

# The Diffusion of Trust and Cooperation in Teams with Individuals' Variations on Baseline Trust

Yi Wang\*

IBM Almaden Research Center  
San Jose, CA, USA  
wangyi@us.ibm.com

David Redmiles

Department of Informatics  
University of California, Irvine, CA, USA  
redmiles@ics.uci.edu

## ABSTRACT

Baseline trust, which refers to the personality aspect of trust and varies with different individuals, is essential for understanding the development of trust and cooperation in a team. At the same time, informal, non-work-related conversations (aka, cheap talk) have positive influences on the diffusion of trust and cooperation in global software engineering (GSE) practice. This paper seeks to develop an understanding of the influences of individuals' baseline trust on the diffusion of trust and cooperation, in the presence of cheap talk over the Internet. We employ a novel approach, designing a virtual experiment that integrates abstract agent-based modeling and simulation with realistic, empirical network structures and baseline trust data from two large open source projects (Apache LUCENE and Google CHROMIUM OS). The results highlight the significant impact of baseline trust on the diffusion of trust and cooperation, for instance, the emergence of non-traditional diffusion trajectories. The results also demonstrate that proper seeding strategies can improve the effectiveness and efficiency of diffusion of trust and cooperation.

## Author Keywords

Cheap talk; agent-based modeling; baseline trust; trust and cooperation; global software engineering (GSE); diffusion.

## ACM Classification Keywords

H.5.3 [Information Interfaces and Presentation]: Group and Organization Interfaces - Organizational design; K.4.3 [Computers and Society]: Organizational Impacts - Computer-supported collaborative work

## INTRODUCTION

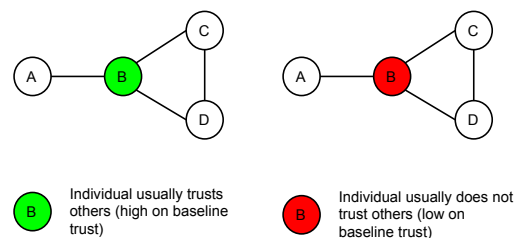
Trust, as an expectation that the other will behave cooperatively in dyadic social interactions, ensures the cooperation among individuals [2, 5, 29]. Trust and cooperation is crucial to the success of global software engineering (GSE) projects. Trust and cooperation can diffuse and be seeded through proper social and technical mechanisms in a networked GSE

\*The work was done while the author was at Department of Informatics in University of California, Irvine.

team context, for example, [3, 4]. As one of these mechanisms, informal, non-work-related conversations (aka, cheap talk) have been proven to have positive influences on the emergence and the diffusion of trust and cooperation through theoretical and empirical means [66]. However, these studies have two limitations. First, they often assume, explicitly or implicitly, individuals in a team are similar. Thus, they neglect the individuals' variations on "baseline trust". Second, most current studies on the diffusion of social behaviors typically consider social network structures generated from random or small world graphs rather than real-world interpersonal networks. Due to the lack of basis in reality, these results may not be relevant to or applicable in practice.

## The Importance of Individual Variations on Baseline Trust

"Baseline trust" refers to an individual's general, global tendency to perceive the trustworthiness of other individuals (or other entities, such as organizations) [16]. Baseline trust is evident in statements such as: "most people are trustworthy" or "most people are basically good and kind" [15, 69]. As the personality trait of "trust," its influences on the diffusion of *situational* trust from one team member to another cannot be neglected. Moreover, the influence of baseline trust on the development of trust and cooperation may be very significant when considering the social network structure of GSE teams. Although CSCW literature [25, 29, 72] has revealed the importance of baseline trust (or propensity to trust others), the analysis unit is almost always in individual level without considering connections among individuals, hence may not reflect the group level dynamics.



**Figure 1. An example of baseline trust's influences on the development of trust and cooperation in a simple 4-node network. The links indicate peer influence.**

In a networked team, an individual's baseline trust will not only influence his or her behavioral choices but also impact others in the same network. For example, figure 1 abstractly depicts a team in which the node "B" has a differing baseline

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

CSCW '16, February 27-March 02, 2016, San Francisco, CA, USA  
Copyright 2016 ACM. ISBN 978-1-4503-3592-8/16/02\$15.00  
DOI: <http://dx.doi.org/10.1145/2818048.2820064>

trust. Without considering baseline trust, there are no structural differences between the two networks, and one might draw the conclusion that the development of trust and cooperation exhibits a similar pattern. However, when considering baseline trust, the situation changes. For instance, the development of trust and cooperation may fail if B strongly prefers to be uncooperative. Hence, C and D may not switch to “cooperate.” Although baseline trust is important for developing *situational* trust and cooperation, it is often neglected to simplify the complexity, especially when considering it in a social network context.

Our prior interviews [2] confirm that global software engineering practitioners vary when it comes to baseline trust. A few interviewees tended to trust everyone without any reservation. One interviewee’s comment represents this attitude: “I trust everyone. Even if they do something wrong, I still believe people are usually trustworthy.” There are also many interviewees who prefer to “give others the benefit of doubt.” In other words, they trust until proven otherwise. By contrast, some individuals need their remote colleagues to prove they are trustworthy. He or she trusts others only when there is a reason. There are still a few people who believe they can never trust any of their collaborators, and may always prefer to be distrustful. Moreover, incorporating baseline trust also enables a new seeding strategy. With baseline trust, it is possible to identify the small fraction of agents with the lowest trust. Triggering the diffusion of trust and cooperation from them may be a good alternative.

### Empirical Networks vs. Artificially Generated Networks

Investigating the influences of individual variations on baseline trust requires an empirical network. Although we can arbitrarily assign any baseline trust to any node in an artificially generated network, there is no way to assess how much this method distorts reality. Without empirical data to constrain them, under diverse, unrealistic assumptions, the simulations may yield different results. Furthermore, due to the lack of basis in reality, these results may not be relevant to, or applicable in, practice. Using empirical networks provides a way to incorporate empirically observed distributions of trust propensities or correlations between individuals’ personality and social network characteristics.

An artificially generated network provides a great flexibility for investigating social interactions in a network. As already mentioned, though, this flexibility may not correctly reflect the reality. On the other hand, although the empirical study is more realistic than simulation, it usually fails to assess global characteristics and the effects of multiple strategies due to the needs to maintain experimental control and precision [63].

### Research Statement

Given the important role of baseline trust, we design a study to investigate its influences on the diffusion of trust and cooperation in networked GSE teams, with online cheap talk moderating the process. We adopt agent-based modeling and simulation, grounded by real world observations of empirical networks and individuals’ baseline trust. This approach has not yet been widely adopted but does provide several benefits

[4]. It leads to ready-to-use practical implications and avoids the troublesome mismatch between artificially generated and real-world networks. We also aim to examine whether two specific seeding strategies (seeding from the hubs, and seeding from the distrustful) would be helpful when considering baseline trust. In short, our research merges the flexible, but abstract, simulation-based approach with the more realistic, yet limited, empirical approach, aiming to answer following questions:

*With the presence of cheap talk over the Internet,*

**RQ1:** *how does baseline trust influence the diffusion process of trust and cooperation in the empirical network context?*

**RQ2:** *what are the different seeding strategies’ impacts on trust diffusion and cooperation in the empirical network context?*

The research distinguishes itself from the prior work in social network analysis and trust in following aspects. First, compared with the prior research on the social network analysis on organization process [9, 10, 33, 45, 58, 59, 12], we do not attempt to identify causality between network’s structural features and organizational outcomes. Our focus is identifying the dynamic patterns of the diffusion of trust and cooperation under specific network contexts without considering detailed topological characteristics (e.g., structural hole). Second, compared with the prior empirical work on trust such as Jarvenpaa et al. [29], we do not focus on the antecedence and consequence of individual’s trust. We aim to answer whether and how the trust and cooperation could or could not be developed in a networked team consisting of individuals of different baseline trust. Lastly, although we utilized empirical networks and empirically-derived individual characteristics as the infrastructures for the simulation, the research approach is essentially generative. In another word, the empirical data was used to provide the context of the study rather than evidence. Via the models and simulations, we deductively develop propositions characterizing real-world phenomena [14]. This approach comports well to current mainstream empirical inquiries ([29, 73]) on trust in globally distributed team.

The remainder of this paper is organized as follows. The next section presents some background. Next, we provide a high-level overview of the research process, followed by a section that discusses how we solve our major research tasks’ technical challenges. Then, we introduce the design of the virtual simulation experiment. Then, we present the results and findings. The remaining two sections respectively discuss related issues and conclusions.

## BACKGROUND

### Adapting Stag Hunt Game to Study Trust in GSE

The influences of informal, non-work-related conversations (aka, cheap talk<sup>1</sup>) on trust and cooperation development have been studied previously [11, 23, 24]. Recently, we [66, 67] proposed an approach that utilizes game theory (specifically,

<sup>1</sup>The cost of cheap talk in the economics literature is assumed to be zero. However, we allow the cost to be minimal in this paper.

stag hunt game) to investigate cheap talk and trust in the context of GSE practices. This approach provides opportunities for researchers to specify and simulate individuals' behavior to study the dynamics of the GSE team. The research described in this paper adapted this approach to develop a behavioral model, extending it to incorporate individuals' baseline trust. In this subsection, we will first briefly introduce the basics of using stag hunt game to describe trust and cheap talk in GSE.

#### Classic Stag Hunt Game

The classic stag hunt game is a non-zero-sum, two-player game in which each player has two strategic choices: *cooperate* or *defect* (see fig. 2). Stag hunt derives from an ancient hunting scenario. In ancient times, two men hunt for food. If both of them defect, they will hunt individually, and each would get a hare. If both cooperate, they could kill a stag, and each would receive one-half of a stag. If one cooperates but the other defects, the *cooperator* would receive nothing or very small payoff and the *defector* would receive a hare. Formally, the stag hunt game can be represented by the first payoff matrix in fig. 2 if  $R > T > P > S^2$ . For the numerical example, if both *cooperate*, each individual's payoff is 2 (total payoff = 4). If one *cooperate* while the other *defect*, the cooperator will only receive 0.5, and the defector will get 1.5 (total payoff = 2). If both *defect*, each will receive 1 (total payoff = 2). The state of (*cooperate*, *cooperate*) is a payoff-dominated equilibrium (best total payoff), while (*defect*, *defect*) is a risk-dominated equilibrium (no risk). Which one can be achieved is determined by players' belief in their opponent: namely, their trust [53, 54, 56].

	C	D		C	D
C	$R$	$S$	C	2	0.5
D	$T$	$P$	D	1.5	1

a. Stag hunt game ( $R > T > P > S$ )      b. A numerical example

Figure 2. The stag hunt game and a numerical example. The elements of matrix represent the payoffs of the "row" player.

#### Software Development and Stag Hunt

The stag hunt game is a natural metaphor of dyadic (one-to-one) collaborations<sup>3</sup> in software engineering activities. In many cases, developers do not necessarily need to cooperate with other team members to complete their jobs ("receive a *Hare* as payoff"), even when their work items are highly interdependent. However, low cooperation may influence their work's quality. Cataldo et al. [13] pointed out that communication among developers can significantly influence the quality of a software system, even if work items can be independently completed. Thus, collaboration can produce higher quality work ("receive a half *Stag* as payoff"). In some cases,

<sup>2</sup> $T$  may also equal to  $P$ , and  $S$  is not necessary to be 0 although it should be the smallest number.

<sup>3</sup>In the scope of this paper, cooperation is restricted to dyadic interactions to simplify the analysis and discussion. In fact, multi-person cooperation can be conceptualized as a series of dyadic ones.

a software engineer may believe that her colleague will cooperate, but things do not go as she expects. Thus, she may experience some "unfavorable" results (e.g., fail to deliver a commitment on time) due to the other's "defect," whereas the other can still achieve the utility of individual action ("receive a *Hare* as payoff"). Hence, dyadic collaboration in software development can be analogous to stag hunt, allowing us to use standard Evolutionary Game Theory (EGT) techniques to investigate software development collaborations.

#### Stag Hunt Game with Cheap Talk over the Internet

We have proposed an extension to classic stag hunt game that allows the existence of cheap talk over the Internet [66, 67]. Figure 3 depicts the new game's payoff structure. A new strategy "C-C" was added to describe the "cooperation" after "cheap talk." Different from face-to-face cheap talk, cheap talk in GSE projects usually happens over the Internet. Therefore, it is reasonable to assume this would lead to some small cost ( $e$ ) for cheap talk participants. Besides, since the communication records are publicly accessible, uncooperative behavior (after the other player shows some cooperative signal with cheap talk) may be punished ( $g$ ). The cooperative individual may receive compensation. We assume the compensation equals to the punishment ( $g$ ), so the top left cell changes to  $S - e + g$ . Therefore, if a person using cheap talk meets a cooperator, he or she will receive  $R - e$  (benefit from cooperation ( $R$ ) minus the cost of cheap talk ( $e$ )), and the cooperator will keep all benefit  $R$  from cooperation as payoff. If both use cheap talk, they will share the cost of cheap talk, so their payoffs are same ( $R - e/2$ ). If he or she meets a defector, he or she will receive  $S - e + g$ , and the defector will get  $T - g$  for  $g$  is the punishment. Above changes on payoff structure are highlighted in red in figure 3. The study adopted that payoff structure and the numerical model to specify the payoff of using different strategies.

	C-C	C	D		C-C	C	D
C-C	$R-e/2$	$R-e$	$S-e+g$	C-C	1.9	1.8	1.3
C	$R$	$R$	$S$	C	2	2	0.5
D	$T-g$	$T$	$P$	D	0.5	1.5	1

a. New stag hunt game with cheap talk.      b. A numerical example,  $e=0.2$ ,  $g=1$ .

Figure 3. The new stag hunt game with cheap talk and a numerical example. The elements of matrix represent the payoffs of the "row" player.

#### Investigating Socio-Technical Systems with Simulation in HCI & CSCW

The main research approach is agent-based modeling and simulation. Leveraging stag hunt game with cheap talk, the model expresses the strategic changes at both individual and team levels. By simulating an individual behavioral decision process in interactions, we capture a complex dynamic systems' behaviors and properties [46]. Agent-based modeling and simulation have been increasingly applied in HCI and CSCW to develop theoretical knowledge and practical design

implications [41, 47]. It also has the potential to link related social theories in different research streams together to investigate complex social phenomena [18]. The research realizes these benefits. Moreover, the research also advances this approach by integrating agent-based modeling and simulation with empirical data to develop rigorous and relevant studies.

## RESEARCH PROCEDURE

### Data Collection and Cleaning

The research utilized empirical data from two large open source projects: Apache LUCENE<sup>4</sup> (data from 04/2005 to 12/2014), and Google CHROMIUM OS<sup>5</sup> (data from 11/2009 to 04/2011). The reason we do not include all available Chromium OS data is that Chromium OS' IRC was "polluted" by the some end users' messages around the release of the first Chromebook in mid-2011. The main reason we chose them is that both are large enough to provide real globally distributed team setting, while the two projects still has many differences. The two projects are different in terms of size and application domain. Apache LUCENE (~900 KLoc) is an open source information retrieval framework. Google CHROMIUM OS (~6,000 KLoc including a Linux kernel) is a Linux-based open source operating system. They are also different from each other in several other aspects. First, LUCENE is a pure open source project, while CHROMIUM OS is driven by Google. Second, the contributions to LUCENE are all voluntary, while contributing to CHROMIUM OS is a part of some Google employees' routine job. Third, LUCENE is built from scratch, while CHROMIUM OS reused the Linux kernel that already has a huge code base. These differences enable us to ensure the results may be still applicable to a wide-range of large open source projects. However, they may not well represent small open source projects and global teams in a traditional organization.

### Collecting Communication Records

There are some off-the-shelf tools available for collecting online data such as mailing lists. However, these tools may have some drawbacks, such as obscuring email addresses for privacy reasons [7]. Moreover, most of them cannot support multi-data sources. We wrote a crawler in Python to download the public communication records from various sources. In total, we collected 83,627 HTML documents.

### Data Extraction and Cleaning

The original HTML documents were not ready for use. We leveraged Python BeautifulSoup to analyze the HTML files and extract the information we needed. During this step, we also excluded all auto-generated information, such as the commit and build messages that were automatically sent to mailing list subscribers. To ensure the reliability, we manually examined 100 crawled records for each type of data in each project (total: 700). Then, we compared them with the automatically extracted and cleaned data. Overall, the automatic process provided satisfying results (Precision: 98% 100%, Recall: 97% 100%, varies over different categories). We labeled every cleaned piece of information according to its category and stored it in a MongoDB database

<sup>4</sup><https://lucene.apache.org/>.

<sup>5</sup><https://www.chromium.org/chromium-os/>.

as a JSON Object. A JSON object records message content, authors, time, original URL, and any other necessary information (e.g., mentions in a message). In total, we have 121,539 JSON objects. Then, we built a simple index to associate users with the text content they produced. Using the data, we derived the two projects' empirical network structures and inferred their members' baseline trust through Natural Language Processing (NLP) techniques.

### Study Procedure and Main Task

The study design follows standard agent-based modeling and simulation procedures [37]. Before running the virtual experiment, we built the environment (the networks), inferred agents' baseline trust from their communication records, and specified the decision rules for them to interact with each other. Then, we performed the virtual experiment, analyzed the data collected from the experiment, and then summarized our results and findings.

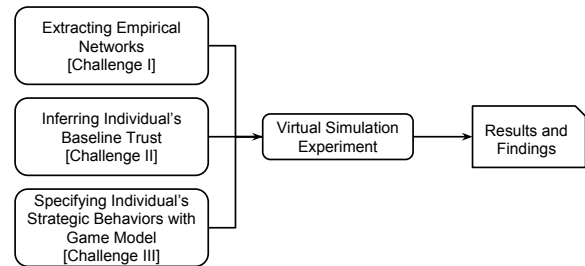


Figure 4. The process of performing the major research tasks.

## TECHNICAL CHALLENGES AND SOLUTIONS

There are some technical challenges associated with the major tasks (as we marked in figure 4). In this section, we introduce these challenges and corresponding solutions.

### Challenge I: Building Empirical Networks

Software engineers' social networks have been the focus of various studies [8, 12, 21, 26]. In this present work, we only build a social network for those who contributed source code to the repositories, but excluded peripheral contributors who may report bugs, but have not contributed code. Following [8], we only considered those who contributed code and have exchanged more than 150 messages (in all communication mediums such as mailing list, IRC, issue tracking, and discussions). We reasonably assume that people who have not exchanged a substantial number of messages over a few years can hardly have influenced their peers. This also ensures that we can precisely figure out people's baseline trust from the relatively large amount of records for every individual. The major challenge is how to re-establish the individual's identity, because individuals may use different names for different communication mediums, even in a single project.

For example, a developer named Tyrion Lannister may use the name "The Imp" in IRC chatting, while using an email address "tlan@casterly-rock.com" in the versioning system. It is almost impossible to automatically infer whether or not these two identifiers represent the same person for it requires some human judgment. In this example, a person who



watches the popular television show *Game of Thrones* likely understands the link between “The Imp” and “tlan@casterly-rock.com.” However, computers cannot automatically make such associations. Although many developers tend to use all or part of their email addresses as their usernames, a substantial proportion uses multiple names (LUCENE: 31 in 82, CHROME-OS: 35 in 129). The multiple identities must be resolved to a single identifier for purposes of assigning network nodes.

#### *Solution-Entity (Name) Resolution*

Although we can manually identify this study’s mapping, a manual method is not scalable for very large teams. Consider this simple heuristic: if an individual contributor uses the combination of `< username, email >` in an open source project, he or she might use this combination in other online occasions (e.g., community Q&A platform such as stackoverflow.com, a game community, etc). For a specific username, the corresponding email is probably the email address that has the highest co-occurrence with the name on the Internet.

We developed a method that leverages Google search to retrieve the number of search results for the different combinations of `< username, email >`. We wrote a script that automatically sends search requests to Google.com by manipulating the search [URL](#). Then, we analyzed the returned HTML to retrieve the number of search results can be retrieved from the corresponding HTML elements. For a given user name, we assume the combination has the highest number of search results is the right mapping.

Although it is quite simple, the method yields strong results. We performed a simple experiment using manually matched pairs of *name* and *email* as the ground truth. We first randomly selected 100 pairs of manually matched `< username, email >`. Then, we used the above method to identify the mappings over these pairs’ two sets of username and email. The results were encouraging, as it returned 96 pairs of mapping, 93 of which were correct (*precision* = 0.97). For this specific study, we also manually matched names and emails while experimenting with the above automatic method. If a developers’ social networks are very large, the automatic method is a good option for avoiding time-consuming and costly manual efforts. To further evaluate this method, we applied the six 1-attribute non-learning entity resolution algorithms (e.g., COSY) [35], the precision of these algorithms are between 0.42 and 0.79. We also tried a commercial entity resolution product: Alchemy’s entity disambiguation API<sup>6</sup>. It yielded 0.71 precision and 0.65 recall. Our simple algorithm outperforms all of them on this specific username matching problem.

#### *Solution-Network Building*

The entity (name) resolution enabled us to build mappings between unique individuals and their communication records. Using these records, we built the network among developers. As we pointed out before, the network only contains those who were code contributors and had a substantial amount of

communication records. This excluded those ad hoc contributors. The diffusion of trust and cooperation in a network relies on interpersonal interactions within the network. When defining the links, we need to make sure the interactions represented by a link are meaningful and substantial. Moreover, since the data comes from hybrid sources, we need to define the clear standard of “interaction” before defining “connection/link” between individuals.

We followed the guidelines of extracting networks from hybrid data sources to define *interaction* and then use it to identify *links* [28]. In this study, an “interaction” could be one of the following four types of communications: (1) a send-reply pair in mailing list; (2) an explicit mention in IRC discussion; (3) both commenting on an issue; (4) both discussing in the same discussion thread. Based on this definition, two individuals is considered to be connected if the following two conditions hold. First, the interaction between them is greater than once per month on average during their shared tenure. Second, the total number of all type of interaction is no less than 30 during their shared tenure. This ensures the exclusion of casual relationships among developers. The links are unidirectional, so the network is undirected. The network is static, reflects the status quo in the last day of the available data. Although using the static network is a common practice, the main limitation is that it cannot reflect the dynamic evolution of the network structure. Considering dynamic network structure will enhance the study in future.

#### **Challenge II: Extracting Individuals’ Baseline Trust**

Conventionally, standard questionnaire surveys based on mature psychometric models are the typical method used to identify an individual’s trust [15]. However, it is very difficult to ensure open source project members’ participation, especially considering that most surveys’ response rates fall below 20%. It is highly likely that we would be unable to assess the baseline trust of the majority of developers’ social networks, which would inevitably lead to highly distorted results. Moreover, a questionnaire-based survey is difficult to automate, meaning it would not likely be used to develop automated decision support tools for GSE practitioners, which would thus limit this study’s potential practical value.

#### *Solution*

For each individual, we collected his or her communication records from the IRC channel, mailing list, and issue tracking system. Then we organized the communication records by month. Using NLP methods [31, 32], we calculated the trust score for each month. For each month, we required a substantial number of total messages to ensure the reliability of the trust score. Otherwise, we simply assigned a “0” to this month. The two-tuple `< month, trust >` form a time series. We performed a de-trending transformation on the time series using `pracma` package in R [44]. The detrending step is necessary because trust might exhibit an increasing trend that results from continuous interactions with other team members. Obviously, the increasing trend is irrelevant to “baseline” trust that is a stable personality trait.

To ensure the reliability of inferring trust through text records, we used two linguistic lexicons: LIWC (Linguistic Inquiry

<sup>6</sup><http://www.alchemyapi.com/api/entity/disambiguation>.

and Word Count: LIWC 2007 [42, 62]) and NRC Emotion Lexicon [39]. Each contains multiple dimensions for a single word; we only used the dimensions related to “trust.” To optimize the results’ reliability, we took two measures. First, we compared the level of precision using *unigram*, *bigram*, and *trigram*. Using *unigram* is problematic. For instance, in the sentence “*I do not believe his commitment*,” if one only uses *unigram*, we would miss the negative descriptor “*not*,” and incorrectly label the statement as an indicator of *high* trust. We experimented with different combinations and found that *unigram+bigram* almost always yielded the best results. This is consistent with the prior research such as [36]. Second, we computed trust with the LIWC and NRC Emotion Lexicon, compared the results, and found that they are quite consistent. We computed the correlation of two trust score sequences for each individual, and found most of the pairs were significantly correlated. We also compared their normalized means and confirmed no significant differences. Hence, we used the average trust of both lexicons as the final value.

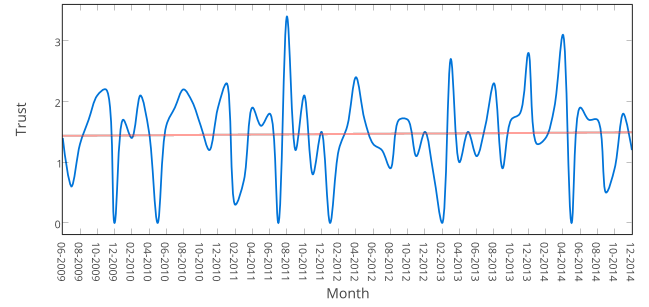
Figure 5 describes the dynamics of a developer’s de-trended trust inferred from their word use from 06/2009–12/2014. Although trust changes over the time, it fluctuates either way about the average (the horizontal line in figure 5). The average of trust hence approximates baseline trust.

To further validate the measurements, we randomly picked five individuals from each project (total = 10). We manually coded their trust through reading their text generated in each month. Since we could not assign numerical scores to it, we categorized every individual’s monthly trust into five categories: {Very High, High, Medium, Low, Very Low}. Each category was mapped to a number from 2 (Very High) to -2 (Very Low). Hence, we developed a manually coded time series for every individual. Then, we pulled out the corresponding ten time series generated with NLP-methods. Thus, we had a pair of manually coded time series and NLP-generated time series for every sampled individual. We tested the similarity of each pair using cointegration [17]. The results show that eight out ten pairs of time series are cointegrated ( $p < 0.05$ ), and another one pair is marginal cointegrated ( $p < 0.1$ ). Only one pair of time series is not cointegrated. Since cointegration indicates two time series are mutual predictable, most of the pairs are cointegrated enhance the confidence towards the NLP-generated trust score.

We normalized the individual trust to the interval  $[-1, 1]$  using the sequential combination of *z-score* normalization and *feature scaling*. These transformations are mainly for the convenience of specifying how the baseline trust influences individual behaviors (see the next section for details). Table 1 shows the basic statistics of baseline trust in LUCENE and CHROMIUM OS.

Project	Sample Size	Median	St. Dev.
LUCENE	81	0.329	0.380
CHROMIUM OS	126	0.261	0.517

**Table 1. The basic statistics of baseline trust.** For some individuals, we cannot resolve their names even using manual mapping, so the sample size is slightly smaller than the total number of individuals (see above subsection “Challenge I.”)



**Figure 5. The dynamics of a developer’s de-trended trust inferred from their word use from 06/2009–12/2014. The line indicates the average trust over this period.**

The approach of inferring baseline trust is still only an approximation of people’s true baseline trust. In fact, even the standard trust measurements’ validity is not fully guaranteed because various factors may influence it various factors, including respondent’s interpretation, survey execution, and so on [60]. Explicit and implicit self-reporting biases may further undermine the survey’s validity because people may prefer to show that they trust rather than distrust others. Literature [52, 62] has established the acceptable level reliability of analyzing an individual’s communication records to infer his or her personality traits. The approach is also supported by psycholinguistic models, for example, [62] and [52]. However, it is worthwhile to explore and evaluate the approach when deploying it in other domains and scenarios.

### Challenge III: Specifying Individuals’ Decision Dynamics

Since we have extracted an individual’s baseline trust, we face another challenge; that is, how can we specify how an individual’s baseline trust influences their strategic behaviors? We assume an agent’s decision-making is probabilistic rather than deterministic. A specific strategy’s resulting higher payoff does not guarantee the agent will switch to it. Therefore, we need to figure out how to properly reflect baseline trust’s influence in decision models. Should baseline trust directly influence the probability of an individual’s strategy selection, or only their subjective judgment of utility (and then indirectly their behavior)? And what is this influence’s extent? All these concerns should be properly addressed.

#### Solution

Obviously, one can define a mechanism that allows baseline trust (as a belief of the world, e.g., whether people are usually trustworthy) to alter the probability of behaviors directly [20]. However, arbitrarily defining such a mechanism is risky, because different mechanisms may yield quite different dynamics, and there is no easy way to evaluate the results’ sensitivity. Performing sensitive analysis over a series of functions is very difficult.

We took a more conservative approach in this study. We applied the Belief-Preference-Constraints (BPC) model [22] to treat “baseline trust” as a type of constraint that influences an individual’s subjective evaluation of his or her payoff. In the payoff structure, baseline trust’s influence will be expressed

by an idiosyncratic payoff. We assume that the utility functions in equation 1 to satisfy von Neumann-Morgenstern's utility theorem [65], which allows us to avoid changing the decision dynamics or arbitrarily altering the probability of a specific strategy, which may lead to unfavorable noise in global level dynamics.

Let  $S$  be the set of all possible strategies, and  $s \in S$  be a specific strategy. In this study, there are three possible strategies:  $\{\text{cooperate (C)}, \text{cheap talk-cooperate (C-C)}, \text{or defect (D)}\}$ .  $U_i(s)$  denotes the overall value an individual  $i$  received by using strategy  $s$ . According to the above discussions,  $U_i(s)$  is determined by two parts as follows:

$$U_i(s) = \begin{cases} P_{\text{interaction}}(s, -s) + P_{\text{trust}} & \text{if } s = C \\ P_{\text{interaction}}(s, -s) & \text{if } s = C - C \\ P_{\text{interaction}}(s, -s) - P_{\text{trust}} & \text{if } s = D \end{cases} \quad (1)$$

In equation 1,  $P_{\text{interaction}}(s, -s)$  refers to the (*expected*) payoff received from interacting with one's neighbors. In this study, we only consider the direct influence, i.e., two directly connected individuals, A and B. Let's suppose he or she has  $M$  neighbors, and  $(m_C, m_{C-C}, m_D)$  denotes the numbers of his or her neighbors who choose three possible strategies at period  $t - 1$ .  $P_{\text{interaction}}(s, -s)$  can be written in the following form:

$$P_{\text{interaction}}(s, -s) = \sum \frac{m_{-s}}{M} \times p(s, -s) \quad (2)$$

In equation 1,  $P_{\text{trust}}$  refers to the idiosyncratic payoff resulting from different baseline trust levels. We simply use a linear function to describe it:

$$P_{\text{trust}} = c \times \text{baseline trust}. \quad (3)$$

The constant  $c$  is in the range of  $[0, 1]$  where "0" means baseline trust has no influence, and "1" indicates baseline trust has full influence. This structure has been used in literature such as [55] to address individual subjective bias or preferences' influence on payoff evaluation. We can simply parameterize the constant  $c$  to examine the results' sensitivity.

Now let's take a closer look at equation 1. As we mentioned before, baseline trust ranges from  $[-1, 1]$ . If an individuals' baseline trust is positive, they receive an extra idiosyncratic payoff from using "cooperate" strategy. This is intuitive because the "cooperate" strategy fits their personality (they tend to trust others). If they select "defect," the overall value will be less than the value they could get from interacting with their neighbors, because they may feel unhappy for selecting a strategy that does not fit their personality. Individuals who have a negative baseline trust tend to distrust others and work independently. Therefore, the overall value of using "defect" will increase, whereas they may feel uneasy using the "cooperate" strategy. For "C-C," we assume neither population has a special preference for it. Therefore, their payoffs are solely determined by the interactions.

Since baseline trust's influence is only reflected by payoff changes, we use the logistic learning rule to specify the probability of switching strategies [64], parameter  $\beta$  represents the sensitivity towards payoff change: the larger  $\beta$  is, the more likely it is that he or she will choose the best reply, given the

actions of his or her neighbors.

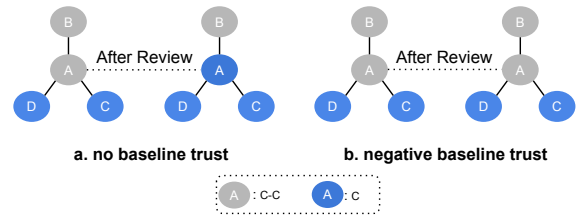
$$\frac{e^{\beta U_i(s)}}{\sum_S e^{\beta U_i(s)}}, \beta \geq 0 \quad (4)$$

Figure 6 shows an example of an individual's decision-making process. In the original example (figure 6.a), A changes her strategy from "C-C" to "C," since the latter results in a better payoff (1 vs 0.916). But in figure 6.b, if we assume the baseline trust is -0.2 and  $c = 0.5$ , the expected payoff of playing "C" becomes  $1 - 0.5 \times -0.2 = 0.9$ . However, the expected payoff of playing "C-C" is still 0.916 according to equation 1. Obviously,  $0.9 < 0.916$ , so the probabilities of using "C" and "C-C" are specified in equation 5:

$$P(C) = \frac{e^{0.9\beta}}{e^{0.9\beta} + e^{0.916\beta}} < \frac{1}{2} \quad (5)$$

$$P(C - C) = \frac{e^{0.916\beta}}{e^{0.9\beta} + e^{0.916\beta}} > \frac{1}{2}$$

If  $\beta \rightarrow +\infty$ , the learning process is deterministic, and A will definitely keep using C-C.



**Figure 6. An comparison of applying the decision model to make decision ( $\beta \rightarrow +\infty$ ). In this example, A changes her strategy after the review process.**

A minor challenge is to determine who will be the next individual to review their strategy. We implemented simple social learning in a network. That is: (1) if an individual is selected to review her strategy and changes it in period  $t$ , the next selected individual should be one of her neighbors; (2) otherwise, randomly pick one from all individuals. We also allow individuals to make mistakes with a small probability (5%) to reflect their bounded rationality [30]. It also ensures the process will eventually reach stable absorbing states [70].

### Summary

After solving the above challenges, we built infrastructures (Networks, Individuals' Baseline Trust, and decision rules) for the virtual simulation experiment. Figure 7 visualizes the LUCENE's network with individual's baseline trust depicted in different levels of grayscale. The CHROMIUM OS' network is similar. In this study, we only considered the largest connected component<sup>7</sup> of the network, and removed those individuals who do not belong to it. Please note that, although the average of baseline trust in each group is exactly "0" due to the  $z$ -score normalization, both groups have more members with positive baseline trust than those with negative baseline trust (see table 1, both medians are positive).

<sup>7</sup>Informally, it refers the largest set of nodes and edges in which there is a path formed by edges between every pair of nodes.



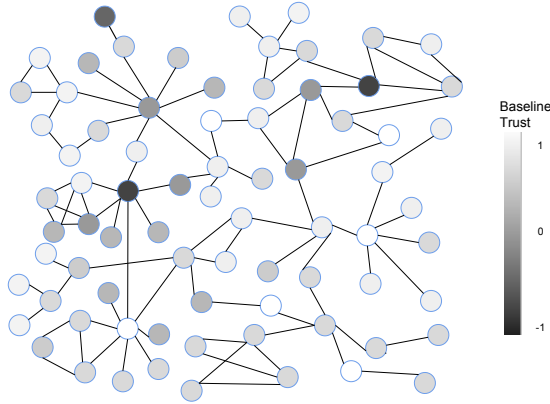


Figure 7. The developer network of LUCENE (largest connected component only). The gray-scale indicates each individual's baseline trust.

### VIRTUAL EXPERIMENT DESIGN

The virtual experiment contains two parts, which correspond to this study's two foci (baseline trust and seeding strategy) and two research questions.

Obviously, if  $c = 0$ , the baseline trust has no influence. Therefore, it serves as a benchmark for assessing the influence of different degrees of baseline trust. We sequentially manipulated baseline trust's influence ( $c$ ) with the interval of 0.1 from 0.1 to 1. Thereby, we have 10 conditions:  $c = 0.1, c = 0.2, \dots, c = 1$ . For each condition, we perform 1,000 independent trials for each of LUCENE and CHROMIUM OS networks. In total, we have  $10 \times 2 \times 1,000 = 20,000$  independent simulation trials. At the beginning of each trial, 10% individuals are randomly selected as seeds who use the "cooperate" strategy. To reduce the complexity, we set all individual's learning factor  $\beta = 10$  for all simulations<sup>8</sup>. Sensitivity analyses ensure the robustness of the results for a relatively broad range of  $\beta$  ( $5 \leq \beta \leq 50$ ).

We tested two seeding strategies: **seeding from the hubs**, and **seeding from the distrustful**. As opposed to the first part, seeds are not randomly assigned. We rank all individuals according to their hub score<sup>9</sup> and their baseline trust. Then, for seeding from the hubs, we choose 10% of individuals with the highest hub score and begin simulation with them. For seeding from the distrustful, we choose 10% individuals who are lowest in baseline trust, and begin simulation with them. All other settings are kept intact. As in the first part, for each seeding strategy in each baseline trust condition ( $c$ ), we perform 1,000 independent trials for each network. We reuse a discrete event simulator to manage the simulation process. We keep detailed records of every simulation trial's state in every period.

## RESULTS AND FINDINGS

### Overview of Results

In this section, we present the results of the virtual simulation experiments. The analyses yield six propositions, which

<sup>8</sup>The distribution of learning factors may correlate with the distribution of baseline trust, future research may need to address this point.

<sup>9</sup>The hub score was calculated with the method introduced in Manning et al.[38], we made slight changes.

provide answers to the two main research questions. Table 2 summarizes the correspondence between the research questions and these propositions. The results are aggregated from all 1,000 trials in a given experiment condition.

### Diffusion Trajectories

We then examined the influence of coefficient  $c$ , which determines baseline trust's different degrees of influence. If  $c = 0$ , baseline trust has no influence (see equation 1). With the increase of  $c$ , the influence of baseline trust becomes more significant.

#### Diverse Trajectories of Diffusion

Figure 8 (a-f) shows the six trajectories of full diffusion of trust and cooperation in LUCENE's network. These six trajectories were also observed in CHROMIUM OS. There are no significant differences except the total periods in the diffusion. To keep the conciseness, we will not plot them again. The diffusion processes are generally quicker in LUCENE's network since it is smaller. We cannot rule out the possibility that different network typologies and baseline trust distributions also contribute to the difference in the speed of diffusion. However, they are beyond the scope of this paper.

First of all, cheap talk is still important. Almost all diffusion curves start with a relatively flat part (see figure 8). During these periods, agents are most likely to switch to cheap talk first. It is natural to be safe after realizing your neighbors have not been cooperative. Then, the trajectories become fairly diverse and non-classic with the increase of baseline trust's influence. When  $c = 0$ , the majority of diffusions exhibit the classic S-curve [1]. However, more diffusion trajectories appear with the increase of  $c$ , which indicates that baseline trust diversifies trust and cooperation's diffusion. It is reasonable for introducing the influence of baseline trust to make the payoff structure personalized and no longer static (see equation 1 for reference).

Perhaps the most surprising trajectory is that which exhibits a "staged" pattern (figure 8.f). We examined the detailed process behind these patterns and noticed a few highly "distrustful" individuals cause the "platforms" in the diffusion trajectories. The process is stuck and only moves forward after they "mistakenly" change their behavior; it becomes exhaustively long. There are not many processes that demonstrate this trajectory; however, its frequency becomes non-trivial when  $c$  is large ( $c \rightarrow 1$ ). Compared with other patterns, this pattern is less investigated. Even in literature that documents several non-classic diffusion trajectories [49, 50], the "staged" trajectory is not covered. For the empirical network of LUCENE, all trajectories are observed when  $c \geq 0.5$ . For the empirical network of CHROMIUM OS, all trajectories were observed when  $c \geq 0.4$ . This suggests that the critical value for the diffusion process expresses that all trajectories may depend on the profiles of baseline trust distribution and the network.

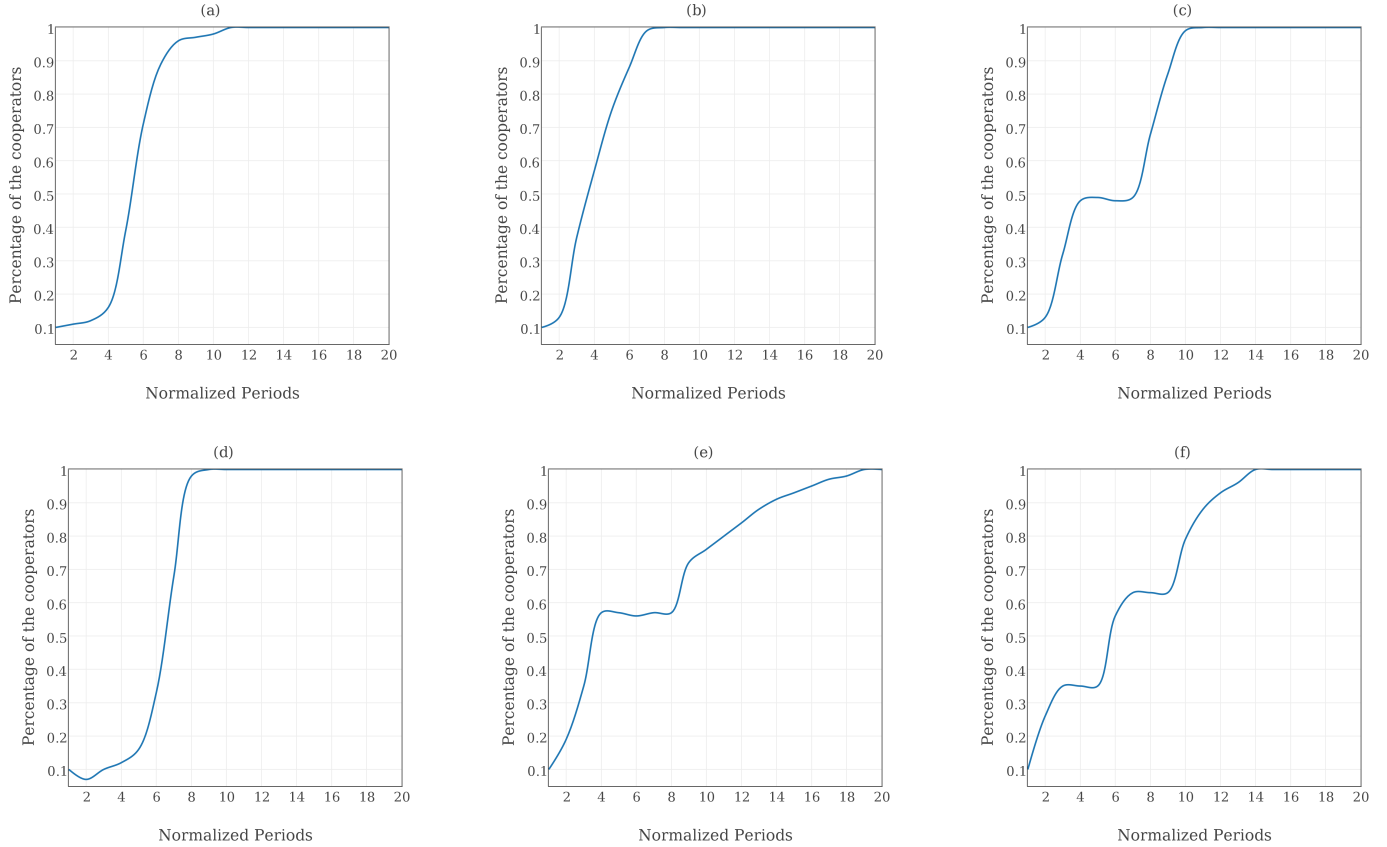
#### The Effectiveness of Diffusion

Figure 9 shows the frequency of the individual simulation process reaches a stable homogeneous trust state (full diffusion) under different  $c$ . Both LUCENE and CHROMIUM OS show similar patterns. An apparent pattern is that there are



RQs	Proposition	Key Point
<b>RQ1</b>	PROPOSITION I, II, & III	When considering baseline trust, the simulation results show: (I) <i>C-C</i> is still important at the beginning of diffusion and possible to be a long term stable strategy, and more diverse diffusion trajectories appear in later phases, (II) diffusion is more limited when baseline trust's influence becomes substantial, and cheap talk becomes a stable strategy in the long run, (III) the average speed of diffusion improves, while it varies more significantly.
<b>RQ2</b>	PROPOSITION IV, V & VI	(IV & V) Both seeding strategies (seeding from the hubs and from the distrustful) positively influence the effectiveness and speed of diffusion. (VI) Using them together may yield even better results.

Table 2. Summary of findings and corresponding research questions.



**Figure 8.** Different possible full diffusion trajectories on LUCENE network ( $c = 0.6$ ). We pick the case of ( $c = 0.6$ ) for it is large enough to allow the 1000 simulated diffusion processes exhibit all six trajectories. Each curve is a typical representative simulation for a subset of 1000 simulations which share the similar trajectory.

fewer processes reaching full diffusion with the increase of baseline trust's influence. These processes often end with hybrid states in which two or three strategies still exist. In  $c = 0$  situation, the majority of simulations achieve full diffusion. However, when  $c = 1$ , almost 29% simulations reach limited diffusion for LUCENE network, and around 37% for CHROMIUM OS. The decrease of the diffusion's effectiveness may be non-linear (see figure 9). The full diffusion rate drops faster when  $c$  is between 0.4 and 0.6, which indicates that there may be some qualitative change when  $c$  is in this interval. Among those limited diffusion processes, we observed

the existence of *Cheap talk-Cooperate (C-C)* as a long run stable state. That is because switching to cooperator becomes less attractive for some individuals when the extra payoff is offset by their idiosyncratic payoff related to baseline trust.

#### The Speed of Diffusion

Figure 10 shows the average number of periods (normalized) to reach full diffusion<sup>10</sup> in each condition. In figure 10, the average speed of diffusion is faster with the influence of baseline trust, although the improvements are not very significant.

<sup>10</sup>We ignored all limited diffusion simulations.

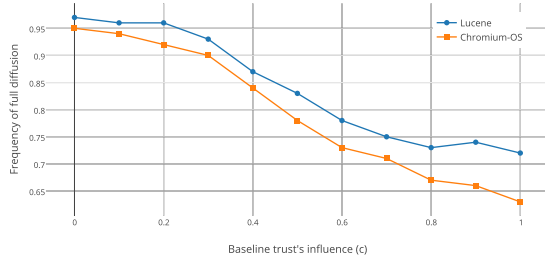


Figure 9. The change of frequency of full diffusion under different  $c$ .

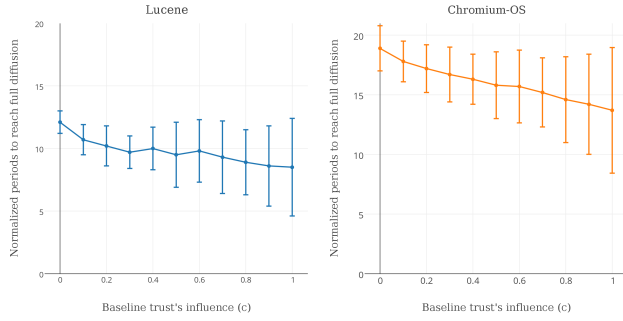


Figure 10. The change of the number of normalized periods to reach full diffusion under different  $c$ .

This may result from the fact that the majority of individuals have positive baseline trust in two projects. However, the speed of diffusion varies much more significantly. We performed a simple ANOVA test on the sample of full diffusion in three conditions ( $c = 0$ ,  $c = 0.5$ , and  $c = 1$ ), and the results suggest there are significant differences in the variances.

#### Summary of Findings

The main findings can be summarized as the following three PROPOSITIONS:

**PROPOSITION I:** *C-C is important at the diffusion process' outset. The diffusion of trust and cooperation exhibits non-standard trajectories when baseline trust has substantial influence on an individual's subjective payoff evaluation.*

**PROPOSITION II:** *The probability of a limited diffusion of trust and cooperation becomes greater when baseline trust substantially influences an individual's subjective payoff evaluation. Also, strategy C-C may become a stable strategy in the long run.*

**PROPOSITION III:** *Suppose the baseline trust has substantial influence, then the average speed of diffusion improves if the majority of individuals have positive baseline trust; however, the speed of diffusion varies more significantly.*

Baseline trust matters! PROPOSITION I, II, & III not only re-confirm the importance of cheap talk but also illustrate how baseline trust shapes the diffusion of trust and cooperation. We performed sensitivity analysis on payoff structures. Under the condition that the payoff from interaction is at the

same level of baseline trust, the results are robust enough when punishment/compensation is comparable to cost.

#### Seeding Strategies

##### Seeding from The Hubs

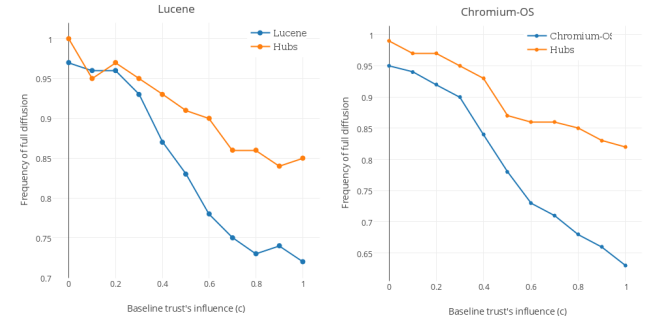


Figure 11. Comparisons of between random seeding and seeding from the hubs (frequency of full diffusion).

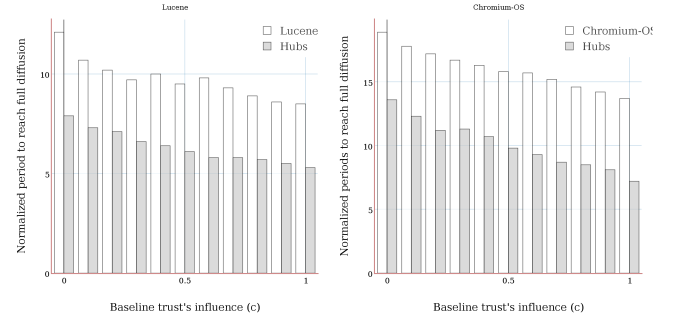


Figure 12. Comparisons of between random seeding and seeding from the hubs (periods to reach full diffusion).

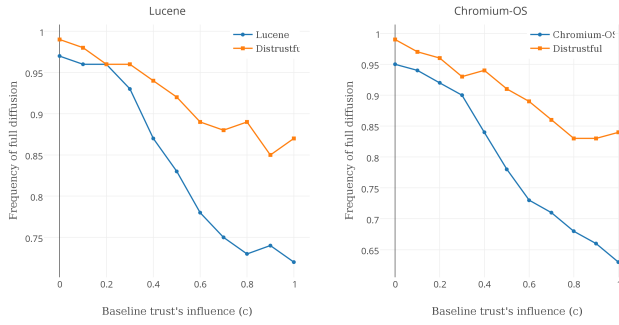
Figure 11 shows the influence of seeding from the hubs on the effective development of cooperation and trust. Obviously, this effect is more significant when baseline trust's influence is large ( $c \rightarrow 1$ ). Similarly, the speed of diffusion also improves by using this strategy (see figure 12, and simulation results from both LUCENE and CHROMIUM OS networks show the same patterns. Seeding from the hubs also helps to reduce the uncertainty about how long it takes to reach full diffusion in the worst case.

##### Seeding from The Distrustful

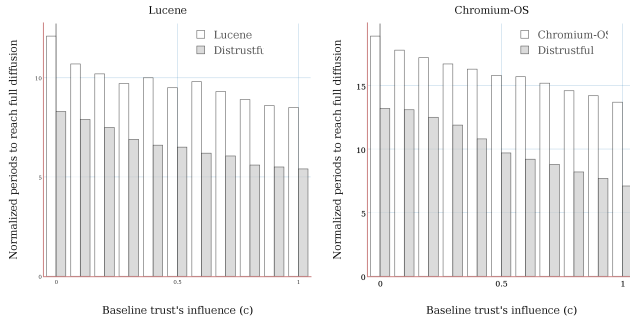
The second seeding strategy examined in this study is seeding from the distrustful. The simulation results suggest it is also an effective and efficient way to improve the diffusion of trust and cooperation. Figure 13 shows that for both networks, seeding from the distrustful always brings better than random results for almost all conditions. The only exception is  $c = 0.2$  for LUCENE network. For the speed of diffusion, figure 14 indicates the exactly same patterns.

##### Joint and Independent Effect

There may be some correlations between “hubs” and “the distrustful” (see figure 7). The distrustful may be slightly more likely to appear in the hub positions, which is why we do not



**Figure 13.** Comparisons of between random seeding and seeding from the distrustful (frequency of full diffusion).



**Figure 14.** Comparisons of between random seeding and seeding from the distrustful (periods to reach full diffusion).

simply put the term “Ceteris Paribus” in PROPOSITION IV & V. Due to the restrictive empirical network structures, we cannot fully evaluate their effects independently. However, it is reasonable to assume that “seeding from the distrustful” has at least a moderate level of independent positive effects. Intuitively, for example, when  $c = 1$ , seeding from the distrustful yields better results on both the effectiveness and speed of diffusion. Therefore, there must be an effect resulting from the independent influence of seeding from the distrustful strategy. The independent effect of seeding from the hubs can be established by similar arguments.

#### Summary of Findings

The main findings can be summarized as the following three PROPOSITIONS:

**PROPOSITION IV:** *The effectiveness and the speed of diffusion improves when seeding from those in the hub positions.*

**PROPOSITION V:** *The effectiveness and the speed of diffusion improves when seeding from those who are distrustful. The effect becomes increasingly significant with the higher influence of baseline trust.*

**PROPOSITION VI:** *The combination of both strategies yields better results, although they both have independent positive effects on the speed of diffusion.*

Seeding strategy matters! PROPOSITION IV, V, & VI show that using proper seeding strategy would help improve the effectiveness and speed of the diffusion of trust and cooperation. Although there are some correlations between the hubs and the distrustful, both of them have independent impact. Again we performed sensitivity analysis on payoff structures. Under the condition that the payoff from interaction is at the same level of baseline trust, the results are robust enough when punishment/compensation is comparable to cost.

## DISCUSSION

### Discussion of Findings

#### The Implications of Diverse Diffusion Trajectory

The study reveals that the diffusion trajectory becomes diverse when considering individual variations on baseline trust. One of the key challenges in CSCW research is to facilitate the diffusion of social and technical innovations within the distributed team. Conventionally, researchers often assume the successful diffusion of social and technical innovations follows S-shape curve. The S-shape curve actually reflects the importance of critical mass in accelerating the diffusion process [48]. The diverse curve indicates that only focusing on critical mass may be neither sufficient nor efficient. A few “powerful” or “extreme” individuals may be equally important. Particularly, in the processes exhibiting a “staged” pattern, almost all halts in diffusion are triggered by a few individuals who have extremely low baseline trust. Therefore, identifying the “staged” pattern through simulation has particular value for CSCW research. This helps to identify the major blockers of the diffusion. Then, researchers may design specific mechanisms (e.g., incentives, extra connections, seeding, etc.) to overcome the negative effects of these individuals. Doing so before the initiation of innovations may improve the efficiency and effectiveness while saving resources.

Moreover, the diverse diffusion trajectories also may lead to rethinking the use of statistical techniques used in empirical social network analysis. Currently, linear regression techniques are widely used in exploring the relationship between network attributes and the diffusion of innovation or other social constructs [34, 59, 68]. Since the subjects’ characteristics may vary a lot in empirical data, non-linear regression models may be a worthwhile alternative for CSCW researchers [40]. Applying such models may bring more insights and also help to alleviate the validity concerns of using linear regression techniques in social network analysis [27, 50].

#### The Role of Cheap Talk

As we mentioned before, there are only initial seeds using *cooperate* at the very beginning. In the first few normalized periods, switching from *defect* to *cheap talk* is the mainstream dynamic. Then, some individuals start to switch to *cooperate* either directly from *defect* or indirectly. [66] reported that cheap talk works as a catalyst in trust and cooperation development, and tends to disappear gradually once trust and cooperation are established. In general, this argument still largely holds. However, there are two differences. First, [66] requires that cheap talk achieves the majority at the global level before the occurrence of switching to *cooperate*. In this present study, this requirement becomes unnecessary. Under

some specific network structures (e.g., cluster or star) or specific composition of baseline trust, a local majority of cheap talk, or even a single individual may lead to the emergence of trust. Second, the cheap talk may not disappear. Since some individuals may have very low trust, they may stay with cheap talk and have no interest to *cooperate*. In this case, the incentive (extra interaction payoff) may be not good enough to offset their unwillingness towards *cooperate*.

Moreover, the role of cheap talk becomes even more significant yet more subtle. In the next subsection, we will show that the establishment of trust may require very specific conditions of distribution of cheap talk. Sometimes, it may require an individual to stay with cheap talk for a while to allow his or her neighbors' to develop trust first (see CASE 2 in next subsection). To sum up, the role of cheap talk is much more complicated than it seems to be. Future research is necessary to uncover its influence.

#### Why Seeding from the Distrustful Works?

Our study reveals that seeding from the hub and seeding from the distrustful are efficient interventions for developing team level trust. Seeding from the hub is pretty straightforward because the hubs are usually more influential. It also has been consistently confirmed by prior literature [4]. However, the underlying mechanism why seeding from the distrustful is also efficient is not very clear. Now we are going to conceptually discuss the rationale of seeding from the distrustful. In the following discussions, without loss of generality, let's assume  $c = 0.6$  and use the other parameters as they are in the simulation directly.

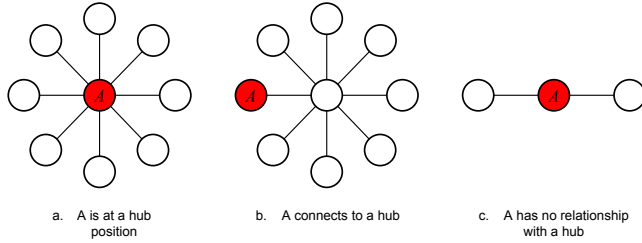


Figure 15. The possible positions for a low trust individual.

First, let's consider an individual  $A$  who has very low baseline trust (*baseline trust* = -1). In any social network,  $A$  may be at a hub position ( $A@Hub$ , figure 15.a), connects to it ( $A \leftrightarrow Hub$ , figure 15.b), or has no relationship with a hub ( $A \nleftrightarrow Hub$ , figure 15.c).

#### CASE 1: $A@Hub$

In this case,  $A$  connects to a number of other individuals. Without loss of generality, let's assume  $A$  has 8 links as shown in figure 15.a to simplify the discussions. Among these 8 neighbors,  $x$  neighbors play *C-C*,  $y$  neighbors play *cooperate*, and the rest ( $z$ ) play *defect*. At the very beginning,  $A$  plays *defect*. Now, let's calculate how difficult for  $A$  to convert to *cooperate*. First of all, let's write the  $A$ 's expected utilities of

using three strategies using equation 1 and 2:

$$\begin{aligned} U_A(C) &= \frac{2x + 2y + 0.5z}{8} - c \\ U_A(C - C) &= \frac{1.8x + 1.9y + 1.3z}{8} \\ U_A(D) &= \frac{1.5x + 0.5y + z}{8} + c \end{aligned} \quad (6)$$

Suppose  $c = 0.6$ , the utility differences between these three strategies are:

$$\begin{aligned} U_A(C) - U_A(C - C) &= \frac{0.2x + 0.1y - 0.8z}{8} - 0.6 \\ U_A(C - C) - U_A(D) &= \frac{0.3x + 1.4y + 0.3z}{8} - 0.6 \\ U_A(C) - U_A(D) &= \frac{0.5x + 1.5y - 0.5z}{8} - 1.2 \end{aligned} \quad (7)$$

In this case, switching from  $D$  to  $C-C$  requires very specific conditions to be satisfied. Let's have a close look at  $U_A(C - C) - U_A(D)$ , it can be rewritten as  $\frac{1.4y + 0.3(x+z)}{8} - 0.6 = \frac{1.4y + 0.3(8-y)}{8} - 0.6 = \frac{1.1y}{8} - 0.3$ . To ensure it is positive (more likely to switch),  $y$  must  $\geq 3$ . So  $A$  has to wait at least 3 of his neighbors playing  $C-C$  before he or she can actually give up the *defect* strategy. It may take some extra steps. Anyway, it is still possible for  $A$  to switch to  $C-C$  although it could be slow and take many steps.

The real trouble happens when  $A$  tries to switch from  $C-C$  to  $C$ . The maximal possible value of  $U_A(C) - U_A(C - C)$  is -0.4 when  $x = 8, y = 0, z = 0$ . Therefore, becoming cooperative could never be better than using  $C-C$ . Suppose  $\beta = 10$ , according to equation 4, the probability of switching from  $C-C$  to  $C$  is:  $\frac{e^{14}}{e^{14} + e^{18}} = \frac{1}{1 + e^4} = 0.018$  even when all  $A$ 's 8 neighbors have become cooperators! Obviously, it is almost impossible for  $A$  to switch from  $C-C$  to  $C$ . It is very likely that  $A$  will get stuck with strategy  $C-C$ . The whole process hence cannot reach all-cooperator state.

Let's see what will happen if seeding from  $A$ . First, seeding from  $A$  reduces the time waiting for its satellites to adopt *cheap talk*.  $A$  will become a cooperator at the very beginning, and keep using *cooperator*. Second, it also accelerates the process for its satellites to be cooperative. For many of its satellites may only connect to  $A$ ,  $A$  becomes a cooperator may immediately lead them switch to *cooperate* also.

#### CASE 2: $A \leftrightarrow Hub$

In the second case,  $A$  only connects to the hub node. Let's assume the hub is still using *defect*. With the same parameters in CASE 1, its expected payoffs are: -0.1 (*cooperate*), 1.3 ( $C-C$ ), 1.6 (*defect*). It would strongly prefer to be a defector. The probability of switching to *cooperate* and  $C-C$  are 0 and 0.047 respectively. It is almost impossible to directly switch to *cooperate*. We can expect it takes  $1/0.047 \approx 21$  reviews to switch to  $C-C$ . For an  $N$ -member team, this would take  $21 \times N$  periods on average. This would be extremely slow. If the hub has already been a cooperator,  $A$ 's expected payoffs are: 1.4



(*cooperate*), 1.8 (*C-C*), 2.1 (*defect*). A's best choice is still to be a defector.

Only when the hub uses *C-C*, A will quickly switch to *C-C*. However, since the hub's strategic choice also depends on a set of other individuals, it may not stay with *C-C* for a long time. If A miss this short interval, A might get stuck in *defect*.

#### CASE 3: A ↔ Hub

The discussion is very similar to above two cases; we omit it to keep this section concise.

To sum up, the presence of A as an individual of extremely low baseline trust is potential to cause significant delays or even failures in all these three cases. Seeding from the distrustful will avoid these low-trust individuals to get stuck with their "preferred" *defect* strategy. Especially, when the distrustful are at the hub, seeding from the distrustful will also help to accelerate the process for its satellites to be cooperative. The discussions for Case 1 also at least partially illustrates the benefits of using both seeding strategies.

#### Implications to Research

This work has implications for future research in distributed collaboration and trust. First, we explore how baseline trust, as an important individual characteristic, shapes the diffusion of trust and cooperation in a social network setting. The state-of-the-art social network research usually focuses on the aspects resulting from different positions of individuals, while assuming individual nodes are as same as each other [61]. However, individuals differ in many aspects. They have various beliefs, preferences, and hence may behave differently in social interactions [22]. Our work thus demonstrates the importance of considering individual characteristics.

We developed a new way to conduct data-driven, agent-based simulation study, which combines abstract, flexible simulation and empirical, observational data collected from real-world software projects. We demonstrate that it achieves both methods' advantages, allowing us to explore dynamics in individual and team level while keeping the results and findings high relevant to practical applications. This method extends the current simulation approach that is an important way of knowing in HCI [47]. Researchers may leverage it in theory development and empirical inquiry, since it produces empirically testable hypotheses, especially for research targeting complex, dynamic social and technical systems in which individual variations cannot be ignored.

Another contribution is the model itself, which represents extensible theoretical knowledge for understanding both how people interact and influence each other in a social network, and how their baseline trust influences their behavioral decisions. Integrating these social theories into a consistent and comprehensive model enables us to examine a rich set of factors together in a unified platform. Moreover, the model can be extended by incorporating other social theories by simply coding these theories to the decision rules. In developing the infrastructure for the simulation experiment, we adapted, developed, and invented several methods to extract social structure and individuals' baseline trust from team communication

records. Other theoretical and empirical studies may also apply the methods developed in this paper.

#### Implications to Practice

Some findings, especially our seeding strategy insights, can be directly applied to distributed collaboration practices. Prior research demonstrates how identifying team hubs and investing more resources to help them adopt cooperation first (seeding from the hubs) may be an effective and efficient way to improve the diffusion process. This study confirms the usefulness of this strategy in empirical network settings. Moreover, we show that "seeding from the distrustful" is also an effective strategy; one might even combine them to achieve better results. Our study reveals that some individuals with lower baseline trust may potentially block the progress of diffusion. Therefore, another possible implication lies in designing organizational communication networks to minimize such individuals' influence. This can be achieved by adding new links between unconnected people; for example, connecting A to C in figure 1 may avoid the negative influence of B, and may eventually force B to switch to cooperate.

#### Design Implications

This study opens possibilities for designing tools that support team collaboration. The agent-based model can be expanded and augmented with a rich user interface to serve as a decision-making tool for GSE practitioners. By changing the model's parameters (e.g., payoff structure, social learning factors, etc), project managers and team leaders can run "what-if" experiments to navigate the mechanism design space and explore different scenarios, and evaluate the potential influences of decisions within the team's context. In this way, the team may be able to select the best mechanism, such as a combination of different seeding strategies, to proactively facilitate the teamwork process. The agent-based model is dynamic, which enables tools based on it to identify the influence of a specific scenario, as well as develop insights into the long-term consequences of complex social-technical processes.

#### Threats to Validity

**Construct Validity** The main construct in this study is "baseline trust," which refers to an individual's general, global tendency in perceiving the trustworthiness of other individuals (or other entities, such as organizations) [16]. It is a dimension of personality, yet we did not use traditional psychometric approaches to assess it due to practical restrictions. We adopted an unconventional method based on text analytics, for which there is some early experimental evidence to confirm its validity [52, 62]. More evaluations in future will help establish the confidence in this method.

The private conversations among developers may also influence the assessment of baseline trust. For LUCENE, the influence may be minimal for it is a pure open source project and most of the developers communicate with each other only through the public mediums. LUCENE also enforces that all discussions should be logged and public. However, Google has internal communication channels for their employees. Therefore, for CHROMIUM OS, the influence of the absence

of private conversations may be significant. In general, the extracted baseline trust may be slightly lower without considering the internal records. People tend to assume their colleagues are trustworthy because they often reason that their colleagues are at least qualified to work in a prestigious software development organization (Google) [2]. Shared identity also enhances trust. These may be reflected by their word use in interaction. However, for personality is relatively stable, and the word use is largely unconscious [19], the extracted baseline trust should be valid.

**External Validity** This study utilized empirical data from two open source projects. We cannot guarantee that the results and findings of this study are generalizable to other projects. However, replicating this study with different empirical settings would inductively develop solid knowledge. Researchers would become more confident in a theory when similar findings emerge in different contexts [6]. However, it is possible that the findings may be still valid for other open source project, or even global software team in traditional organizations. There are a few reasons. First, we only consider the “developers” who contribute the code, which enable us to avoid the limitations and problems associated with the hierarchical structure, “small-core and very-large-periphery” distribution of work. Second, CHROMIUM-OS is a company-driven open source project where many contributors are from the same company but located distributively. It more or less reflects the nature of distributed work in commercial software development organizations.

**Internal Validity** There is no significant threat to internal validity. The empirical data used in this study are public communication records that are collected and analyzed by computer programs, and thus there is almost no human judgment involved in the data collection, extraction, and cleaning process. The agent-based simulation process is autonomous. The agents’ position in the networks are derived from empirical data, and we only specify the rules that are applied to all agents without any manipulation on a specific individual agent.

## CONCLUSION

In this paper, we describe a study focused on identifying the influence of individuals’ baseline trust on the diffusion of trust and cooperation with the mediation of cheap talk over the Internet. Using data from two large globally distributed open source projects, we design and perform an experiment in the form of a virtual simulation. Our results demonstrate that: (1) cheap talk over the Internet is still important for the development of trust and cooperation when considering individual variations on baseline trust, (2) baseline trust impacts the diffusion of trust and cooperation significantly, and yields very rich and non-traditional diffusion trajectories, and (3) seeding from the hubs and seeding from the distrustful positively influences the effectiveness and speed of diffusion. Combining these together is likely to provide better diffusion of trust and cooperation.

This research makes multiple contributions. First, the research revealed the importance of considering individual variations on baseline trust and developed an understanding to-

wards the influence of baseline trust. Second, we developed an extensible, theoretical model that can be applied in future research investigating trust and cooperation dynamics in GSE teams. Third, the study provides ready-to-use suggestions (i.e., seeding strategies) for managing collaborations in GSE. Finally, we developed a data-driven, agent-based modeling and simulation approach, integrating it with observational, empirical data. This integration has great potential to help CSCW researchers to develop flexible, rigorous, and relevant theories. For the future, we will continue our efforts on investigating collaborations in GSE team with data-driven, agent-based modeling and simulation approach. We will explore the integration of dynamic network to enable the co-evolution of trust and network structure [58, 57]. Moreover, trust is a complex construct that can be influenced by many factors [51]. One limitation of this study is that we have not considered these factors. We plan to extend our research to incorporate more social and organizational factors, for example, social identity, cultural background [71], organizational constraints [43], etc. The simulation infrastructures developed in this paper can be easily adapted to incorporate them in future.

## ACKNOWLEDGEMENT

We would like to thank Prof. Debra Richardson and Prof. Brian Skyrms for discussing the research idea and the early version of this paper. We also want to thank the ACs and reviewers for their insightful comments and suggestions.

## REFERENCES

1. E. Abrahamson and L. Rosenkopf. 1997. Social network effects on the extent of innovation diffusion: A computer simulation. *Org. Sci.* 8, 3 (1997), 289–309.
2. B. Al-Ani, M. Bietz, Y. Wang, and et al. 2013. Globally distributed system developers: their trust expectations and processes. In *Proc. CSCW*. 563–574.
3. B. Al-Ani, S. Marczak, D. Redmiles, and R. Prikładnicki. 2014. Facilitating contagion trust through tools in global systems engineering teams. *Inf. Softw. Technol.* 56, 3 (2014), 309–320.
4. S. Aral, L. Muchnik, and A. Sundararajan. 2013. Engineering social contagions: Optimal network seeding in the presence of homophily. *Net. Sci.* 1 (2013), 125–153.
5. B. Barber. 1983. *The logic and limits of trust*. Rutgers.
6. V. Basili, F. Shull, and F. Lanubile. 1999. Building knowledge through families of experiments. *Soft. Engin., IEEE T.* 25, 4 (1999), 456–473.
7. N. Bettenburg, E. Shihab, and A. Hassan. 2009. An empirical study on the risks of using off-the-shelf techniques for processing mailing list data. In *Proc. ICSM*. 539–542.
8. C. Bird, A. Gourley, P. Devanbu, M. Gertz, and A. Swaminathan. 2006. Mining email social networks. In *Proc. MSR*. 137–143.

9. S. Borgatti and P. Foster. 2003. The network paradigm in organizational research: A review and typology. *J. Manage* 29, 6 (2003), 991–1013.
10. D. Brass. 1984. Being in the right place: A structural analysis of individual influence in an organization. *Admin. Sci. Quart.* (1984), 518–539.
11. J. Cassell and T. Bickmore. 2003. Negotiated Collusion: Modeling Social Language and its Relationship Effects in Intelligent Agents. *User Model. User-Adap.* 13, 1-2 (2003), 89–132.
12. M. Cataldo and J. Herbsleb. 2008. Communication networks in geographically distributed software development. In *Proc. CSCW*. 579–588.
13. M. Cataldo, J. Herbsleb, and K. Carley. 2008. Socio-technical congruence: a framework for assessing the impact of technical and work dependencies on software development productivity. In *Proc. ESEM*. 2–11.
14. L-E. Cederman. 2005. Computational Models of Social Forms: Advancing Generative Process Theory. *Am. J. Sociol.* 110, 4 (2005), 864–893.
15. J. Delhey, K. Newton, and C. Welzel. 2011. How general is trust in “most people” Solving the radius of trust problem. *Am. Sociol. Rev.* 76, 5 (2011), 786–807.
16. J. Driscoll. 1978. Trust and participation in organizational decision making as predictors of satisfaction. *Acad. Manage. J.* 21, 1 (1978), 44–56.
17. R. Engle and C. Granger. 1987. Co-integration and error correction: representation, estimation, and testing. *Econometrica* (1987), 251–276.
18. D. Etzion. 2014. Diffusion as classification. *Organization Science* 25, 2 (2014), 420–437.
19. Lisa A Fast and David C Funder. 2008. Personality as manifest in word use: correlations with self-report, acquaintance report, and behavior. *Journal of personality and social psychology* 94, 2 (2008), 334.
20. N. Feltovich. 2000. Reinforcement-based vs. Belief-based learning models in experimental asymmetric-information Games. *Econometrica* 68, 3 (2000), 605–641.
21. D. Fisher and P. Dourish. 2004. Social and temporal structures in everyday collaboration. In *Proc. CHI*. ACM, 551–558.
22. Herbert Gintis. 2014. *The bounds of reason: game theory and the unification of the behavioral sciences*. Princeton University Press.
23. J. Herbsleb, D. Atkins, David G. Boyer, M. Handel, and T. Finholt. 2002. Introducing instant messaging and chat in the workplace. In *Proc. CHI*. 171–178.
24. J. Herbsleb and A. Mockus. 2003. An empirical study of speed and communication in globally distributed software development. *Soft. Engin., IEEE T.* 29, 6 (2003), 481–494.
25. P. Hinds and S. Kiesler. 2002. *Distributed work*. MIT.
26. Q. Hong, S. Kim, S. Cheung, and C. Bird. 2011. Understanding a developer social network and its evolution. In *Proc. ICSM*. 323–332.
27. J. Howison, A. Wiggins, and K. Crowston. 2011. Validity Issues in the Use of Social Network Analysis with Digital Trace Data. *J. Assoc. Inf. Syst.* 12 (2011).
28. M. Jackson. 2010. *Social and Economic Network*. Princeton University Press.
29. S. Jarvenpaa, K. Knoll, and D. Leidner. 1998. Is anybody out there? Antecedents of trust in global virtual teams. *J. Manage. Inf. Syst* (1998), 29–64.
30. D. Kahneman. 2003. Maps of Bounded Rationality: Psychology for Behavioral Economics. *Am. Econ. Rev.* 93, 5 (2003), 1449–1475.
31. A. Kanavos, I. Perikos, P. Vikatos, I. Hatzilygeroudis, C. Makris, and A. Tsakalidis. 2014. Conversation Emotional Modeling in Social Networks. In *Proc. ICTAI*. 478–484.
32. R Kempter, V. Sintsova, C. Musat, and P Pu. 2014. EmotionWatch: Visualizing fine-grained emotions in event-related tweets. In *Proc. ICWSM*.
33. M. Kilduff, W. Tsai, and R. Hanke. 2006. A paradigm too far? A dynamic stability reconsideration of the social network research program. *Acad. Manage. R.* 31, 4 (2006), 1031–1048.
34. F. Kivran-Swaine and M. Naaman. 2011. Network Properties and Social Sharing of Emotions in Social Awareness Streams. In *Proc. CSCW*. 379–382.
35. H. Köpcke, A. Thor, and E. Rahm. 2010. Evaluation of Entity Resolution Approaches on Real-world Match Problems. In *Proc. VLDB*. 484–493.
36. E. Kouloumpis, T. Wilson, and J. Moore. 2011. Twitter sentiment analysis: The good the bad and the omg!. In *Proc. ICWSM*. 538–541.
37. C. Macal and M. North. 2010. Tutorial on agent-based modelling and simulation. *J. Sim.* 4, 3 (2010), 151–162.
38. C. Manning, P. Raghavan, and H. Schütze. 2008. *Introduction to Information Retrieval*. Cambridge.
39. S. Mohammad and P. Turney. 2013. Crowdsourcing a word-emotion association lexicon. *Comput. Intelli.* 29, 3 (2013), 436–465.
40. S. Myers and J. Leskovec. 2014. The bursty dynamics of the twitter information network. In *Proc. WWW*. 913–924.
41. Judith S. Olson and Wendy A. Kellogg (Eds.). 2014. *Ways of Knowing in HCI*. Springer New York.
42. J. Pennebaker, R. J Booth, and M. Francis. 2007. Linguistic inquiry and word count: LIWC 2007. *LIWC.net* (2007).

43. V. Perrone, A. Zaheer, and B. McEvily. 2003. Free to be trusted? Organizational constraints on trust in boundary spanners. *Org. Sci.* 14, 4 (2003), 422–439.
44. R Development Core Team. 2008. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
45. R. Reagans, E. Zuckerman, and B. McEvily. 2004. How to make the team: Social networks vs. demography as criteria for designing effective teams. *Admin. Sci. Quart.* 49, 1 (2004), 101–133.
46. Y. Ren and R. Kraut. 2014a. Agent-Based Modeling to Inform Online Community Design: Impact of Topical Breadth, Message Volume, and Discussion Moderation on Member Commitment and Contribution. *Hum-Comput. Interact.* 29, 4 (2014), 351–389.
47. Y. Ren and R. Kraut. 2014b. Agent Based Modeling to Inform the Design of Multiuser Systems. In *Ways of Knowing in HCI*. Springer New York, 395–419.
48. E. Rogers. 2003. *Diffusion of innovations* (5th ed.). Simon & Schuster.
49. G. Rossman. 2012. *Climbing the charts: What radio airplay tells us about the diffusion of innovation*. Princeton University Press.
50. G. Rossman, M. Chiu, and J. Mol. 2008. Modeling diffusion of multiple innovations via multilevel diffusion curves: Payola in pop music radio. *Sociol. Methodol.* 38, 1 (2008), 201–230.
51. D. Rousseau, S. Sitkin, R. Burt, and C. Camerer. 1998. Not so different after all: A cross-discipline view of trust. *Acad. Manage. R.* 23, 3 (1998), 393–404.
52. J. Shen, O. Brdiczka, and J. Liu. 2013. Understanding email writers: Personality prediction from email messages. In *User Modeling, Adaptation, and Personalization*. Springer, 318–330.
53. B. Skyrms. 2001. The Stag hunt. *Presidential Address, Proceedings and presidential Addresses of the APA* 75 (2001), 31–41.
54. B. Skyrms. 2004. *The stag hunt and the evolution of social structure*. Cambridge University Press.
55. B. Skyrms. 2005. Dynamics of conformist bias. *The Monist* 88, 2 (2005), 260–269.
56. B. Skyrms. 2008. Trust, risk, and the social contract. *Synthese* 160, 1 (2008), 21–25.
57. Br. Skyrms and R. Pemantle. 2000. A Dynamic Model of Social Network Formation. *P. Natl. Acad. Sci. USA* (2000), 9340–9346.
58. G. Soda, A. Usai, and A. Zaheer. 2004. Network memory: The influence of past and current networks on performance. *Acad. Manage. J.* 47, 6 (2004), 893–906.
59. K. Starbird and L. Palen. 2012. (How) will the revolution be retweeted?: information diffusion and the 2011 Egyptian uprising. In *Proc. CSCW*. 7–16.
60. P. Sturgis and P. Smith. 2010. Assessing the validity of generalized trust questions: What kind of trust are we measuring? *Intl. J. Publ. Opi. Res.* 22, 1 (2010), 74–92.
61. A. Sundararajan, F. Provost, G. Oestreicher-singer, and S. Aral. 2013. Information in Digital , Economic , and Social Networks. *Inf. Syst. Res.* 24, 4 (2013), 883–905.
62. Y. R. Tausczik and J. Pennebaker. 2010. The psychological meaning of words: LIWC and computerized text analysis methods. *J. Lang. Soc. Psychol.* 29, 1 (2010), 24–54.
63. S. Taylor, E. Bakshy, and S. Aral. 2013. Selection effects in online sharing: Consequences for peer adoption. In *Proc. EC*. 821–836.
64. K. Train. 2009. *Discrete choice methods with simulation*. Cambridge University Press.
65. J. Von Neumann and O. Morgenstern. 2007. *Theory of games and economic behavior (60th Anniv. Comm. Ed.)*. Princeton University Press.
66. Y. Wang and D. Redmiles. 2013. Understanding Cheap Talk and the Emergence of Trust in Global Software Engineering: An Evolutionary Game Theory Perspective. In *Proc. CHASE*. 149–152.
67. Y. Wang and D. Redmiles. 2015. Cheap Talk, cooperation, and trust in global software engineering: An evolutionary game theory model with empirical support. *Empirical Software Engineering* (2015), Accepted, to appear.
68. B. Xu, Y. Huang, H. Kwak, and N. Contractor. 2013. Structures of broken ties: exploring unfollow behavior on twitter. In *Proc. CSCW*. 871–876.
69. T. Yamagishi and M. Yamagishi. 1994. Trust and commitment in the United States and Japan. *Motiv. Emotion* 18, 2 (1994), 129–166.
70. H. Young. 1998. *Individual Strategy and Social Structure: An Evolutionary Theory of Institutions*. Princeton University Press.
71. M. Yuki, W. Maddux, M. Brewer, and K. Takemura. 2005. Cross-cultural differences in relationship-and group-based trust. *Pers. Soc. Psychol. B.* 31, 1 (2005), 48–62.
72. J. Zheng, E. Veinott, N. Bos, J. Olson, and G. Olson. 2002. Trust without touch: jumpstarting long-distance trust with initial social activities. In *Proc. CHI*. 141–146.
73. R. Zolin, P. Hinds, R. Fruchter, and R. Levitt. 2004. Interpersonal trust in cross-functional, geographically distributed work: A longitudinal study. *Inf. Org.* 14, 1 (2004), 1–26.