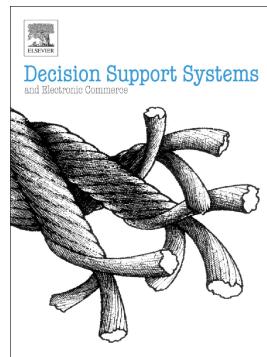


Accepted Manuscript

A Visual Analytics System to support Tax Evasion Discovery

Walter Didimo, Luca Giamminonni, Giuseppe Liotta, Fabrizio Montecchiani, Daniele Pagliuca



PII: S0167-9236(18)30056-3
DOI: [doi:10.1016/j.dss.2018.03.008](https://doi.org/10.1016/j.dss.2018.03.008)
Reference: DECSUP 12943

To appear in: *Decision Support Systems*

Received date: 7 November 2017
Revised date: 6 March 2018
Accepted date: 25 March 2018

Please cite this article as: Walter Didimo, Luca Giamminonni, Giuseppe Liotta, Fabrizio Montecchiani, Daniele Pagliuca , A Visual Analytics System to support Tax Evasion Discovery. The address for the corresponding author was captured as affiliation for all authors. Please check if appropriate. Decsup(2018), doi:[10.1016/j.dss.2018.03.008](https://doi.org/10.1016/j.dss.2018.03.008)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

A Visual Analytics System to support Tax Evasion Discovery

Walter Didimo^{a,*}, Luca Giamminonni^a, Giuseppe Liotta^a, Fabrizio Montecchiani^a, Daniele Pagliuca^{a,b}

^aDip. Ingegneria, Università degli Studi di Perugia, Via G. Duranti 93 06125 (PG), Italy

^bAgenzia delle Entrate, Arezzo, Italy

Abstract

This paper describes TAXNET, a decision support system for tax evasion discovery, based on a powerful visual language and on advanced network visualization techniques. It has been developed in cooperation with the Italian Revenue Agency, where it is currently used. TAXNET allows users to visually define classes of suspicious patterns, it exploits effective graph pattern matching technologies to rapidly extract subgraphs that correspond to one or more patterns, it provides facilities to conveniently merge the results, and it implements new ad-hoc centrality indexes to rank taxpayers based on their fiscal risk. Moreover, it offers a visual interface to analyze and interact with those networks that match a desired pattern. The paper discusses the results of an experimental study and some use cases conducted with expert officers on real data and in a real working environment. The experiments give evidence of the effectiveness of our system.

Keywords: Tax Evasion, Network Analysis, Graph Visualization, Visual Analytics, Graph Pattern Matching, Graph Database

1. Introduction

Tax evasion represents one of the major problems of many governments, because of its strong economic, political, and social impact (see, e.g., Basta et al. 2009; González & Velásquez 2013; Goumagias et al. 2012; Matos et al. 2015; Tian et al. 2016; Wu et al. 2012). Italy is among the

*Corresponding author.

Email addresses: walter.didimo@unipg.it (Walter Didimo), giuseppe.liotta@unipg.it (Giuseppe Liotta), fabrizio.montecchiani@unipg.it (Fabrizio Montecchiani), daniele.pagliuca@agenziaentrate.it (Daniele Pagliuca)

countries that are particularly affected by this phenomenon: In the period 2007-2013, the Italian government estimated an annual average of tax evasion of about 91.4 billion EUR. As a consequence, the Italian Revenue Agency (IRV in the following) devotes every year a great amount of human and economic resources to contrast the problem. Italian tax officers have to continuously deal with an overwhelming number of heterogeneous data sources, managed through different software applications. This causes information redundancy and makes it extremely difficult for the analysts to preserve their mental map during the fiscal audit process. Moreover, the current software applications used at the IRV are based on a taxpayer-centric paradigm which does not allow for an easy exploration of relations between different subjects. Hence, tax evasion involving groups of subjects, rather than single individuals, is typically more difficult to discover.

This paper deals with the design of a new decision support system for tax evasion discovery, able to assist analysts to overcome the above problems. Our contribution is twofold:

(i) We present the system TAXNET, developed in cooperation with the IRV. It supports the work of tax officers by means of a powerful visual language and network visualization techniques. It models the set of data as a unified network, whose nodes represent taxpayers and whose edges are different types of economic and social relationships between them. The user can visually define classes of suspicious patterns, based both on topological properties and on node/edge attributes. TAXNET exploits effective graph pattern matching techniques to efficiently extract subgraphs that correspond to one or more suspicious patterns, it provides facilities to conveniently merge the results, and it implements new ad-hoc centrality indexes to rank taxpayers based on their fiscal risk. The system also offers visual tools to interact with those subgraphs that match a desired pattern, so to get more details or to filter out less relevant information. To efficiently execute graph pattern matching routines on large networks, data are conveniently stored in a graph database instead of a traditional relational database.

(ii) We assess the validity of TAXNET on real data and within a real working environment. Namely, the system is currently adopted at the IRV. We analyze the results of an experimental study and two use cases on data handled at the IRV to estimate the system's effectiveness. The

experimental data suggest that using TAXNET has a strong impact on the fiscal risk analysis process: It reduces the time needed to execute fundamental analysis tasks, facilitates the retrieval of suspicious patterns, and increases the reliability of the results.

The remainder of this paper is organized as follows. Section 2 discusses work related to our research and highlights differences and novelties of our system with respect to previous approaches. Section 3 illustrates examples of suspicious or fraudulent patterns in the tax evasion domain and it clarifies the importance of visualization both for the definition of such patterns and for the discovery of additional information related to subjects that match them. Section 4 recalls basic definitions about graphs and networks, used in the paper. Section 5 summarizes the typical workflow of a fiscal audit at the IRV, which guided us in the design of the system. In the same section, we formalize the data model of TAXNET, based on the considerations of Section 3. Sections 6-7 describe the visual language of TAXNET and its visual analysis tools. Sections 8-9 present the experimental studies and use cases of TAXNET conducted at the IRV. Section 10 concludes the paper and describes future work.

2. Related Work

Since traditional tax auditing methods typically require a significant amount of time and human resources, much effort has been made to use data mining techniques to automatically or semi-automatically discover tax evasion (González & Velásquez 2013; Wu et al. 2012) and financial frauds (Basta et al. 2009; Carneiro et al. 2017; Michalak & Korczak 2011; Ravisankar et al. 2011; Zhou & Kapoor 2011). See also the survey of Ngai et al. 2011. As observed by Tian et al. 2016, the main drawback of these approaches is that they are based on supervised methods, which usually need large sets of training data that are difficult to obtain. Moreover, the trained models become out-of-date when behaviors in tax evasion change and they need additional training. Besides the above general approach, different methods and systems tailored to the economic and fiscal reality of various countries have been developed, including a support model to predict the behavior of risk-neutral enterprises in Greece (Goumagias et al. 2012) and

social network analysis techniques to prevent money laundering phenomena within the data of an Italian factoring company (Colladon & Remondi 2017). The importance of fighting tax frauds by means of methods that analyze networks of economic transactions between subjects has been recently highlighted by the Council of the European Union 2016. Works in this direction include the analysis of networks about companies in Belgium (Vlasselaer et al. 2017), the identification of frequent fraud patterns in the Brasilian fiscal environment (Matos et al. 2015), and a graph-based method to discover specific families of suspicious transactions in China, which mainly involve medium and large companies (Tian et al. 2016).

In Europe, and in particular in Italy, the economic environment is predominantly characterized by small-medium sized enterprises (see, e.g., Bank 2013), and the continuous evolution of tax evasion and financial fraud schemes requires the use of flexible methodologies. In this direction, a promising approach for the analysis of economic and financial activities, which has received a growing consensus in the last decade, is based on the use of visualization techniques. Examples include: Systems for financial and social crime detection (Didimo et al. 2014; Drezewski et al. 2015; Xiang et al. 2005); systems to detect fraudulent activities by employees of business organizations (Argyriou et al. 2014); applications of visual analytics to financial stability monitoring and fraud detection in financial markets (Huang et al. 2009; Flood et al. 2015); comparative studies of different visualization methods for business ecosystem analysis (Basole et al. 2016).

Our system TAXNET makes use of graph visualization techniques but, differently from previous works, its intuitive visual language offers high flexibility in the definition of fraudulent patterns, and it allows us to model and capture a wide variety of suspicious fiscal schemes (including fake invoices, VAT missing trader fraud, and “controlled transactions”). Furthermore, TAXNET is tailored to the tax evasion domain and implements new effective risk indexes for this context, which are different from classical centrality indexes frequently used in social network analysis. We remark that other papers address the problem of defining visual query languages for graph data sets. Notable examplea are the system GRAPHITE (Chau et al. 2008) and the

system PROXIMITY, based on the query language of Blau et al. 2002. We also mention a system developed by Koenig et al. 2010, which returns a set of candidate results approximating a desired pattern and which allows for a subsequent interaction with these results in order to refine them. However, all these systems adopt ad-hoc data structures and pattern matching algorithms and are not conceived to take advantage of modern graph database technologies. Conversely, TAXNET interacts with NEO4J¹, one of the most popular graph database management systems; the visual language of TAXNET and its visual analytics engine are a substantial evolution of a previous prototype system called KOJAPH (Didimo et al. 2015). This evolution significantly increases the expressiveness of the visual language, the functionalities of the visual analytics tools, and the scalability of the visual interface. A more recent system related to TAXNET is VISAGE (Pienta et al. 2017, 2016). Similarly to our system, VISAGE has an intuitive interface that supports users in the definition of graph patterns. However, the visual language of VISAGE does not support some important features for rule definitions, such as conditions on edge attributes, bounds on the lengths of paths between nodes, numeric annotations for nodes and edges, logical operators different from the AND operator, regular expressions, and operations like merge and split of results, which can be effectively used in the analysis phase to manage the subgraphs that match a given query. TAXNET supports all the above features and it also automatically rank the results of a query according to a set of fiscal risk indexes specifically designed for the tax evasion discovery scenario.

3. Suspicious Patterns and the Role of Visualization

To better illustrate the specific application domain of our research, we show some real cases of interaction between taxpayers that highlight “pathological” economic behaviors (Section 3.1). According to a careful analysis conducted by the public authorities (including the IRV), these cases actually resulted in fiscal evasion activities. We then discuss how it is possible to abstract

¹<http://neo4j.com>

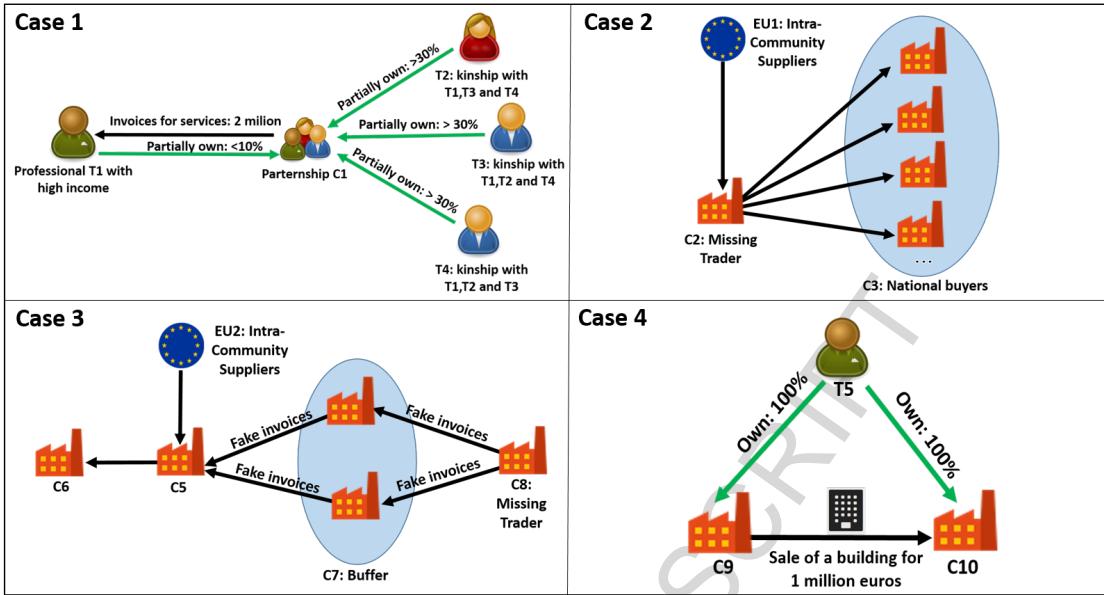


Fig. 1. Real cases of fiscal risky schemes.

the presented cases by identifying the underlying core schemes, which we refer to as *suspicious patterns*. In Section 3.2 we clarify the importance of visualization approaches to define suspicious patterns and to analyze data that match them.

3.1. Suspicious Patterns

We discuss four different real cases. Refer to Fig. 1 for a schematic illustration.

Case 1. Taxpayer T1 is a well-known professional who declares high incomes, subject to the highest rate of the personal income tax. T1 establishes a partnership (company) C1 with three relatives who have low incomes: T2, T3, and T4. The percentage of participation is lower for T1 (under 10%) and higher for T2, T3, and T4 (over 30% each). C1 issues invoices to T1 for service supplies, for over two million euros (EUR). The IRV found that this scheme was implemented to transfer a large part of the income from T1 to C1 and in turn to T2, T3, and T4. Moreover, thanks to this scheme, T1 unduly benefited from a lower tax rate. The tax evasion assessed by the IRV was over one hundred thousand euros.

Case 2. C2 is a recently established company in the trade of electronic products. The amount of invoices issued by C2 increases sharply: From a few hundred euros in 2013 to over ten million euros in 2014. In this last year, C2 receives invoices of significant amount for intra-community purchases of products from a European company EU1, under the non-taxable VAT regime. Then C2 resells the purchased products at a slightly lower price to various national companies C3, thus issuing new invoices, which this time are subject to VAT. This scheme was found to be an intra-community VAT fraud, where C2 acted as a missing trader; C2 did not submit the tax declaration and evaded VAT for more than 10 million euros. At the same time, C3 could resell the products at a competitive market price.

Case 3. C5 is a company that trades *white certificates*, i.e., documents that certify energy savings of market actors as a consequence of energy efficiency improvement actions. C5 makes significant intra-community purchases of white certificates from EU2, under the non-taxable VAT regime. C5, in turn, sells the aforementioned white certificates to C6, a big national company in the business of the electric service distribution. C5 receives several million euros of fake invoices from various companies having the role of “buffer” (C7), so to compensate the VAT debt of C5. At the same time C7 receives invoices from a missing trader C8 so that its VAT balance becomes close to zero. C8 does not submit the tax declaration and evades the VAT due. Public authorities assessed a fraud with fake invoices issued for over 17 millions euros, related to a false trade of white certificates. The amount of VAT evaded was over 3.5 million euros.

Case 4. C9 is a company in the building sector. C9 sells a building to a company C10 (operating in the building rental sector), issuing an invoice of one million euros. C9 declares its VAT debt for the aforementioned invoice, but it does not pay this debt. C10, in turn, performs a refund request for the VAT credit of its purchase. The IRV analyzed the refund request and discovered that C9 and C10 were owned by the same subject T5. The IRV denied the VAT refund, as this scheme exhibited a high fiscal risk.

Real cases such as those described above have some similarities with one another, but are also characterized by some specific peculiarities. As we shall see, these aspects can be taken

into account in the definition of patterns that capture a broad spectrum of suspicious situations. For example, in all the presented cases the fundamental types of relationships are economic transactions between persons and/or companies. In some cases (see Cases 1 and 4), there are also relevant shareholding relationships, which may highlight common interests among different economic counterparts (seller and buyer). Therefore, economic and shareholding relationships can be used as the main types of links between taxpayers in the definition of a suspicious pattern. At the same time, adding many details to the structure of a pattern may limit too much its scope. For example, kinship relationships may or may not be present in a scheme like the one underlying Case 1; thus we can avoid to specify such relationships in the core structure of a suspicious pattern, while they may be considered in a subsequent phase of exploration of the results that match the pattern. Also, some real cases are characterized by different details but a common key element. For example, Cases 2 and 3 have different numbers of subjects linked to a missing trader and in Case 3 there is an additional level of subjects that act as a “buffer”. However, the presence of a missing trader in both situations is a clear common key element. In these cases, a succinct definition of a pattern that includes the key element and that excludes additional details may help to efficiently capture a large variety of fiscal risky situations.

To give a concrete idea of the considerations above, in the following we briefly discuss examples of suspicious patterns abstracting Cases 1–4, defined on the basis of the experience of a specific IRV provincial office. Alternative abstraction processes, which lead to slightly different patterns for the same cases, may be followed according to the territorial economic peculiarities of other offices. Each pattern is informally described in terms of taxpayers, relationships, and additional rules. Refer to Figs. 2(a)–2(c), where taxpayers are represented by circles and their relationships are directed links (black for economic transactions and green for shareholdings). An economic transaction link is oriented from the seller to the buyer; a shareholding link is oriented from the shareholder to the participated company. Additional rules on an element (taxpayer or relationship) are described by textual labels associated with the element. A more formal data model that describes the network of taxpayers in terms of an algebraic graph is introduced

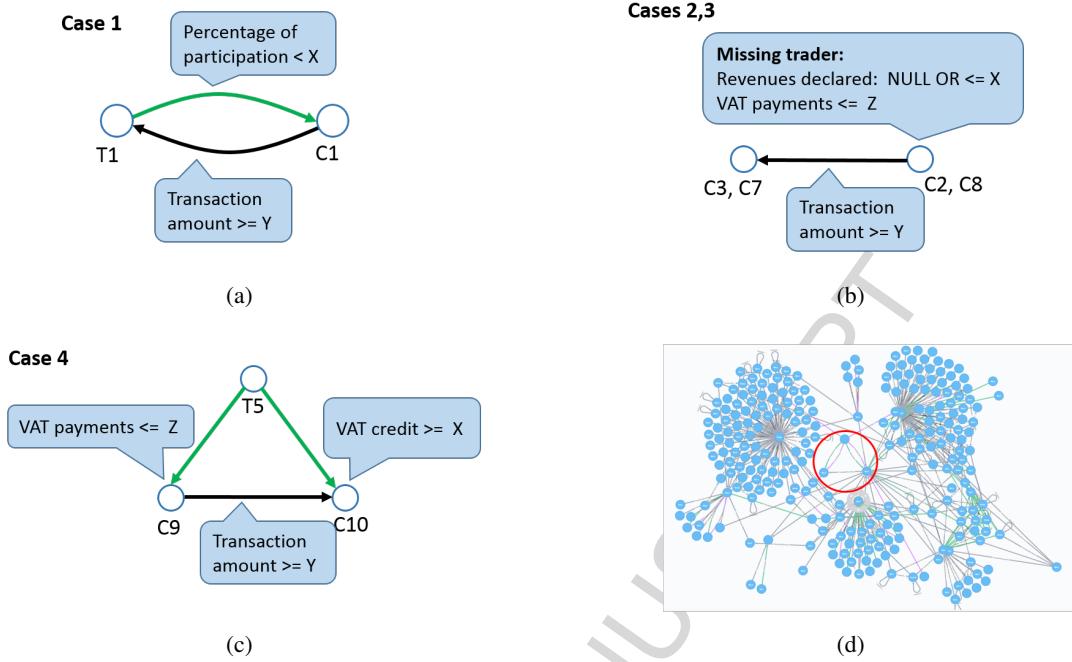


Fig. 2. (a)-(c) Suspicious patterns for Cases 1–4. (d) Matching the pattern of Case 4 in a real network.

in Sections 5. The visual language used to define suspicious patterns is described in Section 6.

Case 1: PurchaseFromRelated pattern. The fundamental characteristic of this case is the presence of a participated company (C1) that sells products/services to one of its shareholders (T1), which can be either a person (like in Case 1) or a company itself. Additionally, the amount of the sale and the shareholding percentage may be relevant information. Therefore, we can define a pattern composed of two taxpayers and a double relationship between them, one expressing an economic transaction and the other expressing the shareholding. The pattern can also be refined with the following rules: The economic transaction must be greater than or equal to a desired value; the shareholding percentage may be set below a desired value.

Cases 2-3: MissingTraderSuppliers pattern. The main characteristic of these cases is the presence of economic transactions in which the seller (e.g., C2, C8) is a missing trader with serious tax irregularities. It is relevant that the economic transactions towards the buyers (e.g., C3, C7) are greater than or equal to a given threshold, while tax irregularities of the seller corre-

spond to one or more of the following situations: The seller does not submit the tax declaration or declares revenues of small amounts; the VAT payments of the seller are below a given threshold. In the definition of the pattern, we do not add rules on the buyers, because they might limit too much its scope according to the experience of the IRV officers.

Case 4: `SuppliesFromAssociated` **pattern.** This case exhibits a triangular scheme, in which both the seller (C9) and the buyer (C10) of an economic transaction are participated by the same subject (T5). The main additional rules are: The economic transactions are greater than or equal to a given threshold; the VAT payments of the seller are below a given threshold; the VAT credit declared by the buyer exceeds a given threshold. Conversely, in the pattern definition we do not specify a threshold for the shareholding percentages, because, according to the experience of the IRV officers, this rule may filter out a relevant number of risky situations.

3.2. *The Role of Visualization*

The cases described above suggest that an effective fiscal risk analysis should not only look at the single taxpayers but should also look at the interplay between them. To this aim, our system conveniently models taxpayers and their relationships as an algebraic graph (see Section 5), which is physically stored in a graph database to facilitate the retrieval of entities that match a suspicious pattern. For example, Figure 2(d) shows a portion of a real network of taxpayers and highlights a subset of entities that match the `SuppliesFromAssociated` pattern underlying Case 4 (the formal definition of graph pattern matching is given in Section 6).

In our approach the role of visualization is crucial and twofold. First, the complex task of coding suspicious patterns into the graph database native language is replaced by the use of a visual language, which can be intuitively adopted by tax officers who do not have specific knowledge about databases and query languages technicalities. For example, in order to search in the database those instances that match the `PurchaseFromRelated` pattern, the tax officers can simply “draw” the structure of Fig. 2(a), which is then automatically translated into the more complex query expressed in the database’s language (see. e.g., Fig. 3). Second, the results of a

```

MATCH (n1)-[e1]-(n2),(n1)-[e2]-(n2)
WHERE ((n2)-[e1]-(n1)) AND (type(e1)='TransazioneEconomica') AND
(type(e2)='Partecipazione') AND ((n1)-[e2]-(n2)) AND ((e1.ImponibileComDaCes >=
value1) OR (e1.ImponibileComDaCed >= value2)) AND NOT ID(n1) = ID(n2)
RETURN n1, labels(n1), ID(n1), e1, type(e1), ID(e1), ID(STARTNODE(e1)) as
STARTNODE_e1, ID(ENDNODE(e1)) as ENDNODE_e1, n2, labels(n2), ID(n2), e2, type(e2),
ID(e2), ID(STARTNODE(e2)) as STARTNODE_e2, ID(ENDNODE(e2)) as ENDNODE_e2 ORDER BY
id(n1) SKIP 0

```

Fig. 3. A query in Cypher (the query language of Neo4J) matching the PurchaseFromRelated pattern.

query can be visually displayed as diagrams, which can be subsequently used by tax officers to explore the taxpayers network and to acquire additional information about the involved elements. Looking at these diagrams and interacting with them results in a more effective and efficient process with respect to interpreting big tables of textual data, as witnessed by the results of our experiments and use cases presented in Sections 8 and 9.

4. Basic Definitions on Graphs and Networks

A *graph*, also called *network*, is a structure $G = (V, E)$, where V is a non-empty finite set of elements called the *nodes* or the *vertices* of G , and E is a finite set of elements called the *edges* of G . An edge $e \in E$ is a pair of nodes $u, v \in V$, which expresses a binary relationship between u and v . Nodes u and v are also called the *end-nodes* (or *end-vertices*) of e , and we write $e = (u, v)$; nodes u and v are said to be *adjacent* and edge e is said to be *incident* to both u and v . The *degree* of a node is the number of its incident edges. An edge $e = (u, v)$ is *directed* if the pair of its end-nodes is an ordered pair; in which case, u and v are also called the *source* and the *target* of e , respectively. Two or more edges of G having the same end-nodes are called *multiple edges*. An edge (u, v) such that u and v coincide is a *self-loop* of G . A node or an edge of a graph G may have multiple *attributes*, which define values of different types associated with that node or edge. When an attribute is a numeric value, it is also called a *weight* of the node or of the edge. A graph $G' = (V', E')$ is a *subgraph* of G if $V' \subseteq V$ and $E' \subseteq E$. A *path* in a graph G is an alternating sequence of nodes and edges $p = \langle v_1, e_1, v_2, e_2, \dots, v_k, e_k, v_{k+1} \rangle$, where k is a positive integer and $e_i = (v_i, v_{i+1})$ for $i = 1, \dots, k$; k is said to be the *length* of p . Path p is a

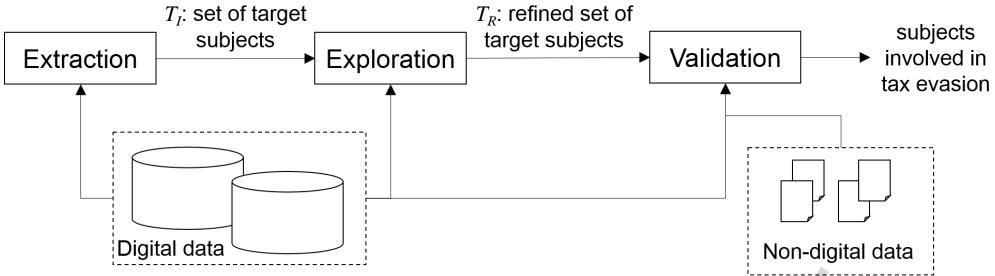


Fig. 4. Workflow of a fiscal audit process.

cycle if $v_1 = v_k$. G is *connected* if there exists a path between every pair of its nodes. A *tree* T is a connected graph with no cycle. A *rooted tree* is a tree T with a distinguished node v , called the *root* of T ; the *leaves* of T are the nodes of degree one distinct from the root; the nodes that are not leaves (including the root) are the *internal nodes* of T .

5. Workflow and Network Data Model

Crucial to the design of our visual analytics system is the modeling of the workflow of a fiscal audit process at the IRV, which is summarized as a pipeline of three phases (see Fig. 4):

- **Extraction.** Starting from a bulk of data stored in the IRV's database in a given time window (usually one year), an initial set of suspicious positions is extracted. This provides a set T_I of involved *target subjects* (either individuals or companies). During this phase, tax officers follow the guidelines of central and regional directorates and query the digital information system in search of suspicious patterns.

- **Exploration.** A deeper analysis of the data is performed and the initial set T_I of target subjects is refined: Some of the target subjects selected in the previous phase may be ruled out, while some new target subjects may be inserted in T_I if an involvement with other target subjects is discovered. This is done by exploring, for each subject $s \in T_I$, additional digital data associated with s and possible connections of s within or out T_I . The refined set T_R of target subjects is then inserted in the final *work plan*.

- **Validation.** A third level analysis is executed on all target subjects in T_R , in order to

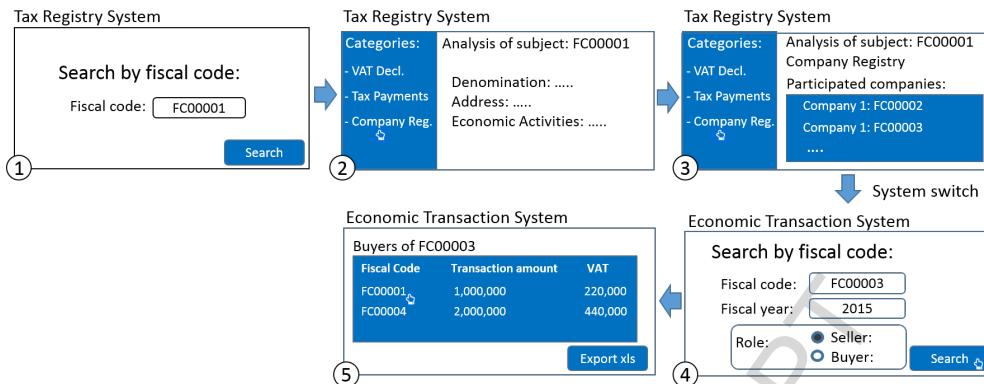


Fig. 5. Example of interaction with the traditional software at the IRV during the Exploration phase.

establish which one of them is actually involved in tax evasion. This is done by collecting and analyzing further data and documents other than those directly accessible through the digital information system of the IRV.

Currently, the Extraction and the Exploration phases at the IRV are executed using a collection of heterogeneous software systems (more than ten systems) that are accessible through a Web interface. These systems are taxpayer-centric querying oriented and the user has to switch from a system to another to retrieve different types of data related to the same taxpayer. This approach is time consuming, and it may cause mental-map loss and information redundancy. Furthermore, each system provides a restricted set of query functionalities and the output is returned in the form of textual data or spreadsheets, which makes it difficult to gather information about relationships between different taxpayers. From now on we refer to the current collection of software systems used at the IRV as the *traditional software*. Figure 5 shows an example of interaction with the traditional software during the Exploration phase, in order to verify if a target taxpayer is involved in a *PurchaseFromRelated* pattern (see Section 3). The user inserts the fiscal code of the taxpayer in a first system (“Tax Registry System”), and then selects the category “Company Registry” to get the list of companies participated by the taxpayer. After this, for each company in the list, the user switches to a different system (“Economic Transaction System”) to check if the target taxpayer is in the list of buyers of the participated company.

TAXNET is specifically designed to speed-up the activities of the Extraction phase and the Exploration phase, by overcoming the limits of the traditional software. It gives the IRV officers a set of graph mining and graph visualization facilities that help them to efficiently execute the related tasks and to increase the effectiveness (reliability) of the outputs. Unlike the Extraction and the Exploration phases, the Validation phase is mainly performed out of TAXNET.

Data Model. The data sources queried by the IRV officers are modeled in TAXNET as a unified network G . Each node v of G is a single taxpayer, which can be either an individual or a legal person, like a private company or a public institution. Many attributes are associated with v , including the type of economic activity, the geographic location and territorial scope, the declared income, the amount of VAT credits/debts and of VAT refunded/paid, and the amount of economic exchange within the European Union. The edges of G are directed edges. An edge (u, v) can model different types of relationships between u and v . From the economic point of view, consistently with the examples discussed in Section 3, the two main types of relationships considered in TAXNET are *economic transactions* and *shareholdings*. For an economic transaction (u, v) , the source node u is the seller and the target node v is the buyer: The two main attributes for such an edge are the transaction amounts declared by the two subjects in the considered time window (when they do not coincide a potential risk factor is highlighted). For a shareholding (u, v) , the source u is the shareholder and the target v is the participated company: The main attribute for this edge is the percentage of share. The data model of TAXNET also allows the addition of other types of relationships such as kinship and participation in legal acts.

6. The Visual Query Language of TAXNET

Tax evasion and tax fraud continuously evolve over time, thus generating schemes of increasing complexity. Hence, it is often ineffective to contrast them by relying on a fixed set of suspicious patterns. TAXNET gives the user the possibility of defining customized suspicious graph patterns (see, e.g., Section 3), through an intuitive visual language. Formally, a pattern P is specified by a pair $\langle G_P, R_P \rangle$, where $G_P = (V_P, E_P)$ is a graph that defines the topology of P ,

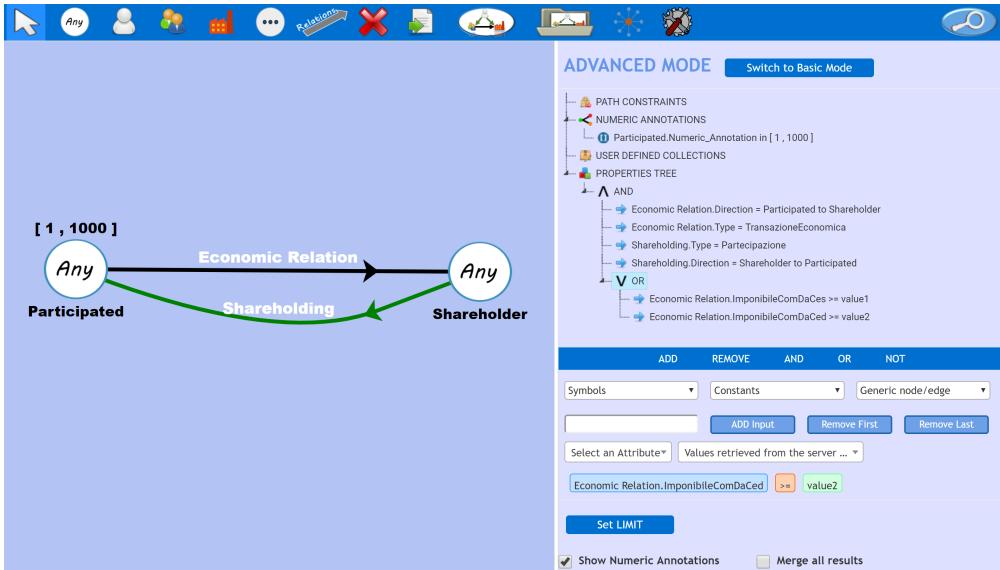


Fig. 6. Example of the graphical interface used to define a pattern.

and R_P is a set of rules on the nodes and the edges of G_P . An edge of E_P corresponds to a single edge of G or to a path whose length is within a desired range. This correspondence is established by a specific type of rules of R_P , which we call *path constraints*. Other types of rules in R_P are used to describe desired properties for node/edge attributes of G_P ; these properties can then be combined with logical operators AND, OR, NOT to form a tree, called the *properties tree*: The internal nodes correspond to logical operators and the leaves are atomic rules of R_P .

In the visual interface, the topology G_P is easily specified by means of a graph editing panel on the left side, while the properties tree is shown on the right side of the interface; see Fig. 6, where it is shown a pattern similar to the PurchaseFromRelated pattern discussed in Section 3. The graph editor supports multiple edges and self-loops, which are automatically drawn avoiding overlaps; a self-loop on a node v can be useful, for instance, to refer to a cycle that passes through v . The set of rules R_P is defined by interacting with the elements of G_P and with another panel, which we refer to as the *prop-def* panel (the bottom-right panel in Fig. 6). More in detail, each property can be defined as follows: (i) The user can see the list of attributes for a node/edge of G_P by simply clicking with the mouse right button on it; to this aim, when

the system starts, it automatically inspects the database and extracts from it the set of possible attributes for every type of node/edge. (ii) When the user selects an attribute, it is added to the prop-def panel. (iii) Attributes can be correlated to specific user-inputs or to constant values (the system automatically suggests the possible constant values for some attributes); they can also be combined together to form complex expressions, using a variety of operators, accessible from a list of predefined symbols in the graphical interface. These symbols consist of *comparison*, *inclusion*, and *mathematical operators*, including parenthesis to define association rules and arrays. It is also possible to associate an attribute value with a regular expression. During the definition of a pattern, the properties tree is kept consistent with the view of G_P : If a node/edge is removed in the graph editing panel, all rules involving it are automatically removed from the tree. The user can also act on a leaf of the tree to directly modify the corresponding rule.

Furthermore, in order to increase the power and the expressiveness of the visual language, and in order to make it more suitable for the specific application domain of tax evasion discovery, TAXNET supports two important features: *numeric annotations* for nodes/edges and the possibility of defining and sharing a collection of template patterns, called *pattern library*. These two features are described below.

Once a pattern P has been defined, the user can ask TAXNET to run its graph pattern matching procedure to retrieve from the network all subgraphs that match P , or up to $k > 0$ of these subgraphs for some desired k . TAXNET will automatically translate P into a pattern specification in Cypher, and it will collect the results returned by the graph pattern matching engine of Neo4J, based on graph traversals. In some cases, TAXNET has to execute specific procedures in addition to the Neo4J query execution, for example when numeric annotations are used in P (see below).

Numeric annotations. In the definition of a pattern $\langle G_P, R_P \rangle$, the visual query language of TAXNET allows the user to add on any element (node/edge) el of G_P a numeric annotation, i.e., a pair $[min, max]$, where min and max are two non-negative integers, with $min \leq max$. These values specify the minimum and maximum number of occurrences of el in a subgraph that matches P . Default values are $min = max = 1$. If $min = 0$ then el is optional; if $max = n$

then the maximum number of occurrences of el can be arbitrarily high. If $\min = \max = 0$, then an element like el is *not* allowed in a subgraph that matches the pattern, and we call el a *negated element*. Numeric annotations are a powerful tool, which gives high flexibility. In the specific application domain of TAXNET, numeric annotations can be used to aggregate multiple companies participated by the same subject or to include a person in a suspicious pattern only if she is connected to at least a certain number of desired types of subjects. For example, in the MissingTraderSuppliers pattern examined in Section 3, one can add a numeric annotation rule to aggregate multiple buyers of the missing trader (like the national companies C3 in Case 2 or the companies C7 that act as a “buffer” in Case 3). However, there are some restrictions on the use of numeric annotations, and, since Cypher does not provide a direct support to efficiently handle numeric annotations, we implemented ad-hoc procedures that must be combined with the results of one or more Cypher queries. For example, suppose that a node v of G_P has a numeric annotation $[0, \max]$, with $\max > 0$. First, our system generates a Cypher query corresponding to P , but where v is removed from G_P ; denote by S the set of results (subgraphs) returned by Neo4J. Then, a second Cypher query is generated, where v appears exactly once; let S' be the set of results returned by Neo4J. At this point, for each result $G \in S$, TAXNET determines the subset $\{G'_1, G'_2, \dots, G'_k\} \subseteq S'$ of results such that $G'_i \setminus G$ is just an occurrence of v , and generates a final set of distinct results, each obtained from G by adding at most \max occurrences of v taken from $\{G'_1, G'_2, \dots, G'_k\}$. As another example, if e is an edge with annotation $[0, 0]$, a first superset S of results is obtained by querying Neo4J for a pattern that ignores e ; then, for each $G \in S$, a simple query is applied to check whether e exists in the graph: In the affirmative case, G is discarded, otherwise it is kept as a final result.

Pattern library. Depending on the specific economic context and on the considered countries, some types of tax evasion patterns are more frequent than others. In these cases, it is worth having the possibility of defining abstract patterns like those introduced in the Section 3, which can be easily reused and completed with the values of the specific attributes and/or constraints involved in the rules (i.e., expressed by the properties tree). The idea is to speed up the defini-

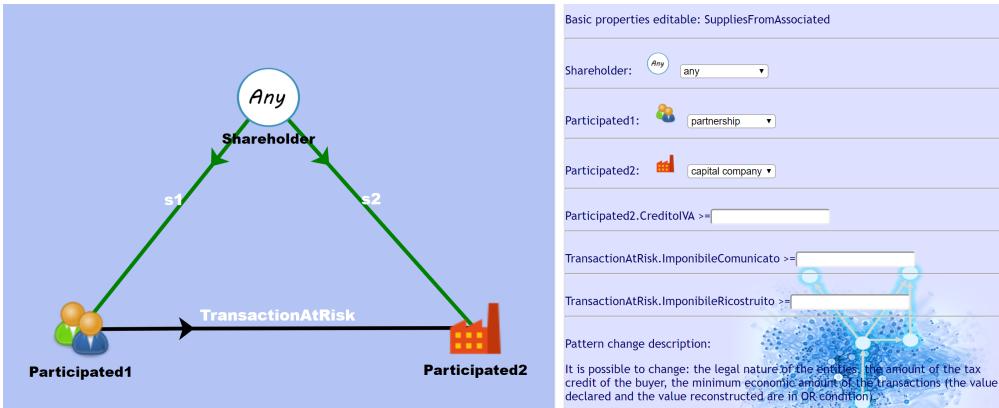


Fig. 7. Simplified interface used to complete the definition of a library pattern.

tion of suspicious patterns, and at the same time to give tax officers the possibility of sharing a library of patterns based on their experience and on official guidelines. TAXNET offers such a possibility. Once a pattern $P = \langle G_P, R_P \rangle$ has been defined, the user can decide to include it in a central library managed by the system. This operation requires to specify which values among those that form the whole set of rules can be modified and whether a default value is applied for some of them. When a user accesses a pattern P in the central library, she can decide to interact with a *simplified interface* in which the topology G_P cannot be changed. The list of all modifiable values is shown to the user, so that she can easily complete the specification of P before executing the query. Figure 7 shows the simplified interface used to complete the definition of a *SuppliesFromAssociated* pattern (see Section 3). The labels for nodes and edges are decided by the user who defines and inserts the pattern in the library. By default, each node in the pattern is of type *Any*, i.e., it can be either a person or a company; the user can change its value.

7. TAXNET: Network Analysis and Exploration

TAXNET provides two main tools, explained in the following, to analyze and explore the results returned by the pattern matching algorithms: (i) An automatic procedure to compute different *risk indexes* for the nodes (taxpayers) of the network; (ii) a user interface to visualize, merge, and explore the subgraphs that result from a specific query. These tools can be used in

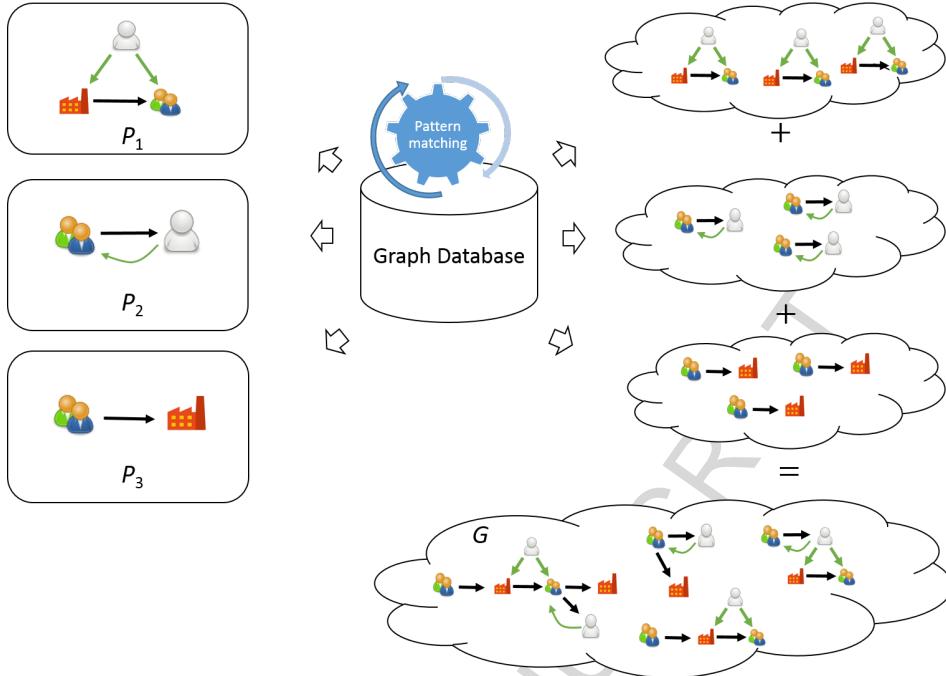


Fig. 8. Schematic illustration of the process that computes the risk indexes.

combination; for example a specific type of risk index can be used to order (rank) the subgraphs that match a specific suspicious pattern.

Risk indexes. These indexes express a fiscal risk for each taxpayer in the graph database. They measure the centrality of a taxpayer in the network structure, with respect to a desired set of suspicious patterns. The user can select a desired subset $\mathcal{P} = \{P_1, P_2, \dots, P_k\}$ of library suspicious patterns (potentially all those defined in the pattern library), assign each pattern P_i a *relevance weight* $\alpha(P_i)$, and ask the system to associate risk indexes with the nodes of the network, based on \mathcal{P} and α .

To this aim, TAXNET works as follows. Using the pattern matching engine, for each $i = 1, \dots, k$, it retrieves from the database all subgraphs $G_1^{(i)}, G_2^{(i)}, \dots, G_{h_i}^{(i)}$ that match P_i , and merges all these subgraphs in a unique network G ; see Fig. 8. Each edge e of G is then assigned a weight $p(e)$ equals to the number of graphs $G_j^{(i)}$ in which e is present, weighted according to $\alpha(P_i)$. Also, each edge e that models an economic transaction of (maximum) value $w(e)$ is assigned a second

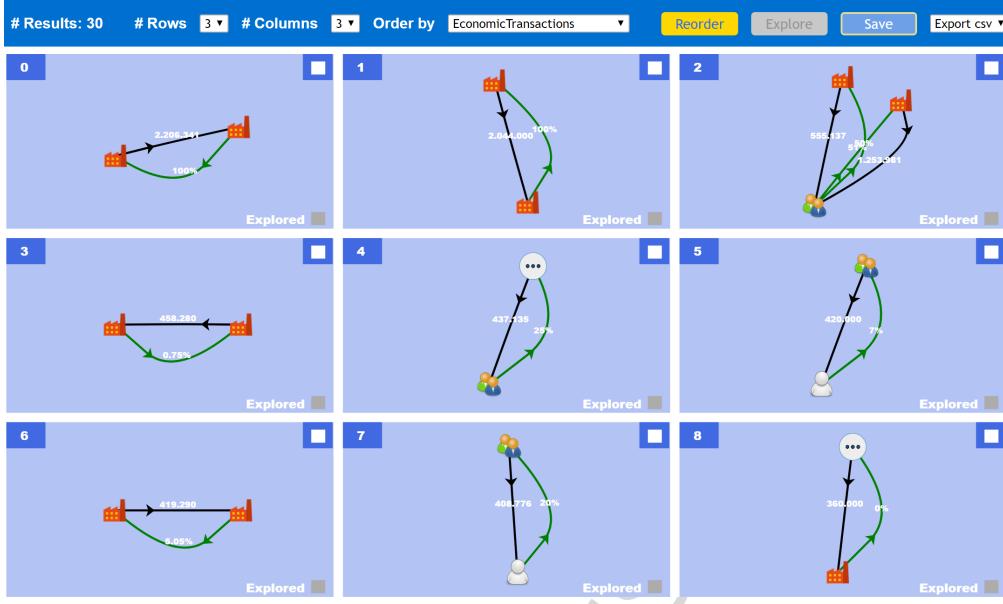


Fig. 9. Visualization of the subgraphs that match a given pattern.

weight $t(e)$, obtained by summing up $w(e)$ over all $G_j^{(i)}$ that contain e , still weighted according to $\alpha(P_i)$. More formally:

$$p(e) = \sum_{i=1}^k \sum_{j=1}^{h_i} \alpha(P_i) \chi_e(G_j^{(i)}) \quad t(e) = \sum_{i=1}^k \sum_{j=1}^{h_i} \alpha(P_i) w(e) \chi_e(G_j^{(i)}), \quad (1)$$

where $\chi_e(G_j^{(i)}) = 1$ if $e \in G_j^{(i)}$ and $\chi_e(G_j^{(i)}) = 0$ otherwise. Finally, each node u of G has two risk indexes, the *pattern centrality* and *transaction-pattern centrality* of u , denoted as $pdc(u)$ and $tpdc(u)$, respectively, and defined as:

$$pdc(u) = \sum_{e \in E(u)} p(e) \quad tpdc(u) = \sum_{e \in E(u)} t(e), \quad (2)$$

where $E(u)$ is the set of edges incident to u in G . For each of the two indexes, we also define the variants $pdc_{in}(u)$, $tpdc_{in}(u)$, restricted to the incoming edges of u , and $pdc_{out}(u)$, $tpdc_{out}(u)$, restricted to the outgoing edges of u .

Visual Exploration. When a user queries TAXNET for a specific pattern P , a set of subgraphs G_1, G_2, \dots, G_h is returned, ranked, and visually displayed by the system. Different ranking

criteria can be applied, based on node/edge attributes. The ranking can be based either on the total amount of the economic transactions in each G_i or on one of the risk indexes defined above. For example, the user can rank the results according to the maximum value or to the average value of $pdc(u)$, over all nodes u in each G_i . Once the ranking has been defined, all subgraphs G_i are automatically drawn and arranged in a matrix, as in Fig. 9. The user can customize the dimensions of the matrix, so to magnify the visualization of each subgraph. The graphs are drawn with the force-directed algorithm in the D3.js library Bostock et al. 2011, using a post-processing ad-hoc step to avoid overlap of multiple edges.

The user can also decide to merge some of the subgraphs together, in order to aggregate in a unique view elements that appear in multiple graphs. The system implements a heuristic algorithm to find maximal subsets of subgraphs that can be merged together without creating conflicts. For example, suppose that a query for the `SuppliesFromAssociated` pattern produces two subgraphs that share the identity of the participated companies, while the owner is different, as in Fig. 10(a). Merging these subgraphs will result in the graph of Fig. 10(b). Note however that merging subgraphs may create ambiguous situations in some cases. Namely, suppose that G_1 and G_2 are two subgraphs that match a pattern P and that share a node u . If u plays different roles with respect to P in G_1 and G_2 , then merging the two subgraphs makes it impossible to consistently assign u a unique role in the new graph. In these cases, TAXNET automatically notifies a conflict situation, so that the user can decide whether to go ahead with the merge operation or not. The system also implements an algorithm to automatically find maximal subsets of subgraphs that can be merged together without creating conflicts. Such an algorithm first constructs a conflict graph C where the nodes are all the subgraphs matching the desired pattern and where two nodes are connected if and only if merging their corresponding subgraphs causes a conflict. Then, it applies the $(\Delta + 2)/3$ -approximation greedy algorithm described by Halldórsson & Radhakrishnan 1997 (where Δ is the maximum node-degree of C) to iteratively detect maximal independent sets of nodes, corresponding to subgraphs that can be merged without conflicts.

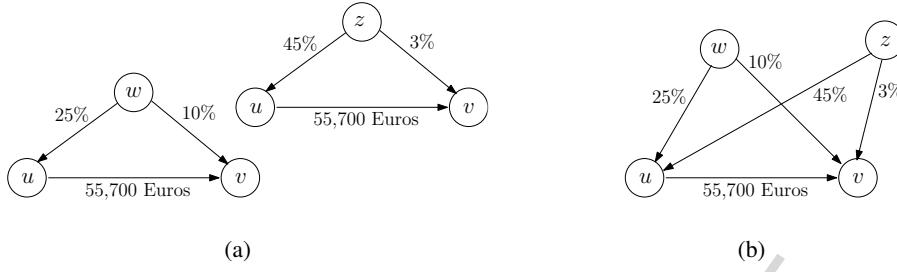


Fig. 10. (a) Subgraphs that match the same pattern. (b) Merging the subgraphs.

By interacting with the matrix of results, the user can select one of them (possibly corresponding to merged subgraphs). In this case a new panel is opened where the user can explore the graph by iteratively expanding the neighbors of the displayed nodes; see Fig. 11(a). Different interaction tools are provided. Among them, one can decide which kind of attributes must be shown for each type of element (node/edge) and it is possible to filter out elements based on the values of some attributes. This is crucial to keep the visual complexity low or to highlight some specific information. For instance, Fig. 11(b) shows a network obtained from that in Fig. 11(a) by keeping only the economic transactions above 220,000 EUR. Edges involved in some patterns are marked with specific labels, and it is possible to filter out all the edges that come from one or more patterns. Also, the user can enter a comment for each visualization, which will be used to automatically generate a report of the analysis if required.

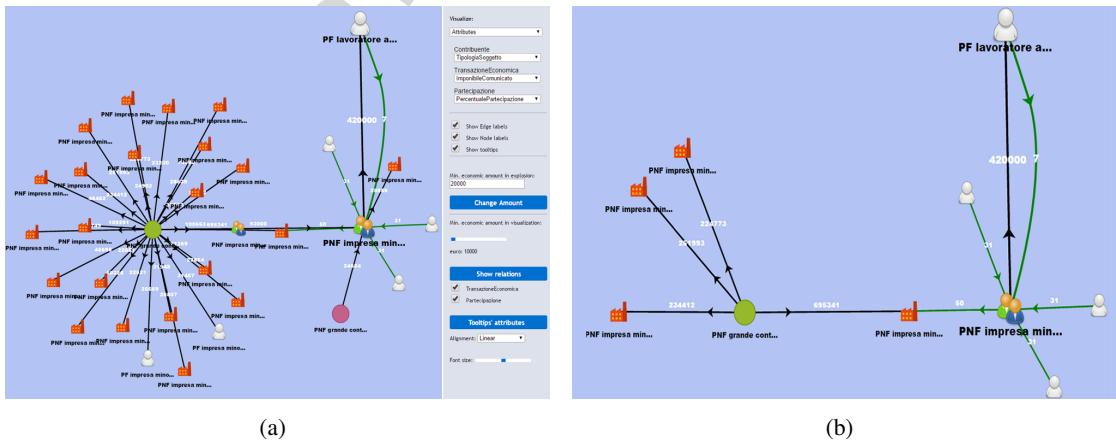


Fig. 11. (a) Exploring a result. (b) Filtered network.

7.1. Running time

We conclude this section by giving some data about the running time of TAXNET. We recall that graph pattern matching is in general an NP-hard problem, as it is related to subgraph isomorphism (see Cook 1971). However, when the topology of a graph pattern is relatively simple and when such a topology is complemented by some attribute rules (which may significantly restrict the scope of the search), retrieving those subgraphs that match the desired pattern can result, in practice, in a fast process with Neo4J. In particular, on the suspicious patterns most frequently used at the IRV, TAXNET exhibits quick response times, both for the execution of the pattern matching algorithm and for ranking and presenting the results according to the matrix arrangement described in the previous section. To have an idea of the practical computational complexity required by TAXNET, Table 1 reports the running times needed to retrieve and present all subgraphs matching the suspicious patterns `PurchaseFromRelated` and `SuppliesFromAssociated` (already described in Section 3) on a suitable dataset of real networks. Namely, we considered networks of increasing size. The largest network is a real network with 670,812 nodes (taxpayers) and 1,819,263 edges (relationships), concerned with data of the Tuscany region. The other networks are subgraphs of the largest one, which consider data of progressively reduced geographic areas. For each network: $|V|$ and $|E|$ denote the number of nodes and edges, respectively; the number of subgraphs that match the query is reported; T_Q and T_P denote the query execution time and the presentation time, respectively, expressed in seconds. Both the server and the client side of TAXNET ran on a machine having an Intel I5 - 2.5GHz processor and 8GB of RAM. The table also reports the number of subgraphs extracted in response to the query. It can be seen that the time needed for executing the query scales very well with the size of the network; it takes less than ten seconds, even for the largest network. We remark that the query execution time includes the time needed by TAXNET to translate the pattern specification from its visual query language to the Cypher language and the time spent for the pattern matching phase. The presentation time is higher than the query execution time, but it is still reasonable; it ranges from a few seconds to about one minute depending on the size of the

		PurchaseFromRelated			SuppliesFromAssociated		
$ V $	$ E $	Subgraphs	T_Q	T_P	Subgraphs	T_Q	T_P
75,724	123,704	85	0.685	1.480	64	1.043	1.495
290,434	530,444	401	2.013	5.471	318	2.113	5.585
411,329	829,868	661	2.853	10.527	653	5.665	16.179
670,812	1,819,263	1,488	7.568	38.098	1,565	9.213	67.758

Table 1

Running times taken by TAXNET for query execution and result presentation on two suspicious patterns.

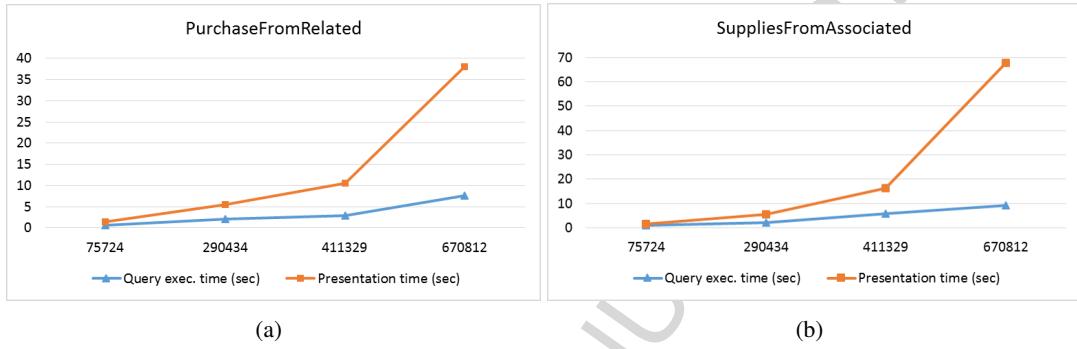


Fig. 12. Running times of TAXNET for query execution and result presentation on two common suspicious patterns.

network. We observe that the presentation time includes the time needed to rank the extracted subgraphs based on the risk indexes plus the time needed to compute and rendering the drawings of these subgraphs. Hence, the presentation time strongly depends on the number of extracted subgraphs, other than on the type of pattern. A plot of the running times is also shown in Fig. 12; the x -axis reports the size of the network in terms of number of vertices, while the y -axis reports the running time in seconds.

8. Experimental Study with TAXNET

TAXNET is currently adopted by the Italian Revenue Agency (IRV), under a pilot project of the provincial office of Arezzo (Tuscany). In order to assess the effectiveness of our system on real data and in a real working environment, we performed a user study that involves expert tax officers of the IRV. The user study is aimed to estimate the usability of TAXNET and its impact on the execution of fundamental analysis tasks performed during the Exploration phase. The

impact is measured both in terms of time needed to execute a task and in terms of accuracy of the corresponding results.

Overview. Our goal is to study the following research question: *Are tax investigations conducted with TAXNET more efficient and more accurate than those conducted with the current software?* To answer this question we asked the participants to execute two types of tasks: (i) Find relevant subjects related to a given taxpayer; (ii) Find suspicious fiscal relationships of a given taxpayer. In the following, we give details about the different aspects of our experiment.

Participants. We recruited 32 participants, with age ranging from 34 to 65 (the average age is 47), equally distributed from males and females. Every participant had an experience of at least 3 years in tax evasion discovery and currently works at the IRV's fiscal audit office.

Design. We used a between-participants design with two different environments (conditions) to execute tasks: 1) TAXNET and 2) traditional software used by the IRV (NO TAXNET). The 32 participants were randomly allocated to one of the two environments, so to create two environment-groups, each composed of 16 participants. The way in which the user interacts with the traditional software has been already clarified in Section 5. We remark that the digital data available in TAXNET and in the traditional software for the experiments were the same, although the two environments significantly differ from the conceptual data model and the interaction perspectives. The user study consisted of two experiments, each having one task and four instances per task. The two experiments are as follows:

Experiment 1. The hypothesis for this experiment was: *Participants using TAXNET spend less time and achieve more accurate results to find relevant subjects related to a given taxpayer, than participants using traditional software.* The task used to check this hypothesis was: *For a given taxpayer t , find the fiscal code of: A) the suppliers of t for over than the amount X ; B) the shareholders of t ; C) other companies owned by the subjects referred to in point B).*

Experiment 2. The hypothesis for this experiment was: *Participants using TAXNET spend less time and achieve more accurate results to find suspicious fiscal relationships of a given taxpayer, than participants using traditional software.* The task used to check this hypothesis

was: *For a given taxpayer t , find the fiscal code of:* 1) *the companies participated by t that are suppliers of t for more than the specified annual amount X ;* 2) *the subjects that do not declare revenues and that are suppliers of t for more than the specified annual amount Y ;* 3) *the subjects that omit VAT payment for more than the amount Z and that are suppliers of t for more than the annual amount Y ;* 4) *the European service providers of t for more than the annual amount Z .*

Procedure. Since the types of tasks were relatively complex, we decided to conduct both the experiments on a one-to-one basis, with the simultaneous presence of a single participant (user) and the experimenter. This method ensures that the participants fully understand the tasks and hence the collected data are not affected by user's misunderstandings. Before starting the experiment, the participant was informed about the data set to explore. Also, if the participant had to use TAXNET then: (i) she was informed about the network model adopted by the system; (ii) we provided the participant with a full overview of TAXNET and with a tutorial session to ensure that she understood the necessary concepts. We also gave the participant a few training examples and indications of the correct answers. We remark that, the analysts at the IRV have to follow a specific fiscal analysis training course of about 14 hours per year to effectively use the combination of the different heterogeneous interfaces that compose the traditional software. In contrast, the overview and tutorial session for the use of TAXNET took about one hour. The participant was invited to pose clarifying questions during these introductory phase. Finally, we specified to the participant that she had unlimited time to complete the given tasks, but at the same time she was encouraged to work quickly and accurately. For each participant, Experiment 1 was executed before Experiment 2. For each instance, the participant received as input the fiscal code of a taxpayer and she was asked to provide the fiscal codes of the related subjects, according to the requested criteria; for each fiscal code in the output, the participant specified the corresponding matching criterion through a checkbox. We observe that, in the usual analysis flow at the IRV, tasks like those executed in Experiment 1 actually precede the more complex tasks executed in Experiment 2. The experience acquired by the analysts during Experiment 1 may increase the efficiency in executing Experiment 2; however, since this reflects

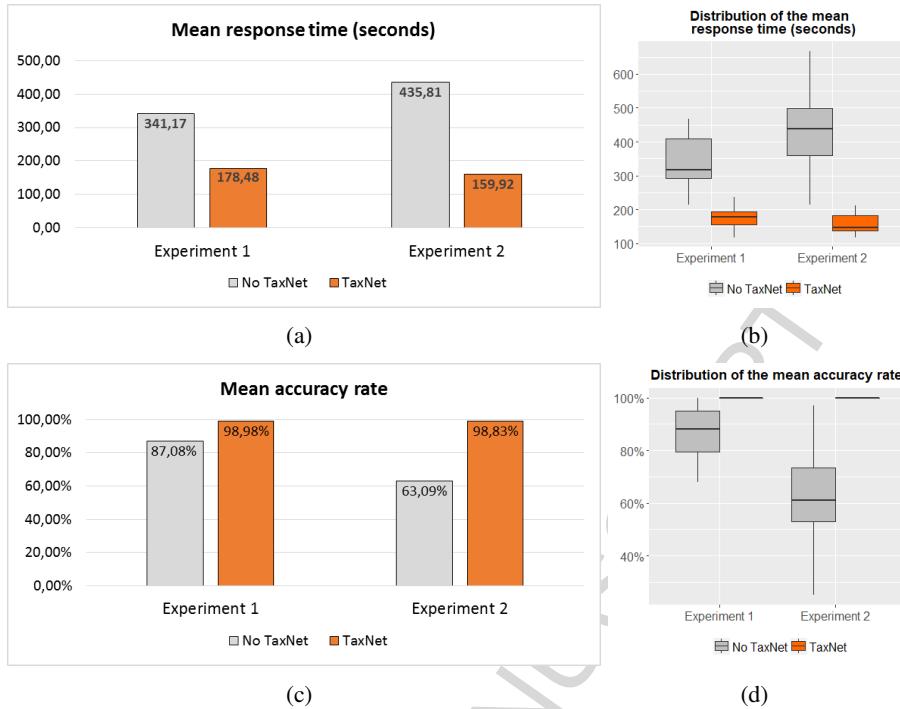


Fig. 13. (a)-(c) Mean response time and accuracy rate. (b)-(d) Distribution of the response time and accuracy rate.

the typical real analysis flow, both groups of users (those using TAXNET and those using the traditional software) executed the two experiments in this order, which also guarantees fairness in the comparison.

Results. During the experiments, for each instance and for each participant, we collected the *response time* in seconds and the *accuracy rate*. The accuracy rate is given by the ratio between the number of subjects correctly detected by the participant and the total number of related subjects according to the required criteria². For each of the two measures, Fig. 13(a) and Fig. 13(c) report the mean values in each experiment, over all participants and task instances of each environment. The results suggest that using TAXNET has a relevant positive impact for the two experiments, both in terms of time response and in terms of accuracy rate. The participants

²We decided not to count as a penalty a wrong subject reported by a participant for a given criterion. Indeed, the total number of wrong subjects is negligible for the TAXNET environment (just 2 wrong subjects over all users and task instances) and relatively small for the NO TAXNET environment (24 subjects over all users and task instances).

who used TAXNET were faster than the other participants: In Experiment 1, they used in the average only the 52% of the time spent by the participants that adopted traditional software; the improvement is even more evident in Experiment 2, where the time spent using TAXNET is only 37% of the time spent without TAXNET. More in detail, the use of TAXNET, compared with the use of traditional software, saved for each task instance a mean absolute time of about 2.7 minutes in Experiment 1 and of about 4.6 minutes in Experiment 2; therefore, for each instance, the total absolute time saved for the two experiments is of about 7.3 minutes. Based on statistical data reported by the IRV, the number of taxpayers corresponding to companies and self-employed is of about 6,000,000 of instances. Hence, the experiments indicate that a hypothetical massive monitoring of all these types of instances using TAXNET would allow the IRV to save more than 90,000 man days per year, only in the Exploration phase of the fiscal audit process.

About the accuracy, the participants who used TAXNET improved the results of the other participants of about 13.7% in Experiment 1 and of about 56.6% in Experiment 2. In particular, the high improvement of the second experiment witnesses how much TAXNET may help to monitor multiple risk factors simultaneously. The boxplots in Fig. 13(b) and Fig. 13(d) report the distributions of the mean response time and of the mean accuracy rate for each user, over all the executed tasks. From the charts it is possible to see that the values obtained with TAXNET are close to the median value; in particular, the accuracy rate for TAXNET is very close to 100% (in Fig. 13(d), the corresponding box of values is collapsed on the median value).

To evaluate if the data are statistically significant, we used the nonparametric Mann-Whitney's U test, as the data did not follow a normal distribution. We obtained p -values much smaller than 0.01, which indicate that the data are statistically significant with high probability. More precisely, we have the following p -values (we also report the U values): Experiment 1. $p = 2.329E - 08$ and $U(16, 16) = 3$ for the response time; $p = 7.086E - 05$ and $U(16, 16) = 34.5$ for the accuracy rate; Experiment 2. $p = 3.327E - 09$ and $U(16, 16) = 3$ for the response time; $p = 2.329E - 08$ $U(16, 16) = 0$ for the accuracy rate.

9. Use Cases with TAXNET

To further assess the validity of TAXNET, we collected and analyzed the results of two use cases on real data sets, handled at the IRV with our system, and we compared them with results on the same data handled with the traditional software.

Use Case 1. In this use case, the IRV officers used TAXNET for the Extraction phase. They defined seven typical suspicious patterns through the visual language of the system, and stored them in the central pattern library. Then they took past data referring to a portion of tax declarations for the year 2011, for which they already performed a fiscal audit process with the traditional software, and asked TAXNET to compute the risk indexes for the nodes of the corresponding taxpayers network, with respect to all seven patterns. They finally ranked in a list T_I all subjects in the subgraphs that matched some patterns, according to decreasing values of index $tpdc_{in}$. From a comparison with previous data, they observed that about 45% of the total amount of tax evasion successively payed by subjects in T_I comes from the first quartile of T_I . The median of this amount is higher of about 93% than the value computed on all fiscal audits of the same type performed by the office for the year 2011. This indicates that TAXNET is effective to model suspicious patterns and to rank the results based on its risk indexes.

Use Case 2. In this use case, the tax officers of IRV utilized TAXNET for a fiscal audit process on new data concerned with a portion of tax declarations of the year 2013. They performed both phase Extraction, from which a first list T_I of 262 target subjects raised, and phase Exploration, which reduced T_I to a list T_R of 141 subjects. These subjects have been inserted in the work plan for the Validation phase. The ratio between the size of the list at the end of phase Extraction and the size at the end of phase Exploration is similar to recent fiscal audit processes. However, a comparison with the time required to execute the two phases in previous audits without TAXNET revealed for the new job a speed-up of 134% (on average, 1.33 subjects per hour were inserted in T_R , against 0.57 with the traditional software). This time improvement is a further confirmation of the results observed in the experiment of Section 8.

10. Conclusions and Future Work

We presented TAXNET, a decision support systems for tax evasion discovery, developed in cooperation with the IRV (Italian Revenue Agency). TAXNET makes use of a powerful and flexible visual query language for graph databases, which allows users to rapidly define and extract suspicious patterns in a suitably defined network of taxpayers. It also implements new risk centrality indexes to automatically rank the results that match a specific query and visual tools to present and further explore these results and other possible related elements of the network. The system is currently used at the IRV in the region of Tuscany. The results of a user experiment and those emerging from two use cases performed at the IRV on real data indicate that TAXNET may have a strong impact in the fiscal audit process, both in terms of efficiency (saved time) and in terms of accuracy and reliability of the suspicious situations discovered.

There are several research directions that we plan to further investigate in the near future. Among them: (i) The assessment of TAXNET has been done on real data and in a real working environment. In the long-term plan, we will extend the experiments to a larger geographic area in Italy and will collect and analyze data that can give additional insights on the effectiveness of the system. For example, it would be interesting to evaluate it in terms of the economic impact directly related to tax recovery. (ii) Following the indications of the Council of the European Union, 20 May 2016, 9046/16 (FISC 77 ECOFIN 404), we plan to explore a possible cooperation with other countries in order to extend our methodologies and technologies to a broader context. (iii) Combining machine learning techniques with our current visual query language tools could further increase the capability of the system to support analysts in tax evasion discovery. For example, training data for a machine learning technique could come from the Validation phase, which may confirm or not the validity of a suspicious position.

Acknowledgements

This work is in cooperation with the Italian Revenue Agency (IRV). We thank the director Mario Landolfi and the staff of the IRV provincial office of Arezzo. We acknowledge Carlo

Palumbo and Giuseppe De Luca of the IRV, regional directorate of Tuscany, for their support and suggestions. We also thank the anonymous reviewers for their valuable comments. Funding: Work partially supported by the project: “Algoritmi e sistemi di analisi visuale di reti complesse e di grandi dimensioni - Ricerca di Base 2017, Dep. of Engineering, University of Perugia”.

ACCEPTED MANUSCRIPT

References

- Argyriou, E. N., Symvonis, A., & Vassiliou, V. (2014). A fraud detection visualization system utilizing radial drawings and heat-maps. In R. S. Laramee, A. Kerren, & J. Braz (Eds.), *IVAPP 2014* (pp. 153–160). SciTePress.
- Bank, E. C. (2013). Economic and monetary development, monthly bullettin.
- Basole, R. C., Huhtamäki, J., Still, K., & Russell, M. G. (2016). Visual decision support for business ecosystem analysis. *Expert Systems with Applications*, 65, 271–282. doi:10.1016/j.eswa.2016.08.041.
- Basta, S., Fassetti, F., Guarascio, M., Manco, G., Giannotti, F., Pedreschi, D., Spinsanti, L., Papi, G., & Pisani, S. (2009). High quality true-positive prediction for fiscal fraud detection. In Y. Saygin, J. X. Yu, H. Kargupta, W. Wang, S. Ranka, P. S. Yu, & X. Wu (Eds.), *IEEE ICDM 2009* (pp. 7–12).
- Blau, H., Immerman, N., & Jensen, D. (2002). A visual language for querying and updating graphs. *University of Massachusetts Amherst Computer Science Technical Report*, 37, 2002.
- Bostock, M., Ogievetsky, V., & Heer, J. (2011). D³ Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17, 2301–2309. doi:10.1109/TVCG.2011.185.
- Carneiro, N., Figueira, G., & Costa, M. (2017). A data mining based system for credit-card fraud detection in e-tail. *Decision Support Systems*, 95, 91–101. doi:10.1016/j.dss.2017.01.002.
- Chau, D. H., Faloutsos, C., Tong, H., Hong, J. I., Gallagher, B., & Eliassi-Rad, T. (2008). GRAPHITE: A visual query system for large graphs. In *ICDM 2008* (pp. 963–966). IEEE.
- Colladon, A. F., & Remondi, E. (2017). Using social network analysis to prevent money laundering. *Expert Systems with Applications*, 67, 49–58. doi:10.1016/j.eswa.2016.09.029.
- Cook, S. A. (1971). The complexity of theorem-proving procedures. In M. A. Harrison, R. B. Banerji, & J. D. Ullman (Eds.), *3rd ACM Symposium on Theory of Computing* (p. 151158).

Council of the European Union (2016). 9046/16. vat action plan "towards a single eu vat area".

Didimo, W., Giacchè, F., & Montecchiani, F. (2015). Kojaph: Visual definition and exploration of patterns in graph databases. In E. Di Giacomo, & A. Lubiw (Eds.), *GD 2015* (pp. 272–278). Springer volume 9411 of *LNCS*.

Didimo, W., Liotta, G., & Montecchiani, F. (2014). Network visualization for financial crime detection. *Journal of Visual Languages and Computing*, 25, 433–451. doi:10.1016/j.jvlc.2014.01.002.

Drezewski, R., Sepielak, J., & Filipkowski, W. (2015). The application of social network analysis algorithms in a system supporting money laundering detection. *Information Sciences*, 295, 18–32. doi:10.1016/j.ins.2014.10.015.

Flood, M. D., Lemieux, V. L., Varga, M., & Wong, B. L. W. (2015). *The Application of Visual Analytics to Financial Stability Monitoring*. Technical Report Office of Financial Research, USA.

González, P. C., & Velásquez, J. D. (2013). Characterization and detection of taxpayers with false invoices using data mining techniques. *Expert Systems with Applications*, 40, 1427–1436. doi:10.1016/j.eswa.2012.08.051.

Goumagias, N. D., Hristu-Varsakelis, D., & Saraidaris, A. (2012). A decision support model for tax revenue collection in Greece. *Decision Support Systems*, 53, 76–96. doi:10.1016/j.dss.2011.12.006.

Halldórsson, M. M., & Radhakrishnan, J. (1997). Greed is good: Approximating independent sets in sparse and bounded-degree graphs. *Algorithmica*, 18, 145–163. doi:10.1007/BF02523693.

Huang, M. L., Liang, J., & Nguyen, Q. V. (2009). A visualization approach for frauds detection in financial market. In E. Banissi, L. J. Stuart, T. G. Wyeld, M. Jern, G. L. Andrienko,

- N. Memon, R. Alhajj, R. A. Burkhard, G. G. Grinstein, D. P. Groth, A. Ursyn, J. Johansson, C. Forsell, U. Cvek, M. Trutschl, F. T. Marchese, C. Maple, A. J. Cowell, & A. V. Moere (Eds.), *IV 2009* (pp. 197–202). IEEE.
- Koenig, P., Zaidi, F., & Archambault, D. (2010). Interactive searching and visualization of patterns in attributed graphs. In D. Mould, & S. Noël (Eds.), *GI 2010* (pp. 113–120).
- Matos, T., de Macêdo, J. A. F., & Monteiro, J. M. (2015). An empirical method for discovering tax fraudsters: A real case study of brazilian fiscal evasion. In B. C. Desai, & M. Toyama (Eds.), *IDEAS 2015* (pp. 41–48). ACM.
- Michalak, K., & Korczak, J. J. (2011). Graph mining approach to suspicious transaction detection. In *FedCSIS 2011* (pp. 69–75).
- Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*, 50, 559–569. doi:10.1016/j.dss.2010.08.006.
- Pienta, R., Hohman, F., Tamersoy, A., Endert, A., Navathe, S. B., Tong, H., & Chau, D. H. (2017). Visual graph query construction and refinement. In S. Salihoglu, W. Zhou, R. Chirkova, J. Yang, & D. Suciu (Eds.), *ACM SIGMOD 2017* (pp. 1587–1590).
- Pienta, R., Tamersoy, A., Endert, A., Navathe, S. B., Tong, H., & Chau, D. H. (2016). VISAGE: interactive visual graph querying. In P. Buono, R. Lanzilotti, M. Matera, & M. F. Costabile (Eds.), *AVI 2016* (pp. 272–279).
- Ravisankar, P., Ravi, V., Rao, G. R., & Bose, I. (2011). Detection of financial statement fraud and feature selection using data mining techniques. *Decision Support Systems*, 50, 491–500. doi:10.1016/j.dss.2010.11.006.
- Tian, F., Lan, T., Chao, K., Godwin, N., Zheng, Q., Shah, N., & Zhang, F. (2016). Mining suspi-

cious tax evasion groups in big data. *IEEE Transactions on Knowledge and Data Engineering*, 28, 2651–2664. doi:10.1109/TKDE.2016.2571686.

Vlasselaer, V. V., Eliassi-Rad, T., Akoglu, L., Snoeck, M., & Baesens, B. (2017). Gotcha! network-based fraud detection for social security fraud. *Management Science*, 63, 3090–3110. doi:10.1287/mnsc.2016.2489.

Wu, R., Ou, C., Lin, H., Chang, S., & Yen, D. C. (2012). Using data mining technique to enhance tax evasion detection performance. *Expert Systems with Applications*, 39, 8769–8777. doi:10.1016/j.eswa.2012.01.204.

Xiang, Y., Chau, M., Atabakhsh, H., & Chen, H. (2005). Visualizing criminal relationships: comparison of a hyperbolic tree and a hierarchical list. *Decision Support Systems*, 41, 69–83. doi:10.1016/j.dss.2004.02.006.

Zhou, W., & Kapoor, G. (2011). Detecting evolutionary financial statement fraud. *Decision Support Systems*, 50, 570–575. doi:10.1016/j.dss.2010.08.007.

Biographical Note

Walter Didimo received the PhD degree in computer science from the University of Rome "La Sapienza" in 2000. He is currently an associate professor of computer science in the Department of Computer Engineering at the University of Perugia. His research interests include graph drawing, information visualization, algorithm engineering, and computational geometry. He collected more than 100 international publications in the above areas. He served and chaired organizing and program committees of international conferences, including the International Symposium on Graph Drawing, the ACM Symposium on Computational Geometry, and the Pacific Visualization Symposium.

Luca Giamminonni is a Computer Engineer and he received his master degree cum laude at the University of Perugia. The topic of his master thesis was the design and development of a visual query language for graph databases.

Giuseppe Liotta received a PhD in computer science from the University of Rome "La Sapienza" in 1995 and is currently a professor in the Department of Engineering at the University of Perugia. His research interests include information visualization, graph drawing, and computational geometry. On these topics, he published more than 200 papers and gave invited lectures worldwide. He served and chaired program committees of international symposiums and has served in the editorial board of international journals. His research has been founded by the Italian National Research Council, by the Italian Ministry of Research and Education, by the EU, and by industrial sponsors.

Fabrizio Montecchiani received a PhD in Information Engineering at the University of Perugia in 2014 and is currently a Postdoc fellow at the University of Perugia, Engineering Department. His main research interests lie within the fields of graph algorithms and graph drawing, computational geometry, information visualization and visual analytics, algorithm engineering and system development. He collected more than 50 international publications in the above areas. He served in organizing and program committees of international conferences, including the International Symposium on Graph Drawing.

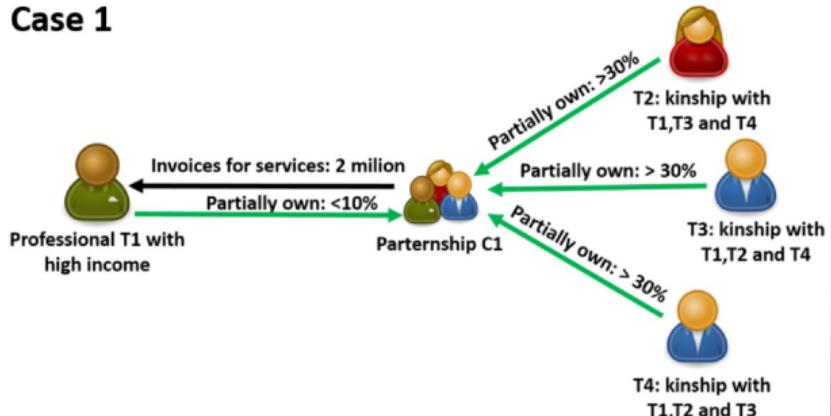
Daniele Pagliuca is a PHD student at the University of Perugia in Information Engineering and, at the same time, he works in the fiscal audit office of the Italian Revenue Agency. Since the beginning of his PHD program he investigated the use of visual languages and graph databases to design effective decision support systems for tax evasion discovery.

ACCEPTED MANUSCRIPT

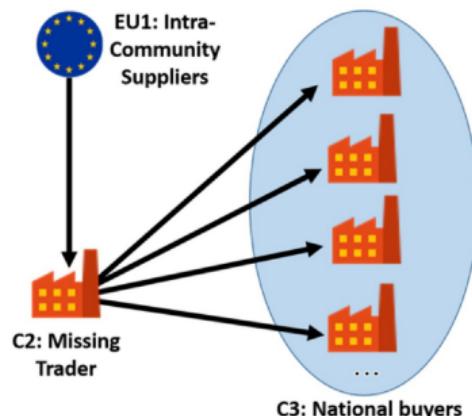
Highlights

- We present TaxNet, a new decision support system for tax evasion discovery
- It is based on a powerful visual language and on advanced network visualization techniques
- It has been developed in cooperation with the Italian Revenue Agency, where it is used
- To evaluate TaxNet we present an experimental study and use cases with experts

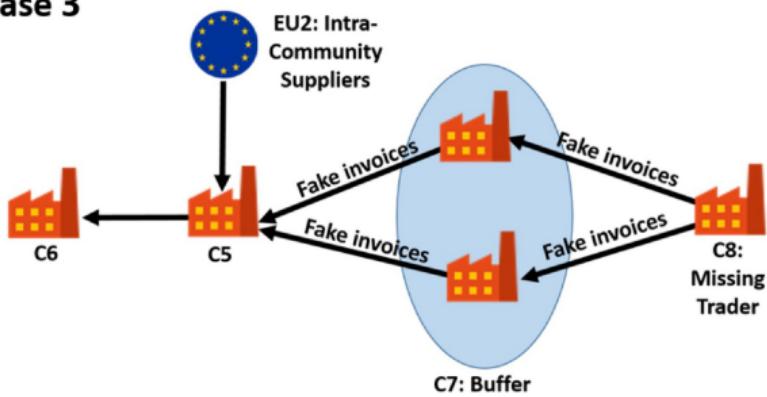
Case 1



Case 2



Case 3



Case 4

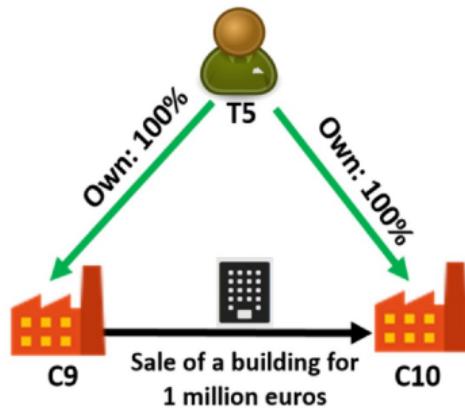
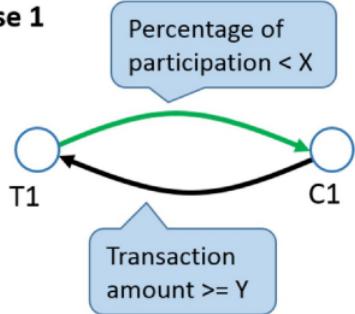


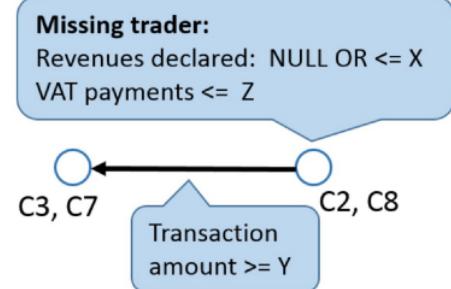
Figure 1

Case 1



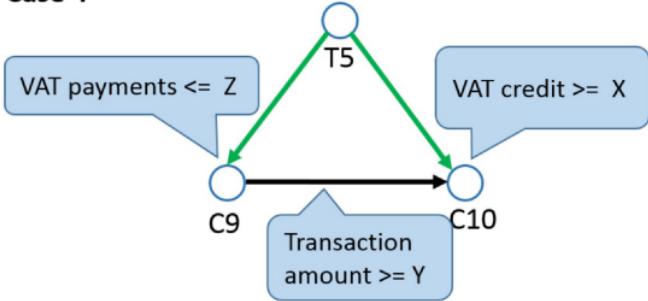
(a)

Cases 2,3

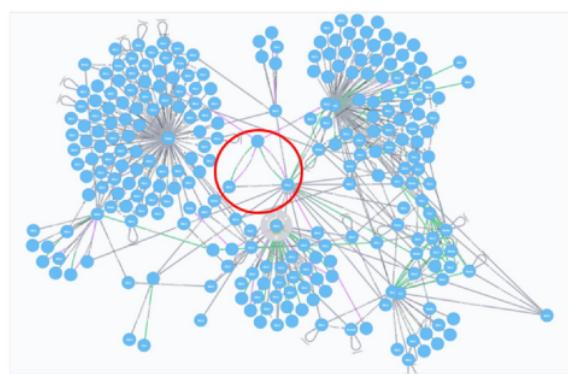


(b)

Case 4



(c)



(d)

Figure 2

```
MATCH (n1)-[e1]-(n2),(n1)-[e2]-(n2)
WHERE ((n2)-[e1]->(n1)) AND (type(e1)='TransazioneEconomica') AND
(type(e2)='Partecipazione') AND ((n1)-[e2]->(n2)) AND ((e1.ImponibileComDaCes >=
value1) OR (e1.ImponibileComDaCed >= value2)) AND NOT ID(n1) = ID(n2)
RETURN n1, labels(n1), ID(n1), e1, type(e1), ID(e1), ID(STARTNODE(e1)) as
STARTNODE_e1, ID(ENDNODE(e1)) as ENDNODE_e1, n2, labels(n2), ID(n2), e2, type(e2),
ID(e2), ID(STARTNODE(e2)) as STARTNODE_e2, ID(ENDNODE(e2)) as ENDNODE_e2 ORDER BY
id(n1) SKIP 0
```

Figure 3

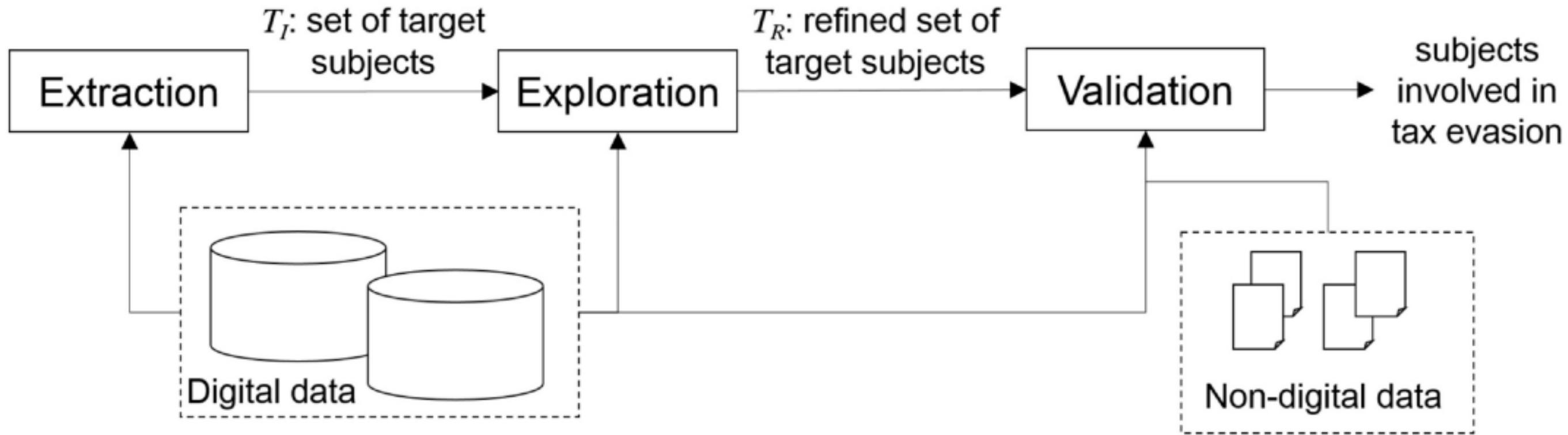


Figure 4

Tax Registry System

Search by fiscal code:

Fiscal code: FC00001

Search

1

Tax Registry System

Categories:

- VAT Decl.
- Tax Payments
- Company Reg.
↳

Analysis of subject: FC00001

Denomination:

Address:

Economic Activities:

2

Tax Registry System

Categories:

- VAT Decl.
- Tax Payments
- Company Reg.
↳

Analysis of subject: FC00001

Company Registry

Participated companies:

Company 1: FC00002

Company 1: FC00003

....

3

System switch

Economic Transaction System

Buyers of FC00003

Fiscal Code	Transaction amount	VAT
FC00001 ↳	1,000,000	220,000
FC00004	2,000,000	440,000

Export xls

5

Economic Transaction System

Search by fiscal code:

Fiscal code: FC00003

Fiscal year: 2015

Role:
 Seller:
 Buyer:

Search ↳

4

Figure 5

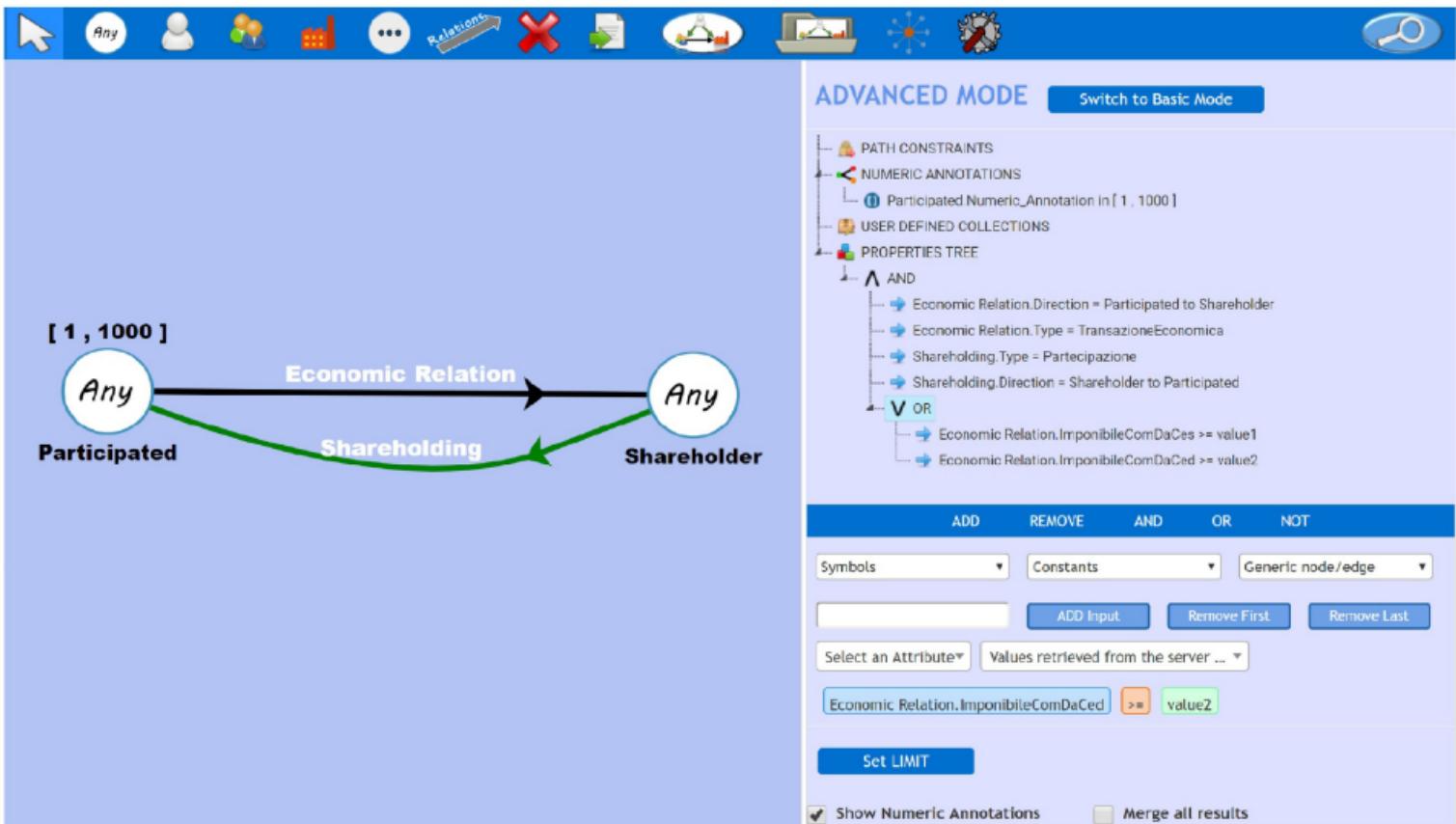
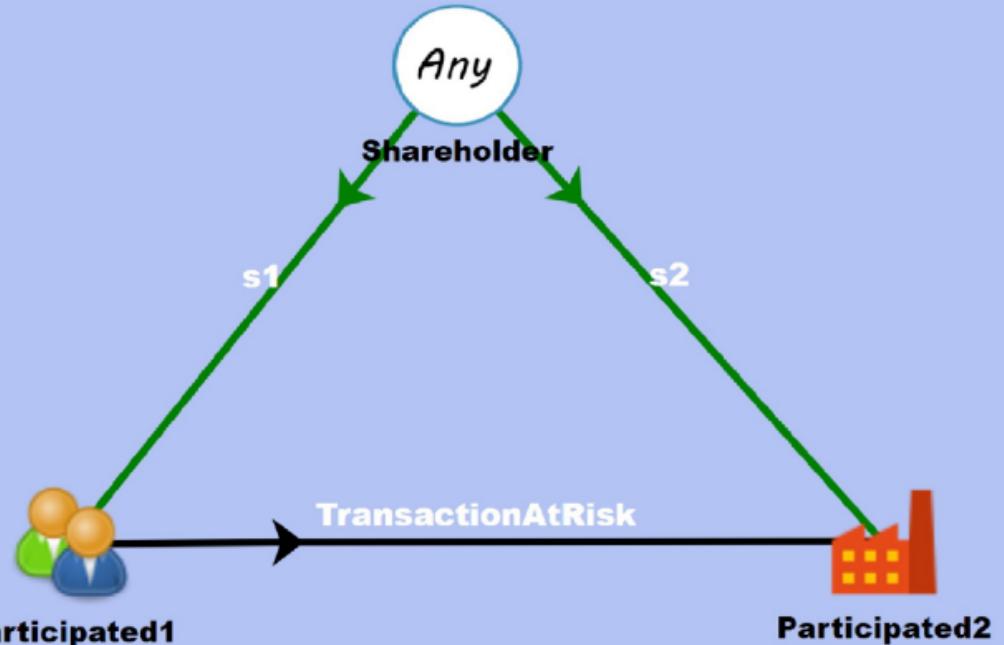


Figure 6



Basic properties editable: SuppliesFromAssociated

Shareholder: **Any**

Participated1: **partnership**

Participated2: **capital company**

Participated2.CreditolVA >=

TransactionAtRisk.ImponibileComunicato >=

TransactionAtRisk.ImponibileRicostruito >=

Pattern change description:

It is possible to change: the legal nature of the entities, the amount of the tax credit of the buyer, the minimum economic amount of the transactions (the value declared and the value reconstructed are in OR condition).

Figure 7

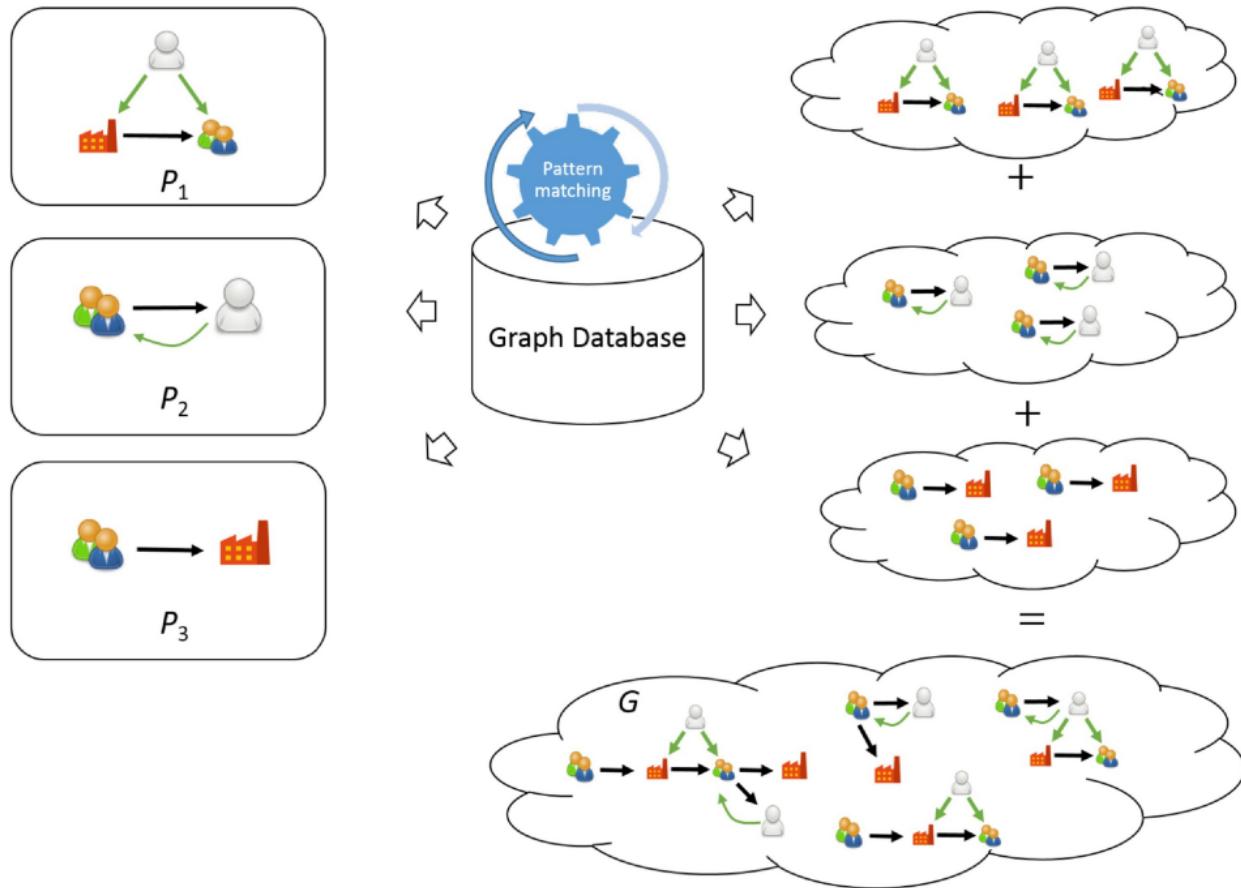


Figure 8

Results: 30

Rows 3

Columns 3

Order by EconomicTransactions

Reorder

Explore

Save

Export csv

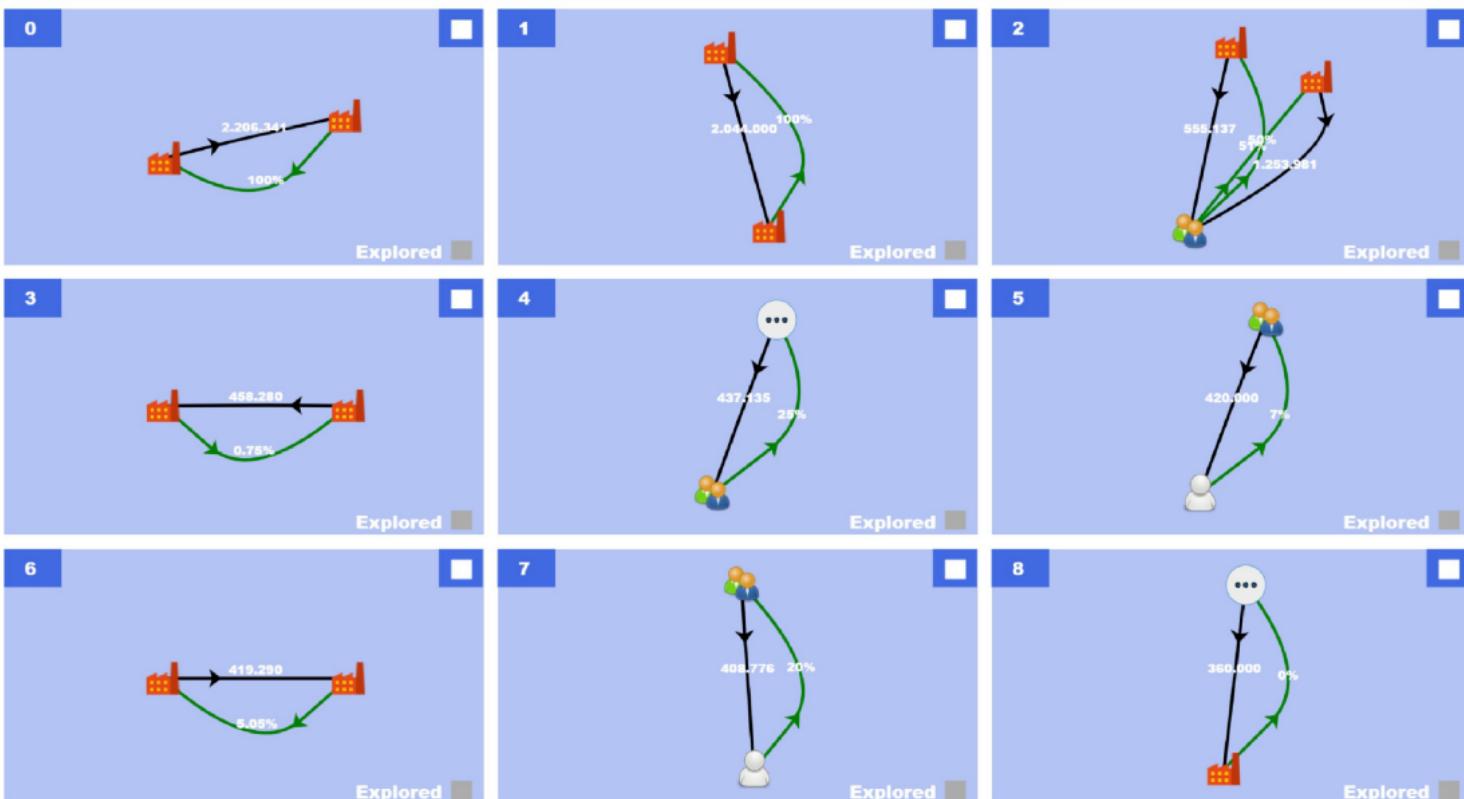
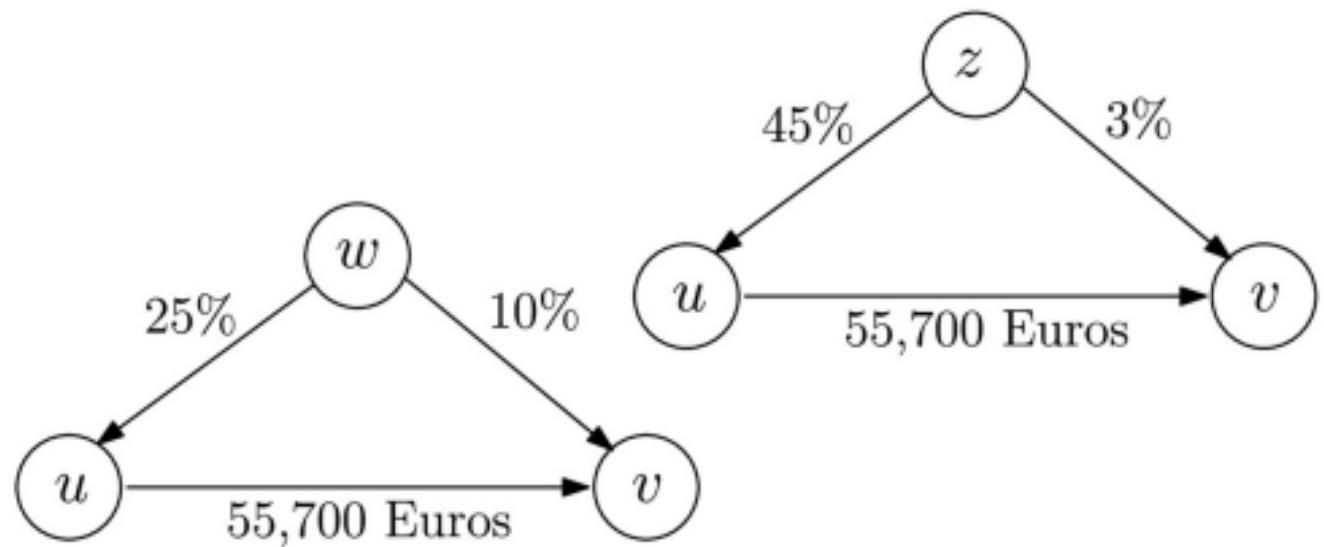
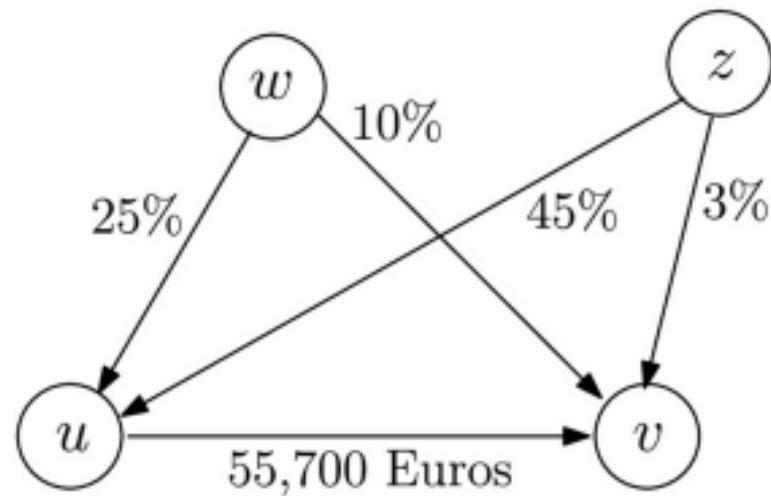


Figure 9

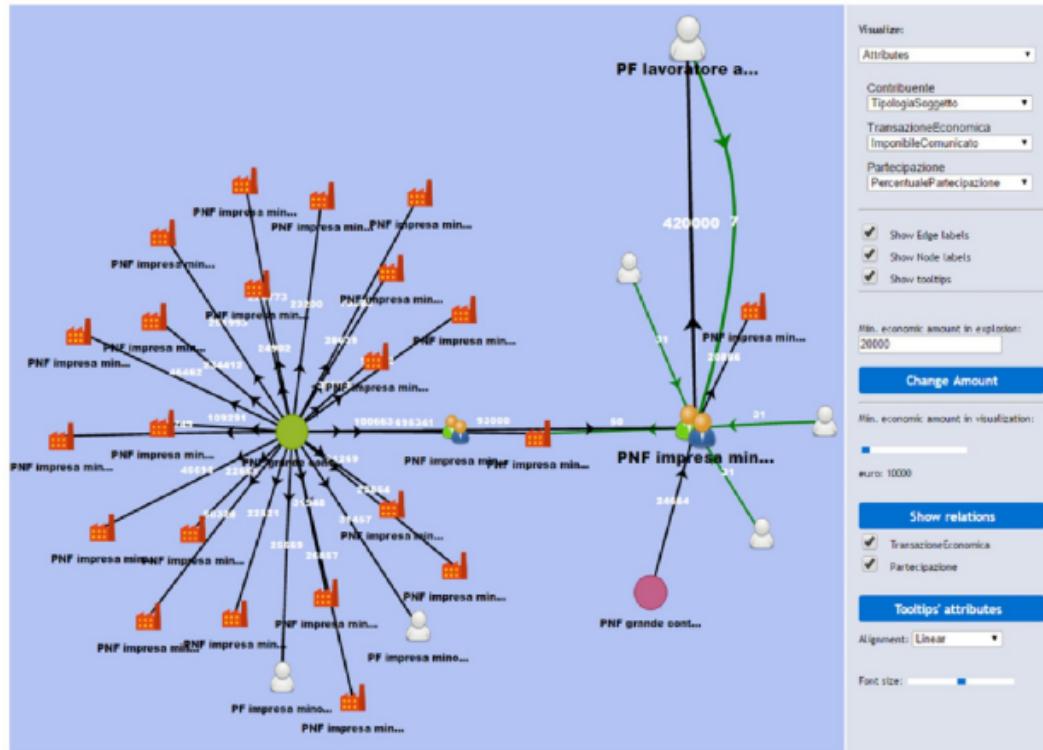


(a)

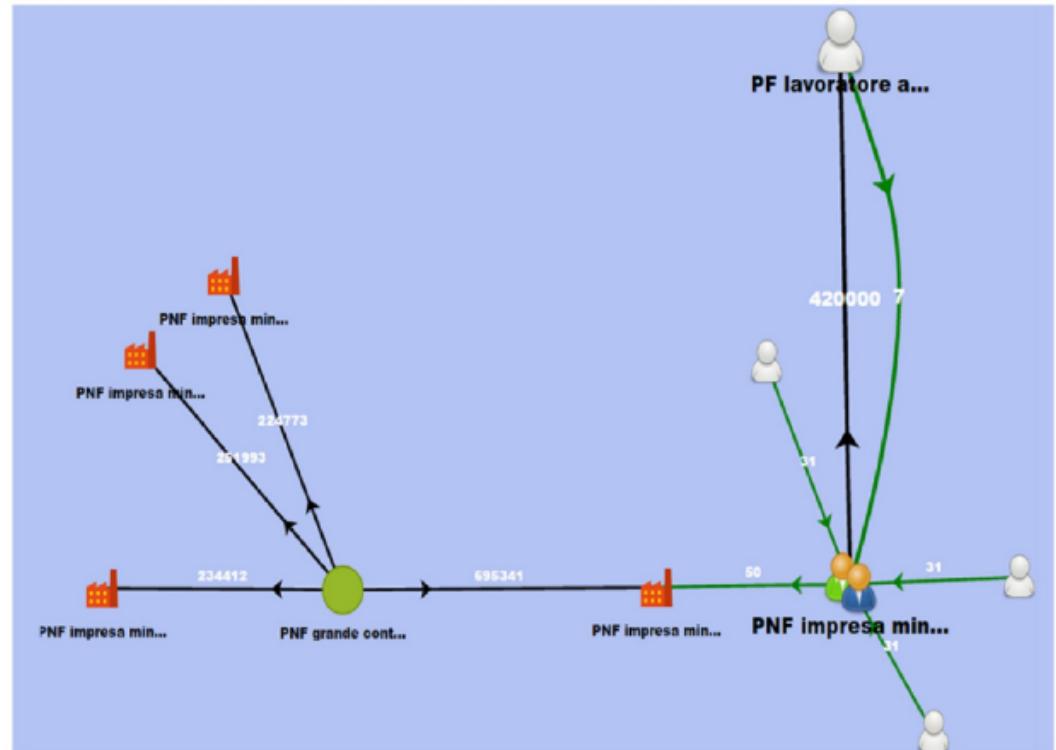


(b)

Figure 10



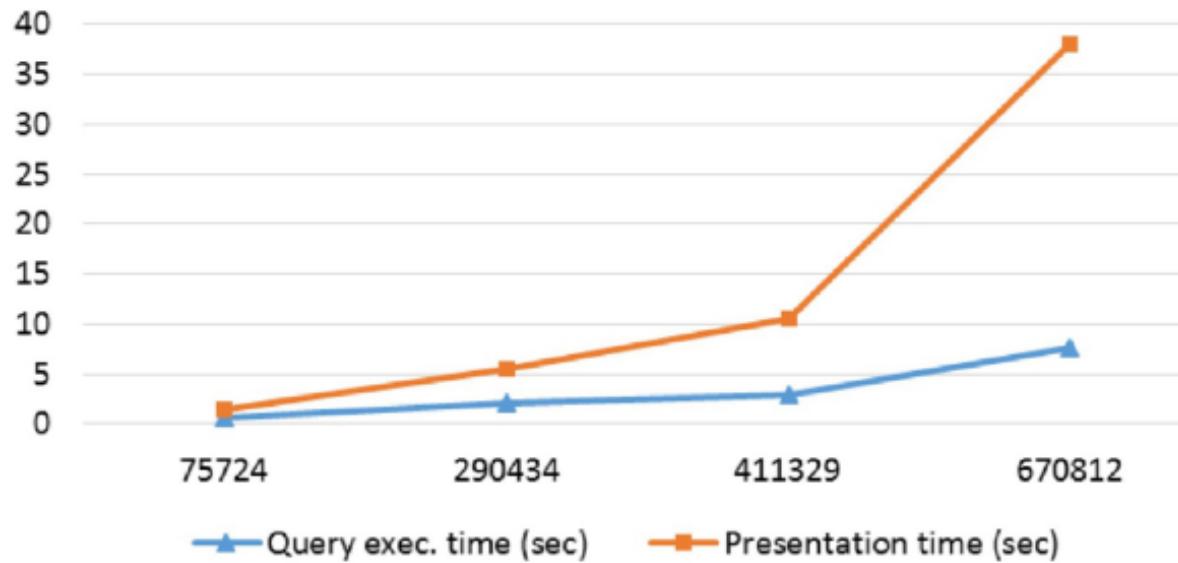
(a)



(b)

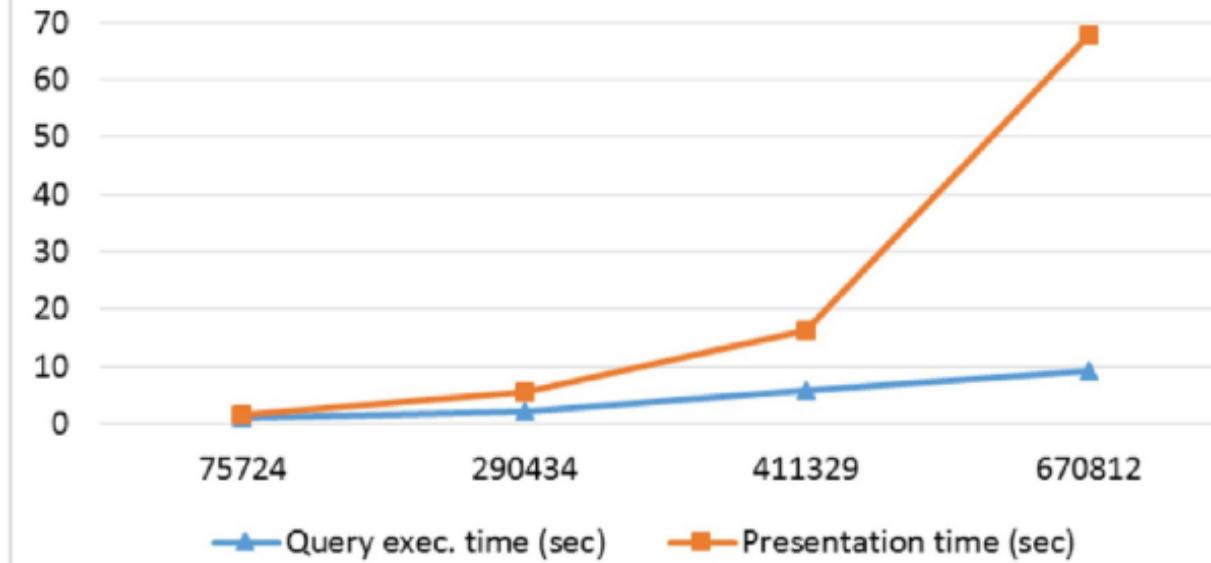
Figure 11

PurchaseFromRelated



(a)

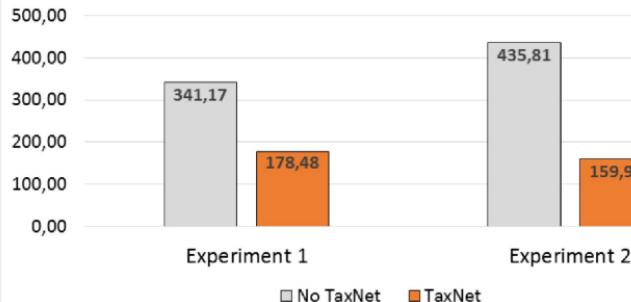
SuppliesFromAssociated



(b)

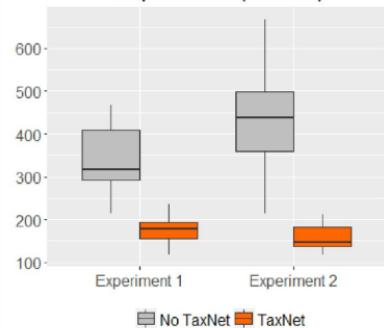
Figure 12

Mean response time (seconds)



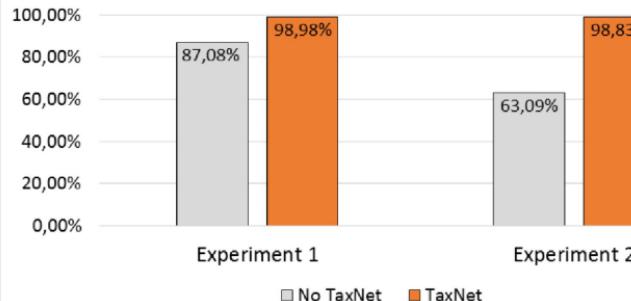
(a)

Distribution of the mean response time (seconds)



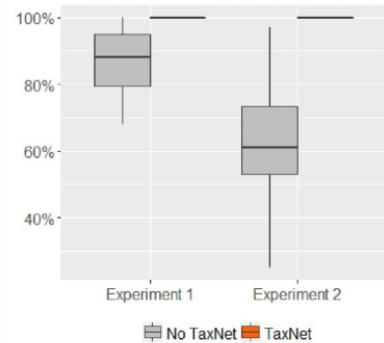
(b)

Mean accuracy rate



(c)

Distribution of the mean accuracy rate



(d)

Figure 13