selects the entry of the matrix Θ; for example, if $a_i(u) = 0$ and $a_i(v) = 0$, then $\Theta_{a_i(u)a_i(u)} = \alpha$. The product is then taken of all these entries. If the values on the diagonal of Θ are large, then the link probability is high when nodes share the same attributes. For instance, the matrix

$$\begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.8 \end{pmatrix}$$

represents homophily, while

$$\begin{pmatrix} 0.2 & 0.9 \\ 0.9 & 0.1 \end{pmatrix}$$

represents heterophily. It is assumed that $r = d \log n$ for some constant d (see also the Logarithmic Dimension Hypothesis in item of (1) of Sect. 12.4).

The MAG model generates graphs which satisfy a densification power law.

Theorem 12.9 ([17]). *The expected number of edges of graphs generated by $M(n,r,\mu,\Theta)$ is*

$$\frac{n(n-1)}{2}(\mu^2\alpha + 2\mu(1-\mu)\beta + (1-\mu)^2\gamma)^r + n(\mu\alpha + (1-\mu)\gamma)^r.$$

The diameter of MAG graphs is also low.

Theorem 12.10 ([17]). *If* $(\mu\beta + (1-\mu)\gamma)^d > 1/2$, *then* aas $M(n,r,\mu,\Theta)$ *has constant diameter.*

Under certain assumptions, the MAG model follows a log-normal degree distribution. With more parameters, a variation of the model *aas* generates graphs whose degree distribution follows a power law.

12.3.5 The GEO-P Model

Our next and final model [6] uses both the notions of embedding the nodes in a metric space (geometric), and a link probability based on a ranking of the nodes (protean). We identify the users of an OSN with points in m-dimensional Euclidean space. Each node has a region of influence, and nodes may be joined with a certain probability if they land within each others region of influence. Nodes are ranked by their popularity from 1 to n, where n is the number of nodes, and 1 is the highest ranked node. Nodes that are ranked higher have larger regions of influence, and so are more likely to acquire links over time. For simplicity, we consider only undirected graphs. The number of nodes n is fixed but the model is dynamic: at each time-step, a node is born and one dies. A static number of nodes is more representative of the reality of OSNs, as the number of users in an OSN would typically have a maximum (an absolute maximum arises from roughly the number of users on the internet, not counting multiple accounts). For a discussion of ranking models for complex networks, see [12, 14, 15, 26].

We now formally define the GEO-P model. The model produces a sequence $(G_t : t \geq 0)$ of undirected graphs on n nodes, where t denotes time. We write $G_t = (V_t, E_t)$. There are four parameters: the *attachment strength* $\alpha \in (0,1)$, the *density parameter* $\beta \in (0, 1-\alpha)$, the *dimension* $m \in \mathbb{N}$, and the *link probability* $p \in (0,1]$. Each node $v \in V_t$ has rank $r(v,t) \in [n]$ (we use $[n]$ to denote the set $\{1, 2, \ldots, n\}$). The rank function $r(\cdot, t) : V_t \to [n]$ is a bijection for all t, so every node has a unique rank. The highest ranked node has rank equal to 1; the lowest ranked node has rank n. The initialization and update of the ranking is done by *random initial rank* (Other ranking schemes may also be used. We use random initial rank for its simplicity.) In particular, the node added at time t obtains an initial rank R_t which is randomly chosen from $[n]$ according to a prescribed distribution. Ranks of all nodes are adjusted accordingly. Formally, for each $v \in V_{t-1}$ that is not deleted at time t,

$$r(v,t) = r(v,t-1) + \delta - \gamma,$$

where $\delta = 1$ if $r(v,t-1) > R_t$ and 0 otherwise, and $\gamma = 1$ if the rank of the node deleted in step t is smaller than $r(v,t-1)$, and 0 otherwise.

Let S be the unit hypercube in $\mathbb{R}^m$, with the torus metric $d(\cdot,\cdot)$ derived from the L_∞ metric. More precisely, for any two points x and y in $\mathbb{R}^m$, their distance is given by

$$d(x,y) = \min\{||x - y + u||_\infty : u \in \{-1, 0, 1\}^m\}.$$

The torus metric is chosen so that there are no boundary effects.

To initialize the model, let $G_0 = (V_0, E_0)$ be any graph on n nodes that are chosen from S. We define the *influence region* of node v at time $t \geq 0$, written $R(v,t)$, to be the ball around v with volume

$$|R(v,t)| = r(v,t)^{-\alpha} n^{-\beta}.$$

For $t \geq 1$, we form G_t from G_{t-1} according to the following rules.

1. Add a new node v that is chosen *uniformly at random* from S. Next, independently, for each node $u \in V_{t-1}$ such that $v \in R(u,t-1)$, an edge vu is created with probability p. Note that the probability that u receives an edge is proportional to $p\, r(u,t-1)^{-\alpha}$. The negative exponent guarantees that nodes with higher ranks ($r(u,t-1)$ close to 1) are more likely to receive new edges than lower ranks.

2. Choose uniformly at random a node $u \in V_{t-1}$, delete u and all edges incident to u.
3. Node v obtains an initial rank $r(v,t) = R_t$ which is randomly chosen from $[n]$ according to a prescribed distribution.
4. Update the ranking function $r(\cdot,t) : V_t \to [n]$.

Since the process is an ergodic Markov chain, it will converge to a stationary distribution. (See [25] for more on Markov chains.) The random graph corresponding to this distribution with given parameters α, β, m, p is called the *geo-protean* graph (or *GEO-P* model), and is written GEO-P(α, β, m, p).

Let $N_k = N_k(n, p, \alpha, \beta)$ denote the number of nodes of degree k, and $N_{\geq k} = \sum_{l \geq k} N_l$. The following theorem demonstrates that the geo-protean model generates power law graphs with exponent

$$b = 1 + 1/\alpha. \tag{12.2}$$

Note that the variables $N_{\geq k}$ represent the cumulative degree distribution, so the degree distribution of these variables has power law exponent $1/\alpha$.

Theorem 12.11 ([6]). *Let* $\alpha \in (0,1)$*,* $\beta \in (0, 1-\alpha)$*,* $m \in \mathbb{N}$*,* $p \in (0,1]$*, and*

$$n^{1-\alpha-\beta} \log^{1/2} n \leq k \leq n^{1-\alpha/2-\beta} \log^{-2\alpha-1} n.$$

Then aas *GEO-P*(α, β, m, p) *satisfies*

$$N_{\geq k} = \left(1 + O(\log^{-1/3} n)\right) \frac{\alpha}{\alpha+1} p^{1/\alpha} n^{(1-\beta)/\alpha} k^{-1/\alpha}.$$

Geo-protean graphs are relatively dense.

Theorem 12.12 ([6]). Aas *the average degree of GEO-P*(α, β, m, p) *is*

$$d = (1 + o(1)) \frac{p}{1-\alpha} n^{1-\alpha-\beta}. \tag{12.3}$$

Note that the average degree tends to infinity with n; that is, the model generates graphs satisfying a *densification power law*. While the diameter is not shrinking, it can be made constant by allowing the dimension to grow as a logarithmic function of n.

Theorem 12.13 ([6]). *Let* $\alpha \in (0,1)$*,* $\beta \in (0, 1-\alpha)$*,* $m \in \mathbb{N}$*, and* $p \in (0,1]$*. Then* aas *the diameter D of GEO-P*(α, β, m, p) *satisfies*

$$D = \Omega(n^{\frac{\beta}{(1-\alpha)m}} \log^{\frac{-\alpha}{m}} n), \textit{ and } D = O(n^{\frac{\beta}{(1-\alpha)m}} \log^{\frac{2\alpha}{(1-\alpha)m}} n). \tag{12.4}$$

In particular, aas *the order of the diameter can be expressed as:*

$$\log D = \frac{\beta}{(1-\alpha)m} \log n + O\left(\frac{\log \log n}{m}\right).$$

If $m = C\log n$, for some constant $C > 0$, then *aas* we obtain a diameter bounded above by a constant.

Aas the GEO-P model, for some values of m, generates graphs with higher clustering coefficient than in a random graph $G(n,d/n)$ with the same expected average degree. We use the notation $\lfloor x \rfloor_2$ to denote the largest *even* integer smaller than or equal to x.

Theorem 12.14 ([6]). Aas *the clustering coefficient of G sampled from GEO-P*(α,β,m,p) *satisfies the following inequality*

$$c(G) \geq (1+o(1))\left(\frac{3}{4}\left(1-\frac{2}{3K}\right)\right)^m \left(\frac{1-\alpha}{1+\alpha}\right)p$$
$$= (1+o(1))\exp\left(-f\left(\frac{m}{K}\right)\right)\left(\frac{3}{4}\right)^m \left(\frac{1-\alpha}{1+\alpha}\right)p,$$

where $f(\frac{m}{K}) = \Theta(\frac{m}{K})$*, and*

$$K = \left\lfloor \left(\frac{n^{1-\alpha-\beta}}{\log^3 n}\right)^{1/m} \right\rfloor_2 .$$

Note that if

$$m \leq (1-\alpha-\beta)\frac{\log n}{\log\log n}\left(1-\frac{1}{\log\log n}\right) = (1+o(1))(1-\alpha-\beta)\frac{\log n}{\log\log n},$$

then $K \gg m$, and the clustering coefficient of GEO-P(α,β,m,p) is *aas* at least

$$(1+o(1))\left(\frac{3}{4}\right)^m \left(\frac{1-\alpha}{1+\alpha}\right)p = n^{o(1)} \gg (1+o(1))\frac{p}{1-\alpha}n^{-\alpha-\beta} = c(G(n,d/n)).$$

Hence, the clustering coefficient is larger than that of a comparable random graph.

The next theorem represents a drastic departure from the good expansion found in binomial random graphs, where $\lambda = o(1)$ [8,9].

Theorem 12.15 ([6]). *Let* $\alpha \in (0,1)$*,* $\beta \in (0,1-\alpha)$*,* $m \in \mathbb{N}$*, and* $p \in (0,1]$*. Let* $\lambda(n)$ *be the spectral gap of the normalized Laplacian of GEO-P*(α,β,m,p)*. Then* aas

1. *If* $m = m(n) = o(\log n)$*, then* $\lambda(n) = 1 + o(1)$*.*
2. *If* $m = m(n) = C\log n$ *for some* $C > 0$*, then*

$$\lambda(n) \geq 1 - \exp\left(-\frac{\alpha+\beta}{C}\right).$$

Given an OSN, we describe how we may estimate the corresponding dimension parameter m if we assume the GEO-P model. In particular, if we know the order n, power law exponent b, average degree d, and diameter D of an OSN, then we can calculate m using our theoretical results. Formula (12.2) gives an estimate for

α based on the power law exponent b. If $d^* = \log d/\log n$, then Eq. (12.3) implies that, asymptotically, $1-\alpha-\beta = d^*$. If $D^* = \log D/\log n$, then formula (12.4) about the diameter implies that, asymptotically, $D^* = \frac{\beta}{(1-\alpha)m}$. Thus, an estimate for m is given by:

$$m = \frac{1}{D^*}\left(1-\left(\frac{b-1}{b-2}\right)d^*\right) = \frac{\log n}{\log D}\left(1-\left(\frac{b-1}{b-2}\right)\frac{\log d}{\log n}\right). \tag{12.5}$$

This estimate suggests that the dimension is proportional to $\log n/\log D$. If D is constant, then this means that m grows logarithmically with n. Recall that the dimension of an OSN may be roughly defined as the least integer m such that we can accurately embed the OSN in m-dimensional Euclidean space. Based on our model we conjecture that the dimension of an OSN is best fit by approximately $\log n$.

The parameters b, d, and D have been determined for samples from OSNs in various studies such as [2, 16, 20, 28]. The following chart summarizes this data and gives the predicted dimension for each network. We round m up to the nearest integer. Estimates of the total number of users n for Cyworld, Flickr, and Twitter come from Wikipedia [35], and those from YouTube comes from their website [36]. When the data consisted of directed graphs, we took b to be the power law exponent for the in-degree distribution. As noted in [2], the power law exponent of $b = 5$ for Cyworld holds only for users whose degree is at most approximately 100. When taking a sample, we assume that some of the neighbours of each node will be missing. Hence, when computing d^*, we used n equalling the number of users in the sample. As we assume that the diameter of the OSN is constant, we compute D^* with n equalling the total number of users.

Parameter	OSN			
	Cyworld	Flickr	Twitter	YouTube
n	2.4×10^7	3.2×10^7	7.5×10^7	3×10^8
b	5	2.78	2.4	2.99
d^*	0.22	0.17	0.17	0.1
D^*	0.11	0.19	0.1	0.16
m	7	4	5	6

12.3.5.1 The GEO-P Tension Model

A variant of the GEO-P model was presented in [31] that warrants further study. In the GEO-P model, if a node v falls in an influence region of two nodes u_1 and u_2, then v can join to u_1 and u_2 with equal probability. We consider a variant where the probability depends on the volume of the corresponding influence regions. Consider a fixed *tension parameter* $h \in (-\infty, 0)$. For a given $t \geq 0$ and vertex u, define

$$T(u,t) = r(u,t)^h.$$

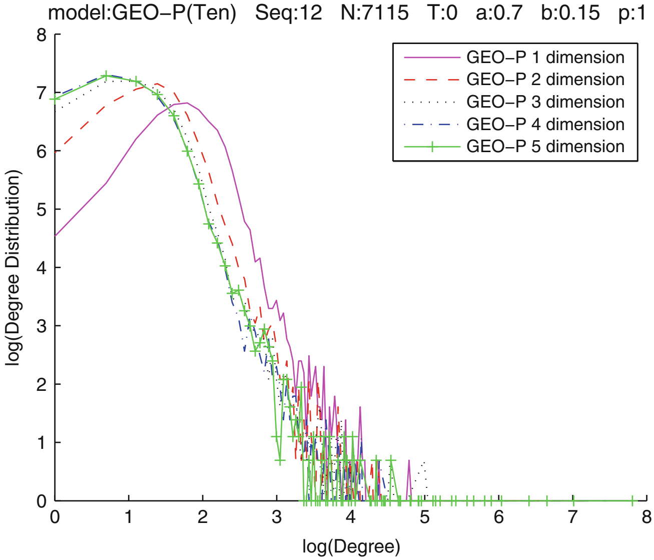


Fig. 12.2 Degree distribution of graph generated by the GEO-P Tension model, $h = -0.1$

Given two nodes u and v, define

$$T(u,v,t) = \frac{T(u,t) + T(v,t)}{2}.$$

The definition of the GEO-P Tension model is analogous to the GEO-P model, but at time $t > 0$, independently, for each node $u \in V_{t-1}$ such that $v \in R(u,t-1)$, an edge vu is created with probability $pT(u,v,t)$. Hence, if v is in the influence region of both u_1 and u_2 with the rank of u_1 higher than u_2, then it is more likely to join to u_1.

Preliminary simulation results indicate that the GEO-P Tension model captures many of the properties of OSNs described in Sect. 12.2. Figures 12.2–12.4 display the log-log plots of the degree distribution of graphs of order 7,115 simulated by the GEO-P Tension model in dimensions 1–5 inclusive. We set the tension parameter $h = -0.1, -0.3$ and -0.7, respectively. Tables 12.1 and 12.2 list the diameters (where $|V(C)|$ is the order of the largest connected component) and spectral gaps (with respect to the adjacency matrix) of the corresponding graphs.

For the GEO-P(Ten) model, it remains to rigorously prove the properties outline in Sect. 12.2.

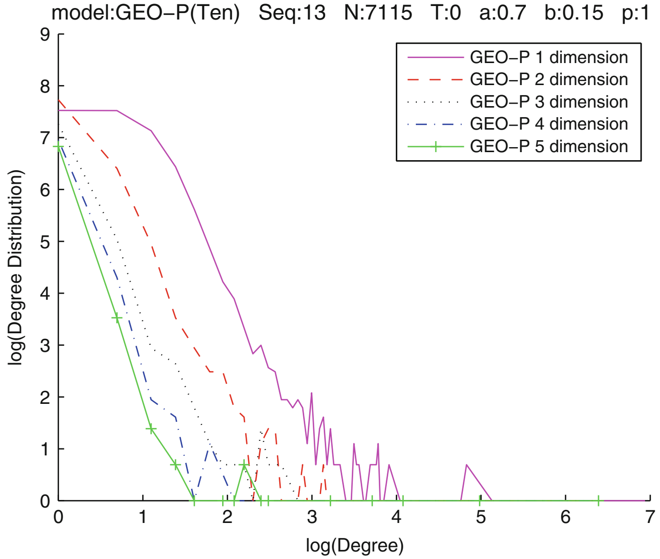


Fig. 12.3 Degree distribution of graph generated by the GEO-P Tension model, $h = -0.3$

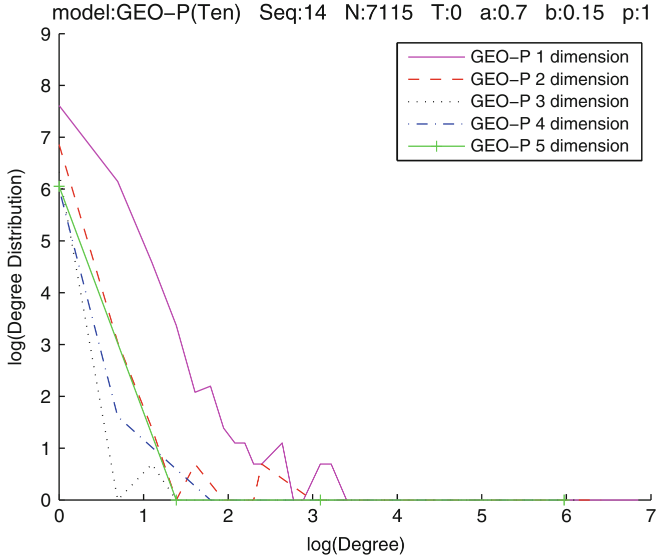


Fig. 12.4 Degree distribution of graph generated by the GEO-P Tension model, $h = -0.7$

Table 12.1 Spectral gaps of graphs generated by the GEO-P(Ten) model, in dimensions 1–5 inclusive

GEO-P(Ten) $N = 7{,}115$, $\alpha = 0.7$, $\beta = 0.15$, $p = 1$

$h = -0.1$		$h = -0.3$		$h = -0.7$	
Dim	Gap	Dim	Gap	Dim	Gap
1	14.170589	1	9.048678	1	13.048242
2	0	2	9.286437	2	12.406163
3	11.410871	3	10.354512	3	13.150765
4	15.54475	4	10.186001	4	14.226928
5	20.845548	5	12.422439	5	15.133812

Table 12.2 Diameters of graphs generated by the GEO-P(Ten) model

GEO-P(Ten) $N = 7{,}115$, $\alpha = 0.7$, $\beta = 0.15$, $p = 1$

	$h = -0.1$		$h = -0.3$		$h = -0.7$	
Dimension	Diam	$\mid V(C) \mid$	Diam	$\mid V(C) \mid$	Diam	$\mid V(C) \mid$
1	20	7,098	6	6,300	4	2,673
2	20	6,953	7	3,149	4	990
3	20	6,847	7	1,648	3	571
4	15	6,744	6	1,152	3	415
5	12	6,747	5	977	2	428

12.4 Open Problems

Many questions remain in modelling OSNs and other complex networks. We collect these here for future reference.

1. The *Logarithmic Dimension Hypothesis* (or *LDH*) [6] conjectures that the dimension of an OSN is best fit by about $\log n$, where n is the number of users in the OSN. The motivation for the conjecture comes from both the GEO-P and MAG models. Both models posit $\log n$ attributes for each user so as to provably ensure that certain properties found in OSNs (such as constant diameter and bad spectral expansion) are satisfied. Given the availability of OSN data, it may be possible to fit the data to the model to determine the dimension of a given OSN. Initial estimates in [31] from sampled OSN data indicate that the spectral gap found in OSNs correlates with the spectral gap found in the GEO-P model when the dimension is approximately $\log n$, giving some additional credence to the LDH. See also the MAG model as discussed in Sect. 12.3.4.
2. Another interesting direction would be to generalize the GEO-P to a wider array of ranking schemes (such as ranking by age or degree), and determine when similar properties (such as power laws and bad spectral expansion) provably *aas* hold. Simulations with the GEO-P Tension model show promising data [31], but the rich dependence structure of this model may make rigorous analysis a challenge.

3. As discussed in Sect. 12.2, the recent work [7] indicates that social networks lack high compressibility, especially in contrast to the web graph. Note that property (v) bad compressibility has not been explicitly studied in any of the models presented here. It would be interesting to study compressibility in these models, and to devise a model which provably has all five properties.
4. Anecdotal evidence from everyday experience with Twitter and Facebook shows that news and gossip spread quickly in such networks. An epidemiological model, such as SIS or SIRS, or even a deterministic model such as firefighting and seepage [5] would be worth exploring in real OSN data and in the models.

References

1. L.A. Adamic, O. Buyukkokten, E. Adar, A social network caught in the web, *First Monday* **8** (2003).
2. Y. Ahn, S. Han, H. Kwak, S. Moon, H. Jeong, Analysis of topological characteristics of huge on-line social networking services, In: *Proceedings of the 16th International Conference on World Wide Web*, 2007.
3. A. Bonato, *A Course on the Web Graph*, American Mathematical Society Graduate Studies Series in Mathematics, Providence, Rhode Island, 2008.
4. A. Bonato, N. Hadi, P. Horn, P. Prałat, C. Wang, Models of on-line social networks, *Internet Mathematics* **6** (2011) 285–313.
5. A. Bonato, R.J. Nowakowski, *The Game of Cops and Robbers on Graphs*, American Mathematical Society, Providence, Rhode Island, 2011.
6. A. Bonato, J. Janssen, and P. Prałat, The geometric protean model for on-line social networks, In: *Proceedings of the 7th Workshop on Algorithms and Models for the Web-Graph (WAW2010)*, Lecture Notes in Computer Science 6516, Springer, 2010, 110–121.
7. F. Chierichetti, R. Kumar, S. Lattanzi, M. Mitzenmacher, A. Panconesi, P. Raghavan, On compressing social networks, In: *Proceedings of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'09)*, 2009.
8. F.R.K. Chung, *Spectral Graph Theory*, American Mathematical Society, Providence, Rhode Island, 1997.
9. F.R.K. Chung, L. Lu, *Complex Graphs and Networks*, American Mathematical Society, U.S.A., 2004.
10. E. Estrada, Spectral scaling and good expansion properties in complex networks, *Europhys. Lett.* **73** (2006) 649–655.
11. Facebook: statistics. Accessed September 1, 2011. http://www.facebook.com/press/info.php?statistics.
12. S. Fortunato, A. Flammini, F. Menczer, Scale-free network growth by ranking, *Phys. Rev. Lett.* **96** 218701 (2006).
13. O. Frank, Transitivity in stochastic graphs and digraphs, *Journal of Mathematical Sociology* **7** (1980) 199–213.
14. A. Henry, P. Prałat, Rank-Based Models of Network Structure and the Discovery of Content, In: *Proceedings of the 8th Workshop on Algorithms and Models for the Web Graph (WAW 2011)*, 2011.
15. J. Janssen, P. Prałat, Protean graphs with a variety of ranking schemes, *Theoretical Computer Science* **410** (2009), 5491–5504.
16. A. Java, X. Song, T. Finin, B. Tseng, Why we twitter: understanding microblogging usage and communities, In: *Proceedings of the Joint 9th WEBKDD and 1st SNA-KDD Workshop 2007*, 2007.

17. M. Kim, J. Leskovec, Multiplicative attribute graph model of real-world networks, In: *Proceedings of the 7th Workshop on Algorithms and Models for the Web Graph (WAW 2010)*, 2010.
18. J. Kleinberg, The small-world phenomenon: An algorithmic perspective, In: *Proceedings of the 32nd ACM Symposium on Theory of Computing*, 2000.
19. R. Kumar, J. Novak, A. Tomkins, Structure and evolution of on-line social networks, In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.
20. H. Kwak, C. Lee, H. Park, S. Moon, What is Twitter, a social network or a news media?, In: *Proceedings of the 19th International World Wide Web Conference*, 2010.
21. S. Lattanzi, D. Sivakumar, Affiliation networks, In: *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, 2009.
22. J. Leskovec, J. Kleinberg, C. Faloutsos, Graphs over time: densification Laws, shrinking diameters and possible explanations, In: *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2005.
23. J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, Realistic, mathematically tractable graph generation and evolution, using Kronecker multiplication, In: *Proceedings of European Conference on Principles and Practice of Knowledge Discovery in Databases*, 2005.
24. J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, Z. Ghahramani, Kronecker Graphs: An approach to modeling networks, *Journal of Machine Learning Research* **11** (2010) 985–1042.
25. D.A. Levin, Y. Peres, E.L. Wilmer, *Markov Chains and Mixing Times*, American Mathematical Society, 2009.
26. T. Łuczak, P. Prałat, Protean graphs, *Internet Mathematics* **3** (2006), 21–40.
27. M. Mahdian, Y. Xu, Stochastic Kronecker graphs, In: *Proceedings of the 5th Workshop on Algorithms and Models for the Web-Graph, 2007*
28. A. Mislove, M. Marcon, K. Gummadi, P. Druschel, B. Bhattacharjee, Measurement and analysis of on-line social networks, In: *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, 2007.
29. M.E.J. Newman, J. Park, Why social networks are different from other types of networks, *Phys. Rev. E* **68** 036122 (2003).
30. J.P. Scott, *Social Network Analysis: A Handbook*, Sage Publications Ltd, London, 2000.
31. Yanhua Tian, Models and Mining of On-line Social Networks, M.Sc. Thesis, Ryerson University, 2011.
32. Twitaholic. Accessed September 1, 2011. http://twitaholic.com/.
33. D.J. Watts, S.H. Strogatz, Collective dynamics of ‘small-world’ networks, *Nature* **393** (1998) 440–442.
34. H. White, S. Harrison, R. Breiger, Social structure from multiple networks, I: Blockmodels of roles and positions, *American Journal of Sociology* **81** (1976) 730–780.
35. Wikipedia: List of social networking websites. Accessed September 1, 2011. http://en.wikipedia.org/wiki/Listofsocialnetworkingwebsites.
36. YouTube, Advertising and Targeting. Accessed September 1, 2011. http://www.youtube.com/t/advertising_targeting.

Chapter 13
NAVEL Gazing: Studying a Networked Scholarly Organization

Dimitrina Dimitrova, Barry Wellman, Anatoliy Gruzd, Zack Hayat, Guang Ying Mo, Diana Mok, Thomas Robbins, and Xiaolin Zhuo

Abstract Many North Americans now work in a global economy where corporations foster networked work – with employees participating in multiple teams and often for multiple purposes – and they do so in networked organizations – whose workers may be physically and organizationally dispersed. We analyze networked workers in one networked scholarly organization: the GRAND Network Centre of Excellence. Drawing on qualitative and social network data, we present

D. Dimitrova (✉) • B. Wellman • G.Y. Mo
Department of Sociology, University of Toronto, 725 Spadina Avenue, Toronto, ON,
M5S 2J4, Canada
e-mail: dima@chass.utoronto.ca; wellman@chass.utoronto.ca; oliviamo@hotmail.com;
zhuo.jolin@gmail.com

A. Gruzd
School of Information Management, Dalhousie University, 6100 University Avenue,
Halifax, NS, B3H 4R2, Canada
e-mail: gruzd@dal.ca

Z. Hayat
Faculty of Information, University of Toronto, 45 Willcocks Street, Toronto, ON,
M5S 1C7, Canada
e-mail: tsahi.hayat@utoronto.ca

D. Mok
Department of Geography, University of Western Ontario, Social Science Centre, London,
ON, N6A 5C2, Canada
e-mail: dmok3@uwo.ca

T. Robbins
Department of Sociology and Social Anthropology, Dalhousie University, 6100 University
Avenue, Halifax, NS, B3H 4R2, Canada
e-mail: th698731@dal.ca

X. Zhuo
Department of Sociology, Harvard University, 33 Kirkland Street, Cambridge, MA, 02138
e-mail: zhuo.jolin@gmail.com

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_13,

our preliminary findings at the early stages of GRAND. Early discussions viewed networked organizations as the antithesis of traditional bureaucratic organizations and expected bureaucratic characteristics such as hierarchy, centralization and formalization to be absent and cross-boundary flows – the hallmark of networked organizations – to be prominent. Our research shows that reality is more complex than the early deductive expectations for networked organizations. The GRAND network is well positioned for cross-boundary flows but they are not yet extensive. In the distributed GRAND network, researchers communicate mostly via now-traditional email although in-person contact is almost as frequent. GRAND is designed with few formal hierarchical differences. Yet hierarchy matters when it comes to communication – researchers in higher positions have higher centrality in communication structures, both GRAND-wide and within projects, suggesting consistent advantages in their communication. Cross-disciplinary exchanges in GRAND are low at the network's early stages, with little collaboration between Computer Science and Engineering, on the one hand, and Social Sciences and Humanities, on the other. Researchers in Arts and Technology emerge as the most active collaborators in the network both internally and externally. Work within provinces is still the norm.

13.1 Introduction: Networked Work in Networked Organizations

The world is becoming networked: work and organizations as well as computers and friendships [43]. Many North Americans now work in a global economy where corporations deal agilely with turbulent market environments by fostering networked work. They participate in multiple teams – often for multiple purposes [36]. And they do so in networked organizations – whose workers may be physically and organizationally dispersed. This is a major change from the situation since the Industrial Revolution of the 1800s, where the organization of work has been that of employees in large factories and offices, or in small bounded groups such as retail shops. Although there are still plenty of such factories, offices and shops, a host of forces has been transforming work from individual or group activities to networked activities.

We analyze in this chapter networked workers in one networked organization: the GRAND Network Centre of Excellence. Although a scholarly network may seem different from other organizations, it has some analytic advantages. All researchers are computer literate. Researchers are used to collaborating, and in GRAND, all researchers are expected to contribute to other projects in the network. Productivity is visible through such outputs as scholarly papers, presentations, research grants, and media coverage: all data that GRAND collects. All of this collaboration leaves a clearly-marked trail.

GRAND, an acronym for Graphics, Animation and New Media, is a Canadian network of scholars – computer scientists, social scientists, and humanists – that stretches 4,440 km from Dalhousie University in Halifax on the Atlantic Coast to the University of British Columbia and Simon Fraser University in Vancouver on the Pacific Coast. GRAND is designed to be a networked organization. Each of the more than 100 faculty members in GRAND is encouraged to work in multiple research projects, and each project seeks to include researchers from several universities and several disciplines.

One of the 34 projects in the network is GRAND's NAVEL-gazing project – intended to analyze GRAND itself. (NAVEL officially stands for "Network Assessment and Validation for Effective Leadership".) In this chapter, we present the results of NAVEL's research at the initial stage of the network when GRAND had just become operational. We examine why scholars have joined the GRAND network, what ties connect them to the network and to their projects, how they work in multiple teams on multiple projects, and how they use various communication media for connectivity.

Although the research presented here is preliminary, it is pioneering both in the study of dispersed scholarly networks and in the more general study of networked organizations. That is because management gurus' assertions and advocacy about how networked organizations can – and should – operate outweighs the evidence and analysis about how they actually operate. By contrast, our analysis draws on qualitative and survey data to reveal patterns of collaboration and communication. It contributes to the study of networked organizations a detailed description of how networked scholars are connected and how such organizations function.

After a review of the current state of knowledge about networked work in networked organizations, we briefly described how our NAVEL team is studying GRAND. We then proceed to discuss our preliminary findings at the early stages of GRAND about:

- Rationales for participating: pragmatic, intellectual, and networking
- Overlapping types of networks in GRAND, such as Working With, Friendship, Advice, and Coauthorship
- Means of communication: in-person, email, phone, and social media
- The extent to which GRAND functions as a networked or hierarchical organization
- GRAND's networked relationships across projects, disciplines, and provinces

13.2 Literature Review

13.2.1 The Turn to Networked Organizations

Networked organizations cross boundaries: their employees work in multiple fluid teams – that are often ad hoc and temporary – and they switch between teams, organizational units, or organizations. Compared to traditional bureaucracies, networked organizations can be flatter and more decentralized. Such organizations are often geographically distributed and facilitated by technology; they become both networked and virtual. They can take the form of work groups, large-scale project teams, and inter-organizational strategic alliances [11, 13, 28, 50].

Five related trends are encouraging the turn to networked work in networked organizations:

First, the globalization of work, consumerism, and travel has expanded and diversified the reach and purview of organizations.

Second, the shift in developed countries from growing, mining, making, and transporting things – atom-work in the material economy – to selling, describing, and analyzing things via words and pictures – bit-work in the information economy. A rising number of people belong to what Richard Florida calls the "creative class":

> People in science and engineering, architecture and design, education, arts, music and entertainment, whose economic function is to create new ideas, new technology, and/or new creative content [21, 37].

In short, they are people who usually manipulate bits on computer networks more than atoms [43]. For example, GRAND is entirely composed of bit-workers: faculty, students, support staff, and corporate, non-governmental and government partners.

Third, although the shift to networked work and organizations began before the Internet and Mobile Revolutions, these two revolutions have accelerated the shift, because these two revolutions allow bit workers to have more ability to network – in multiple senses of that word – than atom workers on assembly lines. Information and communication technologies (ICTs) make it easier to connect and collaborate when workers are pushing bits by calculating, searching, drawing, and writing. Some of the most ICT-connected workers have jobs built around creative effort rather than manufacturing or standardized paper-pushing [43].

Fourth, the Internet allows people to communicate, collaborate, and access shared information, databases, software and hardware at a distance – from publicly available libraries to secret corporate records [22].

Fifth, the Mobile Revolution allows bit-workers to be productive with their laptops and smartphones while being away from their desks. While the purchase of desktop computers has leveled off and that of wired-in landline phones has declined, the purchase of mobile phones, tablets, and laptops has soared. It is often as easy to push bits at home or in the coffee shop as it is in the office. "Road warriors," including scholars, do their work at homes, offices, hotels, planes, trains, and automobiles.

Neither all workers nor all organizations in North America have become connected. Many people still work on assembly lines, sit in separate cubicles – white-collar assembly lines – or work alone in shops or trucks and taxis. There are plenty of traditional bureaucratic organizations. Even so, networked work has become the norm in many workplaces. Networked work and networked organizations are common in research where they are well aligned with the information-related and creative nature of work and with scholarly traditions of collaboration [41]. Several trends specific to scientific research encourage the emergence of networked scholarly organizations.

First, ever since the emergence of big science in the 1930s and 1940s, scientific research has become the domain of large collaborative projects [23]. They are often distributed across several locations and even globally. Collaborating researchers rely on an array of technology to collaborate and share results encouraging the emergence of e-science [27].

Second, concerns about the broader recruitment of experts, coordinating research, pooling data, or efficiently using expensive equipment, foster multi-organizational involvement [6]. Sometimes scholars from different organizations become members of communities of practice – virtual learning communities through whose networks they can access tacit knowledge and lore, earn professional reputations, and develop useful concepts [6, 52].

Third, the scope and complexity of research issues today often benefit from multi-disciplinary solutions. Many projects recruit experts from several disciplines [17]. Research collaboration is becoming larger and more complex, increasingly multi-institutional, multi-site, multi-disciplinary, and reliant on technology.

Fourth, in addition to collaboration in informal scholarly networks – "invisible colleges" [14], research networks are becoming more formally structured. Large-scale complex collaboration among scholars today entails organizational issues such as negotiating goals and priorities or providing administrative and technological support [6]. That is why collaborative efforts foster more formal structures, including networked organizations [46].

Thus, many scholarly networks today function as networked organizations with flexible cross-boundary information flows and decentralized flatter structures. Such organizations may significantly improve creation but also entail substantial coordination costs.

13.2.2 Benefits and Costs of Networked Organizations

Management analysts have suggested that flexible, decentralized, networked organizations have several advantages, from recruiting the right talent to decreasing office costs and fostering creativity. Networked firms can assemble ad hoc teams with diversified talents and perspectives. Workers have more discretion about the work they accomplish, take greater ownership and pride in their work, and may be more productive. Networked organizations that are geographically distributed and Internet-based, can offer flexibility in the organization of work, reduced real estate costs, more work time through less commuting, and rapid access to information [5, 41, 43].

What sets networked organizations apart from traditional bureaucracies is their potential to encourage knowledge creation, creativity, and collaboration rooted in their flexible cross-boundary flows. Networked organizations are seen as less deterministic than traditional bureaucratic organizations; they are based on fluctuating patterns of association and emergent structures [28]. Members work and network between workgroups and organizations – and at times, between continents. As they move among teams, workers expand the networks of expertise that are “glocal”, with both local interactions and global connectivity [41]. Having connections to different work teams provides more diverse information and helps finding people who are interesting and think differently [7, 10]. The more structurally diverse the networks, the better the performance of both individuals and their work teams [55].

In scientific research, boundary-crossing work and information flows occur along several dimensions: disciplines, institutions, sectors, and locations. Multi-institutional collaboration can bring the benefits of pooling research expertise, coordination of research activities on a broader scope, and more efficient use of infrastructure. Policymakers often assume that collaborative research structures will inevitably foster work across institutions, disciplines, sectors, and distances. They believe that such cross-cutting flows will bring significant gains to society, advance science, and benefit scientists themselves.

Yet, complex research collaboration faces significant challenges and is not always successful [39, 40]. Collaboration can be hindered by competition for funding or efforts to guard intellectual property [6, 7]. Coordinating research activities across large organizations such as universities or government organizations – many of which continue to function as bureaucracies – can be difficult and slow. Some past studies found a negative correlation between the success of collaborative research networks and the number of participating institutions [45]. Other research, in contrast, links fewer participating organizations to increased potential of conflict between teams as well as between scientists and project management [46]. When organizational members – such as GRAND scholars – move between multiple projects, their attention and loyalty can be divided. These difficulties intensify when new teams are formed who do not know each other and do not share a common culture: an inherent phenomenon in GRAND’s multi-discipline imperative [38].

Collaboration is especially difficult when participating organizations come from different sectors, such as academia, industry, NGOs, and government. Partnerships across sectors are considered vital for successful knowledge transfer and innovation: engaging future users early in the research process increases the likelihood that the research outcomes will be adopted [18]. Yet, different concerns, institutional constraints, and cultures make such partnerships difficult and research shows that sectoral boundaries can be especially strong [20].

Further, the complexity of current research problems requires contributions from several disciplines. However, researchers from different disciplines do not have a shared understanding of the issues and lack common methodologies and practices created by disciplinary training and by interaction at scientific forums [12, 17]. Ties between them tend to remain limited to information exchange rather than be close collaboration [45]. To work across disciplines, researchers need to establish a shared understanding of issues and common practices, as well as overcome a lack of social bonding. That is why multi-disciplinary teams require management of both tasks and relationships.

Difficulties created by disciplinary and institutional differences are often compounded by distance. When physically dispersed, scholars tend to have more communication and coordination difficulties and may take more time to get work done [17]. Using technology to communicate can increase the opportunity for misunderstanding, slow down communication, decrease the incentive for participants to adapt, and make building trust difficult [6, 30, 38, 56]. Perhaps that is why researchers, who need to communicate novel and complex knowledge, have a strong preference for in-person rather than mediated communication [6, 18, 45]. Despite their collaborative traditions and familiarity with technology, researchers do not make the perfect distant collaborators [17]. Managing dispersed cross-organizational research teams remains difficult even when technology is ubiquitous.

13.2.3 Organization of Work and Collaborative Practices

Individual researchers develop a range of solutions to cope with the difficulties of large scale collaboration. Scientists working with colleagues in large scholarly networks often find themselves engaging in several research projects within or across the network. They have increased access to funding and information yet they juggle multiple deadlines and commitments to different teams and bear higher coordination and communication costs. Since academic institutions do not always reward joint research, joining collaborative endeavors may be problematic [12, 18]. Propensity to collaborate varies across disciplines and is linked to resource concentration, the need to coordinate research, or culture [4]. Individual rationales also play a role: intellectual stimulation or more pragmatic rewards such as revenue related products and services, human capital, self-marketing, peer recognition, and personal needs [42]. Moreover, participants in large collaborative networks tend to be established academics free of the pressures of career building and strongly concerned having others recognize their research [20].

Researchers engaged in complex collaboration develop distinctive work practices to cope with coordination and communication costs. They tend to organize work in ways that minimize interdependence and decrease the need for coordination and communication [26, 39]. Few projects require ongoing interaction. Instead, researchers divide the work into separate, well defined segments that can be done independently. Most projects are additive: collaboration is like a jigsaw puzzle in which pieces fit together at the end. Another common strategy is to invite collaborators with whom researchers have previously worked [18]. Teams where members have known each other in advance have established work practices, and have less conflict [46].

Ensuring successful collaboration in scientific research requires both individual and organizational level solutions. Success depends on the ways in which networked organizations are organized and managed. However, discovering best practices is difficult because networked organizations are conceptualized in different ways. Some analysts maintain that such organizations are loose associations held together by emergent patterns of interaction [28]. Other analysts contend that networked organizations still contain traditional organizational hierarchies [35]. Taylor, drawing on distributed cognition framework, claims senior management becomes especially important in networked organization because they collect and express collective organizational knowledge [49].

Research has yielded disparate findings. While scholarly organizations with centralized decision making – where leaders make decisions without wide consultation – experience more conflicts between scientists and project management, their formal authority facilitates rapid decision-making [46]. Even in more networked structures, centralization and hierarchy can benefit a dispersed group: when a few experts answer inquiries in their field of expertise, communication efficiency can increase [1]. In such situations, decentralized authority co-exists with centralized communication.

In short, trends in scientific research foster the emergence of networked scholarly organizations as a way to conduct large-scale complex collaborations. Such collaborations can bring considerable benefits, yet face significant challenges and foster distinctive practices. Until now, studies of multi-disciplinary multi-site research have focused on project management practices or technology use. Few discussions of networked organizations have actually analyzed the social networks within them. Moreover, studies of scholarly networks are not informed by the concept of networked organizations [31]. This disconnect hinders in-depth understanding of how networked scholars collaborate and how networked organizations operate. NAVEL's study of GRAND addresses these gaps.

13.3 The GRAND Network of Centres of Excellence

GRAND is part of the Networks of Centres of Excellence (NCE) program, a key part of the Canadian government's strategy to encourage knowledge creation and innovation (www.nce-rce.gc.ca). The NCE program is specifically designed to support scientific knowledge that fosters socially and commercially relevant research. It funds multi-discipline and nation-wide research collaboration as well as multi-sectoral partnerships between academia, industry, government, and not-for-profit organizations.

GRAND started functioning at the start of 2010 to serve as a catalyst for research and innovation for new media and information technologies. Its mandate is to encourage innovation in information-intensive industries, increase Canada's capacity to deploy ICT infrastructure, and contribute to the development of its knowledge-based economy. GRAND – as all NCEs – creates a flexible networked organizational form based on less formal ties, boundary-spanning flows, and permeable boundaries. It is a loosely connected network of academics, government and industry decision-makers and researchers, NGOs, and other stakeholders that is united by shared interests in studying new media. GRAND's industry, government, and NGO partners were still being recruited at the time of the data collection, but the activities and relationships among the academic members of GRAND show much of how GRAND functions.

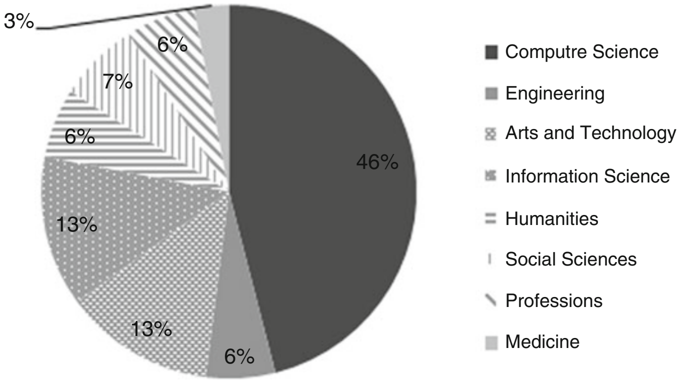


Fig. 13.1 GRAND researchers by discipline

At the early stage that we collected our data, GRAND comprised 143 academics: 56 (39 %) of them project leaders holding the title of Principal Network Investigators (PNI) while the remaining 87 (61 %) are Collaborating Network Investigators (CNI). GRAND members are expected to work in a networked fashion. All members are encouraged to collaborate actively across network projects, thereby pooling resources and information: at the initial stage of GRAND, 52 % of the members participated in several projects. As for all NCEs, GRAND is funded for a limited period of time. In short, while membership in GRAND is somewhat more formal than participating in informal scholarly networks, both the project and network boundaries are porous and temporary.

The composition of the network is diverse in terms of locations and disciplines. GRAND's academic researchers come from 26 institutions of higher education dispersed in seven Canadian provinces. Their disciplinary backgrounds range from Computer Science and Engineering to Art and Design, from Information Science and Journalism to Social Sciences and Humanities. Among GRAND participants, 52 % come from Natural Sciences and Engineering, 45 % from Social Sciences and Humanities, and 3 % from Health research. A more detailed breakdown on Fig. 13.1 shows that almost half (46 %) of GRAND members are computer scientists. Others come from Information Science (13 %), Arts and Technology (13 %), Social Sciences (7 %), Humanities (6 %), and Engineering, Medicine and other professions (15 %).

GRAND's projects are interdisciplinary and dispersed. Among the 34 projects in the network, only three have all members from the same discipline. By contrast, two-thirds of the projects involve three or four disciplines. Project members come from multiple universities and are geographically dispersed. On average, project team members come from five universities and are located in three provinces (Fig. 13.2).

In short, the recruitment and structure of GRAND are intended to encourage multi-disciplinary, multi-university, and inter-provincial collaboration. The diversity of researchers in terms of disciplines, university affiliations, and locations creates

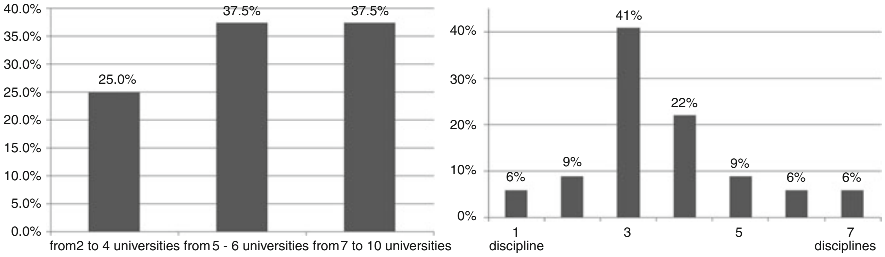


Fig. 13.2 GRAND projects by number of disciplines and participating universities

the precondition for boundary-spanning flows. Moreover, GRAND's formal rules and procedures aim to create links across projects, organizations, disciplines, and locations. Key questions are emerging:

- What processes and relationships does the GRAND networked structure support?
- What researchers does the network attract?
- What ties link researchers to the network?
- How do researchers collaborate when their projects are still new?
- What challenges are emerging, and how are the scholars coping with them?

13.4 Methods

NAVEL conducted a mixed methods study, collecting data in the following ways [15]:

1. An online survey about interactions in GRAND using LimeSurvey open source software. GRAND members are dispersed across the country and an online survey provides secure and convenient access for all GRAND members. Participants describe with which GRAND members they collaborate, exchange advice, share ideas, network, make friends, or would like to meet. In addition, the survey asks about the use of communication media such as landline phones, mobile phones, emails, or instant messages. This approach, starting with a roster of members, is an established procedure in social network analysis.
2. Longitudinal data: NAVEL involves several successive surveys that will examine the early stage, the mid-point, and the final stage of existence of GRAND. This enables the team to examine the evolution of the network. The baseline survey was conducted a few months after GRAND received formal approval. All survey participants were academic researchers as partners and students were still being recruited at that time. All GRAND academic members (143) were invited to participate: 101 of them completed the survey. Analyses were conducted using ORA and UCINET software.
3. Semi-structured interviews with GRAND researchers, students and partners. The interviews focused on respondents' rationale for joining GRAND: work,

coordination, communication practices, and recent developments in projects. Interview data collection is ongoing: the team has conducted 38 initial interviews and 12 follow-up interviews. Analyses use Saturate software. To understand the interviewees' rationales for collaborating, we use a narrative analytic strategy that focuses on the themes within the narrative and searches for experiences that may not be shared by all the interviewees [3].

4. Data from online interactions and publications among GRAND members. Given the dispersal of GRAND members and the popularity of online collaboration and communication tools, much of the interactions among GRAND members are expected to take place in online forums, chats, or with the help of other online collaborative tools. These online interactions are automatically preserved in the form of chat messages, forum postings, wiki pages, etc. and can be collected using automated techniques. In addition, NAVEL collects publications data from various online databases such as Google Scholar, Scopus, and Web of Science.

We use Social Network Analysis (SNA) to analyze connections among researchers. SNA focuses on the structure of relationships among units, be they individuals, groups, or organizations [16], and on the way these relationships affect the processes in a network [24, 51]. By using SNA, NAVEL can demonstrate actual rather than the prescribed exchange among GRAND members, conduct analysis on several levels, capture boundary interactions, identify internal groupings, and study formal positions within GRAND and universities (academic rank) that shape the processes in the organization [44]. In this discussion, we examine ties at two different levels: ties on the level of the GRAND network as a whole, revealing how researchers connect to other GRAND members, and ties on the level of projects, revealing how researchers connect to others within their projects. Network analysis treats exchanges with different contexts as giving rise to distinct networks. For instance, friendships give rise to friendship networks, while working together results in a different work network. It is precisely the overlap and divergence of these different networks that inform the understanding of how processes unfold. In turn, cross-boundary interactions demonstrate the extent to which GRAND functions as a networked organization. Hence, our analysis focuses on: (a) describing existing ties and processes; and (b) revealing cross-boundary interaction.

13.5 Findings

13.5.1 Rationales for Participating

Given the challenges that large collaborative networks such as GRAND present for individual researchers, joining is by no means an obvious decision. It reflects disciplinary constraints and individual attitudes and is a key to understanding collaborative behavior.

13.5.1.1 Pragmatic Rationales

Many researchers point to pragmatic rationales for joining GRAND. Their rationales are consistent with previous studies suggesting that scholarly collaborators are motivated by pragmatic rewards such as revenue related products and services, self-marketing, and personal needs [32].

Academics face pressure to do research and to publish. For many, looking for research funding is a continuing concern. Funding from GRAND lessens this ongoing concern and enables senior and junior researchers alike to maintain their research program; it also reduces some immediate problems such as funding graduate students.

GRAND provides stable funding for up to 15 years. This longevity is especially attractive: it makes it possible for researchers to work on sophisticated projects without interruption. A researcher says, "So I thought the fact that it could extend to 15 years would provide me with base funding in my research area for a good chunk of my career."

GRAND has become a "home" for research that fits well into the network's research program. Some research projects in GRAND continue pre-existing collaborations. "We have been working our ass for 5 years. So for us it was just 'Okay, take what we are doing now anyway and work it into the GRAND proposal'" says a PNI. In other cases, researchers start new collaborations that may shift their research focus or change their team.

Joining GRAND can be useful for building careers. For junior scholars, collaboration can provide multiple publications with leading scholars as well as good reference letters from them. Although the top people may not necessarily be members of the same project, they are usually professors in the same field who are familiar with other scholars' work.

13.5.1.2 Intellectual Stimulation

Pragmatic concerns are neither the only nor the most important rationale for joining GRAND. Many members are senior researchers who often have good access to funding. As other studies have found, academics are often driven by the joys of intellectual challenges, exploring new ideas, developing new paradigms, and finding new methods [42]. Research collaborations, especially large collaborations such as GRAND, allow them to tackle big questions by pooling expertise, student power, facilities and equipment. A PNI says, "Funding is great. I am glad I have the funding. But now you can put more minds to bear on a particular problem you cannot solve by yourself." While researchers have always understood the benefits of collaboration, GRAND membership provides an additional push; "it is supposed to be kind of opportunity – people will be kind of pushed into a direction and I think that will depend more on micro-level collaboration."

GRAND's interdisciplinary nature is also an incentive for researchers who want to expand their intellectual territory. For instance, one of the principal investigators in computer graphics plans to work with computer scientists who would offer new approaches in his area. Other researchers aim to bridge different disciplines.

13.5.1.3 Networking

Although scholars have studied the pragmatic and intellectual rationales of researchers, they have not studied the rationales that researchers have for networking with each other. Yet, GRAND researchers extensively report that they have joined the network and the project because of the caliber of other members. Many researchers want to continue collaborating with colleagues they know, trust and respect. As a PNI says "It's my community. These are people I work with anyway. I was having fun at the reception last night because it's like old friends you're just getting in contact with again." One researcher says, "to be in GRAND is to be in the gang of the cool kids".

GRAND also provides rare opportunities for networking across disciplines and across distance. The sheer scale of GRAND facilitates networking as well as access to resources, knowledge, and ideas from distant collaborators. The variety of disciplines ensures the diversity of ideas and resources. In addition, GRAND can be, as one PNI believes, "the gateway to better research in a global community." Networking opportunities expand in time as well: collaborative networks built in GRAND may last longer than GRAND itself and could be further extended by the graduate students working on projects when they become the next generation of faculty members.

13.5.2 Relationships in the GRAND Network

Examining different ties among members provides insight into what holds GRAND together. GRAND members provided information about a range of social and professional relationships as well as their communication. We examined each type of tie separately, and we did not always assume that ties that went in one direction also went in the opposite direction. For instance, we treated giving advice and receiving advice as different types of ties because such exchanges are often status-based and asymmetrical: people who give advice are not those who receive advice.

The weakest tie, just Knowing another GRAND member, is by far the most numerous. Members of professional communities such as GRAND often know many others because they meet at conferences, exchange graduate students, or collaborate on grant proposals. In addition to such common foci of interaction, GRAND members know each other because they were recruited in a snowball process: the core group of researchers invited their long-term collaborators who in turn invited their own collaborators.

The next most numerous ties are those of Friendship and Work With. Collaboration is the official reason for joining GRAND, and friendship and collaborator

Table 13.1 QAP correlations between social and communication networks in GRAND

	Received advice	Gave help	Received help	Work	Email (all)	Email (strong)	In-person (all)	In-person (strong)
Gave advice	0.789	0.615	0.580	0.579	0.573	0.451	0.510	0.387
Received advice		0.516	0.611	0.296	0.591	0.460	0.515	0.386
Gave help			0.603	0.463	0.466	0.348	0.389	0.257
Received help				0.492	0.509	0.393	0.427	0.290
Work					0.732	0.525	0.621	0.465
Email (all)						0.673	0.802	0.608
Email (strong)							0.579	0.647
In-person (all)								0.766

sometimes coincides [25]. The recruitment practices in GRAND may have also contributed to the relatively strong presence of friendship ties. Further behind in their numbers are Gave Advice, Received Advice, Gave Networking Help, Received Networking Help, and Coauthoring publications.

If the number of ties and the mean density of their interconnections indicate how important each type of ties is in holding the network together, correlations between networks of different types of ties reveal their overlap and shed light on how processes unfold in GRAND. We used QAP (Quadratic Assignment Procedure) to see how correlated two networks are with each other [33]. The higher the QAP correlation between two networks, the more likely it is that the same GRAND members have both relationships, such as "Giving Advice" and "Working Together". Thus, higher correlations show that different relationships connect the same people thereby creating similar networks (Table 13.1).

Among all social networks in GRAND, networks of giving advice and receiving advice are the most highly correlated with other networks. Both the gave- and received advice networks have correlation values higher than 0.5 with all other social networks except for the Coauthorship network. All that connects GRAND members – working, networking, being friends, or even just knowing each other – become opportunities for advice exchanges.

Various types of professional help are reciprocal and flow into one another. GRAND members gave advice and received advice from the same people; this is the QAP highest correlation between all networks (0.789). In many networks, advice exchanges are often status-based and asymmetrical: established experts provide advice to junior colleagues but not vice versa. However, in an interdisciplinary network such as GRAND, advice exchanges become reciprocal: members consult each other in their areas of expertise, both giving and receiving advice from the same colleagues. Moreover, advice exchanges and networking are correlated so that GRAND members give advice and networking help to the same persons (0.615) and receive advice and networking help from the same persons (0.633). One form of professional help easily becomes another form of help. These patterns are similar to those found in other studies of NCEs [19].

By contrast, Coauthoring publications is the least correlated with other GRAND networks. Compared with the rest of the social networks, Coauthoring is not only the most sparsely knit but also the most dissimilar network. Even its strongest correlations – with working together and advice exchanges – are relatively weak. Such low numbers and dissimilarity are consistent with the long cycle and infrequency of publications. At the time the data were collected, GRAND projects had just started and had yet to produce publications. Thus, coauthorship largely reports on past publications. However, previous research has shown that past coauthorship encourages subsequent collaboration, and that both coauthoring and collaboration are related to friendship ties [54].

Patterns of working with colleagues are especially important: collaboration is the reason for the existence of GRAND. Working with someone is strongly correlated with received advice (0.611) and gave advice (0.579): correlations with other networks are weaker. Working together – a more formal tie – thus provides opportunities for consulting colleagues and exchanging advice, although it has yet to provide networking help or coauthorship.

Friendship ties are most strongly correlated with knowing someone in GRAND and with working with them – a pattern that reflects the recruiting practices of the network and the initiation of projects. The people whom GRAND members know in the network tend to be friends and collaborators.

In sum, at the early stage of GRAND that we studied, just knowing someone is the most prevalent type of tie; work and friendship ties are the next most prevalent. Coauthoring is the scarcest. These results are understandable in the light of the goal of the network – collaboration, and the way many members were recruited – by inviting one's long-term collaborators. Advice exchanges are reciprocal and intertwined with all other ties, suggesting knowledge transfer processes among experts. Working with someone – a more structured and formal relationship – tends to encourage advice exchanges, but it has yet to become the main opportunity for other professional exchanges.

13.5.3 Modes of Communication

As GRAND researchers come from across Canada, much of their communication can be with people who do not live within walking or driving distances. Yet, GRAND members tend to be technologically savvy, and they have an array of communication tools available to them. Their communication mainly takes place through email as well as in-person interactions. Other communication media – such as internet phones, landlines or mobile phones, or social networking sites – are used to a much lesser extent.

Email is the most important communication tool of GRAND members, closely followed by in person interaction (Fig. 13.3). Of all communication networks, the email network is most highly correlated with all of the social networks and especially with the work network (0.732). Members use email in all their

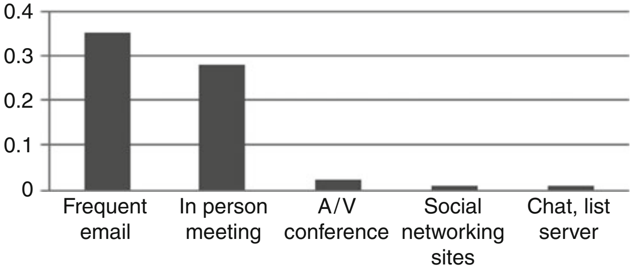


Fig. 13.3 Communication among GRAND researchers by media

relationships and are particularly likely to use email with the people they work with. In-person communication is also highly correlated with working together (0.621). Researchers either find opportunities to meet in person or they work with people who are geographically close to them. Such results confirm past results highlighting importance of in-person contact among scientists [6, 19, 25]. Our findings support and add to a wide body of research that show that email and in-person communication are strongly correlated (0.80): email does not replace but adds on to in-person communication [43].

13.5.4 Relationships Within Projects

GRAND members may connect differently within their own projects than with the overall GRAND network. Projects, as the unit of collaboration in a research network, foster different behavior. Because the average size of a project is 8.5 members, we could not use QAP. Instead, we use OLS to study correlations of network characteristics between projects.

The patterns of ties within the projects are similar to those of the whole network. However, higher densities within projects indicate that projects – the foci of formal collaboration – are more interconnected than the whole network. Researchers know, work with, and exchange professional help with more of their team members than they do with their colleagues in the overall GRAND network. At the same time, the mean densities in projects do not reach above 0.43 (Know): project members do not know all their colleagues nor do they work with all of them. This may suggests a pattern of working in small sub-groups of collaborators. While project members meet at GRAND events or professional forums, they do not necessarily collaborate with every project member they know.

Network-level and project-level density correlations also have similar patterns. Network-wide and within projects, giving and receiving advice are most strongly correlated with other types of social ties. Coauthoring publications remains the most unique network. Exchanges of advice and of networking help are reciprocal (Table 13.2).

Table 13.2 Correlations between social and communication networks in projects

	Know	Work	Gave advice	Received advice	Gave help	Received help	Coauthor	Email	In person
Friends	0.668**	0.528**	0.655**	0.580**	0.176	0.493**	0.715**	0.472**	0.289
Know		0.697**	0.768**	0.744**	0.316	0.527**	0.488**	0.626**	0.273
Work			0.709**	0.889**	0.457**	0.705**	0.452**	0.844**	0.406*
Gave advice				0.844**	0.541**	0.734**	0.446*	0.687**	0.306
Received advice					0.453**	0.793**	0.429*	0.852**	0.319
Gave help						0.612**	0.089	0.444*	−0.034
Received help							0.264	0.764**	0.024
Coauthor								0.337	0.364*
Email									0.279

*$p < 0.05$; **$p < 0.001$

GRAND's projects vary considerably in size, density and centralization [53]. Networks within projects are noticeably different than the networks that extend throughout GRAND. Within projects, people working together are likely to be past co-authors and friends. Friendship ties in projects are also correlated with all of the other types of social networks, indicating that friends working together on projects have especially rich and complex ties. Working with another project member is also correlated with all other social relationships: work ties have become almost as important as advice ties. Work ties are particularly strongly correlated with advice ties.

In short, researchers are more connected to their project team members than to colleagues elsewhere in GRAND. Yet, they neither know nor work with every member of their projects. They have multiple connections to the small subgroup of project team members they actually connect with, and especially to those of the project team members who are friends.

Within projects, as throughout GRAND, researchers rely on both email and in-person communication, with email the most common medium. The frequency of email communication is more strongly correlated than in-person communication with all professional exchanges such as received advice (0.852), work with (0.844), and received networking help (0.764).

In-person communication is more weakly correlated with professional exchanges. The highest correlation is the moderate one with working together (0.406). This suggests that at least some of the project work is either done locally, or that more distant collaborators find opportunities to meet in person. Interview data confirms both patterns: many project members made efforts to visit their project collaborators or work closely with colleagues in the same university.

By contrast to GRAND-wide communication patterns, email and in-person communication are not strongly correlated within projects. While project members use both email and in-person communication, they most likely use each of them to contact different sets of project collaborators. Coauthorship ties on project level offer an interesting twist in communication: in-person communication (0.364) is slightly more strongly correlated with coauthorship than is email (0.337). Researchers contact their co-authors on the project frequently by email and, in addition, meet them in person.

To summarize, all social and professional connections – friendships, work, advice exchanges, and networking help – are more intense within projects, than in GRAND overall. Yet, project members neither know nor work with everyone on their projects. Instead, they connect to sub-groups within projects, where collaboration, exchanges of advice, and networking help are particularly active and ties among project collaborators are especially rich and complex. Such patterns are consistent with past research suggesting that scholars tend to organize their work in small teams within projects [26].

13.5.5 GRAND as a Networked Organization?

13.5.5.1 The Extent of Hierarchical Relationships Among GRAND Members

Although there are relatively few formal hierarchical distinctions in GRAND, these can affect the resources allocated to GRAND members. For instance, a PNI gets more funding than a CNI.

To explore further whether GRAND has a flat, non-hierarchical structure consistent with networked organizations, we examine how the formal position of members affects their communication. We use Closeness Centrality (referred to as Centrality) in the Know network to indicate the position of individual members in the research networks. Higher values of Centrality mean fewer message transmissions, shorter times, and lower costs, thereby facilitating efficiency in communication. Closeness Centrality is measured by two variables: GRAND Centrality – indicating individual researchers' positions in GRAND communication networks – and Project Centrality – indicating their positions within projects. Since both formal position in GRAND and the two centrality measures might be influenced by academic seniority and experience, we use as control variables Academic Rank (assistant professor, associate professor, and professor) and Age (gender was not significant). A two-step analysis demonstrates the impact of formal position on the communication efficiency of researchers. First, social network analysis using ORA software produced the centrality measures of researchers. Second, partial correlation analyses show the relationships between Centrality and Formal position.

The mean value of GRAND Centrality is 0.24, while that of Project Centrality is 0.41. This suggests that researchers communicate with their own project members more efficiently than with other GRAND members since they know more collaborators in their projects and often have shorter geographic distances and disciplinary distances than in the overall GRAND network.

The formal position of researchers in GRAND also affects their communication efficiency. Academic rank and age mediate the relationship between formal position in GRAND and overall Centrality: academic rank is the principal mediator. When controlling for academic rank and age, the partial correlation between Project Centrality and Formal Position increases to 0.36. This indicates a significant relationship between the researchers' formal positions and their centrality in projects.

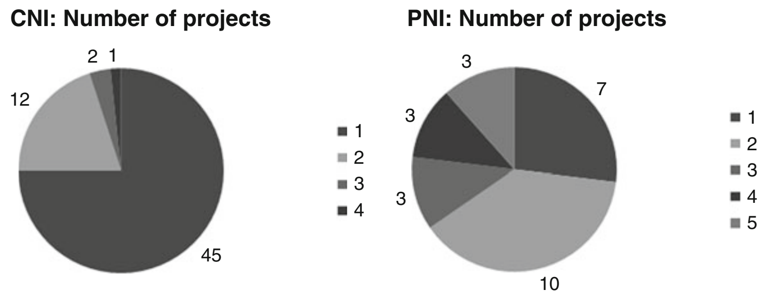


Fig. 13.4 GRAND members' participation in multiple projects

In other words, researchers with higher formal positions in GRAND are more likely to occupy more central positions in the overall GRAND network, either because their formal position led to their centrality or because their centrality in the professional community has resulted in their high formal positions. Similarly, within projects, GRAND members with higher positions communicate more efficiently. Centrality in GRAND and centrality in projects are correlated, suggesting the consistent advantages of higher-position members in their communication. This is in line with interview data indicating that some junior members have trouble accessing information about GRAND activities. As one CNI put it, "sometimes I feel really out of the loop."

Thus, despite GRAND's emphasis on expertise, formal positions in GRAND reflect differences in communication. GRAND members in higher formal positions have advantages in communicating with other members of GRAND as well as with their project members. This is consistent with other research showing that hierarchy in communication structures coexist with relatively flat authority structures [1].

13.5.5.2 Collaboration Across Project Boundaries

Ties across organizational boundaries are a hallmark of networked organizations. GRAND leadership facilitates connections across projects. PNIs – the higher-position researchers – are strongly encouraged to participate in at least three projects, and all GRAND researchers are encouraged to participate in multiple projects. GRAND thus has dual interpersonal and inter-project networks: its researchers are linked by affiliations with projects, and its projects are linked by researchers [3].

Even at the early stages of the network, most GRAND members (52 %) are already working in several teams (Fig. 13.1). PNIs work in a mean of 2.9 projects; some of them have as many as five projects. As Fig. 13.4 shows, 44 % are working in more than three projects and 24 % of them are working with three projects. By comparison, CNIs are working in an average of 1.3 projects; 75 % work in a single project.

Table 13.3 Density of interactions for work ties

	CS	ENGR	A&T	HUM	SS	PRO	IS	MED
Computer sciences	0.12	0.09	0.05	0.02	0.02	0.02	0.02	0.03
Engineering		0.11	0.06	0.05	0.02	0.05	0.04	0.03
Art and technology			0.35	0.13	0.06	0.10	0.05	0.00
Humanities				0.14	0.06	0.08	0.06	0.00
Social sciences					0.02	0.03	0.03	0.00
Professions						0.04	0.02	0.00
Information science							0.05	0.00
Medicine								0.17

CS computer science, *ENGR* engineering, *A&T* art and technology, *HUM* humanities, *SS* social science, *PRO* professions, *IS* information science, *MED* medicine

Thus, GRAND members do not all cross organizational boundaries to the same extent. PNIs – the higher-position researchers – are more engaged in multiple projects than the usually more junior CNIs. In addition, because PNIs bridge inter-group communication across projects, they are able to contact other project members in shorter times and with lower costs. This is an important way in which information flows between projects.

Thus, the design of GRAND, intended to foster ties across projects through PNIs, contributes to a stronger role of hierarchical differences in GRAND and specifically to efficient communication for higher-level members. Paradoxically, fostering one aspect of the non-traditional networked organizations – cross-organizational flows, is associated with another aspect of traditional bureaucratic organizations – hierarchy.

13.5.5.3 Collaboration Across Disciplines

To what extent do GRAND researchers, coming from a range of disciplines, work with colleagues within or outside of their own discipline? Ties within disciplines and sub-disciplines foster efficient work because of shared lore. Yet, bridging ties across disciplines span intellectual boundaries, aiding the transfer of knowledge.

Density measures suggest that disciplines follow two distinct patterns. Researchers in Arts and Technology, Medicine, Computer Science, and Engineering work most actively with colleagues in their own disciplines: the most internally discipline is Arts and Technology (Table 13.3). By contrast, researchers in Humanities, Social Sciences, Professions, and Information Sciences collaborate most actively with colleagues outside their discipline: the strongest external connection is Humanities with Arts and Technology.

While all density values are low, the relatively lower values between disciplines suggest even less interactions. Computer Science, the largest discipline in the network, shows a number of particularly low values for interactions with other disciplines. When computer scientists do collaborate, it is most likely with researchers in Engineering and – to a much lesser extent – with those in Arts and Technology. There is little interdisciplinary collaboration between them and Social Sciences, Humanities, and Information Science: a pattern akin to the proverbial arts and sciences divide.

Arts and Technology researchers, themselves interdisciplinary, play a special place in GRAND collaborations: they have the highest proportion of co-working ties both within their own discipline and with other disciplines. Moreover, Arts and Technology researchers are either the most active collaborators with the Humanities, Professions, and Social Sciences; or the second most active collaborators with Computer Sciences and Engineering. This is all the more important because they work with researchers in Humanities, Professions, and Social Sciences who do not collaborate actively with Computer Scientists. Humanities, Professions, and Social Sciences researchers work outside their own discipline, but much of their interdisciplinary collaboration remains within the liberal arts and professions: only their ties to Arts and Technology take them across the liberal arts boundary.

Information Science, the second largest discipline in the network, has its own distinctive patterns. Similar to disciplines in the liberal arts and traditional professions, a relatively high percentage of Information Sciences affiliations are directed to other disciplines in the liberal arts, traditional professions, and Arts and Technology. Few are to Computer Science. However, Information Scientists also collaborate with engineers. In this way, their work ties cross the divide between sciences and engineering, on the one hand, and liberal arts disciplines, on the other.

To summarize, while ties across disciplines do exist, interdisciplinary collaboration is not extensive and there is some evidence of a division between Computer Science and Engineering on the one hand, and Social Sciences, Humanities, and traditional Professions on the other. Two disciplines – Arts and Technology and Information Science – are more apt to bridge the liberal arts and the sciences while those in the liberal arts are more likely to connect with other researchers in other liberal arts disciplines. These complex results are in line with studies of scientific collaboration suggesting that unique structural and cultural conditions in each discipline encourage collaborative behavior to a different degree [4].

13.5.5.4 Collaboration Across Provinces

In a dispersed organization such as GRAND, connections across distance are a salient part of boundary-spanning flows. GRAND's universities tend to cluster in near-by cities within provinces, and major Canadian universities tend to be separated in different provinces. Hence, we can use work ties across provinces as indicators of distant connections. Ties across distance are captured by the E-I index: a measure that reveals whether the ties of group members are directed internally or externally [34].

The E-I index (Table 13.4) shows that in the four largest provinces, researchers tend to work within their own province. The larger the province and the more network members there are from that province, the more likely they are to work

Table 13.4 E-I Index output for work network by province

	Internal	External	Total	E-I
Ontario	236	182	418	−0.13
British Columbia	286	177	463	−0.24
Alberta	96	68	164	−0.17
Quebec	68	64	132	−0.03
Nova Scotia	0	18	18	1.00
Saskatchewan	6	28	34	0.65

within the province. Disciplinary composition in each province may also affect these results. For instance, network members from computer science-related disciplines work more internally, within their discipline, than externally, with members from other disciplines. In British Columbia and Alberta, where computer scientists comprise about two thirds of the network participants from the province, the propensity for working within the discipline may contribute to the propensity for working within the province. By contrast, in Ontario and Quebec, with large numbers of members from other disciplines, the prevalence of internal ties may be due to the availability of researchers from diverse disciplines.

In short, within-province affiliation is the norm. Work and communication in GRAND are consistent with the glocalization patterns discovered by previous research: networked employees work both locally and globally, but predominantly locally [41].

13.6 Conclusions

First, the GRAND network is well positioned for cross-boundary flows: the members are substantially diverse in terms of disciplines, university affiliations, and locations while the organization's formal rules and procedures foster links across projects, disciplines, and locations. The network has attracted researchers who – in addition to pragmatic considerations of funding and career building – are interested in the intellectual stimulation of diverse collaboration and in networking with the right crowd. Its researchers want to challenge new research questions, find new methods, and build new paradigms. Their rationales bode well for the collaboration in GRAND as past research has found that intrinsic motives encourage more interaction than extrinsic, and intellectual stimulation provides a stronger incentive for collaboration than economic rewards [29, 42]. At the beginning, many members initially connect with their own disciplines and provinces; this is consistent with other research emphasizing difficulties of cross-disciplinary and distant ties [17, 38]. Yet, the nature of GRAND has created a structure that is fostering cross-disciplinary contact and work.

Second, in GRAND's early stage, Knowing, Working With, and Friendship are the most numerous ties connecting the GRAND network. The preponderance of these ties is consistent with the recruitment practices and with the goals of the network. By contrast, the fewest ties are past Coauthorships: a reasonable scarcity in GRAND's early days. Advice exchanges are the ties most strongly correlated with all other ties: whatever else people are doing in GRAND, they exchange advice. This result, congruent with past research on other NCEs, is a strong indicator for knowledge transfer exchanges and learning in scholarly networks [18,19]. Although further research is needed to substantiate such findings, the role of advice exchanges in GRAND suggests that research networks – and perhaps professional networks in general – can function as a distributed Community of Practice.

All types of ties are more numerous within projects, the basic units of collaboration, although project members neither know nor work with everyone on a project. Moreover, ties overlap more within projects: for example, friends exchange more advice. Researchers tend to work with small sub-groups of team members within projects: the projects themselves really are networks of sub-projects. There is a limit to how much connectivity the researchers have, probably because previous research has shown that scholars tend to organize their work to maximize independence and minimize communication and coordination [17, 26].

Within projects, the basic unit for funding and production, all types of networks are more strongly correlated than the overall GRAND network. Friendship and coauthorship are associated as people who like each other, work together. In large part, it is these friendships that led them to GRAND.

Third, in the distributed GRAND network, researchers communicate with their colleagues mostly via email: one-to-one, in small groups, and in larger lists. In-person contact is almost as frequent. These results are congruent with other studies' findings of the importance of in-person scholarly communication [6, 19]. Previous studies also have shown that for some tasks, such as brainstorming, in-person contact is the most effective form of scholarly interaction [38]. Moreover, email and in-person communication are correlated. They are used to contact the same GRAND colleagues. There is media overlap rather than specialization. This is consistent with findings suggesting that complex ties are maintained by several media [25, 43].

Somewhat counter-intuitively, communication in projects relies more on email than does communication throughout all of GRAND. Yet, GRAND projects are dispersed by design, and sub-projects with intense collaboration need not be collocated. Email emerges as the more versatile medium used by intensely collaborating project members.

Despite GRAND members' digital media savvy, they rarely use internet phones, mobile phones, or social networking sites such as forums and wikis. Yet, GRAND members say they are looking for better collaborative tools. In the future, they may incorporate more sophisticated collaborative tools and expand the use of a particular medium. Or, projects may specialize, with each adopting its own informal communication practices.

Fourth, as a networked organization, GRAND is designed with few formal hierarchical differences. Yet, hierarchy matters when it comes to communication. The differences in formal positions are related to centrality in communication structures, both GRAND-wide and within projects. This suggests that GRAND researchers in higher positions have consistent advantages in their communication based on fewer message transmissions, shorter time, and lower costs of communication. Strategic position in communication flows of networked organizations may lead to superior performance and further advantages. The future evolution of the GRAND network will show whether these communication advantages for high-position researchers persist, disappear, or increase.

It is the design of GRAND – intended to foster the permeability of project boundaries – that fosters the impact of hierarchical differences on communication efficiency. Shifting between teams enables networked workers to expand their networks. GRAND's requirement for higher-level researchers to participate in several projects expands their networks, places them in bridging positions, and leads to their communication advantages.

Such a pattern is significant as it may mean a trade-off in the structural characteristics of the networked organizations where cross-boundary flows strengthen hierarchal communication. Early discussions about both online (virtual) organizations and networked organizations expected such organizations to be non-hierarchical [48] and decentralized [2]. Yet, hierarchy and formalization can aid large collaborative networks [46]. As Ahuja and Carley [1] showed, virtual organizations can exhibit considerable hierarchical and centralization tendencies in their communication structures. In virtual organizations, they concluded, decentralized authority structures co-exist with centralization and hierarchy that assists efficiency of communication. Our results support this argument by demonstrating the central position of higher level GRAND researchers in communication despite the relatively flat authority structure. It also points to the design of the GRAND – specifically the deliberate links across projects – that are a mechanism for fostering communication hierarchy.

Fifth, boundary spanning exchanges in GRAND are low at the network's early stages. While ties across disciplines do exist, they are not extensive. There is little collaboration between Computer Science and Engineering, on the one hand, and Social Sciences and Humanities, on the other. Researchers in Arts and Technology emerge as the most active collaborators in the network both internally and externally. They play an important role in the network by working with researchers on both sides of the arts and sciences divide. Information Science, the second discipline that is most apt to cross boundaries, has fewer collaborative ties.

Sixth, collaboration within provinces is the norm. Working across provinces – just like working across disciplines – is not extensive. Work flows across provinces are not abundant. Canada is a big country, and there are few places where participating universities in different provinces are near each other.

The low level of cross-boundary collaboration – both across disciplines and across provinces – is in line with previous findings. It is difficult to achieve cross-disciplinary collaboration, due to language, epistemological, and resource differences that can hinder the formulation of visions, goals, and tasks [39]. When scientists collaborate with others from different sectors, organizations, communities, and countries, additional challenges may arise from different perspectives regarding what constitutes a research goal, realistic tasks, and task completion time frames [47].

In this analysis, the low level of cross-boundary collaboration can also be explained by the early stage at which the network was studied. Yet, such an interpretation may not be the full story. Quan-Haase and Wellman [41] found patterns of glocalization in a networked organization: its members were connected glocally – both locally and globally – with a high proportion of communication occurring within the local work unit and within the organization rather than further afield. GRAND researchers demonstrate similar glocal patterns with respect to disciplines and locations. Such glocal patterns may not be the sign of undeveloped cross-boundary flows but an integral part of the way networked organizations function. The scholarly world is not flat; it is lumpy.

Early discussions viewed networked organizations as the antithesis of traditional bureaucratic organizations. These discussions expected key characteristics of bureaucracies such as hierarchy, centralization and formalization to be absent. Empirical research, scarce as it is, reveals a more complex picture. Traditional bureaucratic properties co-exist with new post-bureaucratic ones, emergent communication structures overlay old authority structure and functional divisions, and performance is differently determined [1, 35, 46]. Our research shows that reality is more complex than the early deductive expectations for networked organizations.

Acknowledgements We thank GRAND for its support of its projects – and NAVEL in particular – and Lilia Smale for her editorial help.

References

1. Ahuja, M. K., & Carley, K. M. (1999). Network structure in virtual organizations. Organization Science, 10(6): 741–757.
2. Baker, W. (1992). The network organization in theory and practice. N. Nohria, R. Eccles, eds. Networks and Organizations. Boston, MA: Harvard Business School Press.
3. Berg, B. L. (2009). Qualitative Research Methods for the Social Sciences (7th edition). Boston: Allyn & Bacon.
4. Birnholtz, J. (2005). When do researchers collaborate? Toward a model of collaboration propensity in science and engineering research. Ph.D. dissertation, University of Michigan.
5. Black, J. & Edwards, S. (2000). Emergence of virtual or network organizations: fad or feature. Journal of Organizational Change, 13(6): 567–576.
6. Bos, N., Gergle, D. Olson, J. & Olson, G. (2001). Being there versus seeing there: trust via video. Proceedings of the CHI 2001 conference, Seattle. http://www.crew.umich.edu/publications.html.
7. Bos, N., Zimmerman, A., Olson, J., Yew, J., Yerkie, J., Dahl, E., & Olson, G. (2008). From shared databases to Communities of Practice: a taxonomy of collaboratories. Journal of Computer-Mediated Communication, 12(2):318–338.
8. Breiger, R. (1974). The duality of persons and groups. Social Forces, 53:181–190.
9. Bresnen, M., Edelman, L., Newell, S., Scarbrough, H. & Swana, J. (2003). Social practices and the management of knowledge in project environments. International Journal of Project Management, 21(3):157–166.
10. Burt, R., (2010). Neighbor Networks. New York: Oxford University Press.
11. Cappelli, P., Bassi, L., Katz, H., Knoke, D., Osterman, P., & Useem, M. (1997). Change at Work. New York: Oxford University Press.

12. Caruso, D. & Rhoten, D. (2001). Lead, follow, get out of the way: sidestepping the barriers to effective practice of interdisciplinarity. Report for the Hybrid Vigor Institute, http://www.hybridvigor.net/publications.pl?s=interdis&d=2001.04.30#.
13. Chen, W., Rainie, L, and Wellman, B. (2012). Networked Work. Chapter 7 in L. Rainie and B. Wellman, Networked: The New Social Operating System. Cambridge, MA: MIT Press.
14. Crane, D. (1972). Invisible Colleges: Diffusion of Knowledge in Scientific Communities. Chicago: University of Chicago Press.
15. Creswell, J. W. and Clark, V. L. P. (2007). Designing and Conducting Mixed Methods Research. Thousand Oaks, CA: Sage.
16. Cross, R., Borgatti, S.P., & Parker, A. (2002). Making invisible work visible: using social network analysis to support strategic collaboration. The Network Roundtable at the University of Virginia, https://webapp.comm.virginia.edu/SnaPortal/portals%5C0%5Cmaking_invisible_work_visible.pdf.
17. Cummings, J. & Kiesler, S. (2005). Collaborative research across disciplinary and organizational boundaries. Social Studies of Science, 35(5): 703–722.
18. Dimitrova, D. & Koku, E. (2009). Research communities in context: trust, independence and technology in professional communities. In D. Akoumianakis (Ed.), Virtual community practices and social interactive media: Technology lifecycle and workflow analysis (pp. 352–377), Hershey, PA: IGI Global.
19. Dimitrova, D., & Koku, E. (2010). Managing Collaborative Research Networks: The Dual Life of A Virtual Community of Practice. International Journal of Virtual Communities and Social Networking, 2(4): 1–23.
20. Dimitrova, D., Koku, E., Wellman, B., & White, H. (2007). Network Mapping Study. Final Report to the Canadian Water Network of Centres of Excellence.
21. Florida, R. (2002). The Rise of the Creative Class. New York: Basic Books.
22. Friedman, T. (2007). The World is Flat. New York : Farrar, Straus and Giroux.
23. Galison, P. & Hevly, B. W. (1992). Big Science: the growth of large-scale research. Stanford, CA: Stanford University Press.
24. Hanneman, R. A. & Riddle, M. (2005). Introduction to social network methods. Riverside, CA: University of California, Riverside http://faculty.ucr.edu/~hanneman/nettext/Introduction_to_Social_Network_Methods.pdf.
25. Haythornthwaite, C. & Wellman, B. (1998). Work, friendship and media use for information exchange in a networked organization. Journal of the American Society for Information Science. 49(12): 1101–14.
26. Haythornthwaite, C. et al. (2003). Challenges in the practice and study of distributed, interdisciplinary collaboration. GSLIS Technical Report No.: UIUCLIS–2004/1+DKRC, http://www.lis.uiuc.edu/~haythorn/hay_challenges.html.
27. Hey, T., & Trefethen, A. (2008). E-science, cyber-infrastructure, and scholarly communication. In G. Olson, A. Zimmerman, and N. Bos (Eds.), Scientific Collaboration on the Internet (pp. 15–33). Cambridge, MA: MIT Press.
28. Hollingshead, A., & Contractor, N. (2002). New media and organizaing at the group level. In Lievrouw, L. A., & Livingstone, S. M. (Eds). Handbook of new media: Social shaping and consequences of ICTs. London: Sage.
29. Howley, I., Chaudhuri, S., Kumar, R. and Ros, C. P. (2009). Motivation and collaboration on-line DOI: http://celstec.org/system/files/file/conference_proceedings/aeid2009/papers/paper_243.pdf.
30. Jarvenpaa, S., & Leidner, D. (1999). Communication and trust in global virtual teams. Organization Science, 10(6): 791–815.
31. Koku, E., Nazer, N., & Wellman, B. (2001). Netting scholars: online and offline. American Behavioral Scientist, 44(10): 1752–74.
32. Kollock, P. (1999). The Economies of Online Cooperation: Gifts and Public Goods in Cyberspace. In M. Smith & P. Kollock (Eds.), Communities in Cyberspace. London: Routledge.
33. Krackhardt, D. (1988). Predicting with networks: nonparametric multiple regression analysis of dyadic data. Social Networks. 10 (4): 359–381.

34. Krackhardt, D. & Stern, R. (1988). Informal networks and organizational crises: an experimental simulation. Social Psychology Quarterly 51(2), 123–140.
35. Krebs, V. (2007). Managing the 21st Century Organization. Institute of Human Resources and Information Management, 11(4): 2–8.
36. Mortensen, M., Woolley, A. W., & O'Leary, M. (2007). Conditions Enabling Effective Multiple Team Membership. International Federation for Information Processing Report No. 236.
37. Negroponte, N. (1995). Being digital. New York: Knopf. Olson, G. & Olson, J. (2003). Mitigating the effects of distance on collaborative intellectual work. Economic Innovation and New Technologies. 12(1): 27–42.
38. Olson, G. M., & Olson, J. S. (2000). Distance matters. Human Computer Interaction, 15: 139–178.
39. Olson, J. S., Olson, G. M., & Cooney, D. (2008). Success factors: Bridging distance in collaboration. In G. M. Olson, A. Zimmerman, & N. Bos (Eds.), Science on the Internet. Cambridge, MA: MIT Press.
40. Olson, J., Hofer, E., Bos, N., Zimmerman, A., Olson, G. D. Cooney, G., Faniel, I. (2008). A theory of remote scientific collaboration. In G. Olson, A. Zimmerman, and N. Bos (Eds.), Scientific Collaboration on the Internet (pp. 73–99), Cambridge, MA: MIT Press.
41. Quan-Haase, A., & Wellman, B. (2004). Groups and networks : local virtuality in a high-tech networked organization. Analyse & Kritik, 26(1): 241.
42. Rafaeli, S., & Ariel, Y. (2008). Online motivational factors: incentives for participation and contribution in Wikipedia. In A. Barak (Ed.), Psychological aspects of cyberspace: Theory, research, applications (pp. 243–267). Cambridge: Cambridge University Press.
43. Rainie, L. & Wellman, B. (2012). Networked: The New Social Operating System. Massachusetts: MIT Press.
44. Reichardt, J. & Bornholdt, S. (2006). Statistical mechanics of community detection, Physical Review, 74, 016110.
45. Rhoten, D. (2003). National Science Foundation BCS-0129573.A multi-method analysis of the social and technical conditions for interdisciplinary collaboration. Report . Hybrid Vigor Institute http://hybridvigor.net/interdis/pubs/hv_pub_interdis-2003.09.29.pdf.
46. Shrum, W., Chompalov, I., & Genuth, J. (2001). Trust, conflict and performance in scientific collaborations. Social Studies of Science, 31(5):681–730.
47. Sonnenwald, D. H. (2008). Scientific collaboration. Annual Review of Information Science and Technology, 41(1): 643–681.
48. Sproull, L., and Kiesler, S. (1986). Reducing social context cues: electronic mail in organizational communication. Management Science, 32(11):1492–1512.
49. Taylor, J. R. (1999). The other side of rationality: socially distributed cognition. Management Communication Quarterly, 13(2):317–26.
50. Walters, D., & Buchanan, J. (2001). The new economy, new opportunities and new structures. Management Decision, 39(10): 818–834.
51. Wasserman, S. & Faust, K. (1994). Social Network Analysis: Methods and Applications. Cambridge: Cambridge University Press.
52. Wenger, E., McDermott, R., & Snyder, W. (2002). Cultivating Communities of Practice. Boston: Harvard Business School Press.
53. Wellman, B. & Zhuo, X. (2012). Structural variation in scholarly teams: size, density and centralization. Presented to the International Social Network Conference, Redondo Beach, CA. March.
54. White, H., Wellman, B., & Nazer, N. (2004). Does citation reflect social structure? Longitudinal evidence from the 'Globenet' interdisciplinary research group. Journal of the American Society for Information Science, 55(2): 111–126.
55. Wu, L., Lin, C-Y., Aral, S., & Brynjolfsson, E. (2009). Value of social network. Presented to the Winter Information Systems Conference, Salt Lake City, February http://smallblue.research.ibm.com.
56. Zheng, J., Veinott, E., Bos, N., Olson, J. & Olson, G. (2002).Trust without touch: jumpstarting long-distance trust with initial social activities. Proceedings of CHI. New York: ACM Press http://www.crew.umich.edu/publications.html

Chapter 14
How Al Qaeda Can Use Order Theory to Evade or Defeat U.S. Forces: The Case of Binary Posets

Jonathan David Farley

Abstract Terrorist cells are modeled as finite partially ordered sets. This paper determines the structure of the terrorist cell most likely to remain intact if a subset of its members is captured at random, provided that the cell has a single leader and no member has more than two immediate subordinates.

14.1 Introduction

I arrived at Ted K.'s cell. "Cell" was the wrong word: As the door swung open, I walked into what you would think of as a plush apartment. All that would strike you as strange was the absence of any windows.

"I still can't get over how well they take care of you here," I said, shaking my head. "Is that flat-screen TV new? I'm surprised they let you watch the news."

Ted was sitting at a large glass table, papers with his neat handwriting littering the surface—ordered chaos. His grey-black beard and hair were as wild as ever. "Don't worry: it's just for playing video games."

I sat down, folded my hands and said, "You have something for me?"

Ted smiled, the wrinkles around his eyes deepening. "As you know, for the last five years I have been working on the problem of creating the perfect terrorist cell."

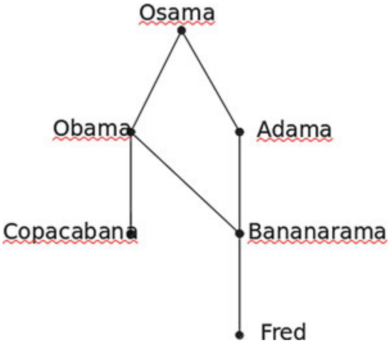
"From inside prison?" I asked with a cold smile.

J.D. Farley (✉)
School of Computing and Information Science, 237 Neville Hall, University of Maine, Orono, ME 04469, USA

Institut für Algebra, Johannes Kepler Universität Linz, Altenbergerstraße 69, A-4040 Linz, Österreich
e-mail: lattice.theory@gmail.com

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_14,

Fig. 14.1 A poset example



“To be precise, I want to determine the structure of the terrorist cell that can best withstand your government’s attempts to destroy it—a cell that, if a certain number of its members were to be captured at random, would still have the best possible chance of being able to carry out its mission.”

“And why would you want to do that?”

“To settle a score. That fellow from the NSA said my research wasn’t useful. Wait till Al Qaeda gets a hold of it: Then your military will wish it had given me funding when they had the chance.”

I usually let Ted go on a while with his *the-fools-I’ll-show-them-all* monologue, but today I had little time. “And you’ve succeeded?”

Ted’s shoulders sagged slightly. “Not quite, but I’m getting close.”

“You remember that we model terrorist cells as partially ordered sets, or *posets*, which are essentially organization charts where commands flow from the top to the bottom.” He sketched a diagram, which he labeled, “Fig. 14.1.”

“In this example, Osama sits at the top of the organization. He can pass plans down to his immediate subordinates, Obama”—I rolled my eyes—“and Adama, but only them. Osama must rely on Obama and Adama to pass his orders further down the organization: Adama to Bananarama, his only immediate subordinate, and Obama to either Bananarama or Copacabana.

“Eventually Osama’s orders reach the foot soldiers, Copacabana and Fred, who are the people who will actually carry out the attacks.

“Your government’s goal,” Ted continued, “is to arrest—”

“Or kill.”

“Arrest or kill enough terrorists so Osama’s plans cannot reach the foot soldiers. For example, you could capture Osama himself, or both Obama and Adama, or both Obama and Fred.”

“But not Copacabana and Adama,” I said impatiently, “because then Osama’s orders could still get to a foot soldier, Fred, via Obama and Bananarama. I read Fraly’s monograph, you know.”

Ted smiled wanly. “Then you know that the subgroups of terrorists you want to capture are called *cutsets*. So the cell with the fewest cutsets will be the hardest for your government to disrupt.”

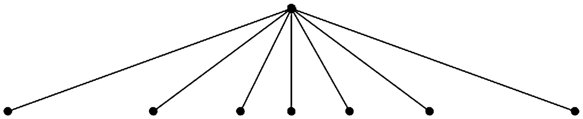


Fig. 14.2 A spider

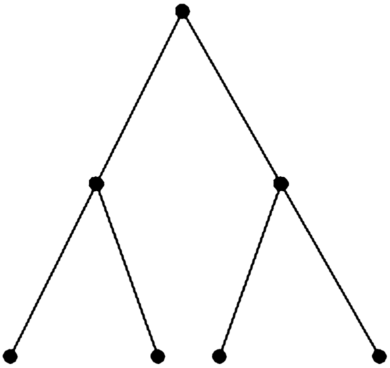


Fig. 14.3 A binary tree with seven terrorists

Ted sat back and put his hands behind his head. "Of course, the cell with a single leader that has the fewest cutsets is clearly a spider: one leader with everyone else as his immediate subordinate." He called this "Fig. 14.2." "To prevent the leader from passing his plans to at least one foot soldier, you either have to capture him or capture every foot soldier.

"But this is unrealistic," Ted continued, "because can one man really supervise 8 or 9 or 18 different people? It is much more practical," Ted said, "to suppose there is a bound, *b*—say 3 or 4—on the number of immediate subordinates a terrorist can have.

"If the cell is a tree—that is, if the cell has one leader and no one has more than one immediate superior, like Fig. 14.3—then some computer scientists in Montreal found that the best tree structure looks like this." He drew Fig. 14.4. "You have a leader with exactly *b* immediate subordinates, all of them foot soldiers except for one, who has *b* immediate subordinates, and so on, until you run out of men." He sighed.

"The problem is, there are lots of possible terrorist cell structures that are *not* trees."

"Figure 14.1, for instance," I chimed in.

"Still," he said, "Fraly looked at all binary cells—"

"'Binary'?" I asked.

"Where *b* equals 2," Ted clarified. "Fraly showed in his monograph that the best binary cell with a single leader and at most six terrorists was always one of the special trees he had discovered, the ones the Montreal group later considered, and Fraly posed a problem." He paused. "But now I have settled the issue."

I stared at him, narrowing my eyes.

"In the binary case," Ted said quietly.

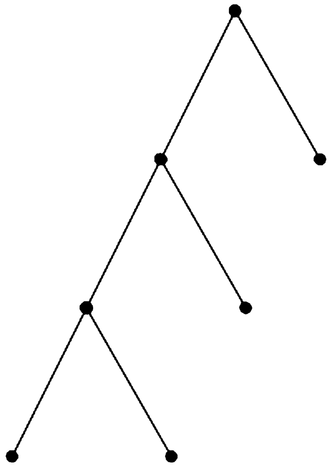


Fig. 14.4 The perfect binary terrorist tree with seven members

14.2 Proof that the Optimal Binary Terrorist Cell with One Leader Is a Pure Fishbone Poset

For order theory terminology, refer to Davey and Priestley [1] and Farley [2].

Definition 14.1. Let P be a finite poset. For p in P, let $\mathrm{LC}(p)$ be the set of lower covers of p.

Definition 14.2. Let P be a finite poset. Let k be a natural number. A k-*cutset* is a k-element subset that intersects every maximal chain of P. Let $\mathrm{Cut}(P,k)$ be the set of k-cutsets of P and let $\mathrm{cut}(P,k)$ equal $|\mathrm{Cut}(P,k)|$. A subset C of P is a *cutset* if it is a $|C|$-cutset.

Observation 14.1. Let P be a finite poset with greatest element $\top$. Assume $\mathrm{LC}(\top)=\{t,x\}$. Then $\{t,x\}$ is a cutset, and, if $\mathrm{LC}(t)$ and $\mathrm{LC}(x)$ are non-empty, $\{t\}\cup\mathrm{LC}(x)$, $\{x\}\cup\mathrm{LC}(t)$, and $\mathrm{LC}(t)\cup\mathrm{LC}(x)$ are cutsets. □

Definition 14.3. Let P be a finite poset. Let b be a natural number. We say P is b-*ary* if, for all p in P, $|\mathrm{LC}(p)|\leq b$; if $b=2$, we call P *binary*.

Definition 14.4. For a natural number n, define the poset $\mathrm{FP}(n)$ as follows:

$$\mathrm{FP}(0) := \emptyset$$

$$\mathrm{FP}(1) := \mathbf{1}, \text{ the one-element poset}$$

$$\mathrm{FP}(k+2) := [\mathrm{FP}(k)+\mathbf{1}]\oplus\mathbf{1} \text{ for all } k\geq 0.$$

Example 14.1. The posets $\mathrm{FP}(n)$ are shown in Fig. 14.5 for $1\leq n\leq 6$.

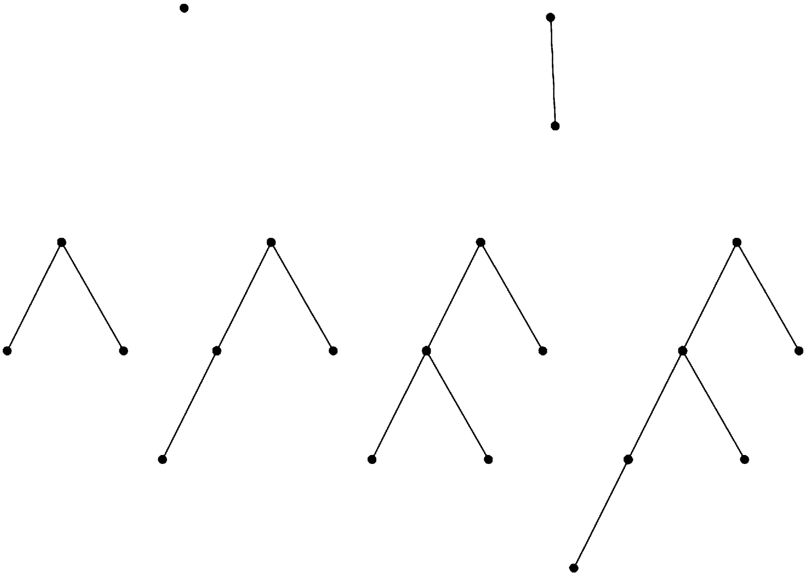


Fig. 14.5 The posets $FP(n)$ for $1 \leq n \leq 6$

Note 14.1. For $n \geq 1$, FP(n) is called in Farley [3], Definition A0.1 the *pure fishbone poset of type* $(\frac{n+1}{2}, \frac{n-1}{2}; 1, 2, \ldots, \frac{n-1}{2}; n)$ if n is odd and $(\frac{n+2}{2}, \frac{n-2}{2}; 1, 2, \ldots, \frac{n-2}{2}; n)$ if n is even.

Proposition 14.1. *For $n \geq 3$, cut(FP(n),1)=1.*

$$\text{For } n \geq 3,\ \text{cut(FP}(n),2)= \begin{cases} n+1 & \text{if } n=4 \\ n & \text{if } n \neq 4. \end{cases}$$

$$\text{For } n \geq 5,\ \text{cut(FP}(n),3)= \begin{cases} \binom{n-1}{2}+\binom{n-3}{1}+2 \text{ if } n=6 \\ \binom{n-1}{2}+\binom{n-3}{1}+1 \text{ if } n\neq 6. \end{cases}$$

Proof. This follows from Corollaries A0.1 and A0.2 of Farley [3] or by direct analysis. □

Corollary 14.1. *Let P be a finite poset with a greatest element $\top$ that has exactly two lower covers, t and x.*

(1) Let $a = 1$ if $|P| = 4$ and let $a = 0$ otherwise. If

$$|\{C \in \text{Cut}(P,2) : \top \notin C \text{ and } C \neq \{t,x\}| > a$$

then cut(P,2) > cut(FP($|P|$),2).

(2) Let $b = 2$ if $|P| = 6$ and let $b = 1$ otherwise. If $|\mathrm{LC}(t) \cup \mathrm{LC}(x)| \geq 2$ and

$$|\{C \in \mathrm{Cut}(P,3) : \top \notin C \text{ and } \{t,x\} \not\subseteq C\}| > b$$

Then cut(P,3) $>$ cut(FP($|P|$),3).

Proof. (1) The cardinality of $\{C \in \mathrm{Cut}(P{,}2) : \top \in C\}$ is $n-1$, so by Observation 14.1,

$$\mathrm{cut}(P{,}2) > n - 1 + 1 + a.$$

Use Proposition 14.1.

(2) The cardinality of $\{C \in \mathrm{Cut}(P{,}3) : \top \in C\}$ is $\binom{n-1}{2}$;

$$|\{C \in \mathrm{Cut}(P,3) : \top \notin C \text{ and } \{t,x\} \subseteq C\}| = \binom{n-3}{1}$$

by Observation 14.1; and cut(P,3)$>$ $\binom{n-1}{2} + \binom{n-3}{1} + b$, so by Proposition 14.1 we are done. □

Lemma 14.1. *Let P be a finite binary poset with greatest element $\top$. Assume that $|P|$ is at least* 3 *and that for k equal to* 1, 2, *or* 3, *cut(P,k)≤cut(FP(|P|),k).*

Then there exists a lower cover x of $\top$ such that $P \setminus \{x, \top\}$ has a greatest element and

$$P = [P \setminus \{x, \top\} + \{x\}] \oplus \{\top\}.$$

Proof. Since $|P|$ is at least 3, $|\mathrm{LC}(\top)|$ is at least 1. If $|\mathrm{LC}(\top)|$=1, then by Observation 14.1, cut(P,1)≥ 2, contradicting Proposition 14.1.

Since P is binary, let LC($\top$)=$\{t,x\}$ where $t \neq x$. Assume for a contradiction that LC(t) and LC(x) are non-empty.

If $|\mathrm{LC}(t)| = 1 = |\mathrm{LC}(x)|$, then Observation 14.1 and Corollary 14.1(1) contradict Proposition 14.1.

Without loss of generality, $|\mathrm{LC}(t)| = 2$. If $|\mathrm{LC}(x)| = 1$, then, since $|P| \neq 4$, Observation 14.1 contradicts Corollary 14.1(1).

If $|\mathrm{LC}(t) \cap \mathrm{LC}(x)|$ equals 2 or 1, then Observation 14.1 contradicts Corollary 14.1 (1) or (2). Thus LC(t)∩LC(x)$= \emptyset$, so $|P| \neq 6$ and hence Observation 14.1 contradicts Corollary 14.1(2).

Without loss of generality, LC(x)$= \emptyset$. Then t is the greatest element of $P \setminus \{x, \top\}$ and $x \parallel y$ for all y in $P \setminus \{x, \top\}$. □

Lemma 14.2. *Let Q be a non-empty finite poset. Let x and $\top$ be distinct elements not in Q and let $P = (Q + \{x\}) \oplus \{\top\}$. Then for all $k \geq 0$,*

$$\mathrm{Cut}(P, k+1) = \{B \cup \{x\} : B \in \mathrm{Cut}(Q,k)\} \cup \{D \cup \{x, \top\}: D \subseteq Q \text{ and } |D| = k-1\}$$
$$\cup \{E \cup \{\top\}: E \subseteq Q \text{ and } |E| = k\}.$$

Proof. Let $C \in \text{Cut}(P, k+1)$. If x and $\top$ belong to C, then let D equal $C \setminus \{x, \top\}$. If $\top$ belongs to C but not x, then let E equal $C \setminus \{\top\}$. If $\top$ does not belong to C, then x belongs to C since C is a cutset. Let B equal $C \setminus \{x\}$.

If B is not a k-cutset of Q, then there is a maximal chain N of Q such that $N \cap B = \emptyset$. Then $M := N \cup \{\top\}$ is a maximal chain of P since N is non-empty, so $M \cap C \neq \emptyset$, i.e., $\emptyset \neq N \cap C = N \cap B$, a contradiction.

Conversely, if F is a subset of Q, then $F \cup \{\top\}$ and $F \cup \{x, \top\}$ are cutsets of P. If B is a k-cutset of Q, let C equal $B \cup \{x\}$. If $C \notin \text{Cut}(P, k+1)$, then there exists a maximal chain M of P such that $M \cap C = \emptyset$. Clearly $x \notin M$ and $\top \in M$, and $M \setminus \{\top\}$ is a maximal chain of Q, so $M \cap C \supseteq M \cap B \neq \emptyset$, a contradiction. □

Theorem 14.1. *Let P be a finite binary poset with a greatest element such that, for all* $k \geq 0$, *cut(P,k)≤cut(FP(|P|),k).*

Then P is isomorphic to FP(|P|).

Proof (by induction on |P|). We may assume that P has at least three elements. By Lemma 14.1, there exist x in P and a subset Q of P with a greatest element such that $P = (Q + \{x\}) \oplus \{\top\}$, where $\top$ is the greatest element of P. Clearly Q is binary.

By Lemma 14.2, for all $k \geq 0$, cut(Q,k)≤cut(FP($|P|-2$),k), so by induction

$$Q \cong \text{FP}(|P|-2).$$

Hence P is isomorphic to FP($|P|$). □

14.3 Conclusion

“So what’s that mean in English?” I pleaded.

Ted sighed. “It means that if a terrorist cell has a single leader, and each terrorist has at most two immediate subordinates, then the cell structure most likely to succeed if some of the terrorists are captured at random looks like something in Figs. 14.4 and 14.5.”

I shuffled my papers and stood up from the glass table. “Okay, we’re done here,” I said: “You’re never getting out.” I smiled.

Ted was unperturbed. “Wait, I’m not dangerous yet. I haven’t figured out how terrorists should organize their cells if each man can have three or more subordinates. There’s still work to be done. You still have time to support my research before it’s too late for your government, before Al Qaeda gets a hold of my work and uses it to evade or defeat your government’s forces.”

“I’m not worried,” I lied. I walked to the door of his cell, pausing with a shock as I opened it.

Man! Even the door handles were nice.

References

1. B. A. Davey and H. A. Priestley, *Introduction to Lattices and Order,* second edition (Cambridge University Press, 2002).
2. Jonathan David Farley, "Breaking Al Qaeda Cells: A Mathematical Analysis of Counterterrorism Operations (A Guide for Risk Assessment and Decision Making)," *Studies in Conflict and Terrorism* **26** (2003), 399–411.
3. Jonathan David Farley, *Toward a Mathematical Theory of Counterterrorism: Building the Perfect Terrorist Cell* (U.S. Army War College, Carlisle Barracks, Pennsylvania, 2007).

Chapter 15
The ABCs of Designing Social Networks for Health Behaviour Change: The VivoSpace Social Network

Noreen Kamal, Sidney Fels, Mike Blackstock, and Kendall Ho

Abstract This chapter presents the Appeal, Belonging, Commitment (ABC) conceptual framework, which describes how online social networks can be designed to motivate positive health behaviour change. The ABC Framework is based on the existing theoretical models that describe the determinants for motivating the use of online social networks and health behaviour change. Common themes are drawn from these theoretical models and combined to provide the determinants for the three emergent themes: Appeal (individual determinants), Belonging (social determinants) and Commitment (temporal determinants). Results from a questionnaire survey and interviews are presented to validate and iterate the ABC Framework. Based on these themes and their determinants, design suggestions are presented. A case study implementation of the ABC Framework is shown through the design of VivoSpace. The design strategies are interpreted to design the online social health system, VivoSpace, and the ABC Framework is used to evaluate the design. This case study shows that the ABC Framework provides the best methodology to design and evaluate an online social network that will lead to a committed user base and motivate health behaviour change.

15.1 Introduction

Maintaining good, healthy behaviour remains elusive for many. While most people realize the importance of maintaining a healthy lifestyle, they often have difficulty in managing their health. At a community level, a healthier population can lower healthcare costs; therefore, it is not surprising that many public health initiatives exist that encourage citizens to lead more healthy lives [16, 19, 44]. Healthcare may

N. Kamal (✉) • S. Fels • M. Blackstock • K. Ho
University of British Columbia, Vancouver, BC, Canada
e-mail: noreenk@ece.ubc.ca; ssfels@ece.ubc.ca; mblackst@magic.ubc.ca; kendall.ho@ubc.ca

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_15,

need to focus on wellness and prevention of illness to ensure future viability [13]. Furthermore, leading a healthy lifestyle is key to avoiding illness and vital for managing chronic diseases. In fact, self-management of one's health has been shown to be of key significance in achieving positive health outcomes for all people, healthy and sick [7, 18, 46]. We intuitively understand that our life choices are heavily influenced by family, friends, colleagues and other connections we have. In particular, our social connections heavily influence our health decisions such as diet, exercise, smoking and drinking. In fact, social networks as far as three degree away (friend of a friend of a friend) have been found to influence us in many ways including health behaviour [8, 9].

Interestingly, the use of online social networks and online social gaming has surpassed everyone's expectations, and led to a committed user-base [30]. Therefore, it is not surprising that there is increasing interest in utilizing online social networks as a technical platform for health behaviour change [31, 32]. Consideration needs to be given to the motivation for use of online social networks, as we need to ensure that users will make use of the online social network system. For this reason, we look at how the motivations for using online social networks can be understood to inform the design of a system that will motivate positive health behaviour change. We take a theoretical approach to understanding the motivation for the use of online social networks and health behaviour change. We will show that existing theoretical models both for motivating health behaviour change and for understanding the motivations in using online social networks provide a conceptual framework, the Appeal Belonging Commitment (ABC) Framework. The ABC Framework is used to inform design strategies for online social networks to motivate health behaviour change, which is a contribution to the domain of Human-Computer Interaction (HCI) literature.

This chapter has the following organization. Related works is first presented in Sect. 15.2, which shows studies that have been conducted in HCI. Then, Sect. 15.3 presents the theoretical models and their determinants for both motivation to use online social networks and motivation to change health behaviour. The ABC Framework is presented in Sect. 15.4, and it is evaluated and iterated in Sect. 15.5. Section 15.6 presents the design strategies, and the case study is presented in Sect. 15.7. Concluding remarks are provided in Sect. 15.8.

15.2 Related Work

In HCI literature, substantial number of papers have been published around the design of technologies to promote health behaviour change. This includes papers that have been specifically based on existing theoretical models. Furthermore, HCI literature has also provided papers that support personal informatics and how to best design and interact with the information. These foundational works are described here. The reason that previous works in personal informatics is being described alongside works in health behaviour change is based on the assumption that personal health informatics can be used to change health behaviour. For example, a user can better understand their current poor nutrition by accessing personal nutritional informatics. Personal health informatics can then also provide trajectories over time for one's illness trajectory [42].

15.2.1 Technologies for Health Behaviour Change

Persuasive technologies to change health behaviours are increasingly being designed and evaluated by the HCI research community [10, 11, 26, 29]. These technologies often assist individuals in increasing their activity level or achieving improved nutrition. The importance of setting goals was explored by Consolvo et al. in the context of a mobile application that encourages physical activity [11]. This study used the Goal-Setting Theory, which specifically looks at the importance of setting goals. In a similar fashion, behavioural economics has been applied to designing technologies to change health behaviour with respect to user's dietary choices [26]. In another example, the design of a mobile phone application was used to encourage physical activity [10], where the design considered social aspects as well as other design aspects. Similarly, an application to monitor, reflect upon and socialize diet, exercise and medicine information was developed and evaluated [29], which filled the gap in previous works through the use *social scaffolding*. The term *social scaffolding* refers to social supports through the sharing of stories and obtaining advice through one's family, friends and other social networks.

15.2.2 Use of Theoretical Models When Designing ICTs

There are also foundational works that highlight the merits of considering theoretical models when designing ICTs (Information and Communication Technologies). These theoretical models include understanding both the motivations for using online social networks and the motivation for changing health behaviour. Furthermore, these theoretical models are used either to understand design strategies or to develop evaluation mechanisms, and they are based in health and psychology disciplines.

HCI literature has revealed the application of foundational theoretical models for understanding motivations in using online communities, which include the Uses and Gratification Theory and Organization Commitment Theory [25]. The uses and gratification for the use of Facebook has been explored [21]. Furthermore, the use of theories from psychology discipline has also been presented for designing technologies to support behaviour change [12], which draws from the Goal-Setting Theory, Transtheoretical Model, Presentation of Self in Everyday Life and Cognitive Dissonance Theory to present design strategies for behaviour change. Understanding the theoretical principles for the motivation to use online social network combined with the motivation for health behaviour change has also been explored [23]. However, there are some theoretical models that are absent from this work of Kamal et al. [23] most notably the Common Sense Model and The Theory of Planned Behaviour. This chapter will include these two theories in our literature review. However, [23] does present the importance and validity for consolidating theoretical models from both the social networking and health motivation domains. This chapter will build on the foundational work of [23] to provide better synthesis of the literature into a conceptual framework that can be applied to the design on online social network to motivate health behaviour change.

15.2.3 Personal Informatics

Understanding and reflecting upon one's behaviour through personal informatics is one way to understand that one's health behaviour needs to change. However, the motivation to enter one's health information is often a challenge. For this reason, an exploration of related works in HCI on personal informatics deserves a review, as this will provide an understanding of some of the key challenges and facilitators in the study of personal informatics.

Understanding personal information especially those that do not fit into existing personal information management systems has been explored, and it was found that the information often is stored in temporary and dispersed locations such as notepad, Post-it notes and temporary text files [6]. Similar work has been done to understand how mobile applications can better support note-taking [14].

The move beyond simply logging of personal information into personal informatics was modelled through a stage-based model [28]. This study proposed a five-stage model for the life-cycle of personal informatics: preparation stage is where people motivate to collect data about themselves; collection stage is where data is collected; integration stage is where information is prepared, combined and transformed; reflection stage is where the user reflects on her/his personal information; and action stage is where people choose what they are going to do with the information.

From these previously mentioned studies it is evident that there are numerous amounts of personal information that are stored in ad-hoc areas and they do not fit into existing personal information management systems. Furthermore, the life logging systems that have been developed require a high level of motivation to use, which makes the system much less useful. None of these studies has combined online social networks. Social network can provide a methodology to gain information from others, promote continuous use and increase the overall usefulness of the system.

Other related works include personal informatics and life logging applied to the health domain. The challenge of managing personal health information from health clinics, insurance information, and home information has been explored in [35]. Further work was done to understand the types of *unanchored* information that needed to be managed by cancer patients [24], which found the large diversity in the information that was required to be handled from various locations, cognitive capacities and comprehension limitations. Additionally, the visualization of clinical information was explored using a horizontal timeline to review personal histories [4, 34].

These health related studies of personal informatics reveals some of the challenges with storing and retrieving information. The studies again did not take the dimension of social network into consideration; however, it was found that connection with social ties was a key component of managing their disease [24].

15.3 Determinants from Theoretical Models

Key determinants of motivation and behaviour change can be understood by studying existing theoretical models. A literature review was conducted in both domains: motivation for using online social networks and motivations for health behaviour change. This review of the literature for theoretical models is then used to develop a conceptual framework to motivate health behaviour change using online social networks.

15.3.1 Motivation for the Use of Online Social Networks

The theoretical models that describe motivation for use of online social networks are *Uses and Gratification Theory*, *Common Identity Theory*, *Common Bond Theory*, *Social Identity Theory*, *Organizational Commitment Theory*, *Behaviour Chain for Online Participation*, and *social network threshold.*

The *Uses and Gratification Theory* presents individual determinants for motivations to use online social networks. This theory was established from social sciences in the 1970s, and there has been renewed interest in it from its applicability to telecommunications, computer-mediated-communication and the internet. The Uses and Gratification Theory aims to understand why people use a specific media and the gratification that they receive from it. Table 15.1 shows a review of the literature for this theory, and presents the determinants for motivation.

Tables 15.1 and 15.2 show the synthesis of these theoretical models by combining similar behavioural determinants. These tables show the unique behavioural determinant from all these theoretical models on the left column. The right column shows the theoretical model(s) where the determinant is derived from, and the right column also shows the exact terminology of the behavioural determinant used in the theoretical model. The purpose of presenting this information is to provide transparency in the derivation of the behavioural determinants.

The Common Bond Theory [37], Common Identity Theory [37], and Social Identity Theory [15, 43] show the socially based motivational determinants for using online social networks. The Common Identity and Common Bond Theory have been applied to the design of online communities. The premise of *common identity* is that an individual feels an attachment to a group as a whole; the other side

Table 15.1 Key determinants from uses and gratification theory

Determinant	Determinant from literature
Entertainment	Diversion [40], pass time [33], entertainment [15,25, 33]
Social enhancement	Social utility [40], interpersonal utility [33], social enhancement [15,25]
Maintaining interpersonal connectivity	Social utility [40], interpersonal utility [33], maintaining interpersonal utility [15, 25], social connection [21]
Self-discovery	Personal identity [40], self-discovery [15,25]
Get information	Surveillance [40], information seeking [33], purposive value [15], get information [25], content [21], social network surfing [21], social investigation [21]
Provide information	Purposive value [15], provide information [25], status updating [21]
Convenience	Convenience [33]
Shared identities	Shared identities [21]

Table 15.2 Key socially based motivational determinants for using online social networks

Determinant	Determinant from literature
Social categorization	*Social categorization*: common identity theory [37], social identity theory [15,43]
Interdependence	*Interdependence*: common identity theory [37]
Social comparison	*Intergroup comparisons*: common identity theory [37] *Social comparisons*: social identity theory [15,43] *Psychological distinction*: social identity theory [15,43]
Social interaction with others	*Social interaction with others*: common bond theory [37]
Personal knowledge of others	*Personal knowledge of others*: common bond theory [37]
Personal attraction to others through similarities	*Personal attraction to others through similarities*: common bond theory [37]
Social identity	*Social identity*: social identity theory [15,43]
Sense of belonging	*Sense of belonging*: social identity theory [15,43]

of the coin is *common bond*, where an individual feels an attachment to individuals within a group. Social Identity Theory is rooted in psychology, and it is based on the psychological process by which individuals perceive themselves as part of a group and how they interact with a group [43]; this theory has been used to understand consumer behaviour in virtual communities [15]. Table 15.2 shows these socially based determinants and the root determinant from its corresponding theoretical model.

The Theory of Organizational Commitment [3] and the Behaviour Chain for Online Participation [17] show the types of attachment and the temporal aspects (respectively) to staying committed to online social networks. Organizational commitment theory is derived from occupational psychology literature and it has been applied to online communities [25]. The three main components to organizational commitment are the following: affective attachments are emotional and often why one wants to stay in an organization; continuance attachments are the perceived

Table 15.3 Key determinants and their antecedents for committing to the use of online social networks [3]

Determinant	Antecedent
Affective attachment	Personal and structural characteristics
Continuance attachment	The magnitude or number of investments and perceived lack of alternatives
Normative attachment	Influenced by experiences both prior to (familial/cultural socialization) and following (organizational socialization) entry into the organization

cost of staying or the perceived need to stay in an organization; and normative attachments are reasons to stay based on obligation [3]. Table 15.3 shows these determinants and their antecedents.

The Behaviour Chain for Online Participation [17] and the social network threshold [45] show the temporal aspects of commitment, as stages over time. The Behaviour Chain provides the determinants of moving between stages of use that include: discovery, superficial involvement and true commitment. In addition to the behaviour chain and commitment, the social network threshold [45] describes the adoption of innovations through influence of ones social networks. Indeed, these innovations can be health habits. The adopter categories include: early adopters, early majority, late majority and laggards.

15.3.2 Motivation for Health Behaviour Change

The theoretical models reviewed for positive health behaviour change are the following: *the Health Belief Model*, *Social Cognitive Theory*, *Theory of Reasoned Action*, *Theory of Planned behaviour*, *Common Sense Model*, and *The Transtheoretical Model*. The *Goal-Setting Theory* [11] as mentioned in Sect. 15.2 has not been included in this review because the *Goal-Setting Theory* is not founded in health, but rather task motivation. The concept of goal-setting, however, is a central component to the theoretical models that are included in the review.

The Health Belief Model was developed to understand disease prevention and uptake of screening tests by social psychologists in the 1950s, which has been the basis of numerous studies to understand health behaviour change from preventative health behaviour to self-management of chronic diseases [20]. *The Social Cognitive Theory* holds that behaviour is determined through expectancies and incentives, and the key expectancies are environmental, outcomes and efficacy [5, 39]. *The Theory of Reasoned Action* is rooted in social psychology, and it suggests that a person's behavioural intention depends on the person's attitude about that behaviour and subjective norms [2,41]. *The Theory of Planned Behaviour* [1] is an extension of the *Theory of Reasoned Action* [2,41], which was made necessary because the previous model's incomplete incorporation of will power. *The Transtheoretical Model* shows the stages of health behaviour change, and the determinants to move between the stages [36].

Table 15.4 Individually based motivational determinants for health behaviour change

Determinant	Determinant from literature
Perceived susceptibility or knowledge	*Perceived susceptibility* to adverse health outcomes: health belief model [20] *Knowledge of health risks and alternative health behaviour*: social cognitive theory [5, 39]
Perceived severity or knowledge	*Perceived severity* of current health behaviour: health belief model [20] *Knowledge of health risk and alternative health behaviour*: social cognitive theory [5, 39]
Expectation about outcomes	*Perceived benefit* of specific health behaviours: health belief model [20] *Expectations about outcomes*: social cognitive theory [5, 39]
Perceived barriers	*Perceived Barriers*: health belief model [20] *Sociostructural factors* (impediments): social cognitive theory [5, 39]
Expectations about self-efficacy	*Expectations about self-efficacy*: social cognitive theory [5, 39] *Perceived behavioural controls*: theory of planned behaviour [1]
Individual incentives	*Individual incentives*: social cognitive theory [5, 39]
Expectations about environmental cues	*Expectations about environmental cues*: social cognitive theory [5, 39]
Goals	*Distal and proximal goals*: social cognitive theory [5, 39] *Proximal goals as targets*: common sense model [27]
Attitude	*Attitude*: theory of reasoned action [2, 41] and planned behaviour [1] *Self re-evaluation*: transtheoretical model [36]
Interaction between emotion and cognition	*Interaction between emotion and cognition*: common sense model [27]

The key determinants for positive health behaviour change can be drawn from these theoretical models. Many of these determinants are individually based, and they are shown in Table 15.4 along with their corresponding theoretical model. Social determinants are also a common theme from the theoretical models, and they are summarized on Table 15.5. Tables 15.4 and 15.5 show the synthesis of these theoretical models by combining similar behavioural determinants. The information displayed in the left and right columns is provided in the same manner as for Tables 15.1 and 15.2, and this methodology is described in Sect. 15.3.1.

The theoretical models for changing health behaviour also show that there are temporal factors that determine change. The Transtheoretical Model and the Common Sense Model present these determinants. The Transtheoretical Model presents the stages of change: pre-contemplation, contemplation, preparation, action and maintenance. The model also states that individuals can revert to a previous stage at any time [36]. The Common Sense Model also states the importance of maintenance or sustaining health behaviour change [27]. The Diffusion of

Table 15.5 Socially based motivational determinants for health behaviour change

Determinant	Determinant from literature
Environmental cues	*Cues to action*: health belief model [20] *Environmental cues*: social cognitive theory [5, 39] *Environmental re-evaluation*: transtheoretical model [36]
Subjective (social) norms	*Subjective norms*: theory of planned behaviour [1]
Moral norms	*Moral norms*: theory of planned behaviour [1]
Self-efficacy (vicariously through others)	Social cognitive theory [5, 39]
Social outcome expectation	Social cognitive theory [5, 39]
Sociostructural factors	*Sociostructural factors*: social cognitive theory [5] *Helping relationships*: transtheoretical model [36]
Sociostructural barriers	Social cognitive theory [5, 39]

Innovation is another model that presents the temporal aspects of accepting new knowledge through the adopter categories [38], which is similar to the work of [45] as described in Sect. 15.3.1.

15.4 ABC Framework

The literature review reveals 13 theoretical models: seven describe the motivations for using online social networks and six describe the motivations for changing health behaviour. The determinants for behaviour change based on these theoretical models reveal three dimensions that emerge around the motivation for using online social networks and changing health behaviour that we call **Appeal**, **Belonging** and **Commitment**. The ABC Framework is based on the behavioural determinants that were described in Sect. 15.3.

Together these dimensions provide the foundation for the Appeal Belonging Commitment (ABC) Framework illustrated in Fig. 15.1 that describes how online social networks can be used to motivate health behaviour change. The ABC Framework is an overview of these three dimensions and their determinants. It shows that motivations for health behaviour change and use of online social networks are complex and are defined by a multitude of factors with significant interplay between the determinants. The brackets in this figure show the interplay between the determinants. The font size for the determinants shows the strength of the determinant, which is described in more detail in Sect. 15.5.4.

15.4.1 Appeal

The online social network needs to **Appeal** to users on an individual level in order for the system to be used. The behavioural determinants for motivating individual to

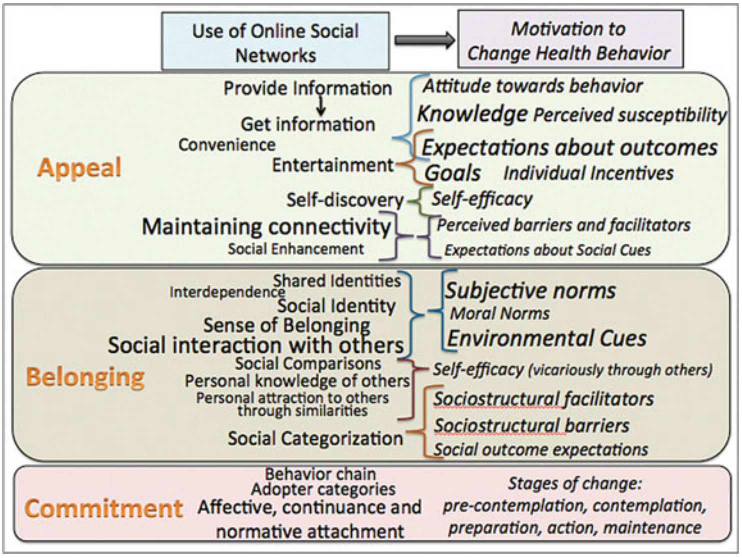

Use of Online Social Networks
Motivation to Change Health Behavior
Appeal
Provide Information
Get information
Convenience
Entertainment
Self-discovery
Maintaining connectivity
Social Enhancement
Attitude towards behavior
Knowledge
Perceived susceptibility
Expectations about outcomes
Goals
Individual Incentives
Self-efficacy
Perceived barriers and facilitators
Expectations about Social Cues
Belonging
Interdependence
Shared Identities
Social Identity
Sense of Belonging
Social interaction with others
Social Comparisons
Personal knowledge of others
Personal attraction to others through similarities
Social Categorization
Subjective norms
Moral Norms
Environmental Cues
Self-efficacy (vicariously through others)
Sociostructural facilitators
Sociostructural barriers
Social outcome expectations
Commitment
Behavior chain
Adopter categories
Affective, continuance and normative attachment
Stages of change:
pre-contemplation, contemplation, preparation, action, maintenance

Fig. 15.1 ABC Framework for using online social networks for health behaviour change

use online social networks is shown in Table 15.1, and the behavioural determinants for motivating health behaviour change is shown in Table 15.4; these determinants make up the first portion of the ABC Framework shown in the top green box in Fig. 15.1, and the interplay between the determinants is shown with the brackets.

Important to health behaviour and personal health informatics is the determinant to *provide information*, which allows others in the social network to *get information*. These two determinants then interplay with the determinants to change health behaviour: *perceived susceptibility*, *attitude*, and *expectations about outcomes*. In other words, the design of the system needs to ensure that the information that is provided allow perceptions and attitudes to be changed. Furthermore, *entertainment* is a determinant for use that can allow one to try to attain health goals by incorporation of *individual incentives*. Additionally, a health system can also use the health information that is provided to display meaningful health information such as calories and carbohydrates (*get information*), which will lead to *self-discovery*, another **Appeal** determinant. *Self-discovery* can facilitate one to better understand their *self-efficacy* in healthy living. By *self-efficacy*, we are referring to one's confidence in their ability to live healthy as defined by the Social Cognitive Theory [5, 39].

Other determinants described in Table 15.4 include *attitude* towards the behaviour and *perceptions about the behaviour change*. These determinants interplay significantly with the determinants for use of online social networks. For example, an online social network that displays raw as well as calculated information about one's health behaviour will lead to the following individually-based determinants of health behaviour change: *knowledge*, *perceived susceptibility*, *attitude towards behaviour*, *expectations about outcomes*, and *self-efficacy*. Further, the individually-based determinants for use of online social networks such as *maintaining connectivity* and *social enhancement* can also allow users to see how to overcome *perceived barriers* through communication exchanges with friends and family, and *expectations about social cues* can be seen if users wish to have influence in their social network (*social enhancement*). The less intuitive interplay exist in the **Appeal** dimension; specifically, maintaining interpersonal connectivity and social enhancement can lead to change in expectations from these *environmental cues* and also changes in *perceived facilitators and barriers* through communication with others, understanding what others are doing, and need to build social status.

15.4.2 Belonging

There are also several socially-based determinants that we have labeled as **Belonging**. The socially-based determinants for use of online social networks are shown in Table 15.2, which includes determinants that are relevant when groups are formed, such as *social comparison*, *shared identities*, *interdependence*, *sense of belonging*, and *social identity*. These factors interplay significantly with socially-based determinants for health behaviour change as described in Table 15.5. Through the development of groups and by providing visibility into the health behaviours of the group through the sharing of information, socially-based determinants of health behaviour change start to emerge; for example, *subjective norms* and *moral norms* are developed as individuals mimic healthy behaviours of their social networks. Further, by *social comparisons* (i.e. with other groups) and gaining *personal knowledge of others*, users can change their own understanding of their *self-efficacy (vicariously through others)*. *Social categorization* as described by the Common Identity Theory [37] and the Social Identity Theory [15, 43] allows a user of an online social network to develop health behaviours through *social outcome expectations*, which is a behavioural determinant from the Social Cognitive Theory citbandura. *Social categorization*, a socially-based determinant for use on online social networks can lead users to overcome *sociostructural barriers* and further develop *sociostructural facilitators*, as well as change *social outcome expectations*. This is shown in the middle brown box on Fig. 15.1.

15.4.3 Commitment

There are several stages, attachments and adoption categories for use and behaviour change related to commitment, which reveals the complexity in motivation, as it is not a simple discrete change but rather a continuum over time with multiple stages that include readiness and action. The literature reveals several temporal stages as well as types of attachment to a system. The stages for use of online social networks from the Behaviour Chain for Online Participation [17] include *discovery*, *superficial involvement* and *true commitment*, and the stages of health behaviour change [36] are *pre-contemplation*, *contemplation*, *preparation*, *action* and *maintenance*. Therefore, a each user's journey will be unique; for example, they may be in the *preparation* stage for their health lifestyle, but be *truly committed* to the online social network system. The types of attachment [3] to the online social network also contributes to the user's commitment level, and they are: *affective*, *continuance* and *normative*. If we use the previous example, this user may only be *truly committed* because all her/his friends are on the system, *normative attachment*. The ABC Framework provides an better understanding of the user's temporal journey as well as attachment to the online social network, which can then be used to design better more persuasive systems tailored to each individual. The Diffusion of Innovation [38, 45] is not listed out explicitly in the Fig. 15.1, since it is not a stage, but rather more of an understanding of the adoption of technology (in this case online social network system) to change health behaviour. The adopter categories are: *innovators*, *early adopters*, *early majority*, *late majority* and *laggards*. The rate at which an online social network system would be adopted can be understood through these adopter categories.

15.5 Evaluation

Although the individual theoretical models have been validated in themselves, the ABC Framework has also been validated as a whole. The theoretical models that have been synthesized for the development of the ABC Framework were often created for entirely different purposes. For example, the Theory of Organizational Commitment [3] was developed to understand employees commitment to their employer, and it was later applied to attachment to online communities [25]; however, it has not been evaluated for online social networks such Facebook$^{\text{TM}}$or Google+$^{\text{TM}}$. This is also true for the Common Identity Theory and Common Bond Theory [37], which is used to describe motivation to participate in online communities. Further, the Social Identity Theory [43] describes the basis of group dynamics, and it was only later applied to online communities [15]. Since the evaluation of most of these theoretical models have not been applied to online social network, we conducted a questionnaire and interview evaluation of the determinants of the ABC Framework to determine their true validity in the motivations to use online social networks. We also evaluated the determinants for health behaviour change to better understand the validity of our synthesis; in other words, the evaluation provides an understanding if we can extract determinants from multiple theoretical models for health behaviour change into a single framework. An initial evaluation was done through questionnaire inquiry for the key aspects of the ABC Framework. Additional interviews were also conducted to obtain a richer understanding of individuals' thoughts on the use of online social networks and health. The points of inquiry for the questionnaire and the interviews were based on the ABC Framework.

Another objective of the ABC Framework is to be valid across age, gender and ethnic boundaries. Healthy living should not be limited to young adults, who are active users of technology, nor should it be for particular ethnicities. The actual validity of the theoretical models used in the ABC Framework across age, gender and ethnic boundaries is unknown. Therefore, we have endeavoured to obtain adult participants in our evaluation that cross various age and ethnic groups. The evaluation also looks further at the differences within these groups, so that the design of an online social network system can focus on aspects that are similar across these groups.

15.5.1 Questionnaire: Recruitment and Respondents

Online and paper questionnaires were developed to learn if the determinants from the ABC Framework were valid. This is especially important for motivation to use online social networks, as many of the theoretical models were based on motivation to participate in online communities and other media. We recruited adult participants from a diverse age range and ethnic identities, and both healthy people and those identifying as having health problems.

Participants were recruited from university listservs, the authors' personal social networks, online social networks, direct outreach to First Nations community, and having a table at a local Punjabi Diabetes forum. We obtained 104 responses to our online and paper questionnaire. Twenty-six respondents completed a paper survey and the balance were completed online.

Of the 104 respondents that completed the questionnaire, 52 % were women and 48 % were men; 15 % 19–24 years old, 29 % 25–34 years old, 29 % 35–49 years old, 12 % 50–64 years old, 12 % 65–74 years old and 3 % were over 75 years old; 26 % identified as Canadian, 24 % as South Asian, 18 % as First Nations, 14 % as Chinese, 7 % as European, and there were eight other ethnic groups that represented the remaining 11 %; 33 % of the respondents identified as having health problems.

The respondents were users of technology and online social networks to varying degrees; 95 % of the respondents used a computer at least once a day; 24 % had never used Facebook; and 9 % had never used any online social network or online community. The respondents to this survey were high use of technology and online social networks, which was likely due to our recruitment methods. This will create a general bias towards technology; however, it will allow us to better understand the motivations to use online social networks to best evaluate the Framework's determinants of for use of online social networks.

15.5.2 Questionnaire: Results

The results from the questionnaire showed strong agreement with the ABC Framework and also revealed some motivational differences between age and ethnic groups. 85 of the 104 respondents answered questions inquiring about their motivation for using online social networks. Respondents were asked to rank their agreement to statements that are reflected by the ABC Framework. For example, respondents were asked for their degree of agreement or disagreement on a five-point likert scale to statements such as, *I use online social networks to get information*, which reflects the *to get information* determinant in the **Appeal** dimension.

The results to the responses around individually based (**Appeal**) determinants for motivation to use online social networks revealed that the strongest agreement was *to maintain connection with people* and *convenience*. The weakest agreement was *to learn about oneself* and *social enhancement*. The **Belonging** dimension for using online social networks was not fully incorporated into the questionnaire due to the complexity in obtaining responses around group behaviour; however, it is interesting to note that "belonging to a group" did solicit positive and negative responses. This does suggest that evaluating social motivation requires alternate inquiry methods. The results also revealed that the strongest **Commitment** determinant is *affective* rather than *continuance* or *normative*, which are described in Table 15.3.

We also looked more deeply into respondents' motivation by understanding the differences between gender, age group and ethnic groups. For this reason, factorial ANOVA was run on the data to better understand any significant difference between these groups, the main effects and the interaction effects. The factorial ANOVA revealed significant differences in the **Appeal** dimension. The motivation to use online social networks for *entertainment* ($\mathrm{F}(4,70) = 2.89, \mathrm{p} = 0.031$) was significantly different for different age groups. The greatest difference in the *entertainment* determinant occurred between the 19–24 years old and 50–64 years old age group ($\mathrm{meandifference} = 1.68, \mathrm{p} = 0.017$). Similarly, *social enhancement* ($\mathrm{F}(2,66) = 3.14, \mathrm{p} = 0.05$) was significantly different for different ethnic groups. The difference between the Canadian and Chinese/South Asian ethnic groups are statistically significant ($\mathrm{meandifference} = -1.39, \mathrm{p} = 0.001$). The analysis also revealed a significant difference in the **Commitment** dimension, as age groups showed significant difference in continuing to use online social networks for their fondness (*affective attachment*) of them ($\mathrm{F}(4,55) = 2.81, \mathrm{p} = 0.034$). The significant difference is between the 19–24 years old and 50–64 years old age group ($\mathrm{meandifference} = 2.17, \mathrm{p} = 0.024$).

Respondents were also asked about their thoughts on their health to gain a better understanding and validation of the ABC Framework, and the results show a strong agreement with the framework. 102 of our 104 respondents answered questions inquiring about their health. Generally, respondents seemed to have agreement with understanding how to live healthy with the greatest concern around exercise. Further, the responses to the **Belonging** and **Commitment** dimensions show good support. Although there is somewhat mixed agreement to social influence on health behaviour, and they are mixed about commitment, since many agree that they can live healthier. We realize that this method of inquiry and the value of responses to this inquiry are limited because asking a person if s/he understands how to live healthily does not mean s/he understands. However, the answers do reveal respondents' perceptions about their understanding of healthy living.

Similar to the feedback on online social network data, the data on the respondents' thoughts on health was also analyzed using factorial ANOVA to better understand the difference between gender, health status (healthy or not), age and ethnicity. Age and ethnic groups were treated the same as was described previously. The results reveal that the **Appeal** dimension does show significant differences. *Knowledge* was one such determinant, as the questionnaire inquiry on "understanding the nutritional value of food" showed statistical difference. The age groups show a statistically significant difference for this determinant ($F(4,43) = 3.29, p = 0.019$). This difference is between the following age groups: 19–24 years old and 25–34 years old ($meandifference = -0.59, p = 0.033$), 19–24 years old and 50–64 years old ($meandifference = -0.70, p = 0.026$), 19–24 years old and 65–74 years old ($meandifference = -0.97, p = 0.003$), and 35–49 years old and 50–64 years old ($meandifference = -0.75, p = 0.009$). This analysis also revealed interaction between age and health status for understanding nutritional content ($F(4,43) = 2.81, p = 0.037$). Concern for one's health in the **Appeal** dimension shows significant difference between healthy and those that have health problems ($F(1,43) = 13.81, p = 0.001$) with a mean difference of -0.76 with the those with health problems being more concerned.

The **Belonging** dimension showed significant difference, especially when they were asked about friends and family influence on their diet. There were significant differences between age groups ($F(4,43) = 2.88, p = 0.034$), health status ($F(1,43) = 6.10, p = 0.018$), interaction between gender and age ($F(4,43), p = 0.008$), interaction between age and ethnicity ($F(8,43) = 3.59, p = 0.003$), interaction between age and health status ($F(4,43) = 13.12, p = 0.001$), and interaction between ethnicity and health status ($F(2,43) = 10.42, p = 0.001$).

The **Commitment** dimension showed significant differences as well in the inquiry, *I ate healthier foods in the past than I do today*. There were significant differences between gender ($F(1,43) = 4.53, p = 0.39$), where the mean difference between male and female was -0.608. Significant differences also existed between ethnic groups ($F(2,43), p = 0.043$), where the post-hoc analysis revealed differences between Canadians and First Nations ($meandifference = -1.02, p = 0.007$), Canadians and Chinese/South Asians ($meandifference = -0.74, p = 0.022$).

15.5.3 Interviews

In order to obtain a richer understanding of people's thoughts around using social networks and their health behaviour, one-on-one in-person interviews were conducted with 11 people. Participants were recruited through university listserv and personal connections. There was no selection criteria except that all participants needed to be over 19 years old. No honourarium or other incentives were given to the participants. There were seven men and four women. Four identified as Canadians, two as Mexican, one as American, one as Indonesian, one as Korean, one as Persian and one as East Indian. Although age was not asked directly, participants were asked to select which age range they belonged to: six were aged 25–34 years old, four were aged 35–49 and one was aged 19–24. They were all users of social networks to varying degrees: one participant checked updates every 20 min, and the others used it at least once a day. As for health problems, four said they had health problems.

There were 13 themes that emerges when users were asked about their usage of online social networks. These themes were: frequency of use (14 comments), change in frequency of use (7 comments), provide information (7 comments), to stay connected to family, friends and other connections (7 comments), view friends activities (7 comments), different uses/purposes for different social networks (7 comments), get information (5 comments), self-promotion (2 comments), participate in social gaming (2 comments), linking two online social networks together (1 comment), research human behaviour (1 comment), build relationships (1 comment), does not contribute to Facebook (1 comment), not concerned with friends' activities (1 comment), entertain friends (1 comment), and not wanting to share information (1 comment).

There were many components of Facebook, Twitter and other online social networks that appealed to the participants. The most common reason to use online social networks was to connect with friends and family. The following quote shows how online social networks allowed the participant to connect with old friends:

> I found that Facebook is a very good method to keep in touch with my friends even old friends as I could find many of my old friends from university or my former colleagues from my previous company, so in this sense it is very useful. (P 6)

Many of the participants said that they use online social networks to get information and view the activity of their friends, such as the following example:

> [I use Facebook] to see what my friends are doing, like what is interesting, what interesting things are going on around my community of friends. (P 4)

This previous quote also provides insight to the **Belonging** dimension, as participants need to view activities that their friends are doing, which builds a group and community network.

Participants also alluded to **Commitment** or lack of it in their use of online social networks. Some participants discussed how they use certain social media less than they did in the past showing a lack of commitment, such as the following examples:

> I use to use Facebook but I disabled it because it wasn't a good use of my time. (P 5)

> It is funny, I actually hated [Facebook] before because of the way it spams bunch of things on to your profile. I know there is a way to control that but its like what's the point, so I actually stopped using it for awhile, a year and a half, but I decided to come back to it because I was getting disconnected to other people who I wanted to keep in touch with. (P 10)

Overall, the interview participants mentioned a number of uses and gratification or **Appeal** determinants for using online social networks, but the frequency of use and level of commitment varies based on the individual and her/his need to use it. Interestingly, the previous quote shows the commitment to Facebook for this person to be more *normative* than *affective*.

A total of 18 themes emerged from a category about living healthy. The themes were mostly based around **Appeal** of living healthy. The 18 themes were: healthy eating (14 comments), doing exercise (8 comments), importance of health (6 comments), need to be organized (3 comments), relieving stress (2 comments), awareness of importance of healthy living (2 comments), motivation to live healthy is to lose weight (2 comments), finds it easy to be healthy (2 comments), reminders on food products pertaining to health (1 comment), living healthy improves one's chronic conditions (1 comment), need to be committed to live healthy (1 comment), balanced lifestyle leads to healthy lifestyle (1comment), need to be motivated to live healthy (1 comment), will power is important (1 comment), need to improve health behaviour (1 comment), clinical information (1 comment), self-initiative is more important than friends (1 comment), and mental health (1 comment).

Much of the discussion was around their own practices in healthy living as it pertained to healthy eating and exercising, for example:

> I do go to the gym very often and I try to exercise because its something that I need as it un-stresses me. (P 1)

The next quote touches on the **Appeal** determinant of *knowledge*, as this participant understands the importance of comprehending one's nutritional intake:

> I think monitoring what you eat is one of the most important [things] and that is one of the things that you should try to do, so [I read] all of the nutritional facts about food that I purchase. (P 2)

The participants also discussed the **Belonging** components or social influences on health in great detail. The following quote show the social norms that occur through one's friends:

> If you are with thin people, your behaviours tend to match up a little better, so I think those are huge influences. (P 9)

Negative social pressures were also mentioned:

> Well if you go out with friends who eat bad stuff, drink alcohol, eat at MacDonald's everyday, eat those fried stuff, well you would eat them too. (P 8)

The difficulty in committing to healthy behaviour was discussed by many of the participants as well as possible reasons why they have not been able to commit to healthy behaviour, as shown by the following quotes:

> I think that it really varies from one period or year to another period. There [is] summertime [when] there are lots of activities and I'm really a big fan of outdoor activities and I have things that I do: I go out and I go mountain climbing and stuff and I feel like I'm doing better and doing more exercise and that makes me feel healthier too. (P 4)

> I think a lot of [healthy living] comes down to being better organized. You know finding time making time one of those expressions for the things you need to be doing maybe also prioritize. (P 9)

Overall, interview participants felt that they generally had the technical knowledge to eat healthy and they knew that they should be exercising more. However, they felt that there were barriers in motivation to maintaining a healthy lifestyle.

15.5.4 Iteration of the Framework

As the results confirm, different population groups' behaviour is affected differently by the various determinants. For this reason, we categorize the determinants between the most and least overlap between the various population groups. These strong determinants lead to design strategies that should have the strongest impact across cultural and age groups versus ones that may need tailoring. For example, respondents across age and ethnic groups said they use online social networks to maintain connectivity with family and friends. We used this strategy across all the determinants, and the results of the evaluation are shown on Fig. 15.1; we use larger fonts for determinants that share the most agreement and smaller for least overlap. There is an important caveat to this iteration of our framework. The questionnaire enquired only about use of online social networks and personal health behaviour. We expect that some of these determinants will be much more important for an online social network that is designed to motivate health behaviour change. As was shown in the ABC Framework, there is significant interplay between the determinants from both domains when put together. Specifically, the three determinants that have a larger impact when the domains are coupled are: (1) *self-discovery*, (2) *expectation about outcomes*, and (3)*environmental cues*. *Self-discovery* can allow one to understand one's health behaviour, which leads to designs that include personal health informatics about the user such as sodium intake over time. *Expectation about outcomes* becomes much more evident through the visibility of one's nutritional intake and exercise level. *Environmental cues* may also be designed into the online social network by displaying evidence-based 'seals of approval' when certain activity meets health and medical criteria. Thus, as seen in the next section, we exploit these determinants in proposing design strategies for an online social network promoting health behaviour change.

15.6 Design Strategies

We derived design strategies for online social networks that aim to motivate health behaviour change based on the ABC Framework and the results from our initial study. There are several design strategies in each of the **Appeal Belonging Commitment** Dimensions of the framework.

15.6.1 Appeal

Connection to Family and Friends The ABC Framework shows that providing *connectivity to friends and family* is the strongest determinant for using online social networks. By providing a means to connect with friends and family, the system can reveal health behaviour for one's social networks to facilitate the development of health norms. Conversation can occur to allow for *social supports* and *reveal barriers and facilitators* to healthy living as is shown by the ABC Framework.

Entertainment: A key determinant from the ABC Framework is *entertainment*; therefore, the system should provide fun and entertaining features such as social gaming to promote commitment toward reaching *goals* and to build *individual incentives*, both of which are determinants from the ABC Framework. This design strategy will overcome barriers for use of the social system and also build commitment to healthy life choices.

Personal Health Informatics: A key determinant for health behaviour change is to understand one's *self-efficacy*. By providing a personal health informatics system, people will have visibility into their own health behaviours, which in turn will promote *self-discovery*. The personal health informatics system is facilitated by the determinant to *provide information* and *get information* both key determinants of the ABC Framework. As shown by the ABC Framework, this information will develop health *knowledge* and change one's *attitude* towards certain behaviour. Not surprisingly, strategies to reduce barriers to acquire personal health information fit within this category; however, motivational factors lead to higher tolerance for the inconvenience of entering personal data.

Information to Promote Education: An important design strategy is to provide educational definitions and information about nutrients and strategies for healthy living on demand. This will overcome *perceptions of outcomes, susceptibility and severity* of health behaviour, which are key determinants of the ABC Framework. This education design strategy will also build knowledge about healthy living.

Goals: Central to health behaviour change is the creation of *goals* as shown by the ABC Framework, which are both a determinant and a design strategy. The online social network should support the ability to create specific time sensitive goals.

15.6.2 Belonging

Groups and Clubs: The online social network should support the creation of groups and/or clubs to promote the following behavioural determinants from the ABC Framework: *social categorization*, *sense of belonging* and *social identity*. This will lead to the development of normative health behaviour, which is reflected by the *subjective norms* determinant.

Social Norms (Friends and Family): Visibility into the health behaviour of friends and family will promote the ability to develop social norms, or mimicking the health behaviour of others. This visibility can also promote the following determinant: *self-efficacy vicariously experienced through others*. Social norms tie in directly to *connection to friends and family* design strategies that is described above.

Approvals (Medical Evidence): The use of evidence-based seals is a design strategy that can support the development of the *environmental cues to action* determinant. The seals can automatically appear if a certain activity meets specific health criteria. These seals need to be applicable to the general population; however, they need special consideration if the approval can be harmful to individual's with certain chronic conditions. Further, specific seals for people with chronic conditions and/or allergies can also be developed for a decision support tool for individual's with special needs.

Social Dialog of Health Behaviour/Information: The system needs to support dialog on the user's health behaviour and its associated health information with the users social networks. This will build *sociostructural facilitators*, *social interaction with others* and build motivation through *expectation of social outcome*, all of which are determinants from the framework.

15.6.3 Commitment

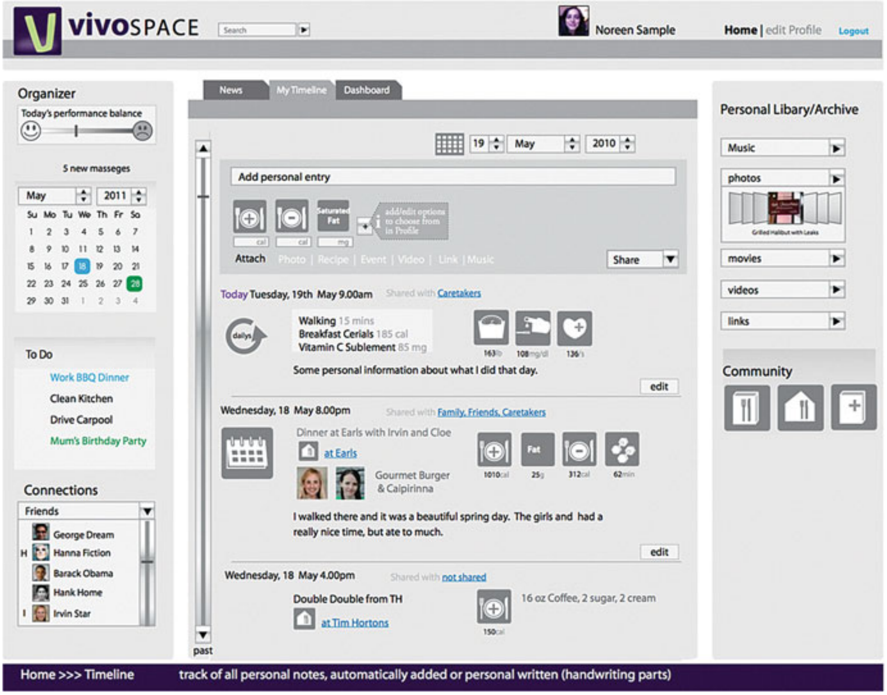
Rewards through Gamification: Users can develop habitual usage patterns through the desire to collect rewards and points (e.g. for achieving goals). These are specific features to design an *entertaining* system, as described above in the Entertainment strategy.

Start and End Dates: Commitment will be built by having firm start and end dates for the achievement of goals, which is in itself a design strategy and a determinant. This can be tied into *reward through gamification* design strategy described above.

Competition: The creation of competitive goals (e.g. challenges) can build commitment for some users, so this option should be available for users who are motivated by competition. Visibility into the leaders of the challenge through such means as a leaderboard will help promote commitment as well as social norms. Competition can help individual to achieve their health *goals*, and it can also help individuals to move from *preparation* or even *contemplation* stage to *action*. For some people, competition can also be an *environmental cue* to action.

15.7 Case Study: VivoSpace

The ABC Framework, based on a synthesis of the top theoretical models and the corresponding design strategies, can start to provide the basis to build an online social networks for healthy living. The design strategies as outlined in Sect. 15.6 can be used to start a user-centred design process, where the evaluation is based on the determinants of the ABC Framework. The application of the ABC Framework has started with the design of VivoSpace. This section will provide a summary of the design of VivoSpace, as a case study for the application of the ABC Framework.



Work BBQ Dinner
Clean Kitchen
Drive Carpool
Mum's Birthday Party

George Dream
Hanna Fiction
Barack Obama
Hank Home
Irvin Star

19 May 2010

Today Tuesday, 19th May 9.00am Shared with Caretakers

Walking 15 mins
Breakfast Cerials 185 cal
Vitamin C Sublement 85 mg

163lb 108mg/dl 136/s

Some personal information about what I did that day.

Wednesday, 18 May 8.00pm Shared with Family, Friends, Caretakers

Dinner at Earls with Irvin and Cloe
at Earls
Gourmet Burger & Calpirinna

1010cal 25g 312cal 62min

I walked there and it was a beautiful spring day. The girls and had a really nice time, but ate to much.

Wednesday, 18 May 4.00pm Shared with not shared

Double Double from TH
at Tim Hortons
16 oz Coffee, 2 sugar, 2 cream
150cal

track of all personal notes, automatically added or personal written (handwriting parts)

Fig. 15.2 VivoSpace: paper prototype for the timeline page

The design strategies were used to develop paper prototypes for VivoSpace. There were 14 pages developed in total in Adobe Illustrator, including: a timeline page that allows users to enter their meals and exercise information; a newsfeed page to be able to see what one's friends are doing; a dashboard page to view one's progress over time; a dashboard to compare one's progress with their friends; a dashboard to view one's activities on a map; and also sidebars with a calendar/to-do list that links to the users other calendars/to-dos and an area to combine one's digital assets including music, videos and pictures. The design attempted to recognize the need for motivation to provide information, which is a key determinant of the ABC Framework, so it provided a combined portal to the user's other assets such as digital assets and calendar events. Figure 15.2 shows the timeline page, and the side bars showing the combination of user's digital assets. The timeline page uses some of the same design features as Facebook^{TM}and Google+^{TM}such as a text entry field to enter a status update, which provides users with familiarity with the interactions and recognition as an online social network; however, the difference with the status update is that the user of VivoSpace would enter their meal or activity in natural language and VivoSpace would provide nutritional information for the meal and calories burned for the activity. The menu bar at the top also shows the links to the other pages.

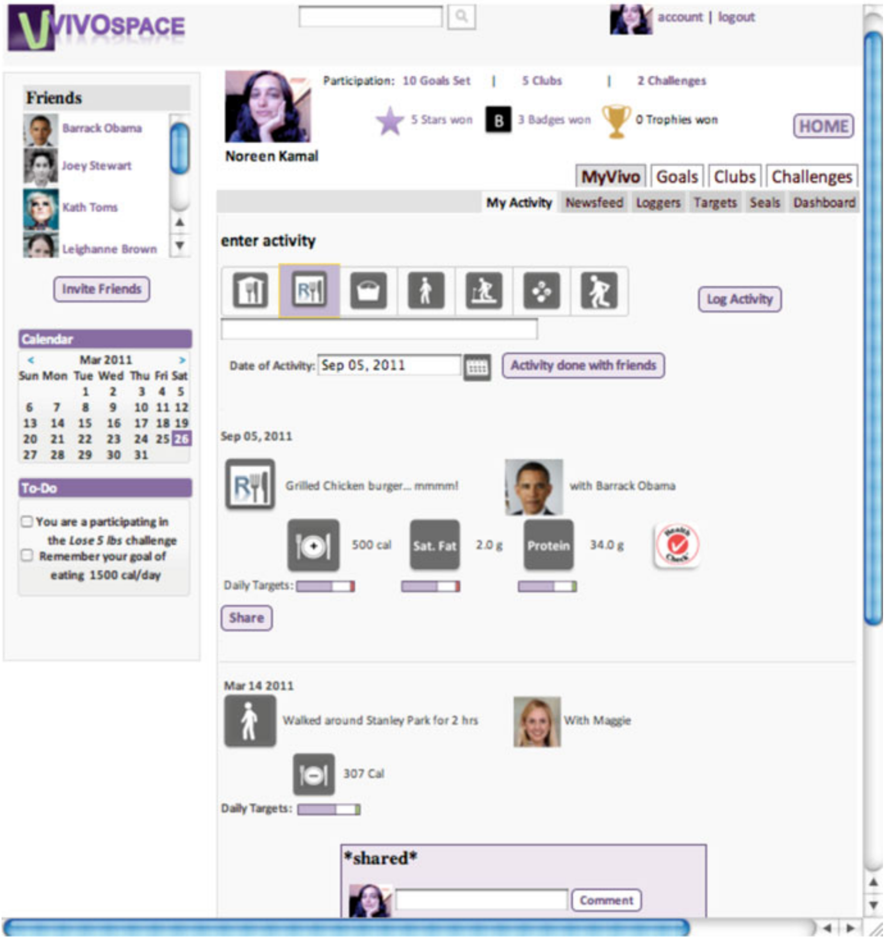
When the paper prototypes were evaluated through one-on-one in-person interviews based on the determinants of the ABC Framework [22], many shortfalls in the design were revealed. The interviewees were concerned about the amount of effort that it would take to enter their health information, and furthermore, they did not feel that combining their digital assets to their health information was of value to them. Some determinants from the ABC Framework that need to be brought out in the design are *personal goals* and the use of *clubs* and *challenges* to build motivation. These are very salient points that the paper designs failed to capitalize on. This shows the validity for the use of the ABC Framework to evaluate the design. The full evaluation revealed 385 comments with 100 themes emerging about the design, and it generally showed promise for the use of online social networks for health behaviour change. The top comments included a suggestion to create clubs and challenges to motivate health behaviour change and expression of concern around the difficulty to enter information; however, participants felt the system would be helpful for their health and the dashboard was appreciated.

These results were used to iterate the design and build a medium fidelity prototype of VivoSpace. The medium fidelity prototype was an interactive prototype developed with HTML, jQuery, CSS and Javascript. There were 32 HTML pages in total and Fig. 15.3 shows one page from the medium fidelity prototype. This page represents the main activity page. Users can also view the newsfeed, which shows the activities that their friends have shared; change their nutritional targets; change what they will log; and change the evidence-based seals that are automatically visible. There is a dashboard page that shows weekly averages for daily intake of nutrients against the target, time series charts for each nutrient of interest; a narrow channel for the users intake of each nutrient is also provided: a green check mark shows that they are on target and a red "high" or "low" as appropriate show those where the target has not been met. The prototype makes goals central to the design and also provides clubs, which are goals that are collaboratively completed with friends and challenges which are competitive. This iteration of VivoSpace simplifies the initial design while still providing the key functionality.

The medium fidelity design of VivoSpace was evaluated in a laboratory with the use of the ABC Framework. Thirty-six study participants were provided with tasks to complete and after each set of tasks they were asked to complete a questionnaire. The questionnaire contained statements that related back to **Appeal** or **Belonging** determinant, such as:

> I would be able to gain information about myself and my capabilities by using a system like this (**Appeal**: self-discovery)
>
> The newsfeed would allow me to view how my friends and family are staying healthy (**Belonging**: social comparison)

Participants were asked how strongly they agree or disagree on a seven-point likert scale. Group belonging and group commitment were also evaluated in the laboratory through indirect methods, as these could not be inquired directly in a laboratory experiment. The results were very encouraging for the ABC Framework, as it identified key determinants that were lacking in the design. The main concern for the



☐ You are a participating in the *Lose 5 lbs* challenge
☐ Remember your goal of eating 1500 cal/day

Date of Activity: Sep 05, 2011

Sep 05, 2011

Grilled Chicken burger... mmmm!

with Barrack Obama

Mar 14 2011

Walked around Stanley Park for 2 hrs

With Maggie

Fig. 15.3 VivoSpace: medium fidelity prototype showing the main activity page

medium fidelity prototype of VivoSpace was that it did not provide enough incentive to provide information, which is central to a social personal informatics system such as VivoSpace. The results also reveal a huge variation in participants' comfort for the competitive challenges function showing that this may need to be combined with the personal goals and group goals (clubs) section in order to appeal to the greatest audience.

The ABC Framework has allowed the design of VivoSpace to evolve in a manner that ensures it will motivate its use and also motivate health behaviour change. The next stage for VivoSpace is to iterate its design based on what was learned from the medium fidelity prototype's lab study, and to build a fully functional prototype, so that it can be tested in the field. Once again the ABC Framework will be used to evaluate if VivoSpace can change health behaviour through an online social network.

15.8 Conclusions

Online social networks provide enormous opportunity to promote health behaviour change, which yields an opportunity to understand what the design strategies for such a system. However, in order for these systems to be designed to adequately promote health behaviour change, the design should be imbedded in theoretical models for both health behaviour change and motivations to use online social networks. The Appeal Belonging Commitment (ABC) Framework incorporates 13 theoretical models and reveals key determinants in each of these three dimensions: (1) the system must **Appeal** to users through individually derived determinants for motivation; (2) it must promote **Belonging** to the online system and their social networks; and (3) it needs to build **Commitment** to good health behaviour and use of the online social network.

The ABC Framework provides the following design strategies. For **Appeal**, the design strategies include: creating connection with friends and family, entertainment, personal health informatics and goals. For **Belonging**, the design strategies are the creation of groups and clubs, social norms, approvals and social dialog of health behaviour. For **Commitment**, the design strategies include: rewards through gamification, start and end dates, and competition. These design strategies have been used to build an online social health system called VivoSpace through a user-centred design process. The ABC Framework then provides a solid foundation to evaluate the design at each stage of the design process to ensure that it meets the determinant to motivate use and health behaviour change.

Acknowledgements The authors would like to thank the Vancouver Aboriginal Friendship Centre Society for providing access to their First Nation staff and clients. The authors are also grateful to the Canadian Network of Centres of Excellence on Graphics, Animation and New Media (GRAND NCE) for funding this research. We thank everyone at the Media and Graphics Interdisciplinary Centre (MAGIC), where this research was conducted.

References

1. Ajzen, I (1991) The Theory of Planned Behavior. Organizational Behavior and Human Decision Processes, 50, 179–211
2. Ajzen, I and Fishbein, M (1977) Attitude-behavior relations: a theoretical analysis and review of empirical research. Psychological Bulletin, 84, 5, 888–918
3. Allen, NJ and Meyer, JP (1990) The measurement and antecedents of affective, continuance and normative commitment to the organization. J Occup Psychol, 63, 1–18
4. Ardito, C, Buono, P, Costabile, MF, and Lanzilotti, R (2006) Two different interfaces to visualize patient histories on a PDA. In Proc. MobileHCI 2006, vol. 159., 37–40
5. Bandura, A (2004) Health promotion through social cognitive means. Health and Edu Behavior. 31, 2 143–164
6. Bernstein, M, Van Kleek, M, Karger, D, and Schraefel, MC (2008) Information scraps: How and why information eludes our personal information management tools. ACM Trans. Inf. Syst. 26, 4, 1–46

7. Bodenheimer, T, Lorig, K, Holman, H, and Grumbach, K. (2002) Patient self-management of chronic disease in primary care. J Am Med Assoc, 288(19)
8. Christakis, NA and Fowler, JH (2009) Connected: How your friends' friends' friends affect everything your feel, think, and do. Black Bay Books, NewYork
9. Christakis, NA and Fowler, JH (2007) The Spread of Obesity in a Large Social Network over 32 Years. N Engl J Med 357:370–379
10. Consolvo, S, Everitt, K, Smith, I and Landay, JA (2006) Design requirements for technologies that encourage physical activity. In Proc. CHI 2006, 457–466
11. Consolvo, S, Klasnja, P, McDonald, DW and Landay, JA (2009) Goal-setting considerations for persuasive technologies that encourage physical activity. In Proc. Persuasive 2009, 8:1–8:8
12. Consolvo, S, McDonald, DW, and Landay, JA (2009) Theory-driven design strategies for technologies that support behavior change in everyday life. In Proc. CHI 2009, 405–414
13. Dubberly,H., Mehta, R., Evenson, S. and Pangaro, P.(2010) Reframing health to embrace design of our own well-being. Interactions, May, Issue 3, 56–63.
14. Dai, L, Lutters, WG, and Bower, C (2005) Why use memo for all?: restructuring mobile applications to support informal note taking. In Proc. CHI 2005. 1320–1323.
15. Dholakia, UM, Bagozzi, RP, and Pearo, LK (2004) A social influence model of consumer participation in network- and small-group-based virtual communities. International Journal of Research in Marketing, 2, 241–263
16. European Food Information Council (2011) Your guide to food safety and quality and health and nutrition for a balanced diet and healthy lifestyle. http://www.eufic.org/.Cited17Jul2011.
17. Fogg, BJ and Eckles, D (2007) The behavior chain for online participation: how successful web services structure persuasion. Persuasive Technology, LNCS 4744, Springer, 199–209
18. Fisher, EB, Brownson, CA, OToole, ML, Shetty, G, Anwauri, VV, and Glasgow, RE, (2005), Ecological Approaches to Self-Management: The Case of Diabetes. Am J Public Health, September 2005, Volume 95, No. 9. 1523–1535.
19. Health Canada (2011) Eating Well with Canada's Food Guide. http://www.hc-sc.gc.ca/fn-an/food-guide-aliment/index-eng.php.Cited17Jul2011.
20. Janz, NK and Becker, MH (1984) The Health Belief Model: A Decade Later. Health Education Behavior, 11, 1, 1–47
21. Joinson, AN (2008) Looking at, looking up or keeping up with people?: motives and use of facebook. In Proc. CHI 2008,1027–1036
22. Kamal, N, Fels, S, Blackstock, M, and Ho, K (2011) VivoSpace: Towards Behavior Change Using Social Gaming. ICEC 2011, LNCS 6972, 319–330.
23. Kamal, N, Fels, S, and Ho, K (2010) Online social networks for personal informatics to promote positive health behavior. In Proc. WSM 2010, 47- 52
24. Klasnja, P, Civan Hartzler, A, Unruh, KT, and Pratt, W (2010) Blowing in the wind: unanchored patient information work during cancer care. In Proc. CHI 2010. 193–202
25. Lampe, C, Wash, R, Velasquez, A, and Ozkaya, E (2010) Motivations to participate in online communities. In Proc. CHI 2010, 1927–1936
26. Lee, KL, Kiesler, S and Forlizzi, J (2011) Mining behavioral economics to design persuasive technology for healthy choices. In Proc. CHI 201 325- 334
27. Leventhal, H, Diefenbach, M, and Leventhal, EA (1992) Illness cognition: Using common sense to understand treatment adherence and affect cognition interactions. Cognitive Therapy and Research. 16, 143–163
28. Li, I, Dey, A, and Forlizzi, J (2010) A stage-based model of personal informatics systems. In Proc. CHI 2010. 557–566
29. Mamykina, L, Mynatt, ED, Davidson, PR and Greenbalt, D (2008) MAHI: Investigation of Social Scaffolding for Reflective Thinking in Diabetes Management. In Proc. CHI 2008, 477–486
30. Martin, A. Trends in Video Games and Gaming (2011) ITU-T Technology Watch Report, September, 2011, 28 pages. http://www.itu.int/ITU-??T/techwatch.

31. Morris, ME, Consolvo, S, Munson, S, Patrick, K, Tsai, J, and Kramer, ADI (2011) Facebook for health: opportunities and challenges for driving behavior change. Ext. Abstracts CHI 2011, 443–446
32. Newman, MW, Lauterbach, D, Munson, SA, Resnick, P and Morris, ME (2011) It's not that I don't have problems, I'm just not putting them on facebook: challenges and opportunities in using online social networks for health. In Proc. CSCW 2011, 341–350
33. Papacharissi, Z and Rubin, AM (2000) Predictors of internet use. Journal of Broadcasting and Electronic Media, 44(2), 175–196
34. Plaisant, C, Milash, B, Rose, A, Widoff, S, and Shneiderman, B (1996) LifeLines: visualizing personal histories. In Proc. CHI 1996. 221- 227
35. Pratt, W, Unruh, K, Civan, A, and Skeels, MM (2006) Personal health information management. Commun. ACM 49, 1, 51–55
36. Prochaska, JO, and Velicer, WF (1997) The transtheoretical model of health behavior change. American Journal of Health Promotion, 12(1), 38–48
37. Ren, Y, Kraut, R, and Kiesler, S (2007) Applying common identity and bond theory to design of online communities. Organizational Studies, 28, 3, 377–408
38. Rogers, E. M. (1983). Diffusion of Innovations. 3rd Edition, The Free Press, a division of Simon and Schuster Inc., New York, NY
39. Rosenstock, IM, Victor, JS, and Brecker, MH (1988) Social learning theory and the health belief model. Health Education Behavior, 15, 175, 175–183
40. Ruggiero, TE (2000) Uses and gratification theory in the 21st century. Mass Communication and Society, 3(1). 3–37
41. Sheppard, BH, Hartwick, J, and Warshaw, PR (1988). The theory of reasoned action: a meta-analysis of past research with recommendations for modifications and future research. The Journal of Consumer Research, 15(3), 325–343
42. Strauss, A, Fagerhaugh, S, Suczek, B and Wiener, C (1985) Social Organization of Medical Work. The University of Chicago Press.
43. Tajfel, H (1974) Social identity and intergroup behaviour. Social Science Information, 13(2). 65–93
44. United States Department of Agriculture (2011) Food and Nutrition Information Center. http://www.nutrition.gov.Cited17Jul2011.
45. Valente, TW (1996) Social network thresholds in the diffusion of innovation. Social Networks, 18(1996), 69–89
46. Wagner, E.H, and Groves, T (2002). Care for chronic diseases. Br Med J, 325(7370), 913–914

Chapter 16
Evolution of an Open Source Community Network

Nilesh Saraf, Andrew Seary, Deepa Chandrasekaran, and Peter Monge

Abstract The study attempts to better understand the evolution of the structure of a network using two snapshots of the developer-project affiliations in an Open Source Software (OSS) community. We use complex networks and social network theory to guide our analysis. We proceed by first extracting separate bipartite networks of projects in each of the five development stages – planning, pre-alpha, alpha, beta and production/stables stages. Then, by analyzing changes in the network using degree distributions, assortativity, component sizes, visualizations and p-star models, we try to infer the project-joining behavior of the OSS developers. Simulations are used to establish the significance of some findings. Highlights of our results are the higher levels of assortativity and networking in the Beta and Stable subnetworks, and a surprisingly higher level of connectivity of the Planning subnetwork. Significant clustering of projects is observed based on the programming language but not on other project attributes, including even licenses.

N. Saraf (✉)
Associate Professor, Beedie School of Business, Simon Fraser University WMC 3317,
8888 University Drive, Burnaby, BC, V5A 1S6, Canada
e-mail: nsaraf@sfu.ca

A. Seary
Research Associate, School of Communication, Simon Fraser University, 8888 University Drive
Burnaby, BC, V5A 1S6, Canada
e-mail: seary@sfu.ca

D. Chandrasekaran
Assistant Professor, College of Business and Economics, Lehigh University RBC 370,
Rauch Business Center, 621 Taylor Street, Bethlehem, PA, 18015, USA
e-mail: dec207@lehigh.edu

P. Monge
Professor, Communication, Annenberg School for Communication and Professor, Management
and Organization, Marshall School of Business, University of Southern California,
3502 Watt Way, Los Angeles, CA, 90089-0281, USA
e-mail: monge@usc.edu

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_16,

16.1 Introduction

We report the results of an exploratory analysis of the networked structure and evolution of the open source software (OSS) development community. Many studies have examined OSS activity by considering each open source project as a separate developer community, in isolation from other open source projects. Other recent studies have also begun to analyze open source projects as a network where the linkages between projects are formed when developers cross-participate in multiple projects [39,51]. The basic premise underlying these studies of the OSS community as an innovation network is that the developers are conduits for knowledge transfer and learning between projects. While this is a useful perspective few studies have attempted to study how and why these networks evolve over time which would have important implications about technology trajectories of OSS products. Our analysis offers several interesting insights that could contribute to a better understanding of OSS community evolution.

We follow two directions in our analyses of network data from an OSS development portal (http://www.sourceforge.net). First, we segregate projects in distinct development stages into different sub-networks because we believe that the key to understanding how the OSS network structure emerges and evolves, is to understand the linking behavior of developers at various stages of software development. Software engineering literature suggests that the resources required by software projects in different development phases differ. Therefore, we conjecture that the manner in which they link with other OSS projects would differ.

In this study, we examine how the network changes after a 9-month lag and how the network properties differ between projects in various stages ranging from planning to pre-alpha, to alpha, to beta, to production/stable, in chronological order. This allows us to infer the behavior of developers as they engage in OSS development. We find that these subnetworks indeed show distinct properties. For example, our simulations reveal that the subnetworks of projects in the latter phases deviate significantly from a random graph whereas the early-stage subnetworks do not. This means, that developers in the latter stages engage in deliberate selection of other projects to participate in more often than those in early stages. And further, this choice of newer projects is not random but is guided by specific criteria which we seek to identify in this study.

Second, with the purpose of identifying rules that guide network change, we explored the evolution of one of the latter stage subnetworks since we found this to be a non-random network. Specifically, to understand what criteria developers in this subnetwork might be using in forming links, we were guided by three logics that underlie tie formation in most types of social networks [1, 20]. One, the rationale of preferential attachment proposes that those OSS projects which are already central in a network of projects accrue more project-project links over time. In the literature, this is labeled as the winner-takes-all or rich-get-richer phenomenon. Two, popular developers seek to participate in a more diverse set of projects rather than specializing in a narrow category of projects. This is referred to in literature as multi-connectivity [1, 20]. Third, the logic of homophily suggests that similar projects would tend to share developers and thus form links with each other more than dissimilar projects would. That is, developers, tend to affiliate with other projects that are similar to their prior projects.

The results of these analyses also suggest a more complex picture. The results indicates support for the homophily logic, but not the preferential attachment or multi-connectivity logics. Specifically, we observe that the largest component of the production subnetwork demonstrates clustering based on different programming languages. This is consistent with recent analysis of mature-only projects [2] where mature projects linked with other mature projects sharing the same programming language. However, the affinity for the same programming language that results in project-project links does not manifest when projects share the same operating systems or the same topics. Finally, a typical property of large networks is the expansion of the main component where previously unconnected nodes in the network eventually coalesce to join the large component. Our comparison of the subnetworks captured at two cross-sections 9 months apart also reveals a visual illustration of such a network trajectory.

16.2 Literature

While the high degree of granularity of the data on the content of OSS project coordination offers an opportunity to advance software engineering methods, the large quantity of data on thousands of OSS projects has helped us better understand the principles underlying complex network emergence. Our analysis following a brief review of relevant literature therefore, will borrow from both these streams in order to better understand the evolutionary patterns in the Sourceforge OSS community. However, the guiding theories for our exploration are the logics of attachment in social networks [1] – homophily, preferential attachment and multi-connectivity. While software engineering literature attempts to advance the software development principles of modularity and coordination in the social context of software development, complex networks research is less concerned with the context. Instead, the latter body of literature attempts to identify topological network patterns that manifest from purposive social behavior, as distinct from patterns that manifest from random linking behavior in networks. Notwithstanding the differences, both these literatures attempt to understand to varying extents how the social motivations represented by these logics proposed by Monge and Contractor [1] manifest in the formation of links among network actors.

16.2.1 Software Engineering and OSS Project Networks

Software engineering research has considered the socio-technical nature of the OSS projects and attempted to understand how the structure of the software code converges with the coordination structures between groups of OSS developers [41]. Thus, the dependencies between different components of the software artifact also are reflected in the hierarchy and coordination among sub-groups of developers. Therefore, assuming this congruence is beneficial for software projects, it can be measured at different points in time in order to optimally guide software development activity [42]. Further, because much of the software development knowledge is implicitly captured in text such as emails, discussion forum and online chats, scholars have also developed tools to extract implicit social networks from such data [43,44]. Comparing the structure of implicit social networks between successful and unsuccessful OSS projects affords greater insights for refining software engineering methods. Similarly, using social network analysis to study the leadership behavior of OSS developers working on specific projects [48], helps to develop a deeper understanding of the types of coordinating activities important for managing such projects. The 'social' dimension of software development also surfaced in another recent study that finds the developers also prefer to associate with others with similar status and experience [47]. Overall, the community level analysis of OSS projects using a social network lens [46] has generated a significant body of knowledge of how these communities succeed in developing successful software artifacts.

16.2.2 Complex Networks of OSS Projects

Open source software researchers have examined the social network of the community in different ways. Valverde et al. [3] analyze the network of debugging-related emails of a specific open source communities such as the Python (a programming language) and TCL project, and they find scale-free behavior where a core of programmers participates in most of the email traffic. Xu et al. [4] examine the topology of the entire set of 87,000 projects and 912,000 developers on an open source development to find that the degree distributions of the (one-mode) developer-network and the (one-mode) project-network are scale-free. They find that the core developer network is far more fragmented than a larger network consisting of not only the core but also peripheral developers and users. This finding is consistent with the predictions of how assortativity affects network structure. Degree-assortative networks are those where nodes associate with nodes of equivalent degree. Assortative networks percolate faster and form smaller giant components. This also suggests that the peripheral developers serve to weakly integrate the entire community into a larger cluster than the core developers themselves.

Thus, depending on the sampling procedure, the open source network is characterized by both as a small world (resulting from its scale-free behavior) and as a highly clustered network (i.e., clustered according to a project attribute, not in terms of link transitivity as is conventionally defined in complex networks literature) [2]. In the same study, while the scale-free distribution is similar confirmed, the project-project network was also found to cluster predominantly according to the programming languages of the projects. Another finding is that the network of mature-only projects is meta-assortative whereas that of early-stage projects is meta-disassortative. That is, mature projects are more likely to link with other mature projects when they have similar attributes, e.g., programming language [2].

Past research [9] suggests that preferential attachment [5–7], clustering and assortativity [8] are key mechanisms that lead to change in the network structure. The theoretical models also apply to and have been tested on bipartite social networks (two-mode) such as the scientific collaboration network of authors collaborating on articles [11], the board of directors network [8], and the movie network [12]. Watts [13] discusses how findings from analysis of one-mode can be extended to understanding bilateral networks.

An important concept used to characterize social networks is its small world structure where the diameter of the network increases as the logarithm of the number of nodes in the network [12, 14]. In small world networks, path distance between most pairs of nodes is extremely small compared with the size of the network. The mechanism that contributes to a small world structure of a network is preferential attachment where it is the central nodes lower the diameter of the network significantly. However, while preferential attachment has been found to be predominant in social networks of scientific patents [15], clustering and assortativity also are characteristic of social networks [10,40]. Intuitively, preferential attachment would contribute to the emergence of a small world structure, while the relationship between assortativity or clustering within small-world structures is more complex [16].

An interesting theoretical prediction in the complex networks literature is the role of the assortativity in percolation of a giant component [10]. Highly assortative networks percolate faster but the size of the giant component is smaller than when the network is disassortative. Conversely, a disassortative network percolates more slowly but the size of the giant component is larger than in assortative networks. For neutral networks with zero assortativity, the emergence and size of the giant component is intermediate between the above two extremes. Guimera et al. [17] discuss how depending on how teams grow by (i) either allowing entry to new entrants or network incumbents, and by (ii) allowing inclusion of past collaborators, the network structure evolves differently. They describe how these parameters affect the percolation of the network and its task performance.

16.3 Data and Exploratory Analysis

We utilize the data from Sourceforge.net, the largest repository of Open Source software on the Internet. As of September 28, 2006, there were about 130,647 registered projects and 1,401,662 registered users on Sourceforge.net. Even though there are several other much smaller sites such as Freshmeat, Rubyforge, or ObjectWeb, Sourceforge can be considered to be representative of the population.

Three types of data from the FLOSSmole website were utilized for this study [18]: (i) Developer-project affiliations: Each project team on Sourceforge consists of a project administrator or owner, who coordinates the efforts among the remaining contributors consisting of both, the core and peripheral developers. Of these, the core-developers are the most important since they are considered experts and have privileged access to the code repositories. We captured data on which developer was registered as the core-developer of each project. (ii) Project characteristics: Sourceforge also provides different classification systems for projects. For example, project characteristics are captured in terms of the database environment, intended audience, license, operating system and programming language. (iii) Project stage: Developers also classify their own projects as belonging to one or more of these stages in a project life cycle: planning phase; pre-alpha phase; alpha phase; beta phase; production/stable category and mature phase of development. Though many projects belong to the same stage of development, they may have been initiated and registered on Sourceforge in different years from 1999 to 2005. We segregated our data into sub-samples using the information about the stage of every project (planning, pre-alpha, alpha, beta, stable and mature). In our analysis, we trimmed the data by excluding projects whose phase information was not available. Next, we proceeded using a combination of network descriptive statistics, network visualizations, and statistical testing typically used in complex networks research, which help us develop a rich picture of the dynamics of OSS community social networks. Graphical visualizations, which we obtained using Multinet [19], are often useful to draw inferences and guide further exploration [20]. Visualizations presented in this study depict the bipartite network. Eigenvectors coordinates were used to graphically locate the nodes [21].

16.3.1 Sampling

Projects which declared themselves simultaneously in two separate phases were recoded into one phase, the latter of the two declared phases. For example, a project XYZ which was declared to be in the alpha and beta phases was assigned to the beta sub-network and excluded from the alpha sub-network. Thus, we obtained a set of 60,164 projects for April 2005 and 71,154 projects for Dec 2005.

Furthermore, projects with only one developer who was not participating in any other project in both time periods, were considered as isolates and excluded from the analysis since these do not contribute to any understanding of the network change. All remaining projects were retained. However, once a project has more than one developer in either April or December, it was retained in both networks. Therefore, a project was removed only if it was an isolate in both April and December. Thus, out of 62,439 projects in April, 1,094 were isolates but these were retained since they were not isolates in December. In the December data no isolates were retained for our network analysis. Line 4 in Table 16.1 shows the number of projects which were connected and also had phase information. It is about 25 % of the initial number, but since the isolates were 30–40 % of all the projects, the percentage of networked projects with their phases declared is actually about 42–44 % (Table 16.2).

Table 16.1 Sampling

	April 2005	December 2005
Total projects	93,514	106,888
Projects with phase	60,164	71,154
Isolated projects	35,680	43,632
Analyzed non-isolates	24,484	27,522
Percent of non-isolates	42 %	44 %

Table 16.2 Distribution of December 2005 connected projects (analyzed)

Phases	Counts	Percentage (%)
Planning	6,662	24.20
Pre-alpha	3,969	14.40
Alpha	4,650	16.90
Beta	6,357	23.10
Production	5,120	18.60
Mature	365	1.30
Inactive	399	1.40

16.3.2 Analysis of Network Degrees

1. Network Degree
   Degree centrality of a project indicates the prominence of a project in the OSS community. Similarly degree centrality of a developer indicates the prominence of a developer and therefore his access to resources within the network [24, 25]. We assess degree centrality of a project simply as the number of participating developers and the degree centrality of a developer simply as the number of projects he or she was participating in. The project degree centrality also is an indicator of the complexity of the project as existing developers recruit additional ones in order to fill project requirements [35]. Developer degree centrality represents the access to diverse workgroups by a developer and thus access to more skills and software knowledge within the community. It is seen from Table 16.3 that the mean developer degree shows an increase especially in the latter groups, namely the beta and stable projects (i.e., from 1.08, 1.05 and 1.06 to 1.22 and 1.13). The increase in mean degree suggests that developers overall activate more links as they work on projects in the latter phases compared to the early stages. From the first rows of Tables 16.4 and 16.5 we note that the number of developers with a degree of one forms a smaller percentage of the total number of developers as we progress from Planning to Stable stage (96–90.5 % for Table 16.4 and 93.40–90.50 % for Table 16.5). It is to be noted that the above computations were done separately for the bipartite networks (subnetworks) for each of the phases. Similarly, the mean degree of projects also increases, which means that on an average projects in the stable stage are connected to a larger number of other projects in the stable stage (compared to the earlier stages). Therefore, this understates the differences between the mean degree for each phases since the cross-phase links (between the two subnetworks) were excluded from the individual phase-wise subnetworks.

Table 16.3 Mean degree

	April 2005		December 2005	
Network measure → Phase	Mean developer degree	Mean project degree	Mean developer degree	Mean project degree
Planning	1.08	3.74	1.08	3.78
Pre-alpha	1.05	3.56	1.05	3.62
Alpha	1.06	3.77	1.06	3.83
Beta	1.22	4.37	1.11	4.40
Stable	1.13	5.04	1.13	5.05

Table 16.4 Developer degree (April 2005)

Distribution	Pre-alpha		Alpha		Beta		Stable	
values	Counts	%age	Counts	%age	Counts	%age	Counts	%age
1	11,489	96	14,033	94.4	20,187	83.4	18,200	90.5
2	437	3.6	739	5	1,595	12.9	1,500	7.4
3	28	0.2	80	0.5	266	2.7	310	0.4
4	5	0	8	0.1	54	0.7	75	0.1
5	2	0	2	0	14	0.1	16	0.0

Table 16.5 Developer degree (December 2005)

Distribution	Planning		Pre-alpha		Alpha		Beta		Stable	
values	Counts	%age	Counts	%age	Counts	%age	Counts	%age	Counts	%age
1	21,769	93.4	13,152	96	15,823	94.5	23,045	83.4	20,761	90.5
2	1,288	5.5	495	3.6	826	4.9	1,822	12.9	1,736	7.6
3	176	0.8	46	0.3	81	0.5	287	2.7	328	1.4
4	44	0	7	0.1	16	0.1	61	0.7	84	0.4
5	10	0	2	0	2	0	18	0.1	19	0.1

2. Visualizations

   Qualitative differences are also observed between the five subnetwork phases. To illustrate, the subnetworks for the early stages were highly tree-like with mostly single paths between connected projects. That is, projects are connected to other projects by single developers (see Fig. 16.1 for the pre-alpha subnetwork). For the Beta subnetwork, it is noted that the central portion is highly meshed with multiple paths connecting several project pairs (Fig. 16.2). This suggests more intensive networking by developers as they participate in multiple projects. This is also seen from the exploded central portion in Fig. 16.3. A similar or slightly higher degree of intermeshing was observed for the Stable subnetwork. Surprisingly, the Planning sub-network appears to have a higher degree of networking than either the Pre-alpha or the Alpha subnetworks. The degree of intermeshing can be observed by removing developers with degree 2 and observing the number of components thus created due to the deleted paths. In Pre-alpha and Alpha subnetworks, a large number of small components are formed by deleting developers of degree 2, where the Beta and Stable subnetworks split into fewer components. For example, Fig. 16.4 shows the component sizes for the giant component in the Beta subnetwork after the developers with degree-2 are removed.

16.3.3 Analysis of Non-random Network Formation Processes

While degree analysis indicates actor-level tendencies to form links within a community, it does not indicate how the network actors are identifying alters to form ties with. If actors randomly form links with others, then complex networks theory predicts that the distribution of component sizes should differ significantly from the distribution that would result if the link formation is non-random. Therefore, we examined the component size distribution of each subnetwork (planning to stable phase) separately in order to understand whether and how they differ, if at all from a random graph. These results were also compared with assortativity computations to develop implications for link formation in the OSS community. In particular, separate analysis for each subnetwork highlighted the differing network formation logics underlying the subnetworks.

1. Component Sizes

   We examined the connectivity of the OSS network by an analysis of the component size and its distribution, where size of a component is the number of projects and developers weakly (directly or indirectly) connected to each other. The distribution of component sizes was obtained separately for projects in five phases. These are weak components and are obtained by using the network links to cluster the projects. The sizes of the components are indicated in terms of the number of nodes. Since this is a bipartite graph, the nodes include both developers and projects [45].

   Table 16.6 shows the size of the top five components for each of the subnetworks. For example, the largest component in the Stable sub-network has 7,088 nodes (developers and projects) of which 6,182 are developers working on 906 projects. This component has 28.6 % of the total number of nodes in the stable subnetwork, significantly more than the relative component size of the largest components in Alpha and Pre-alpha subnetworks. The size of the biggest component in percentage terms decreases from Stable to Pre-alpha but shows a surprising increase for the Planning sub-network. Thus, note that the largest component for the Planning sub-network consists of 9.5 % of the total nodes in that subnetwork indicating that the Planning sub-network is more fragmented than the Stable subnetwork. However, the Planning subnetwork is surprisingly less fragmented than the Pre-alpha and Alpha networks which have only 1.3 and 2.7 % of their nodes in the largest components. While all this broadly suggests that in the latter stages, developers and projects tend to get more interconnected, it is not clear why there should be an unusual increase in the connectivity of the Planning subnetwork (Fig. 16.5) compared to the Pre-alpha projects and also the Alpha projects (not included here).

In order to find statistical support, the theory of random graphs was applied. The theory of random graphs [26] concerns the evolution of complex structures within a graph as the density (average degree) increases. This theory was based on Poisson degree distributions and did not apply to graphs with special structure such as bipartite graphs or to other degree distributions. A central part of this theory predicts percolation of a giant component. That is, before the mean of the degree distribution approaches a critical value, the graph has many small components with simple structure such as paths and trees; then develops triangles and longer cycles as the small components coalesce; after which it very rapidly forms a giant component which absorbs most of the nodes; all other components remain far smaller with a simple structure. For Poisson degree distributions, the critical value of node-degree is 1 at which a giant component percolates [26, 27]. Recent theoretical developments [23, 28, 29] show that important features of the original theory also apply to bipartite graphs and to quite a wide range of degree distributions.

The evidence of non-random processes underlying the network formation can be detected if graphs are randomly generated conditional on the same degree distribution of the actual network. A method for generating bipartite random graphs with given degree distributions is described and demonstrated in [23]. If these simulations yield component size distributions significantly different from the actual data, this suggests that non-random link formation is occurring. Therefore, simulations were run to test whether the structure found in both April and December phases could be explained by simple random assignment of links, with the same separate degree distributions for developers and projects found empirically in the SourceForge data. Each simulation was run 25 times to attempt to get reasonable statistics. The results are shown in Table 16.7 (April 2005) and Table 16.8 (December 2005). The second-largest component sizes are shown for each phase to indicate the distribution of component sizes for that phase.

For both April and December, the observed component sizes of the Planning (9.5 %) and Pre-alpha(1.3 %) phases appear very similar to the random simulations (8.5 and 1.4 % respectively). This means that the components in these phases could arise from random connections of developers and projects. The Alpha phases show the largest component to be about twice as big in the simulations (2.7 and 5.2 %), but the error bar (StdDev) is also large. However, for both Beta and Stable phases, it is clear that the largest components of the data are considerably smaller than what would arise from random connections, implying that the linking of projects in the large components are not the result of random processes. The last line in each table shows the data and simulation results for all the projects (except the isolates), whether they have an assigned phase or not. For example, in Table 16.7, the size of the actual large component (48.8 %) is much smaller than the simulated result (71.2 %).

A possible explanation for the non-random link formation among Beta and Stable projects could be the clustering tendency of developers. Developers with high degree tend to work on projects with other developers of high degree in these phases. This would concentrate the higher degree developers on fewer

Table 16.6 Component distribution of phase sub-networks (only five largest components are shown)

Phase	Component number	Node count	% of total nodes of subnetwork	Developer count	% of total developers of subnetwork	Project count	% of total developers of subnetwork
Stable	1	7,088	28.60	6,182	30.60	906	20.00
	2	195	0.80	180	0.90	15	0.30
	3	182	0.70	164	0.80	18	0.40
	4	169	0.70	163	0.80	6	0.10
	5	126	0.50	114	0.60	12	0.30
Beta	1	6,728	24.20	5,841	26.40	887	15.80
	2	229	0.80	206	0.90	23	0.40
	3	117	0.40	109	0.50	8	0.10
	4	80	0.30	70	0.30	10	0.20
	5	74	0.30	63	0.30	11	0.20
Alpha	1	516	2.70	460	3.10	32	0.70
	2	193	1.00	183	1.20	12	0.30
	3	135	0.70	125	0.80	23	0.50
	4	103	0.50	90	0.60	21	0.50
	5	95	0.50	85	0.60	12	0.30
Pre-alpha	1	209	1.30	177	1.50	32	0.90
	2	72	0.50	65	0.50	7	0.20
	3	68	0.40	67	0.60	1	0.00
	4	51	0.30	49	0.40	2	0.10
	5	49	0.30	46	0.40	3	0.10
Planning	1	2,523	9.50	2,072	10.00	451	7.60
	2	223	0.80	176	0.90	47	0.80
	3	168	0.60	155	0.80	13	0.00
	4	86	0.30	80	0.40	6	0.10
	5	82	0.30	80	0.40	2	0.00

projects in the large component. For example, Newman [10] describes how highly assortative networks percolate faster but the size of the giant component is smaller than when the network is disassortative or random [10]. Thus, the component sizes of the Beta and Stable phases show some indication of assortativity or clustering tendencies. To test this explanation assortativity was computed next.

2. Assortativity

   To gauge the assortativity, the average degree of co-workers was calculated for each degree of the developer degree distributions with at least ten developers. If there is no correlation between the degrees of workers and co-workers, the average co-worker degree should be constant over all degrees. The results for Alpha, Beta and Stable in April and December are shown in Figs. 16.6 and 16.7. There is an upward trend from low worker and co-worker degree to higher worker

Table 16.7 Comparison of empirical and simulated networks for April phases

		Empirical		Simulation from 25 runs		
Phase	Total # of nodes	Largest	Second largest	Largest	StDev	Second largest
Planning	26,623	2,523 (9.5 %)	223	2,263.3 (8.5 %)	840.9	471
Pre-alpha	15,491	209 (1.3 %)	72	199.2 (1.4 %)	64.6	128.3
Alpha	19,063	516 (2.7 %)	193	988.2 (5.2 %)	491.8	336.5
Beta	27,764	6,728 (24.2 %)	229	11,387.7 (41.0 %)	254.3	102.9
Stable	24,748	7,088 (28.6 %)	195	14,169.5 (57.0 %)	153.8	55.5
ALL	128,386	62,671 (48.8 %)	90	98,360.9 (71.2 %)	256.8	37.7

Table 16.8 Comparison of empirical and simulated networks for December phases

		Empirical		Simulation from 25 runs		
Phase	Total # of nodes	Largest	Second largest	Largest	StDev	Second largest
Planning	29,959	2,241 (7.5 %)	228	2,567.0 (8.5 %)	739.3	530.7
Alpha	21,402	329 (1.5 %)	219	1,045.0 (4.9 %)	499.7	308.8
Pre-alpha	17,676	238 (1.3 %)	68	222.2 (1.3 %)	491.8	152.8
Beta	31,606	7,936 (25.1 %)	86	12,997.1 (41.1 %)	254.3	110.5
Stable	28,072	7,735 (27.6 %)	190	15,711.3 (56.0 %)	153.8	67.7
ALL	131,771	65,518 (49.7 %)	116	95,153.4 (72.2 %)	256.8	35.7

and co-worker degree. However, the underlying distributions are very long-tailed since over 80 % of the developers, over all phases, only work on a single project.

The degree distributions of co-workers resemble Pareto distributions [30, 31]. Pareto distributions are of the form: $Pr(X > x) \sim x^{-k}$ (with mean $k/(k-1)$ for $k > 1$), which is the cumulative distribution derived from the Power Law distribution function: $Pr(X = x) \sim x^{-(k+1)}$. A log-log plot of a Power Law frequency distribution is a straight line, and with slope $= -(k+1)$, from which the mean of the Pareto distribution is $k/(k-1)$. Figure 16.7 shows examples of December Stable co-worker degree distributions for developers with degree 1, 3 and 6. The distributions are shown as log-log plots of log degree versus log count of co-workers with that degree, and are approximately linear. As the fitted slopes $-(k+1)$ decrease, the mean degrees $k/(k-1)$ become greater. Figure 16.7 shows the mean degrees if Pareto distributions are assumed. For both tables there is a general increase in mean degree of co-workers with increases in developer degree.

16.3.4 Analysis of Network Change

While the analysis of component size distribution is only cross-sectional, the community social networks across two time periods were first analyzed to identify whether and how clusters interlinked over time by virtue of new links. Exponential Random Graph Modeling, also called P* analysis was performed to understand whether any project level attributes results in clustering of links. The network attachment logics of accumulative advantage and multiconnectivity were examined by enumerating the new links formed and dissolved by a small samples of projects.

1. Network Change
   To understand the evolution of the OSS network we also compared how the clusters in the networks across the two time periods – April and December 2005 – changed. Clustering of the largest component of the Stable subnetwork was done based on the sign patterns of the eigenvectors. The project-project network was clustered for both time periods (April 2005 Stable and December 2005 Stable subnetworks) by considering the number of common developers as a binary link between each project pair. As displayed in Tables 16.9 and 16.10 four clusters emerged of which two were the largest and about 95 % of the projects fell in both these clusters. Interestingly, we observe a very high degree of clustering with over 98 % of the developers within any cluster linked to other projects within the same cluster. Many small components from April Stable (top five are shown in Table 16.9) were added to December Stable. Some of these components merged with the two large April clusters whereas some did not. This illustrates that over time developers tend to participate in other more connected projects thus resulting in the small components of the network coalescing into the large components. However, when the new projects were identified for December 2005, it was found that almost all new projects joined one of the April 2005 clusters without any increase in the number of links across clusters (Fig. 16.8). In other words, the clusters across the 6 month period remained stable with very little inter-cluster links (Table 16.8). This suggests that developers are perhaps restricted to certain clusters based on certain project-specific attributes such as operating systems, programming languages, intended audience or licenses. These differences can impose a technological and knowledge barrier that prevents projects having different attributes from sharing developers. The P* analysis presented next was used to understand such clustering behaviour.
2. Homophily
   Further examination was done by using P* to blockmodel the Stable subnetwork [22]. P* is a class of logit models which are used to model link formation by accounting for the non-independence across dyads (observations) [33]. A key measure of model fit is the change in chi-square, whereas the Wald statistic is used to assess the significance of each parameter. Since the networks under consideration are non-directed project-project networks, only three types of simplistic network tendencies were examined for their effects, apart from the blocks (programming language, operating systems, intended audiences and licenses). Therefore, four blocks were included – programming language, operating systems, intended audience and licenses.

   Since a project can be listed under multiple licenses, operating systems, programming languages and intended audiences, additional simplifying rules

Table 16.9 Network clustering (April 2005)

	Lower-left	Upper-left	Lower-right	Upper-right	Total
LL	570[a]	3	4	0	577
	98.79 %[b]	0.52 %	0.69 %	0 %	7.43 %
	98.28 %[c]	0.07 %	0.15 %	0 %	
LR	7	4,219	3	1	4,230
	0.17 %	99.74 %	0.07 %	0.02 %	54.43 %
	1.21 %	99.83 %	0.11 %	0.39 %	
UL	3	3	2,699	4	2,709
	0.11 %	0.11 %	99.63 %	0.15 %	34.86 %
	0.52 %	0.07 %	99.74 %	1.54 %	
UR	0	1	0	254	255
	0 %	0.39 %	0.00 %	99.61 %	3.28 %
	0 %	0.02 %	0.00 %	98.07 %	
Total	580	4,226	2,706	259	7,771
	7.46 %	54.38 %	34.82 %	3.33 %	

[a] COUNT – Count of links from developers of row cluster to projects of column cluster
[b] ROW % – COUNT as a percentage of total links from the row cluster
[c] COL % – COUNT as a percentage of total links to the column cluster

Table 16.10 Network clustering (December 2005)

	New	Lower-left	Upper-left	Lower-right	Upper-right	Total
New	2,042[a]	78	180	171	18	2,489
	82.04 %[b]	3.13 %	7.23 %	6.87 %	0.72 %	29.35 %
	91.78 %[c]	12.40 %	5.71 %	7.56 %	8.57 %	
LL	12	538	3	2	0	555
	2.16 %	96.94 %	0.54 %	0.36 %	0.00 %	6.54 %
	0.54 %	85.83 %	0.10 %	0.09 %	0.00 %	
UL	90	9	2,967	2	1	3,069
	2.93 %	0.29 %	96.68 %	0.07 %	0.03 %	36.19 %
	4.04 %	1.43 %	94.04 %	0.09 %	0.48 %	
LR	78	3	4	2,087	4	2,176
	3.58 %	0.14 %	0.18 %	95.21 %	0.18 %	25.66 %
	3.51 %	0.48 %	0.13 %	92.26 %	1.90 %	
UR	3	1	1	0	187	192
	1.56 %	0.52 %	0.52 %	0.00 %	97.40 %	2.26 %
	0.13 %	0.16 %	0.03 %	0.00 %	89.05 %	
Total	2,225	629	3,155	2,262	210	8,481
	26.24 %	7.42 %	37.20 %	26.67 %	2.48 %	

[a] COUNT – Count of links from developers of row cluster to projects of column cluster
[b] ROW % – COUNT as a percentage of total links from the row cluster
[c] COL % – COUNT as a percentage of total links to the column cluster

Table 16.11 New link formation

Type of project	New pairs Apr-2005	Old pairs Dec-2005	New links Apr-2005	Old links Dec-2005
Planning	104	226	116	242
Pre-alpha	48	56	52	66
Alpha	92	74	96	78
Beta	436	148	458	148
Stable	496	272	514	278
N/A	140	32	144	34

were used to create and assign blocks (such as the most common license was used to assign the block (dummy variable)). Most projects were listed under a single license, whereas the operating systems were clubbed into three categories POSIX, OS Independent and Windows. This categorization is somewhat consistent with prior exploratory studies [32]. Of these four blocks, only programming language could be used to derive reasonably clean clusters. This is consistent with recent findings on the linking behavior by mature projects on Sourceforge [2] Therefore, in the P* models described next, only programming language was used as a block.

As in the P* output (Figs. 16.9 and 16.10 and Table 16.16), the parameter (left most column) refers to the network variable (1 = choice, 5 = degree and 6 = cyclicity). Separate parameters were assigned to each block for the same network mechanism. As seen 36 % of the '1' links were correctly predicted. The importance of clustering was confirmed when another model was run excluding Cyclicity as a network mechanism (parameter 6). That the chi-square reduced dramatically (doubled) suggests a significant level of clustering of links based on programming language, indicating that developers might seek multiple projects sharing a common programming language (Table 16.11).

3. Preferential Attachment

The rationale of preferential attachment suggests that the more connected projects increase their degree centrality more than the less connected projects would. To test this in an exploratory fashion, we identified the December Stable projects which were also April Stable projects and then calculated the increase in their degrees in the project-project network in two ways: (a) Difference between weighted link counts of December Stable and April Stable projects. The value of a weighted link between two projects is the number of developers they have in common. This difference would then count the number of new developers coming to each project. (b) Difference in the number of links between projects in December and April, ignoring their weights. This counts the number of new projects each project is connected to by developers new to either project.

The top 20 projects with the largest increase in degree amount and degree count are shown in Tables 16.12 and 16.13. Thus, for example, project e107, which has the maximum number of new developer links, had 13 developers in common with other projects in April 2005. Since the rank of this project is

Table 16.12 Project link amounts in the bipartite network (counting binary Project-Project links)

Project name	December new project links	April project link amount	April link amount rank (max 87)
e107	40	13	75
*moodle	33	119	13
Lportal	28	43	45
*collective	26	430	1
*php-blog	25	56	35
wicket-stuff	24	23	65
Tahoe	24	2	86
*sblim	19	18	70
Ebxmlrr	18	25	63
Vtigercrm	16	7	81
typo3xdev	16	51	39
Gate	16	1	87
Wxcode	15	15	73
Bmfo	13	11	77
*wikindx	12	7	81
Jpivot	11	12	76
Devkitpro	11	9	79
Chipmark	11	14	74
*archetypes	11	185	3
*abbot	11	10	78

75, it implies that there were 74 other OSS projects which had more than 13 developers in common with other projects in April 2005. e107 therefore had 40 more developers who joined the project and were at the same time participating in other projects. Based on the preferential logic attachment, we would have instead expected to see the top ranked projects (e.g., collective and archetypes which had a rank of 1 and 3 respectively) acquire the maximum number of new developers in common with other projects. However, this does not appear to be the case from Table 16.12 and also from Table 16.13, which captures the number of new project-project links (instead of the number of developers). Again, we do not find evidence of the top ranked projects forming the most number of new links in the network. (Note: The amount and number of links in the April network, along with their ranks are also included in both Tables 16.12 and 16.13. There is some overlap of project; the seven that appear in both tables are marked with *. In both tables at least one project with low rank (1, 2, 3) appears in the table, and one project (collective) with very low rank appears in both.)

4. Multiconnectivity
The logic of multiconnectivity suggests that over time the prominent developers will engage in more diverse projects. This exploration tendency is proposed to be characteristic of popular developers since it helps them to engage in broader learning. To test this intuition, developers involved with many projects for all

Table 16.13 Project link numbers in one-mode project network (counting binary Project-Project links)

Project name	December new project links	April project link amount	April link amount rank (max 87)
*collective	10	33	2
*php-blog	5	13	17
amis	5	1	29
poedit	4	9	21
*moodie	25	56	35
dirstorage	24	23	65
*archetypes	24	2	86
*abbot	4	2	28
zanebug	3	2	28
xmule	3	1	29
xine	3	10	20
winpooch	3	1	29
*wikindx	3	1	29
wcx	3	1	29
tortoisecvs	3	13	17
syncml ctoolkit	3	3	27
Sotf	3	2	28
Shorewall	3	2	28
*sblim	11	185	3
Plone	11	10	78

the April 2005 data were selected. Of the 96,348 developers involved in multiple projects (across all phases), 86 of them were involved in at least ten projects ("high degree"), and they represent only 0.09 % of all the developers as shown in Table 16.14. These developers were then identified in the large component of the April 2005 Stable developer-project network by using a prominence representation [34]. The projects are colored by the clustering found for this component (Fig. 16.11). Of the 86 developers with at least ten projects, 55 of them appear in this component (and therefore no other), and their degree in this component is shown in Table 16.14. The prominence representation raises each of the 55 "high degree" by an amount proportional to their total April degree above the plane showing the connections among other developers and projects. It is clear that most of these high-degree developers are involved with the two main clusters of projects, although only one of them (Steinbeck) works on projects in both the two main clusters. The top 20 of the high degree developers in this component were then selected for an examination of the diversity of the projects they are involved in based on the project attributes cluster, license, OS, language and audience; the results are shown in Table 16.17. It is observed that there is much more variation in the licenses and operating systems than in the audiences and languages. Table 16.15 shows the number of projects these 20 developers were involved in over all phases of the December 2005 data. There is very little change in the number of projects for these developers between April and December 2005. Thus, the results do not support the multi-connectivity as a logic driving the formation of links in this community.

16.4 Contribution, Limitations and Future Research

Re-organization of the open source community occurs dynamically as software developers participate in multiple projects thus creating an interlinked structure. Such interlinking is beneficial for the software community due to the high potential for software reuse [35]. However, OSS projects do not achieve success spontaneously. Despite the popularity of many OSS products, such as the Mozilla Browser or the Linux operating system, only a small percentage of OSS projects successfully compete with commercial software products. Those that do are considered to be of very high quality and enjoy significant developer and user adoption, whereas significant effort is wasted by developers and software firms by investing in OSS projects that do not become successful and are then discarded. These challenges make it important to understand the evolution of the open source network and thereby infer developers' project-joining behaviors.

The open source community is like a storehouse of software knowledge where each project member's access to the knowledge is determined by how they are connected to the rest of the network. Considering the open innovation paradigm where participation is entirely voluntary, it is tempting to conclude that individuals will be able to broaden their access to the software expertise residing in the community by participating in a diverse portfolio of projects or will engage in exploration at the cost of exploitation [52].

A distinctive feature of the open innovation communities is the emergence of an invisible network of collaborators spread throughout the community [36] termed as the "main component" of the network. Members of this component are presumed to be able to access knowledge elsewhere by accessing the multiple intermediate links to the source of the knowledge. While we are not focusing on understanding how network processes generate positive and negative outcomes for developers, we try to fuse the principles from complex networks with current understanding of social theory.

Overall, results do not indicate support for the two logics of preferential attachment and also multi-connectivity. The absence of preferential attachment in this network is consistent with recent work on online communities [49, 50]. However, we do find homophilous linking behavior in terms of programming languages and assortativity. More specifically, the results suggest that assortativity leads beta projects to form smaller but more robust (more densely connected) giant components than early stage networks. We also find that over time, these components sustain their boundaries and do not coalesce very easily with other components. Perhaps the barriers arising from various project attributes are unsurmountable in that developers restrict themselves to their original clusters. A

Table 16.14 Statistics

Developer degree for April 2005 all phases							
MEAN	VAR	STDEV	MIN	MAX	MED	SIZE	BINS
1.31	0.7514	0.8668	1	50	1	96,348	25
High degree developer projects in April stable large component							
MEAN	VAR	STDEV	MIN	MAX	MED	SIZE	BINS
5.07	15.78	3.972	1	11	4	55	11

Degree distributions					
Developer degree for April 2005 all phases			Higher degree developer projects in April stable large component		
Values	Count	Percentage (%)	Values	Count	Percentage (%)
1	78,245	81.20	1	2	3.60
2	12,096	12.60	2	15	27.30
3	3,523	3.70	3	11	21.80
4	1,307	1.40	4	13	23.60
5	573	0.60	5	4	7.30
6	267	0.30	6	4	7.30
7	134	0.10	7	2	3.60
8	78	0.10	8	1	1.80
9	39	0.00	9	1	1.80
10	25	0.00	11	1	1.80
11	15	0.00			
12	12	0.00			
13	8	0.00			
14	5	0.00			
15	2	0.00			
16	6	0.00			
17	2	0.00			
18	1	0.00			
19	4	0.00			
20	1	0.00			
21	1	0.00			
28	1	0.00			
29	1	0.00			
2	1	0.00			

surprising result is that we do not find much evidence of preferential attachment since neither the top 20 developers were observed to form more links nor did we observe the top 20 projects forming more links than the rest of the projects. Inference of scale-free behavior from past studies, however, are inconsistent with this observation. In fact, we do find highly skewed degree distributions in our data as well.

We acknowledge the following limitations of our analyses. First, Howison and Crowston [37] identify several concerns with the Sourceforge data set, which we have attempted to address in our sampling procedure. However, the importance of including as large a portion of the population as possible is suggested by network

Table 16.15 December 2005 Total degree of top 20 April stable large component developers

Developer (degree)	Total degree	Developer (degree)	Total degree
saunup(50)	50	dreamcatcher(12)	13
hobbs(32)	32	bwarsaw(13)	13
paul-h (29)	27	thusted(12)	12
joeretro(21)	21	limi(12)	12
andreas_kupries(15)	16	aegis(13)	12
timriker(14)	14	steinbeck(10)	11
roock(13)	14	hobbs2(11)	11
loewis(13)	14	das(11)	11
myers_carpenter(13)	13	rinkrank(10)	10
hzeller(13)	13	fdrake(10)	10

literature in order to [38] avoid bias in the analyses. Since the focus of our theory building is explaining the formation of links, exclusion of projects and developers in the initial sample poses the risk that new links from the excluded projects are not captured and thus can give an inaccurate picture. Our exclusion of the isolates across both time periods is based on the premise that these projects reflect a lack of commitment on part of the developers and therefore do not capture the phenomenon of our interest. Second, we have separately analyzed the subnetworks for the various phases since that is computationally feasible. Obviously, therefore, the analysis did not include the links between projects in different subnetworks. We have not been able to quantify in this paper as to how much bias has entered into our results by segregating the entire community in this fashion. Third, in some of our tests, especially those that analyze the preferential attachment and multi-connectivity rationales, we have sampled only the first few developers, which is too small a sample size for statistical validation. Finally, we have defined co-workers simply as those belonging to the same core group of a project. However, more fine-grained analysis can be performed by acquiring more granular data on the collaboration behavior on discussion forums and software versioning systems used on Sourceforge or other open source portals (Tables 16.16 and 16.17).

The findings in this study have several implications. The finding that commonality in programming languages is the only project attribute that leads to cross-participation, highlights the challenges faced by an open innovation community in achieving its key objective, which is to foster and synthesize diverse pieces of knowledge from a very large number of experts. However, such collaboration, mostly facilitated by an online knowledge exchange platform such as Sourceforge, is undermined by the formation of clusters. While formation of clusters of dense ties has its benefits in terms of greater trust and transactive memory systems, it also has pitfalls in that its signifies artificial barriers to the synthesis of diverse expertise. Future research efforts need to be directed towards a deeper understanding of how the clusters in this community network are bridged by specific types of developers, such as experts with niche skills or diverse experience. Understanding this bridging

Table 16.16 Blocking scheme[a]

2	1	1	1	1	1	1	1	1	1
1	2	1	1	1	1	1	1	1	1
1	1	2	1	1	1	1	1	1	1
1	1	1	3	1	1	1	1	1	1
1	1	1	1	4	1	1	1	1	1
1	1	1	1	1	5	1	1	1	1
1	1	1	1	1	1	6	1	1	1
1	1	1	1	1	1	1	2	1	1
1	1	1	1	1	1	1	1	7	1
1	1	1	1	1	1	1	1	1	8

[a] Links 5564, Nodes = 2,192 (Diagonal not included)
−2 Log PseudoLikelihood – 49,427.423 (52,196.1 with single within-block parameter)
Goodness of Fit = 12,808,212.968
Model Chi-squared = 6,508,489.689
df = 24

Parameter	Block	b	Std. error	PLWald	p(df = 1)	exp(b)	Counts	Errors
1	1	−7.703	0.0333	53,344.787	<0.01	0	2,612	0.000
1	2	−5.199	0.096	294.338	<0.01	0.01	858	0.000
1	3	−4.988	0.230	468.882	<0.01	0.01	140	0.000
1	4	−5.373	0.110	2,371.156	<0.01	0.01	428	0.000
1	5	−4.618	0.259	318.816	<0.01	0	214	0.000
1	6	−5.942	0.131	2,045.129	<0.01	0.01	184	0.000
1	7	−5.960	0.067	7,824.120	<0.01	0	730	0.000
1	8	−5.217	0.317	271.336	<0.01	0	378	0.000
5	1	−0.038	0.005	61.075	<0.01	0.01	19,690	0.000
5	2	−0.466	0.028	277.284	<0.01	0.96	3,620	0.000
5	3	−0.354	0.062	32.789	<0.01	0.63	638	0.000
5	4	0.018	0.007	6.358	<0.01	0.7	8,762	0.000
5	5	0.050	0.021	5.461	<0.02	1.02	4,168	0.000
5	6	0.049	0.022	5.058	<0.05	1.05	980	0.000
5	7	−0.101	0.011	81.024	<0.01	1.05	6,636	0.000
5	8	−0.045	0.019	5.543	<0.02	0.9	11,110	0.000
6	1	1.458	0.015	10,125.617	<0.01	0.96	7,902	0.000
6	2	2.122	0.034	3,871.838	<0.01	4.3	2,430	0.000
6	3	1.977	0.076	683.206	<0.01	8.35	348	0.000
6	4	0.487	0.018	710.648	<0.01	7.22	4,920	0.000
6	5	0.277	0.023	145.375	<0.01	1.63	3,346	0.000
6	6	1.313	0.059	494.170	<0.01	1.32	276	0.000
6	7	1.251	0.028	2,041.907	<0.01	3.72	2,934	0.000
6	8	1.061	0.070	231.741	<0.01	3.5	15,036	0.000

	FIT ATP = 0.5				**RESIDUALS**
	Predicted		2,564		Absolute = 7,334.764
	< P		= P		Squared = 3,915.293
Observed	40796,550		558	21.80 %	
5,564	1	3,558	2,006	78.20 %	
		63.90 %	36.10 %		

Table 16.17 Top 20 developers and project affiliations (see abbreviations in Fig. 16.12)

Developer	*Deg*	Cluster	License	OS	Audience	Language
hobbs (32)	11	Ul	BSD	A32, AP, OI, OSX	SA, Dev	Tcl
loewis (13)	9	Ul	GG, GL, PL, O/P	A32, AP, OP, OI	SA, Dev, SR, Info	Python, C++
bwarsaw (13)	8	Ul	BSD, OA, PL, PS	A32, AP, OI	SA, Dev, Oth	Python
timriker (14)	7	Ul	GG, OS, OA, Art, PD	Lin, SGI, XP OI, H/R	SA, EU, Ed, Oth	Assembler, Tcl, Perl, PHP
thusted (12)	7	Lr	AL, AS, O/P	A32, OI	Dev, EU, SR	Java, C# Python, APL
paul-h (29)	6	Lr	BSD, PL, AS, SI	OI	SA, Dev, EU, Oth	Java, Python
fdrake (10)	6	Ul	MIT, PL, GG	A32, AP, OI	SA, Dev	Python, C
andreas-kupries (13)	5	Ul	BSD	A32, OI	SA, Dev	Tcl
roock (14)	5	Lr	IBM,CP	OI	Dev	Java, Python
myers-carpenter (13)	5	Ul	GG, GL, PD, O/P	A32, AP, OP, PI, W32	SA, EU, SR	Assembler, C, C++, Python
hzeller (13)	5	ul	Mozp, GL, GG	AP, OI, Oth	SA, Dev EU, HI	C, Java
hobbs2 (11)	5	ul	BSD	A32, OSX	SA, Dev	Tcl
steinbeck (10)	4	ul, lr	BSD, GL, GG	OI, Sol	SA, EU SR	Java, Javascript
rinkrank (10)	4	lr	OA, Zope, GG	OI	Dev	Java
limi (12)	4	ul	OA, Zope, GG	OI	SA, Dev	Zope, JavaScript
joeretro (12)	4	lr	BSD, GL, AS, MIT	A32, OI	SA, Dev	Java, Python, C++
dreamcatcher (12)	4	ul	OA, Zope, GG, GL	OI	SA, Dev	Zope, Python
das (11)	4	ul	OA, Zope, GG, GL	A32, Mac	Sa, Dev, EU	Tcl
aegis (13)	4	lr	ZL, GL, GL, O/P	W32, OSX, DOS	Dev, EU	Assembler, C, JavaScript

mechanism is of great importance for engendering greater creativity, software reuse and faster software development. The differences in the network topologies of projects in the early versus latter stages is noteworthy and suggests that future research could be directed towards understanding whether alpha stage projects are disadvantaged because of lower connectivity and assortativity, or whether their later percolation compared to beta stage subnetworks, benefits alpha projects because of greater component sizes and thus greater network reach. It is possible that greater reach achieved because of greater component sizes, allows highly distant ideas to reach alpha projects. These implications need to be examined in greater detail in future work. Our results also cast doubt on the preferential attachment mechanism suggesting that in professional expert networks, preferential attachment may be limited simply by the cognitive limitations of individuals and groups to form such ties. The cognitive limitation also manifests as the absence of the multi-connectivity mechanism suggested in theoretical literature. In summary, our work has fruitfully bridged software engineering research on OSS communities with complex networks theory, in particular, by separately examining the subnetworks for each stage. The findings thus substantively contribute to both streams of literature.

Appendix

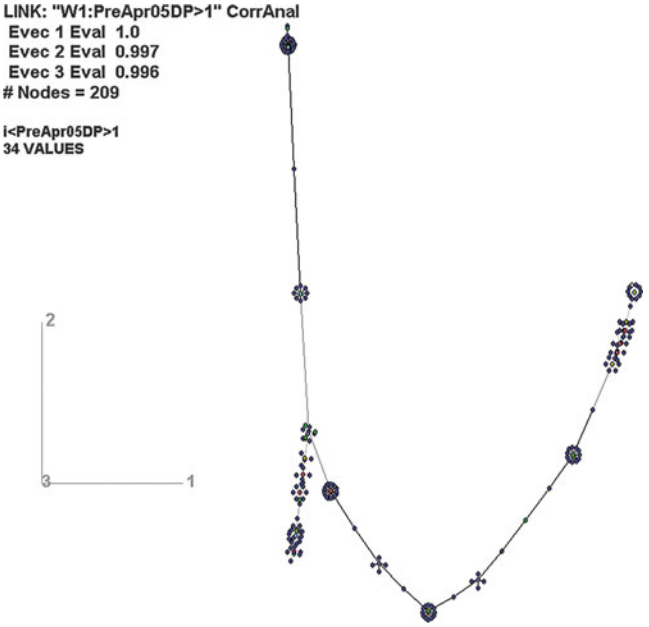


Fig. 16.1 Largest component of the pre-alpha subnetwork (April 2005)

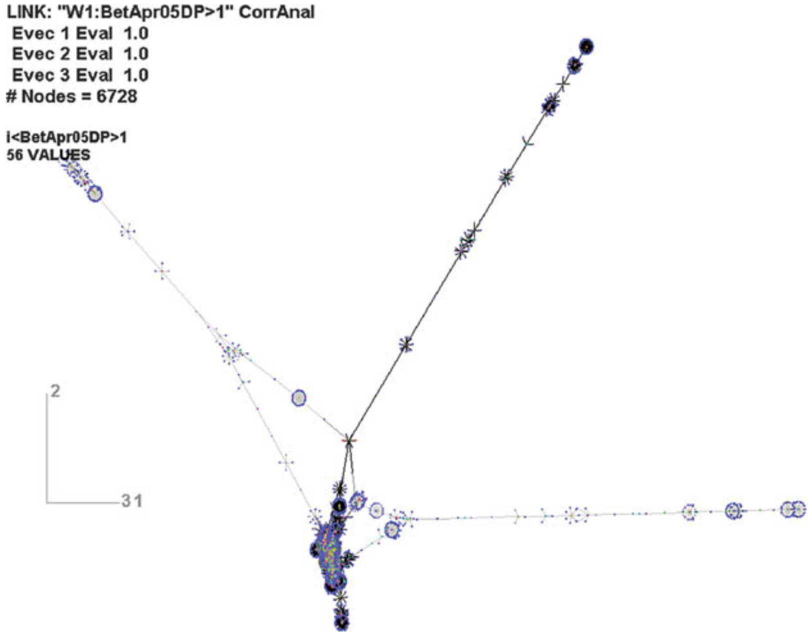


Fig. 16.2 Largest component of the beta subnetwork (April 2005)

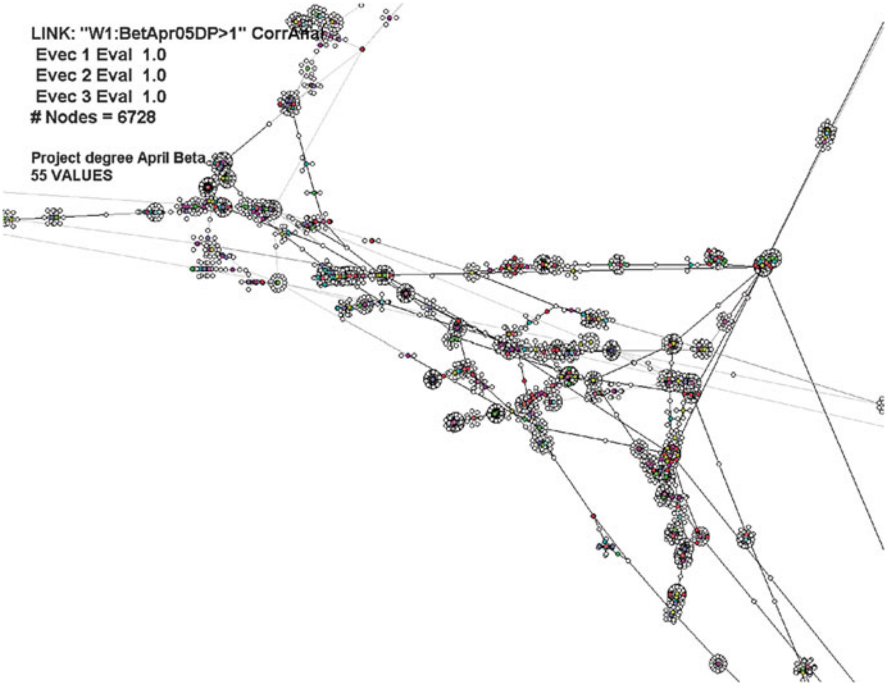


Fig. 16.3 Close-up of the central portion of Fig. 16.2

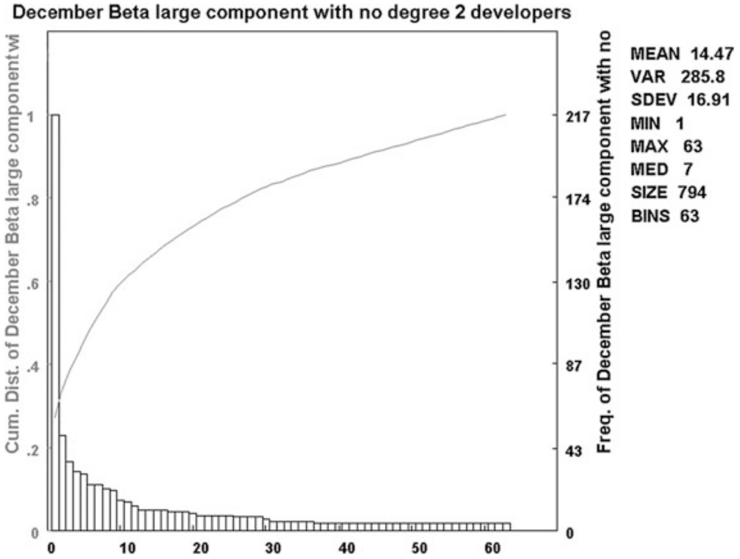


Fig. 16.4 Component sizes for giant component when degree-2 developers are removed (Beta subnetwork)

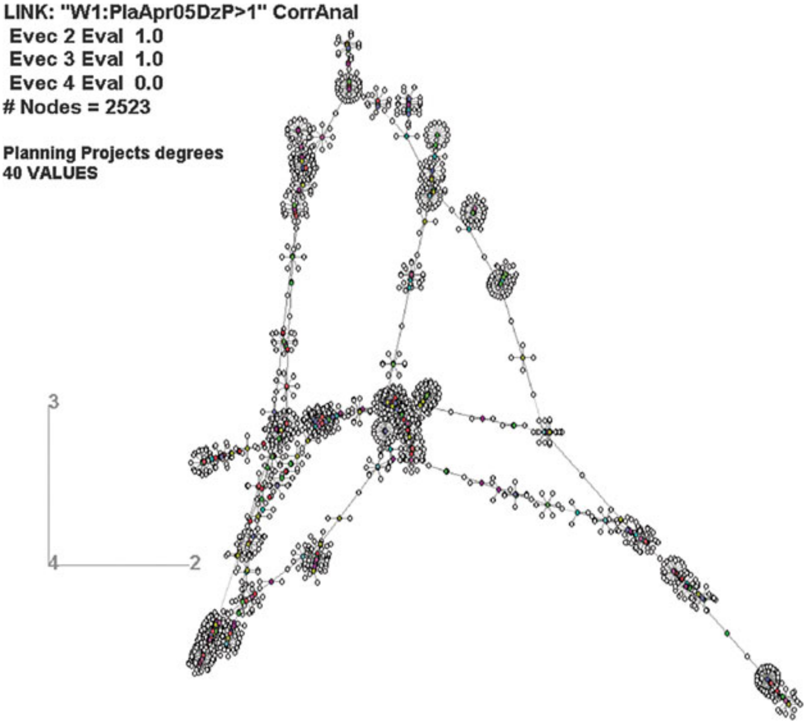


Fig. 16.5 Largest component of the planning subnetwork

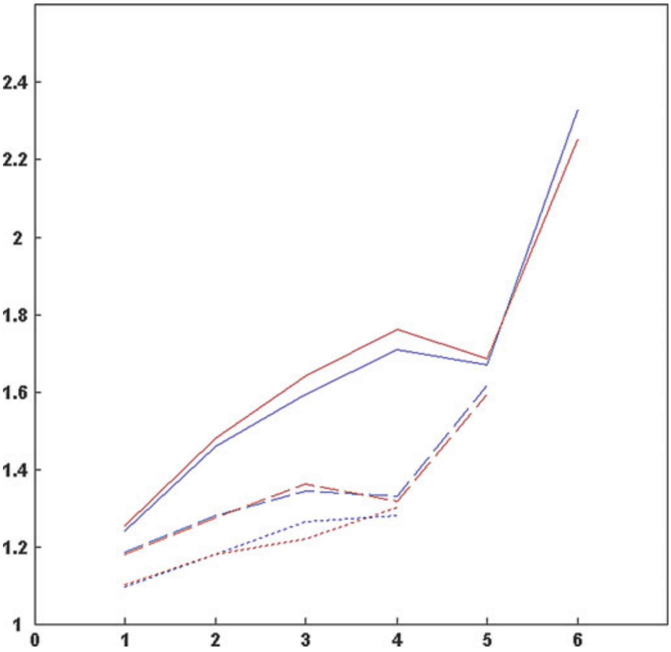


Fig. 16.6 Plots of average degree to co-worker degree (for April + December – Alpha, Beta and Stable sub-networks)

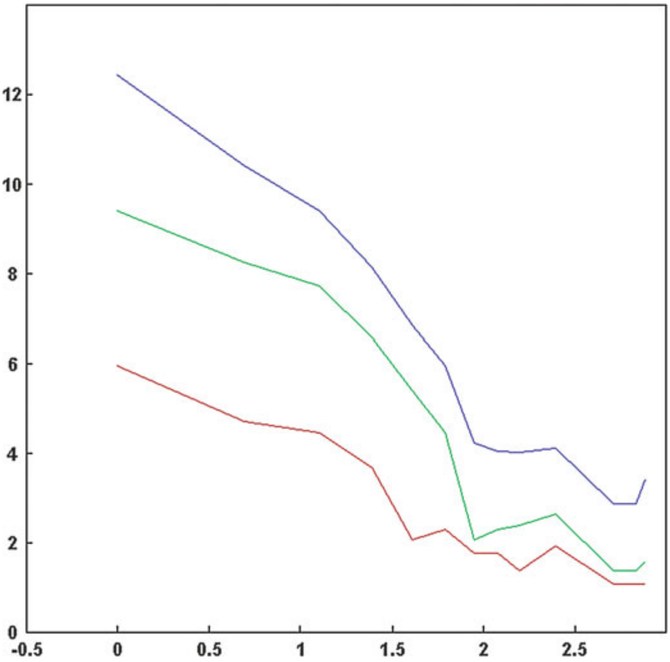


Fig. 16.7 Log-log plot of co-worker degree distributions (log degree vs. log count) (*Dotted*: Alpha (April and December), *dashed*: Beta (April and December), *Solid*: Stable (April and December))

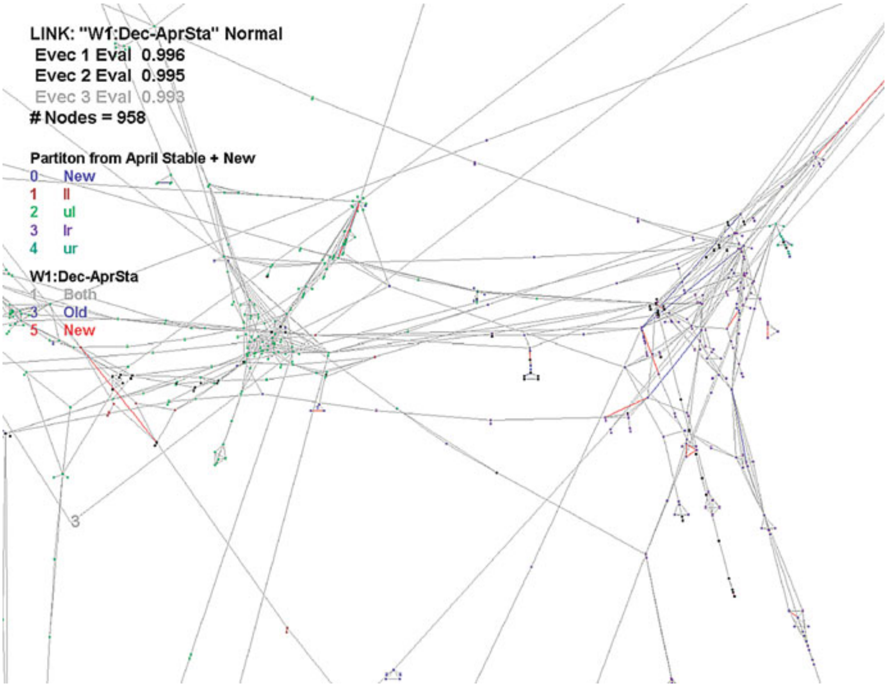


Fig. 16.8 Inter-cluster link (December 2005 stable subnetwork)

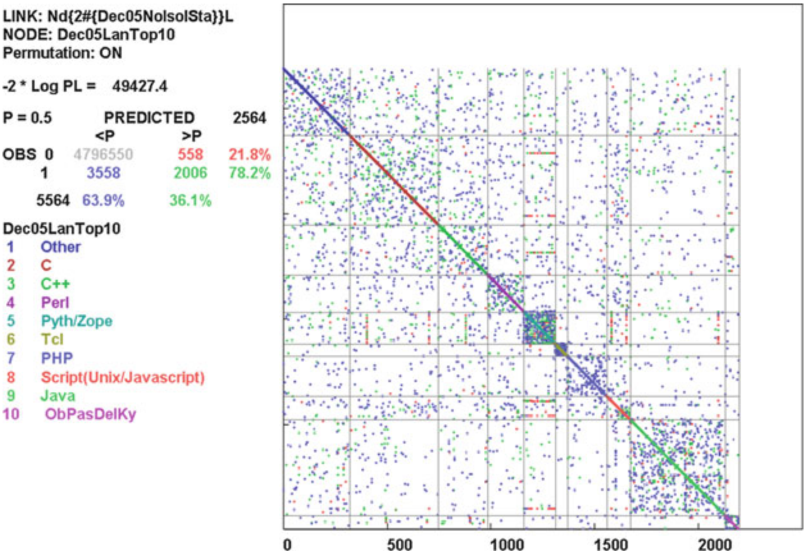


Fig. 16.9 Clustering of projects by programming language (April 2005 stable subnetwork)

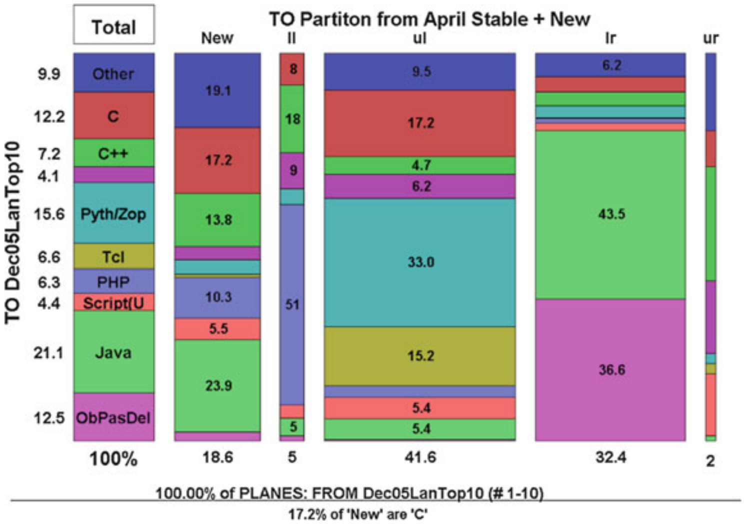


Fig. 16.10 Cross tabs for programming language

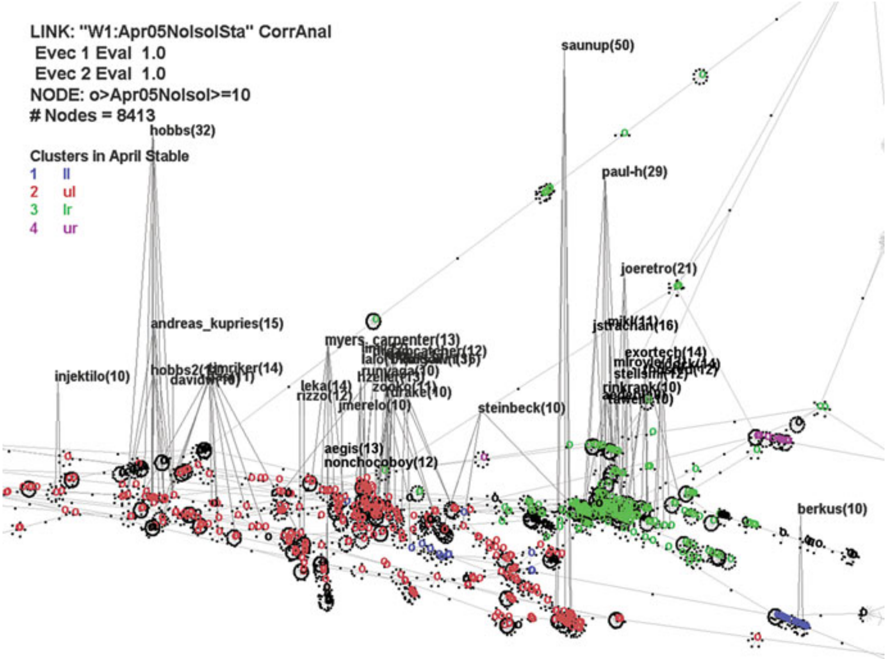
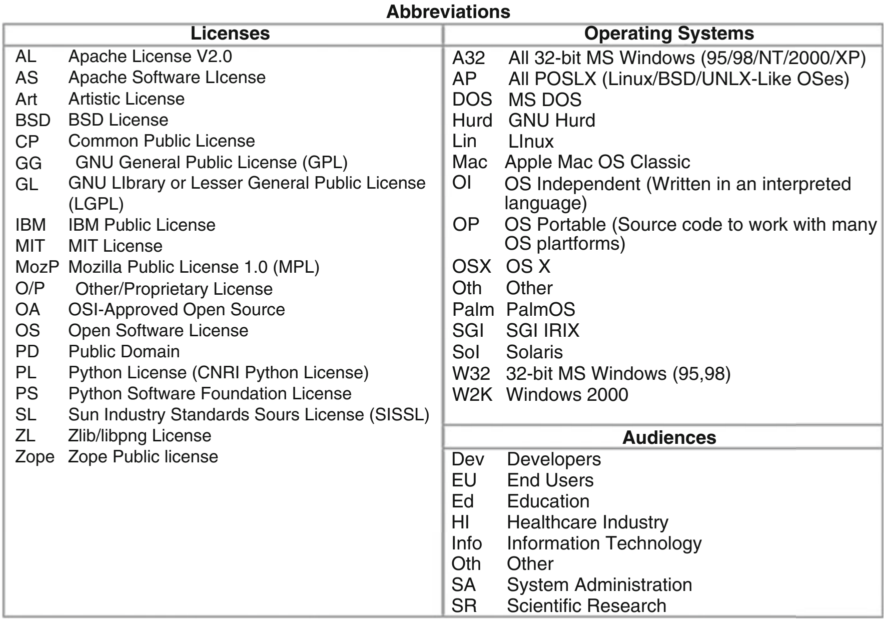


Fig. 16.11 Multiconnectivity of developers



Abbreviations

Licenses	
AL	Apache License V2.0
AS	Apache Software LIcense
Art	Artistic License
BSD	BSD License
CP	Common Public License
GG	GNU General Public License (GPL)
GL	GNU LIbrary or Lesser General Public License (LGPL)
IBM	IBM Public License
MIT	MIT License
MozP	Mozilla Public License 1.0 (MPL)
O/P	Other/Proprietary License
OA	OSI-Approved Open Source
OS	Open Software License
PD	Public Domain
PL	Python License (CNRI Python License)
PS	Python Software Foundation License
SL	Sun Industry Standards Sours License (SISSL)
ZL	Zlib/libpng License
Zope	Zope Public license

Operating Systems	
A32	All 32-bit MS Windows (95/98/NT/2000/XP)
AP	All POSLX (Linux/BSD/UNLX-Like OSes)
DOS	MS DOS
Hurd	GNU Hurd
Lin	LInux
Mac	Apple Mac OS Classic
OI	OS Independent (Written in an interpreted language)
OP	OS Portable (Source code to work with many OS plartforms)
OSX	OS X
Oth	Other
Palm	PalmOS
SGI	SGI IRIX
Sol	Solaris
W32	32-bit MS Windows (95,98)
W2K	Windows 2000

Audiences	
Dev	Developers
EU	End Users
Ed	Education
HI	Healthcare Industry
Info	Information Technology
Oth	Other
SA	System Administration
SR	Scientific Research

Fig. 16.12 Abbreviations

Acknowledgements This project is funded by the Social Sciences and Humanities Research Council of Canada (Grant number 410-2007-1579) and the SFU Discovery Grants program.

References

1. Monge P, Contractor N (2003) Theories of Communication Networks. Oxford University Press.
2. Cottam JA, Lumsdaine A (2008) Extended assortativity and the structure of open source development community. Proc. Sunbelt, St. Pete Beach, FL.
3. Valverde S, Sole RV et al (2007) Self-organization versus hierarchy in open-source social networks. Physical Review E 76 (4): 046–118
4. Xu J, Gao Y Christley S Madey G (2005) A topographical analysis of the open source software development community. In: Proceedings of the 38th Hawaii International Conference on System Sciences.
5. Barabasi L, Albert R (1999) Emergence of scaling in random networks. Science 286: 509–512
6. Kong JS, Sarshar N, Roychowdhury VP (1999) Experience versus talent shapes the structure of the Web. Proceedings of the National Academy of Sciences 286: 13724–13729
7. Ebel H, Mielsh L-I, Bornholdt S (2003) Scale-free topology of e-mail networks. Physical Review E 66 (3): 035–103
8. Newman MEJ (2003) Why social networks are different from other types of networks. Physical Review E 68
9. Newman MEJ (2003) The structure and function of complex networks. SIAM Review E 45 (2): 167–256
10. Newman MEJ (2002) Assortive mixing in networks. SIAM Review E 89 (20)

11. Newman MEJ (2000) Who is the best connected scientist? A study of scientific coauthorship networks.
12. Amaral LAN, Scala A, Barthalamy M, Stanley HE (2000) Classes of small-world networks, Proceedings of the National Academy of Sciences of the United States of America 97 (21): 11149–11152
13. Watts DJ (2004) The new “science” of networks. Annual Review of Sociology, Annual Reviews Inc. 243–270.
14. Watts DJ (1999) Small Worlds: The dynamics of networks between order and randomness. Princeton University Press.
15. Valverde S, Sole RV, Bedau MA, Packard, N (2007) Topology and evolution of technology innovation networks. Physical Review E 76 (5): 056–118
16. Xulvi-Brunet R, Sokolov IM (2005) Changing correlations in networks: Assortativity and disassortativity. ACTA PHYSICA POLONICA B 5: 1431–1455
17. Guimera R, Uzzi B, Spiro J, Amaral LAN (2005) Team Assembly Mechanisms Determine Collaboration Network Structure
18. Howison J, Conklin M Crowston K (2006) FLOSSMOLE: A collaborative repository for floss research data and analysis. International Journal of Information Technology and Web Engineering 1 (3): 17–26
19. Richard W, Seary A (2005) Multinet for Windowns 4.74. In: Editor (ed) Book MultiNet for Windows 4.74 3rd edn.
20. Powell W, Koput KW, White DR, Owen-Smith J (2005) Network dynamics and field evolution: The growth of inter-organizational collaboration in the life sciences. American Journal of Sociology 110 (4): 11–32
21. Seary AJ (2005) MultiNet: An interactive program for analyzing and visualizing complex networks. Disseration/Thesis, Simon Fraser University.
22. Wasserman S, Faust K, Granovetter M (1994) Social Network Analysis. Cambridge University Press.
23. Newman MEJ, Strogatz SH, Watts DJ (2001) Random graphs with arbitrary degree distributions and their applications. Phys. Rev. E, 64.
24. Bonacich P (1987) Power and Centrality: A family of measures. American Journal of Sociology 92 (5): 1170–1182
25. Freeman LC (1979) Centrality in Social Networks. Social Networks 1 (3): 215–239
26. Erdos P, Renyi A (1960) On the evolution of random graphs. Publications of the Mathematical Institute of the Hungarian Academy of Sciences 5: 17–61
27. Bollabas B (2001) Random Graphs. Cambridge University Press.
28. Newman MEJ (2001) Scientific collaboration networks. I. Network construction and fundamental results. Phys. Rev. E, 64
29. Aiello W, Chung F, Lu L (2001) A Random Graph Model for Massive Graphs. IEEE Symposium on Foundations of Computer Science, 64: 510–519
30. Lorenz MO (1905) Methods of measuring the concentration of wealth. Publications of the American Statistical Association 9: 209–219
31. Adamic LA, Huberman BA (2002) Zipf’s law and the Internet. Glottometrics, 3: 143–150
32. Joon Jun S, Barnett G (2005) The Structure of Open Source Software: A Network Analysis of Open Source Software Project. Paper presented at the annual meeting of the International Communication Association, Sheraton New York, New York City, NY.
33. Wasserman S, Pattison P (1996) Logit Models and Logistic Regressions for Social Networks I: An Introduction to Markov Graphs and p*. Psychometrika, 61: 401–425
34. Michael Baur MB, Brandes U, Cornelsen S, Gaertler M, Kpf B, Lerner J, Wagner D (2001) visone – Software for visual social network analysis. In: Editor (ed) Book visone – Software for Visual Social Network Analysis, Springer-Verlag 2002: 463–464
35. Boehm B (1981) Software engineering economics. Prentice Hall.
36. Merton, RK (1996) On social structure and science. University of Chicago Press.

37. Howison J, Crowston K (2005) The perils and pitfalls of mining SourceForge. Workshop on Mining Software Repositories, 26th International Conference on Software Engineering, Edinburgh, Scotland.
38. Laumann EO, Marsden PV, Prensky D, Burt RS, Minor MJ (1983) The Boundary Specification Problem in Network Analysis. in R.S. Burt and M.J. Minor (Eds), Sage Publications. Applied Network Analysis: 18
39. Singh, Param Vir (2010) The Small World Effect: The Influence of Macro Level Properties of Developer Collaboration Networks on Open Source Project Success. ACM Transactions of Software Engineering and Methodology, 20:2, 1–27
40. Rivera MT, Soderstrom SB, Uzzi B (2010) Dynamics of Dyads in Social Networks: Assortative, Relational, and Proximity Mechanisms. Annual Review of Sociology, 36:1, 91–115.
41. Valetto G, Helander M, Ehrlich K, Chulani S, Wegman M, Williams C (2007) Using Software Repositories to Investigate Socio-technical Congruence in Development Projects. Fourth International Workshop on Mining Software Repositories (MSR'07).
42. Ehrlich K, Valetto G, Helander M (2007). Using Software Repositories to Investigate Socio-technical Congruence in Development Projects. International Conference on Global Software Engineering(ICGSE 2007).
43. Ehrlich K, Lin C-Y, Griffiths-Fisher V (2007). Searching for Experts in the Enterprise: Combining Text and Social Network Analysis. GROUP'07, November 4–7, 2007, Sanibel Island, Florida, USA.
44. Chang K, Ehrlich K (2007). Out of Sight but Not Out of Mind? Informal Networks, Communication and Media Use in Global Software. Proceedings of the 2007 conference of the center for advanced studies on Collaborative research.
45. Kuang-Yuan H, Namjoo C (2011) Relating and Clustering Free/Libre Open Source Software Projects and Developers: A Social Network Perspective. In: Proceedings of the 44th Hawaii International Conference on System Sciences.
46. López-Fernández L, Robles G, Gonzalez-Barahona JM. Applying Social Network Analysis Techniques to Community-Driven Libre Software Projects. IJITWE 1(3): 27–48 (2006)
47. Shen C, Monge P (2011). Who connects with whom? A social network analysis of an online open source software community. First Monday (2011), Volume: 16, Issue: 6, Pages: 1–14
48. Porruvecchio G, Uras S, Quaresima R (2008). Social Network Analysis of Communication in Open Source Projects. Agile Processes in Software Engineering and Extreme Programming, pages 220–221
49. Faraj S, Johnson S (Forthcoming). Network Exchange Patterns in Online Communities. Organization Science.
50. Saraf N, Chandrasekaran D, Siddarth S. How Knowledge Overlap Drives (and Doesn't Drive) Developer Preferences for Joining Related Open Source Software Projects. Available at SSRN: http://ssrn.com/abstract=2002366, 2011
51. Hahn J, Moon JY, Zhang C. Emergence of new project teams from open source software developer networks: Impact of prior collaboration ties. Information Systems Research 19(3) 369–391.
52. March JG. Exploration and exploitation in organizational learning. Organization Science 2(1) 71–87.

Chapter 17
SociQL: A Query Language for the SocialWeb

Diego Serrano, Eleni Stroulia, Denilson Barbosa, and Victor Guana

Abstract Social-networking sites are becoming increasingly popular with users of all ages. With much of our social activity happening online, these sites are now becoming the subject of scholarly study and research. Unfortunately, despite the fact that they collect similar content and support similar relations and activities, the current generation of these sites are hard to query programmatically, offering limited views of their data, effectively becoming disconnected islands of information. We describe SociQL, a high-level query language, and a corresponding service, to which social-networking sites can subscribe, that supports the integrated representation, querying and exploration of disparate social networks. Unlike generic web query languages, SociQL is designed specifically to support the integration of networks through a common information model for the purpose of examining sociological questions, motivated by social theories. The paper discusses the design and rationale for the SociQL language elements and syntax, as well as our experience using the SociQL service to query a variety of social-network sites.

17.1 Introduction

We are witnessing a dramatic increase in user participation in social-networking sites, accompanied by a progressive diversification and specialization of their purpose and manner of use. Social tagging, blogging, instant messaging, shared events and fan clubs are just some of the different ways in which people interact online today. In spite of the variety of social-networking platforms available to users today and their distinct user interfaces, they are all built around similar "objects of sociality", namely, *individuals*, connected as *friends* or *followers*, belonging in informal

D. Serrano (✉) • E. Stroulia • D. Barbosa • V. Guana
Computing Science Department, University of Alberta, 221 Athabasca Hall. Edmonton, AL, Canada
e-mail: serranos@ualberta.ca; stroulia@ualberta.ca; denilson@ualberta.ca; guana@ualberta.ca

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_17,

or formal *communities*, and interacting though a variety of *communication channels* among them. Another common feature among most current social-networking sites is the fact that they are “ego-centric”, hiding from their users most of the network except from their immediate neighbourhoods. However, despite their similar underlying conceptual models, such sites are currently insulated from each other, forcing their members to replicate and maintain much of their data in multiple repositories. As a result, the information is siloed in multiple places, preventing us from leveraging the synergies that could arise by sharing and cross-referencing it. We seek to address both limitations with SociQL. On one hand, the SociQL language is designed from the ground up to support the formulation of queries relevant to social-network analysts. On the other hand, the SociQL service proposes a methodology for integrating data from disparate networks and implements the SociQL language in order to answer such queries of interest across the integrated networks.

The benefits of linking social networks are invaluable. Users would be able to have their unique profiles exposed across sites and effortlessly become part of multiple communities. One could easily receive updates from “friends” across platforms, and interests expressed in one community could easily sip through to others. This integration, in addition to increased information dissemination, could also lead to further community differentiation; instead of requiring that new members replicate their personal information just to start participating in a community, users could invest their efforts in taking advantage of the unique features of each platform, providing and accessing more platform-specific information and services.

These opportunities are even more interesting and relevant in our area of interest, i.e., networks of researchers. There exists a variety of research communities today, for announcing conferences, for sharing one’s publications, or for rating and commenting on publications. We believe that the more these communities become integrated, the more substantive the discourse will become, thus contributing to the research activity itself. Towards these goals, SociQL explicitly models the abstractions necessary to represent the elements and relations captured within a typical social network in general, and a research community in particular.

In order to validate the expressiveness and usefulness of SociQL’s conceptual model, we have used it to model the data captured in two research networks: ReaSoN [14] and the GRAND Forum. ReaSoN (REseArcher SOcial Network http://hypatia.cs.ualberta.ca/reason) is not a “live” social network; instead, it is a repository of information about computer-science research, including researcher profiles, their affiliations with organizations, their publications and citations among them, and publication venues. In this context, SociQL enabled us to study research-collaboration patterns. The GRAND Forum is the collaboratory for the researchers[1] involved with the GRAND (Graphics Animation and New Media) Network of Centres of Excellence. The GRAND community includes a number of distinct projects, organized around five themes. Each project involves researchers from numerous universities across Canada and their industrial partners, producing a variety of types of publications and artifacts. Within GRAND Forum, SociQL

[1] At the time this paper is written, this community includes about 500 members.

enabled us to answer a variety of questions, some of them required by the reporting process of NSERC, the agency funding the network. In addition, SociQL is helping the network manager flexibly explore the information in the GRAND Forum, eliminating the need to develop specif-purpose user interfaces for infrequent yet interesting queries. Our experience with ReaSoN and the GRAND Forum demonstrated that SociQL is expressive and readable enough to enable querying and exploration of the information is these repositories, with relatively little effort.

Our work with SociQL makes several contributions to the general field of social-network analysis.

- First, it proposes the conceptualization of social-networking repositories grounded on a socio-material theory of networks, which is a natural framework for the problem at hand. Along these lines, it supports the browsing and querying of these repositories, as Socio-Material Networks [8]. We have based the language design on the identification of actors, relations among them, and their properties. Furthermore, the language syntax is designed to explore social phenomena, such as induction and homophily.
- The SociQL language is implemented in the SociQL service, which can be integrated with a variety of social networking systems. In a nutshell, the integration of the SociQL service with a new social-network repository involves first, the mapping of the repository data model to the SociQL conceptual model, and second the compilation of SociQL expressions either into SQL queries (assuming the repository database is directly accessible to the SociQL service) or into REST APIs (when the repository supports such access to external systems).
- The deployment of the SociQL service is further supported by a visual query editor (VQE) and two special-purpose visualizations for the query results. The SociQL VQE enables users to easily define queries of interest, which are subsequently automatically interpreted to access the data of the underlying resources. The visualizations of query results are designed to better communicate privileged properties and relations of the social network subsets represented in the query results, namely geographic distribution of actors and their interconnections.
- Finally, we discuss the integration of the SociQL service with the GRAND Forum and ReaSoN systems, and the support that it provides towards establishing links across systems. Furthermore, we have experimented with the language and the service through several exploratory queries in our database in concert with information available in other online resources, such as DBpedia [5] and Facebook.

Through these technologies, i.e., (a) the SociQL language, (b) the compilation methods to SQL and REST APIs and (c) the visual SociQL query editor, we aim at advancing the research agenda of interoperability across social networks.

The rest of this chapter is organized as follows. Section 17.2 presents background concepts related with social networking theories, and query languages. Section 17.3 presents SociQL conceptual framework. Section 17.4 shows the language design and data model using the ReaSoN project context. Section 17.5 exposes SociQL architecture and implementation. Section 17.6 portrays SociQL interoperability and querying examples using the GRAND research network. Section 17.7 closes with our conclusions and our future research agenda.

17.2 Background

In this section, we briefly discuss the sociology theories that guided our design and implementation of SociQL, and we review the area of special-purpose query languages for the web.

17.2.1 Social Networking Theories

Social-networking applications share the same conceptual models since they all intend to model similar social interactions, namely their common interest or use of shared objects. Bojars et al. use the term "object-centered sociality" referring to the hypothesis that people are connected by a shared object, which is why sociologists often prefer to talk about "socio-material networks" [8]. Social-network theories attempt to explain the causes of social correlation [3] that, for the case of online social networks, can be roughly categorized in two types: *induction*, and *homophily*.

Induction refers to the influence induced by the social context to a person's behavior. The induction phenomenon in a social network can be recognized as the tendency of a group of actors to exhibit behaviors similar to the source of influence. As an example of this phenomenon consider, for example, the situation of an author who decides to use a keyword because some of his/her colleagues/friends have recently adopted it [3]. The importance of identifying induction as the type of correlation lies in its possible use for recognizing the creation of research paradigms and "fads".

Homophily is the tendency of individuals to associate and bond with similar others. Individuals in homophilic relationships share common characteristics (beliefs, values, behaviors, etc.) that make communication and relationship formation easier. Homophily also takes into account external influences from the environment, such as geography and family ties. The homophily hypothesis is straightforward for individuals [17]. At the individual level, persons are more likely to have a connection, friendship or association if they have common attributes [6]. And while common norms are promoted through common attributes, so are common attributes likely when association or friendship occurs as a result of collocation and commonly situated activities [2].

17.2.2 Web Query Languages

The problem of web query-language design has been of long-term interest to the academic community and industry. In the design of query languages for the web, the notion of a graph data model as an abstraction is fundamental. Pioneering web query languages, such as WebOQL [4], WebSQL [4], and WebDB [19] model the web as a graph over which queries can be expressed using familiar (essentially relational) constructs. In the realm of the semantic web, in which resources are explicitly described as graphs and a multitude of languages have been proposed, SPARQL [24] emerged as the standard, offering an expressive graph pattern-matching metaphor [22]. However, these languages were not designed with social analysts in mind, and thus none of them provides the proper abstractions in terms of which one would consider querying social networks. On the other hand, they deal with graphs consisting of nodes (e.g., web pages) and vertices (e.g., links between web pages) and it is possible to use schemas, to further refine the kinds of nodes and edges that are allowed; for instance, SPARQL allows the use of domain ontologies that restrict the kinds of graphs it can handle.

A more serious limitation of using graph-based query languages in social network analysis is due to their *generality* and their *expressive power*. These languages are said to be general because they were designed to deal with arbitrary graphs, irrespective of what they represent or mean. Therefore, queries in these languages are expressed at the node and edge level; similarly, computations (e.g., finding the centrality of a node in a network) can only be expressed at the algorithmic level. For instance, if one needs to find the *centrality* of a node in a network (which is a basic operation in social-network analysis), one needs to write the algorithm for computing the centrality as part of the query, provided that the language is powerful enough for expressing the necessary algorithm. This is a severe limitation for most social-network analysts who are typically not trained on programming, nor on declarative query processing to write the complex queries required when using these languages.

In terms of expressive power, it is long known that the more expressive a query language is, the harder (computationally) it is to process and optimize queries in that language [1]. Moreover, it so happens that many of the queries of interest to social-network analysts involve computations which are impossible or very hard to optimize in a generic query language. For example, using the SQL query language implemented by most relational database systems,[2] it is impossible to write a query that checks if there is a path connecting two arbitrary nodes in a graph. This is because, for performance reasons, the original SQL standard does not support the notion of *recursion*, which is common in most graph-processing techniques. It must be noted that most graph query languages in the literature are *significantly* more expressive than SQL. The implication here is that they are commensurately harder to process, and more complex to use.

What is needed here is not a generic and powerful query language, which makes queries hard for the user to write, and computationally expensive to process. Instead, we need *specialized* query languages, that provide the right abstractions for their intended users, and can be efficiently implemented.

[2] To be precise, we refer to the SQL:99 edition of the ISO standard, which is supported by the majority of vendors.

17.2.3 Social Networking Data Services

The first steps towards the right querying model for social networks were provided by Yahoo! and Facebook, who have implemented data services with simple query languages allowing client applications to access their data. In 2007, Facebook introduced the FQL query language [12], which allows Facebook application developers to use a SQL-style interface to more easily query the same Facebook social data that can be accessed through other Facebook API methods. The clauses are of the form *select-from-where*, allowing only a single relation in the *from* clause. Joins are not explicitly allowed in the language, but can be simulated by using nested queries.

The Yahoo! approach is known as YQL [25], and is similar to FQL. The grammar of the language is the same; the only difference is in the possibility to use *show* and *desc* clauses to obtain metadata from the source. In both languages (FQL and YQL), the queries are fairly restricted, in order to achieve better performance. Moreover, unlike the web-oriented query languages, YQL and FQL focus on providing "ego-centric" queries in which a small portion of the networks are visited during the query (usually pertaining to a single user of his/her connections).

17.2.4 Special-Purpose Query Languages

Our primary objective in designing SociQL was to develop a domain-specific language that would make sociologically relevant constructs "first class citizens"; the language primitives should reflect the concepts in terms of which sociologists think about social networks and the syntax should make straightforward the formulation of queries, driven by sociological theories. Therefore, we fully embrace the notion of "socio-material networks" [10], where social actors relate to each other through objects.

We also recognize that the same social entities co-exist and interact in many ways and in multiple overlapping networks simultaneously. For example, two researchers may co-author papers and may also cite each other, thus relating to each other through both co-authorship and citation relations. At the same time, in addition to being related within their research social network, they may also be "facebook friends". A secondary objective was to balance the trade-off between expressiveness and readability on one hand, and performance, on the other.

The work most closely related to SociQL is Ronen and Shmueli's [23] SoQL, a new language for querying and creating data in social networks. Like SociQL, this language is designed, with performance in mind, to manage the increasing volumes of data and includes two features for querying social networks: *path* and *group*. A path is an ordered set of network participants in which every consecutive two are friends, and a group is essentially a set of participants. The main element of a query is either a path or a group, with subpaths, subgroups and paths within a group defined in the query. SoQL suffers from two fundamental limitations as compared to

SociQL. First, it relies on an over-simplified conceptual model of the social network, assuming unimodal networks and bidirectional relationships, which is not realistic in modern social networks. Moreover, SoQL does not offer any support for queries across social networks.

17.3 The SociQL Conceptual Framework

In sociology, object-centered sociality characterizes social relations between individuals by means of objects [10]. Essentially, while recognizing the social interaction between individuals, this theory exalts the role of specific objects as the reasons why social actors affiliate with each other. In the same spirit, the theory posits the "objectualization" of social relations in which objects progressively displace persons as relationship partners and increasingly mediate social relationships. For this reason, we define the SociQL data model around the concept of an *object*. For instance, in the context of ReaSoN, we have that a paper (an object) connects the researchers who authored it; similarly, a publication venue (an object) connects authors who publish their work in it. In our model, both objects and *relations* are associated with properties (actual data), such as the name of an author, or the date in which a citation is made from a paper into another. We also distinguish the *context* in which properties are defined to describe the objects and relationships. For instance, the same query might return different email addresses for the same individual depending on the social network context in which the query is asked (professional or personal). In practice, each context corresponds to different social-network system; thus, each context may have its unique data access methods and privacy restrictions, which complicates query processing to a great extent (as discussed below).

Figure 17.1 illustrates (in a schematic way) some key elements of the SociQL model (contexts, objects, properties, and relationships), applied to (fragments of) ReaSoN and Facebook. In this example, ReaSoN (context) describes three kinds of objects, Authors, Papers, and Venues, which are related by the Writes and Appears relationships. Authors in the ReaSoN context are also related to Users in the Facebook context, with the special-purpose relation, sameAs.

In designing SociQL, we have developed a simple, yet comprehensive, model for capturing the semantics of the information contained and extracted from social networks. We define a Social Network and its basic components as a four-tuple $(O;R;PO;PR)$. The object set O is a set of social objects, or individuals, in what we call socio-material networks, for example, "authors" and "papers". The relation set R is a set of links or edges between the objects in O that represent the flow of information of materials, like the relation "writes" or "affiliated-with". All objects and relations have properties that define them, represented by PO and PR respectively, like the "title" and the "date" of publication for a "paper", or the "starting date" in the "affiliated-with" relation. Every property, be it an object property or a relation property, contains a value of a given type (e.g., integer, boolean, string). So, values are objects whose associated meaning is universally

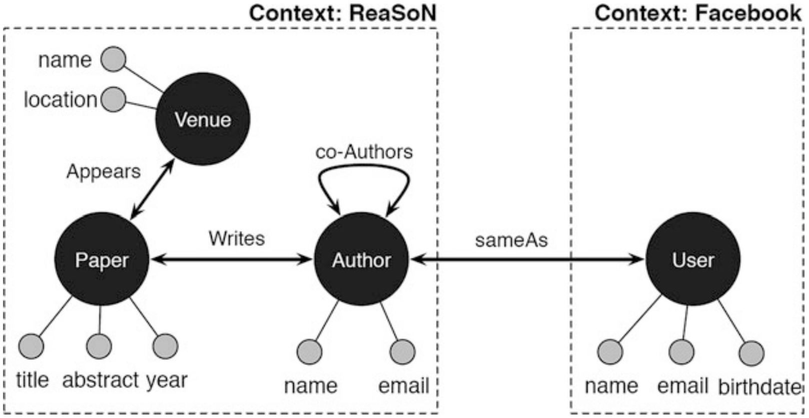


Fig. 17.1 The data models of two contexts, ReaSoN and Facebook, mapped to the SociQL data model

agreed upon. We define the types as a finite non empty set T; and the domain D associated to each type $t \in T$, as $dom(t) \subseteq D$. In contrast to values, social objects represent real or conceptual objects in the world, which are defined in terms of their properties. In the representation of objects, we distinguish a type $Oid \in T$ of object identifiers. We assume an infinite set $Odom$ of object identifiers, Oids, and a disjoint set of values, $Vdom$. The object can be formally defined as a tuple:

$$O = [id : Oid; label : t]$$

Where id is a unique identifier for the object, with a label of type t identifying the type of the object. Each object is characterized by a set of properties, depending on its type. The properties of the objects PO are defined through the object identifier of the associated object, as follows:

$$PO = [id : Oid; label : t1; value : t2]$$

Where id represents the object identifier, $label$ describes the property name and $value$ is the value of the property. Analogously, we can define the relations as:

$$R = [id1 : Oid; id2 : Oid; label : t]$$

Where $id1$ and $id2$ are object identifiers that represent the origin and the end of the relation link,[3] and $label$ identifies the type of the relation. The properties of a relation PR are defined through the two ids that uniquely identify a relation:

$$PR = [id1 : Oid; id2 : Oid; labelRel : t; label : t1; value : t2]$$

Where $id1$, $id2$ and $labelRel$ together represent the relation instance, $label$ describes the property name and $value$ contains the value of the property, just as in the object definition.

[3] Note that in this formulation, relations are directed.

17.3.1 "sameAs" Interlinks

Many objects in one social network (i.e. context) will also "exist" in several other networks. This phenomenon is known as *Pirandello's identity problem* [9] in reference to the novel "One, no one and one hundred thousand", in which the protagonist discovers that everyone he knows has constructed an image of him in their imagination, and that none of these images corresponds to his own image of himself. This identity problem is inevitable: every social-network site exists for a specific purpose and is, thus, narrow in scope. For instance, a researcher may be described in ReaSoN in terms of her academic activities (e.g., her university address, and her co-authors) whereas she will be described in terms of her social activities in Facebook (e.g., her postings and her friends).

As it turns out, this problem is extremely hard to solve in practice. In order to correctly interlink the different communities, different social-network sites describing the same object would have to refer to it with a globally consistent identifier. In practice, however, each site has its own local identifier, unique only in its particular context. This practice results in a proliferation of identifiers, making it hard to merge social networks. SociQL supports a special-purpose relation, "sameAs", to connect object identifiers across contexts, as shown in Fig. 17.1 above. We return to this discussion later when presenting our current implementation.

17.4 The Query Language

In this section, we illustrate the syntax and relevance of SociQL by reviewing a set of sociologically relevant questions, and showing how they can be formulated in SociQL. Queries in SociQL are expressions of the form:

SELECT Oi.p1, . . . , Oj.pn
FROM Object O1, . . . Object Ok
WHERE Predicate P1 AND . . . AND Predicate Pm

The main construct of SociQL is the familiar *select-from-where*. The semantics of *SELECT* queries in SociQL are, essentially, the same as the well known class of relational query languages called Conjunctive Queries [1]. Given an instance of the data model, we find valuations satisfying the first-order formula in the *WHERE* clause, and add the tuples formed by projecting the object properties as indicated in the *SELECT* clause. The *SELECT* clause identifies the objects and their properties of interest to be returned by the query. The *FROM* clause specifies the types of the objects to be examined by the query, some of which will be part of the query result. Finally, the *WHERE* predicate describes a list of predicates relating the objects under examination. A predicate may be (a) a relational predicate, establishing an association between a pair of objects; (b) a selection predicate, filtering out objects based on (the values of) their properties; or (c) an interlinking "sameAs" predicate, pairing object identifiers from different networks that refer to the same real-world object. The Boolean conditions defining the predicates support eight comparison operators: $=$ (equals), $!=$ (not equals), $>$ (greater than), $\geq$ (greater or equal than), $<$ (less than), $\leq$(less or equal than), $><$(contains) and $<>$ (does not contain). The first six operators can be applied to properties corresponding to *ordinal* types (i.e., those for which a partial order can be defined); while only $=$, $!=$, $><$ and $<>$ apply to properties corresponding to *nominal* types (i.e., those for which no inherent order is defined).

Thus applying SociQL in the ReaSoN context, we can retrieve the names of the authors who have co-authored papers with a given author and the corresponding papers, as shown in the following example, which retrieves co-authors of researcher "Gregor Kiczales" and their papers which contain the keyword "Aspect" in the title.

```
Q1: SELECT r1.name, p1.title, p1.venue
    FROM   paper p1, author r1, author r2
    WHERE  writes(r1,p1)
           AND writes(r2,p1)
           AND r2.name='Gregor Kiczales'
           AND p1.title >< 'Aspect'
```

The query above essentially retrieves the *first-order zone* of "Gregor Kiczales", i.e., the ReaSoN authors who are directly related to "Gregor Kiczales" through the "writes" relation. The notion of *order zones* in SociQL refers to increasingly expanding regions of nodes, directly or transitively linked to a focal node. The region of nodes directly linked to a focal node, such as the co-authors of "Gregor Kiczales" above, is the first-order zone; the nodes two steps removed from a focal node, such as the region including the co-authors of his co-authors, constitute the second order zone, and so on.

17.4.1 Influence and Induction

SociQL supports the discovery of zones of influence between two actors, through any relations, with the *NEIGHBORHOOD* construct. This construct defines two participating objects and a maximum number of relationships, necessary to connect them. Consider for example the following query that retrieves all the organizations related to "John Smith", up to his fourth zone. By default, the query results also include the types of objects that serve as intermediaries to reach the target object.

```
Q2: SELECT o1.name
    FROM   author r1, organization o1, NEIGHBORHOOD(r1, o1, 4)
    WHERE  r1.name ='John Smith'
```

A conceptually similar construct is that of *PATH*. Path expressions are a syntactic convenience[4] helpful for finding connections and possible impact zones in social networks. Consider the query in which we want to get the organizations linked to John Smith directly or indirectly.

[4] A path query can always be translated to a series of basic SociQL queries.

```
Q2b: SELECT  PATH(r1,o1)
     FROM    author r1, organization o1, NEIGHBORHOOD(r1, o1, 4)
     WHERE   r1.name = 'John Smith'
```

In the query above, we evaluate the objects named $r1$ and $o1$ and the path expression $r1,m,A,n,B,p,C,q,o1$. This path can be interpreted as: start from 'author' $r1$, follow a relation 'm' that leads to object A, then a relation 'n' to B, then follow a relation 'p' that leads to object C, and finally a relation 'q' to organization $o1$. Furthermore, we limit the length of the path to a maximum of four, allowing shorter paths, like $r1,m,A,n,o1$, to form part of the answer as well.

While, in principle, there can be an infinite number of zones, we assume that the *influence* of each zone on an individual node declines exponentially. For most purposes, the number of effectively consequential zones is between two and three; that is, whatever is being studied, individuals or objects, past the third or at most fourth zone have relatively small effects on the focal individual or structure [17]. This phenomenon of decreasing influence between nodes as the distance between them increases has been well documented. For example, if a direct connection in one's social network is lonely, then this individual is 52 % more likely to be lonely. If one is only connected to a lonely person through an indirect relation (a friend of a friend), then his/her likelihood of being lonely falls to 25 %. At the third order zone, one is linked to a friend who is linked to a friend with a lonely friend, the probability for the original individual to be lonely is 15 % [11].

To support the expression of such decrease in influence, SociQL incorporates syntax to support the filtering of results based on the "importance" of the objects included in the original result set. This importance is measured in terms of a number of social-network centrality metrics, including indegree (the number of links incoming to the object), outdegree (the number of links outgoing from the object), closeness (the inverse of the sum of the object's distances to all other objects in the network), betweenness (the number of shortest paths from all vertices to all others that pass through that node), and pagerank [21], applied to a particular relation.

Consider, for example, a query to obtain the authors and papers, in which authors are affiliated with MIT, and have published more than one paper.

```
Q3: SELECT  writes(r1,p1)
    FROM    author r1, organization o1, paper p1, affiliated(r1,o1), writes(r1,p1)
    WHERE   o1.name='MIT'
    FILTER BY (OUTDEGREE OF r1 ON writes) > 1
```

SociQL supports yet another way to limit the set of results returned by a query. The *LIMIT* clause is used to limit query results to those that fall within a specified range. In other words, the clause can be used to show the first n result patterns. The following example demonstrates how to obtain the first instance of an author-publication relation, where the author's paper is affiliated with "MIT".

```
Q4: SELECT  writes(r1,p1)
    FROM    author r1, organization o1, paper p1, affiliated(r1,o1), writes(r1,p1)
    WHERE   o1.name='MIT'
    LIMIT   1
```

17.4.2 Interlinking Predicates

Let us now revisit our original motivation for interoperation of social-networking sites. An extensive experimental literature in social psychology established that homophily, i.e., similarities in terms of attitude, abilities, beliefs, and values, leads to attraction and interaction. It would make sense then to be able to study one's connections across social network platforms, since the union of these platform-specific sets of connections should be informative about the subject under examination. For example, it is highly probable that among a person's friends in Facebook, one can find individuals publishing on topics of interest to the person in question. In this way, listing the papers of a person's Facebook friends could return a list of papers of interest to the person in question. This query is shown in Q5.

```
Q5: SELECT  r1.name, p1.title, p1.year
    FROM    paper p1, author r1, user u1
    WHERE   writes(r1,p1)
            AND sameAs(r1,u1)
            AND p1.year=‘2009’
```

In this query, $r1$ is a researcher in ReaSoN, and $u1$ is the same person, as a Facebook user. The query essentially returns the papers of all one's Facebook friends that exist in ReaSoN. There is no need to explicitly refer to the Facebook friendship relation, because this query accesses the Facebook API using someone's credentials and therefore returns only the network of this person's friends.

17.4.3 Visualizing Query Results

The result of invoking a SociQL query is always a social network, i.e., a set of objects and relations among them. The SociQL service exports the result to a web-accessible user interface, which, by default, reports the results in a tabular form. There are situations where other means of communicating the information represented in the result might be more appropriate. To that end, the SociQL user interface also supports two alternative representations: a graph and a map-based representation.

The graph-based representation of the results makes more evident the relationships between the objects and the importance of the object, inferred based on the number of connections. Furthermore, this representation is interactive: by right-clicking on a graph node, the user can inspect its properties. To return a graph-based representation of the query results, the line containing the projected properties must be modified: the keyword SELECT must be replaced with EXPLORE. Figure 17.2 shows the graphical representation of the results returned by query Q1.

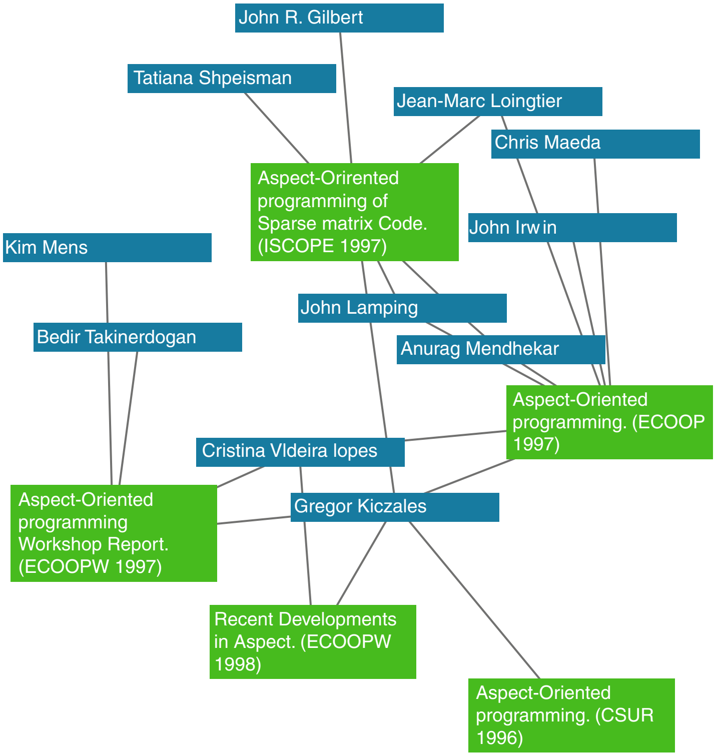
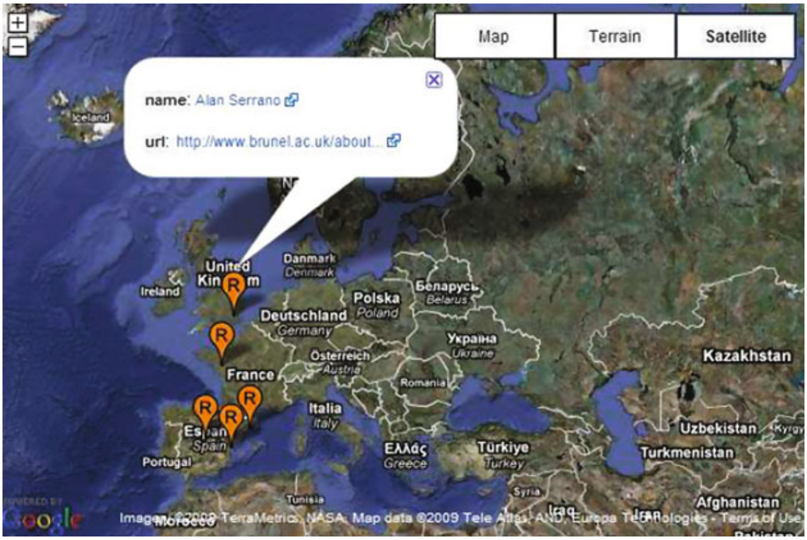

John R. Gilbert
Tatiana Shpeisman
Jean-Marc Loingtier
Chris Maeda
Aspect-Orirented programming of Sparse matrix Code. (ISCOPE 1997)
John Irw in
Kim Mens
John Lamping
Bedir Takinerdogan
Anurag Mendhekar
Aspect-Oriented programming. (ECOOP 1997)
Cristina VIdeira lopes
Aspect-Oriented programming Workshop Report. (ECOOPW 1997)
Gregor Kiczales
Recent Developments in Aspect. (ECOOPW 1998)
Aspect-Oriented programming. (CSUR 1996)

Fig. 17.2 Graphical representation of the Q1 explore query execution

Geographical proximity or collocation is perceived as a fundamental factor in sociological phenomena. According to the propinquity principle, at all levels of analysis, nodes are more likely to be connected with one another, other conditions being equal, if they are geographically near to one another [20]. This is why the SociQL user interface also includes a map-based result visualization, which provides the geographic location of the individual or object, if available. To build a map query, the line containing the projected properties must be modified. Basically, the keyword SELECT must be replaced with MAP, following the name of the object and the properties that the marker will contain, like in the following query.

```
Q6: MAP     r2:name, url
FROM        paper p1, author r1, author r2
WHERE       writes(r1,p1) AND writes(r2,p1)
            AND r1.name=‘John Smith’
```

The result of Q6 is shown in Fig. 17.3.



name: Alan Serrano

url: http://www.brunel.ac.uk/about...

Fig. 17.3 Graphical representation of the Q6 MAP query execution

17.5 Design and Implementation of the SociQL Service

The architecture of the SociQL data query, extraction, and interpretation is exposed in Fig. 17.4. Since the goal of social-network analysis is to help users to take advantage of implicit and explicit relations, it would be convenient to create levels of abstraction around the available information in order to rise our querying and inference capabilities. [16] introduce what they call “semantic social network”, a structure made of three superimposed networks that are assumed to be strongly linked:

- Data layer explicitly relating actors at the lowest level of abstraction;
- Concept layer relating concepts on the basis of implicit similarity and derived relations;
- Network layer relating networks on the basis of explicit conceptual relationships.

In SociQL, we have adapted the ideas of semantic social networks, and apply them to our query language. Figure 17.4 shows a graphic representation of the multi-layer architecture followed in the implementation of the SociQL service, however, due to space restrictions, only a portion of the concept layer is shown in the Figure. The thin slashed arrows represent the mapping between the data and concept layer, the thick slashed lines represent the derived relationships within the concept layer, and the dotted lines depict the mapping between the concept and network layer.

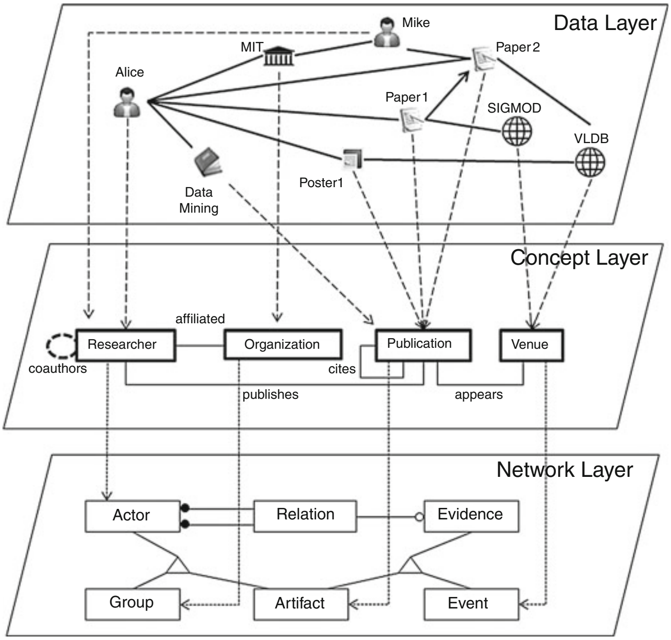


Fig. 17.4 The layered architecture of the SociQL service

17.5.1 The Visual Query Editor

In order to improve the user experience for the creation and execution of SociQL queries, we have created the Visual Query Editor (VQE). The VQE is a web-based application, developed using the Flex [13] framework. It provides a visual and interactive user interface that allow the user to create a visual representation of the actors involved in a query along with their properties and relations.

Inside the editor, a query is represented as a visual graph in which actors and properties are modeled as a set of nodes. The relations between the actors can be expressed as arcs connecting the related actors within the query. Selection, and relation conditions can be created using smart menus that detect the actors and properties involved. This feature helps the users to avoid errors in the creation and edition of SociQL queries. The VQE is capable of translating the graphical model that the user has expressed as a query, generating its respective SociQL expression, and invoking the necessary language services for its execution.

Figure 17.5 depicts the visual representation of query Q10, using the Visual Query Editor.

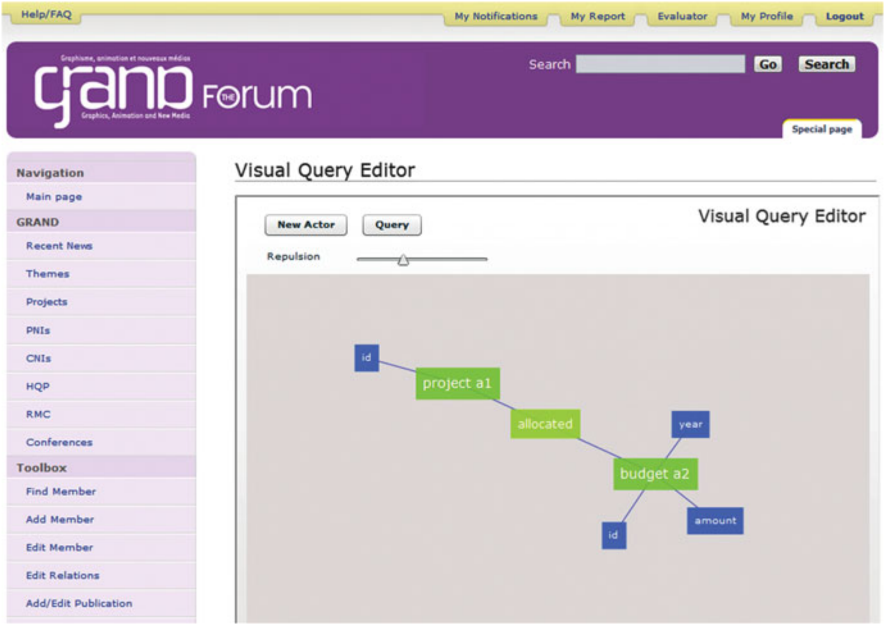


Fig. 17.5 Q10 visual query expression using VQE

17.5.2 Query Execution in the SociQL Service

The query-execution process in the SociQL service is diagrammatically shown in Fig. 17.6. Once a query is received, it is initially validated against the catalog of known social networks in the system. Then, an execution plan is developed for the query, specifying the order in which the statements have to be executed, and, especially, the order in which data from the external sources subscribed to the service should be requested. At the query-execution step, all external data relevant to the query is fetched and stored locally first; then the actual SociQL query is translated into an SQL expression that is executed on the local RDBMS containing all data needed by the query. The final step is to rank the objects in the answer to the query according to their importance. The remainder of this section reviews the technical challenges we had to address in each of these steps.

17.5.3 The System Catalog

Table 17.1 shows (part of) the catalog mapping the ReaSoN social network to the SociQL conceptual model. The catalog describes all the social networks subscribing to the service, i.e., all available *contexts*, and the mechanisms by which they can be accessed. More specifically, these are essentially queries that return

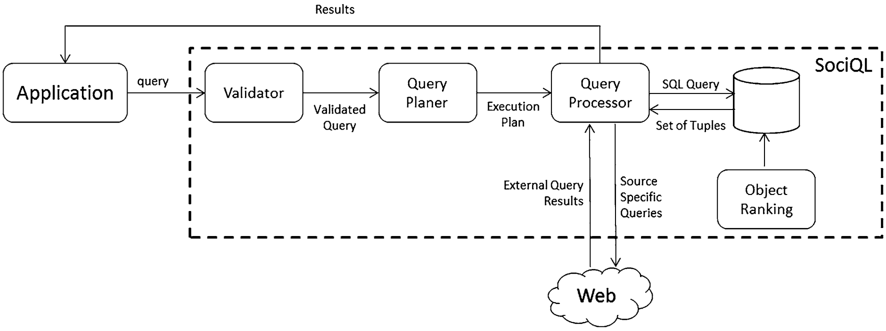


Fig. 17.6 Data representation with SociQL data model at ReaSoN

Table 17.1 Model specification for a research network

Type	Attribute	Value
Context	Name	Reason
	Endpoint	Local
	Type	SQL
Object	Name	Author
	Site	Reason
	Query	SELECT authorId FROM researcher
	ObjectId	researchId
Property	Name	Name
	Object/relation	Author
	Query	SELECT researcherId, name FROM researcher
	type	nominal
Object	Name	Paper
	Site	Reason
	Query	SELECT paperId FROM paper
	ObjectId	paperId
Property	Name	Title
	Object/relation	Paper
	Query	SELECT paperId, title FROM paper
	Type	Nominal
Relation	Name	Writes
	FromProperty	Author.Id
	ToProperty	Paper.Id
	Query	SELECT authorId AS id1, paperId AS id2 FROM writes
Relation	Name	CoAuthor
	FromProperty	Author.Id
	ToProperty	Author.Id
	Query	SELECT a1.id AS id1, a2.id AS id2 FROM author AS a1, author AS a2, paper, writes WHERE a1.id = writes.author id AND paper.id = writes.paper id AND a2.id = writes.author id

- For objects: unique ids within the network;
- For relations: unique pairs of object ids;
- For object properties: an of object id and a value; and
- For relationship properties: triples with two object ids and a value.

For social networks accessible through APIs, such as Facebook or DBpedia, the catalog contains the API calls that produce data in the same format. When a query is executed, if necessary, the SociQL service requests the user issuing the query to authenticate into the remote social network site (e.g., Facebook), thus ensuring access control and privacy settings are respected.

Once this mapping has been established, the SociQL service is able to materialize all the relevant data of all participating contexts locally, and convert SociQL expressions over equivalent SQL ones that identify all objects that belong in the answer to the query.

17.5.4 Visibility and Reputation

A central issue when examining a subset of a social network (such as the result of a SociQL query), is identifying and exposing the relative importance of the objects in that set. In the context of ReaSoN, for instance, when a user searches for researchers in the 'Bay Area', whose 'papers' mention the 'keyword' 'XML', she is most likely interested in knowing about the most influential (or productive) researchers, in the likely numerous set of researchers included in the result. This is in contrast with a search in the database sense where one is interested in every object the satisfies the criteria. Finding the relative importance of objects in social networks is a vast and mature field of research. Because so many "importance" metrics have been proposed to, each with its own nuances in interpretation, we designed SociQL to be orthogonal to the specific notion of reputation used. In our current implementation, however, we employ the established and well understood network "outdegree" and "visibility" [18] as an indicator of reputation. In passing, SociQL's notion of visibility is a generalization of Google's PageRank [21]. Details of how SociQL computes the visibility of objects can be found in [14].

Ranking of objects is done implicitly in SociQL. There are two aspects of the problem to be considered: obtaining the ranking of the objects and visualizing the objects in a way that exposes the ranking. When it comes to visualization, SociQL works as follows. For *SELECT* queries, the results are simply sorted by decreasing visibility, with more visible results appearing earlier in the table. For *MAP* queries, the markers on the map corresponding to the results are assigned different colors, based on a scale for visibility values. With the graph-based visualization of *EXPLORE* queries, however, relative importance is hard to convey because the visual characteristics of the nodes (and edges) in the graph already convey other information (such as the kind of object and relation).

17.5.5 Query Execution

We now discuss the actual execution of queries in SociQL (recall Fig. 17.6). Once the data has been materialized in the local SociQL database, the bulk of the work consists in translating the expression in SociQL into an equivalent executable SQL statement, based on the specifications of the networks in the system catalog. Our goal in doing so is to leverage the several decades of performance improvements on relational query processing engines. The main idea behind the translation is to treat each element of the data model defined in the catalog as a view; the final query is defined in terms of all such views.

We illustrate this translation through an example. Recall query Q1, which returns the co-authors of Gregor Kiczales together with their papers containing the keyword Aspect in their title. Notice that there are two properties of Paper objects that are needed to answer the query: title and year. This means that the final SQL query will have two table expressions in its FROM clause that are individual queries over the base table that stores all data about Paper objects: p1.title and p1.year. In order to make sure that every Paper object is faithfully reconstructed from the database, we define equality joins over object ids (recall the discussion about the system catalog) in every such view. In our example, this is captured by adding p1 title.id = p1 year.id to the WHERE clause of the resulting SQL query. The query relations translation is processed in a similar manner.

As an example, after Q1 is translated to SQL, we have:

```
SELECT r1 name.name, p1 title.title, p1 year.year AS pYear FROM
        (SELECT id, name FROM author) AS r1 name,
        (SELECT id, name FROM author WHERE
                  name=‘Gregor Kiczales’) AS r2 name,
        (SELECT id, title FROM paper WHERE title=‘Aspect%’) AS p1 title,
        (SELECT id, year FROM paper WHERE title=‘%Aspect%’) AS p1 year,
        (SELECT author id, paper id FROM writes) AS writes 1,
        (SELECT author id, paper id FROM writes) AS writes 2
                  WHERE p1 title.id = p1 year.id
                          AND r1 name.id = writes 1.author id
                          AND p1 title.id = writes 1.paper id
                          AND r2 name.id = writes 2.author id
                          AND p1 title.id = writes 2.paper id
```

17.5.6 The Query Planer

The SociQL query planner is concerned with identifying the ordering in which to request data from external social networking sites (which are used in interlinking relations). The goal is to find a strategy for answering the queries that minimizes the amount of external data that is retrieved. Some of the issues that need to be considered are the access control restrictions as well as limitations in the number of answers that API calls are allowed to obtain.

Our solution follows the rationale of [19]. Basically, the planner attempts to find the most cost-effective plan, by examining the costs of the various subqueries. Local queries, i.e., queries directly translatable to SQL and issued to a database on the same intranet are assumed to be the least expensive.

Since local queries, i.e., queries to systems whose repositories are directly accessible to the SociQL service, are most preferable, the planner makes a subquery with only the local actors, in order to get an initial subset of possible results and, based on it, to narrow the queries to the external resources. In case all the actors are external, the planner focuses first on the most constrained objects (those objects which have the highest number of selection criteria defined over them), in an attempt to increase as much as possible the selectivity of the queries to be submitted to external sites. For unconstrained *NEIGHBORHOOD* queries, which examine multiple types of relations, the planner performs a breadth first search on every undefined relation, up to a maximum number of levels defined in the query. Then, every path will create an independent query, that finally will be combined using *UNION* in the SQL query.

17.5.7 Linking Across Social Networks

One last challenge in executing SociQL queries is solving Pirandello's identity problem, that is, finding an effective way of determining when two objects from different networks are indeed the same object. Because each network defines its own internal identifiers for the objects it contains, we resort to linking objects based on equality of some of their properties.

As a crude example, the "sameAs" predicate used in query Q5, establishes that a researcher in ReaSoN and a user in Facebook with the exact same name are the same object. While this solution is clearly brittle, our implementation allows for more sophisticated ones. For instance, we could use duplicate detection algorithms [7] which can be defined over a combination of several attributes (as opposed to a single one, such as the name of a person). However, our ability of using such solutions depends on the kind of data that is available from the external site.

At the time of writing, we are able to successfully link objects from ReaSoN with their counterparts in Facebook and DBpedia. By doing so, in a sense, we are able to enrich the objects in ReaSoN with contextual and social metadata, in a process similar in principle to that of social tagging. A final observation about this issue is that despite the resulting cross-referencing of data, our approach does not fundamentally compromise the privacy of individuals as each context i.e., social networking siteretains its own privacy and access control restrictions. Furthermore, we fully respect data retention and caching restrictions.

17.6 The SociQL Service as a Means for Social-Network Interoperability

The majority of our work on designing the SociQL conceptual model and syntax and implementing the SociQL service was done in the context of our experimentation integrating ReaSoN, DBpedia and Facebook. To validate the usefulness of the language and the service as a means for supporting the interoperability and integrated study across social networks, a new member of our team, working independently, integrated the SociQL service to yet another social network. Our experience with this experiment demonstrates that the integration of SociQL with new contexts that involve different networks is indeed a straightforward and systematic process. Given that the language was designed in a decoupled manner from the networks where it can be deployed, the integration of SociQL with a new context can be done in two steps.

1. Map the data model of the new social network into SociQL's model.
2. Establish correspondences between the new social network objects and other objects from other social networks, already mapped in SociQL, to solve the Pirandello's identity problem.

The first step is concerned with the construction of the social objects, relations and properties of the new social network inside the SociQL system's catalog; the last one is centered on filtering the non-normalized information from the disparate networks through a mapping process.

We applied this process to integrate the GRAND Forum with SociQL. The GRAND Network of Centres of Excellence is a large-scale, national, interdisciplinary project, with academic and industrial partners, with a community of about 500 members, as of September 2011. GRAND includes two meta-level projects, conceived to support, reflect upon, and report on the activities of the GRAND community and its evolution through the lifecycle of the Network. To that end, we are developing the GRAND Forum, a set of tools built on the MediaWiki platform, designed to support the activities of the community and the administration processes of the network. The GRAND Forum contains information about the GRAND projects, its members, the artifacts they produce and their activities. As a general matter, the GRAND Forum contains three main sources of information. The first and biggest one is the database, where relational information about the projects, researchers and teams live in a non-normalized manner. Additionally, many of the network human actors (e.g. researchers and HQP) have Facebook and/or twitter accounts, consequently, there is valuable distributed information that together structure the researchers profiles, their team relations, home organizations, and projects. Thus, the GRAND Forum essentially consists of three SociQL *contexts*: the GRAND Forum database (accessible via SQL queries), Facebook (accessible via REST APIs using FQL), and Twitter (available through the twitter REST API). In this way, we could first define the different contexts where the information comes from, and then, the technical details about the consumption of the services needed for the information extraction.

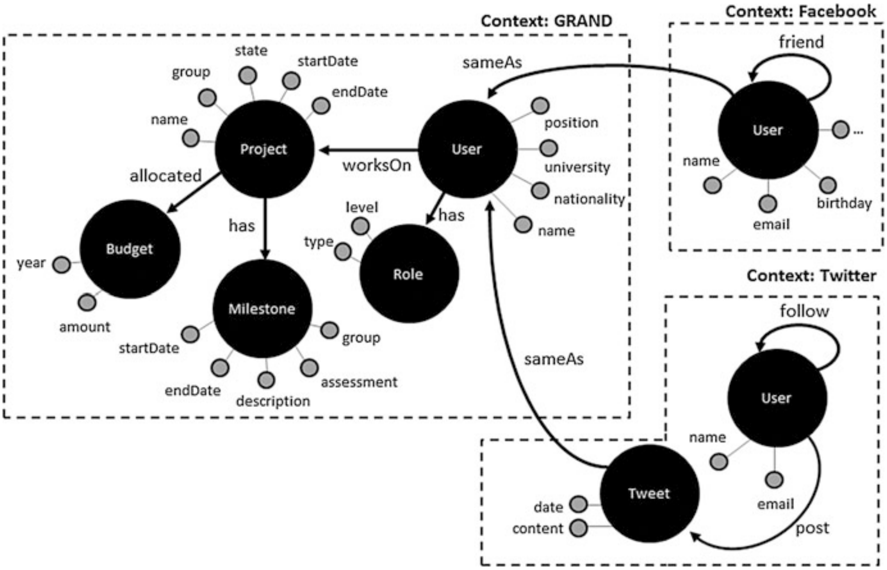


Fig. 17.7 A SociQL data catalog model for the GRAND research network

A partial view of the GRAND Forum catalog model produced after the execution of steps two and three is shown in Fig. 17.7. We have noticed that the redundant information (related with actors and relations) not only comes from the intersection of information between heterogeneous networks, but also from endogenous data integration. For example, because the GRAND Forum database is used to support multiple forums and project-management wikis, its data is structured in a non-normalized manner. Along these lines, we found actors and relations defined in multiple ways within the same context. In those cases, the solution strategy was based on picking the predominant actor (the one with more attached properties) and introduce in his set of properties mappings to his homologous actor properties.

As a result of the integration of SociQL in the GRAND Forum we were capable of generating several valuable queries for the GRAND network manager. In fact, several of the tables required by the NSERC reporting process were produced through SociQL. Below, four queries are presented exposing the utilization of SociQL within the GRAND Forum.

Query Q7 shows how users can follow the milestones of a project inside the network. It reports information about a specific project set of milestones along with their descriptions and current status (e.g. abandoned, completed, new, revised).

```
Q7: SELECT  a1.id, a1.name, a2.group, a2.id, a2.status, a2.title
    FROM    project a1, milestone a2
    WHERE   hasMilestone(a1,a2) AND a1.id=‘140’ AND a2.group=‘7’
```

Query Q8 reveals a view of the network project milestones that have the word *‘sensors’* as a part of their description. In order to highlight and look for the milestones related with an specific topic or resource, a user could introduce more complex keywords (e.g. complete sentences), or a set of them (e.g. ‘sensors’ and ‘3D’ and ‘camera’).

```
Q8: SELECT  m1.id, m1.group, m1.title, m1.assessment, m1.description,
            m1.endDate, m1.startDate, m1.status
    FROM    milestone m1
    WHERE   m1.description><‘sensors’
```

The GRAND network follows a fairly structured and hierarchical role organization for its researchers. From the management perspective, there is a particular interest to follow network-specific human assets in order to distribute them in particular research areas (e.g. full time professors, an area expert, or Ph.D. students). The query Q9 allows the filtering of projects where the highly-qualified personnel (HQP) work, and where the primary qualified human assets are concentrated.

```
Q9: SELECT  a1.username, a1.nationality, a1.position, a1.twitter,
            a1.birthday, a1.university, a2.type, p1.name
    FROM    user a1, role a2, project p1
    WHERE   hasRole(a1,a2)
            AND a2.type=‘HQP’
            AND worksOn(a1,p1)
            AND a1.gender=‘Female’
```

As a final example, query Q10 exposes the different budgets allocated in the network projects classified per year, project id, and/or amount.

```
Q10: SELECT  a1.id, a1.year, a1.amount, a2.name
     FROM    budget a1, project a2
     WHERE   allocated(a2,a1)
```

Together, the information that can be integrated using the presented domain specific language can be exploited as a set of reports for the decision making support. Throughout the GRAND Forum and ReaSoN case studies, we have verified in first place, the viability of the functional aspects attached to the mechanical characteristics of the language: expression, information retrieval and synthesis. And in second place, we have tested its interoperability and decoupling characteristics through the integration of the language with different networks to explore.

17.7 Conclusions and Future Work

In this paper, we discussed our work with SociQL, a query language and its service implementation for querying multiple social networks in an integrated manner, grounded on sound sociological theories. The original motivation of this work is to overcome the common limitations in query languages in dealing with social networks, namely the integration of path finding and the notion of importance in the language. We discussed how SociQL adopts a layer model to support abstractions and hide the complexity of the data. We outlined a method to rank the results according to the knowledge inferred by the topology of the social network. Additionally, we exposed an interoperability strategy in order to use the query language within multiple network contexts.

The litmus test for the validity and usefulness of any query language is whether or not it can be used to perform the common tasks in the application domain for which the language is designed. To achieve this, SociQL was designed from the ground up to help in answering typical queries in social network analysis. Its data model is a formalization of the socio-material network model (recall Sect. 17.3), and the language constructs (Sect. 17.4) all originated from typical social network analysis and/or theories. In order to make SociQL appealing to a larger audience, we chose the core syntactic constructs of the language to be the familiar *SELECT*, *FROM*, *WHERE* from the SQL query language. Furthermore, the language includes intuitive constructs to support the examination of zones of influence, *NEIGHBORHOOD* and *PATH*, the ordering of the results according to importance, *ORDER BY*, and to limit the results, *LIMIT* and *FILTER BY*.

As discussed in Sect. 17.5, SociQL is amenable to an efficient implementation allowing for several optimizations. In particular, our implementation supports three visualization strategies allowing the presentation of the results in terms of three familiar forms, tables, graphs and maps. Finally, our experience with the integration of the GRAND Forum with SociQL, carried out by a third-party developer and used to actually formulate queries defined by an external client, i.e., the GRAND Forum manager, provides persuasive evidence for the systematically of the implementation and the process of the service integration with social networks.

Our experience with SociQL to date is encouraging that our paradigm might be effective in helping social researchers (and other non-experts in database technology) search and explore large, inter-connected social networks. A full usability study to confirm this hypothesis is among the immediate future work for us. Moreover, three technical problems need further investigation. The first pertains to merging ranking scores across networks. There are two facets to this problem: when limited or no global rankings are available for one network, and when two networks use different metrics to rank objects. The second problem concerns the efficient computation of multiple visibility scores at query runtime; this would allow, e.g., computing scores on subsets of the network defined dynamically by the query. While most reputation metrics are expressible within the realms of relational query languages, the resulting performance is typically unacceptable for large graphs [15]. Finally, we plan to investigate more refined ways of defining inter-network links that could be used in SociQL.

References

1. S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, 1995.
2. R. G. Adams. *Placing Friendship in Context*. Cambridge University Press, 1999.
3. A. Anagnostopoulos, R. Kumar, and M. Mahdian. Influence and correlation in social networks. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 7–15, New York, NY, USA, 2008.
4. G. O. Arocena and A. O. Mendelzon. Weboql: Restructuring documents, databases, and webs. In *Proceedings of the Fourteenth International Conference on Data Engineering*, ICDE '98, pages 24–33, Washington, DC, USA, 1998.
5. S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, and Z. Ives. Dbpedia: A nucleus for a web of open data. In *6th International Semantic Web Conference*, pages 11–15, Busan, Korea, 2007.
6. M. Berger and R. M. MacIver. *Freedom and control in modern society*. Octagon Books, New York, 1964.
7. I. Bhattacharya and L. Getoor. Collective entity resolution in relational data. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):5–es, 2007.
8. U. Bojars and A. Passant. Sioc project: Semantically interlinked online communities. In *Proceedings of the Second Multi-Agent Logics, Languages, and Organisations Federated Workshops*, MALLOW '09, pages 7–10, Turin, Italy, 2009.
9. S. Bortoli, H. Stoermer, P. Bouquet, and H. Wache. Foaf-o-matic - solving the identity problem in the foaf network. In *Proceedings of the Fourth Italian Semantic Web Workshop*, SWAP 2007, pages 130–139, Bari, Italy, 2007.
10. K. K. Cetina. Sociality with objects: Social relations in postsocial knowledge societies. *Theory, Culture & Society*, 14(4):1–30, November 1997.
11. N. A. Christakis and J. H. Fowler. *Connected: The Surprising Power of Our Social Networks and How They Shape Our Lives*. Little, Brown and Company, September 2009.
12. Facebook. Fql facebook developers wiki. Retrieved March 5, 2010, from: http://wiki.developers.facebook.com/index.php/FQL, 2007.
13. A. Flex. Computer software. *Adobe Systems Incorporated*, 2009.
14. V. Ganev, Z. Guo, D. Serrano, B. Tansey, D. Barbosa, and E. Stroulia. An environment for building, exploring and querying academic social networks. In *Proceedings of the International Conference on Management of Emergent Digital EcoSystems*, MEDES '09, pages 282–289, Lyon, France, 2009.
15. H. Hwang, V. Hristidis, and Y. Papakonstantinou. Objectrank: a system for authority-based search on databases. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 796–798. ACM, 2006.
16. J. J. Jung and J. Euzenat. Towards semantic social networks. In *Proceedings of the 4th European conference on The Semantic Web*, ESWC '07, pages 267–280, Berlin, Heidelberg, 2007.
17. C. Kadushin. *Making Connections: An Introduction to Social Network Concepts, Theories and Findings*, chapter Some Basic Network Concepts and Propositions. Oxford University Press, 2011. In-press.
18. L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.
19. W.-S. Li, J. Shim, K. S. Candan, and Y. Hara. Webdb: A web query system and its modeling, language, and implementation. In *Proceedings of the Advances in Digital Libraries Conference*, ADL '98, page 216, Washington, DC, USA, 1998.
20. M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.
21. L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

22. J. Perez, M. Arenas, and C. Gutierrez. Semantics and complexity of sparql. In I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors, *The Semantic Web - ISWC 2006*, Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2006.
23. R. Ronen and O. Shmueli. Soql: A language for querying and creating data in social networks. In *Proceedings of the IEEE 25th International Conference on Data Engineering*, ICDE '09, pages 1595–1602, 2009.
24. A. Seaborne and E. Prud'hommeaux. SPARQL query language for RDF. *W3C recommendation (January 2008)*.
25. Yahoo. Query language guide. Retrieved 4 April, 2010, from http://developer.yahoo.com/yql/guide, 2008.

Index

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*,
Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5,