UNIVERSITY OF CATANIA
Department of Mathematics and Computer Science

Doctoral Thesis

New perspectives in criminal network analysis: multilayer networks, time evolution, and visualization

Author:
Salvatore Amato Catanese

Thesis Supervisor:
Prof. Giacomo Fiumara

Head of Ph.D. School:
Prof. Giovanni Russo

A thesis submitted for the degree of
Philosophiae Doctor (PhD) in Mathematics and Computer Science
2016 November

ACKNOWLEDGMENTS

I first thank my advisor, Prof. Giacomo Fiumara. He has been a vast source of inspiration and motivation throughout my graduate career. I owe him all of the things I've done.

I am very grateful to two other special persons (and also main co-authors): Pasquale De Meo, Associate Professor at the Department of Ancient and Modern Civilizations, University of Messina (Italy) and Dr. Emilio Ferrara, Research Assistant Professor at the University of Southern California (USA) - Viterbi School of Engineering, Department of Computer Science. It was a very pleasure for me to work with them.

I would like to express my gratitude for their work to all my other coauthors of the Computer Science research group of University of Messina: Prof. Alessandro Provetti, Dr. Francesco Pagano and Dr. Santa Agreste.

During 2015 I had the chance to spend a month in Tarragona (Spain) at Rovira i Virgili University under the supervision of Professor Alex Arenas. I owe my gratitude to him for the possibility he gave me to stay in his laboratory and to learn the fundamentals on multilayer networks theory and developments. I had the the pleasure of being supported by one of the most talented scientists I have ever met, Manlio De Domenico (Research Fellow at Computer Science Department of Rovira i Virgili University). I want to address him my gratitude for the invaluable suggestions and his availability.

Still in Tarragona I had the chance of meeting Prof. Santo Fortunato, to whom I would like to extend my cordial thanks for the pleasant discussions having with him during breaks.

Catania(Italy), November 2016 Salvatore Amato Catanese.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

PREFACE

This Dissertation comprises the research work that I have carried out during the years of my Ph.D. studies at University of Catania (Italy) - Department of Mathematics and Computer Science, under the supervision of Professor Giacomo Fiumara at the University of Messina (Italy). All my research was carried out with my supervisor and other coauthors: Prof. Pasquale De Meo, Prof. Alessandro Provetti, Dr. Emilio Ferrara, Dr. Francesco Pagano and Dr. Santa Agreste.

The work presented in this Dissertation reflects a long-term human, professional and cultural path started some years ago when I first developed `LogAnalysis`, a tool for the analysis and visualization of criminal and social networks. Since then, I devoted myself to the development of frameworks, algorithms and techniques for supporting intelligence and law enforcement agencies in the task of unveiling the CN structure hidden in communication data, identifying the target offenders for their removal or selecting effective strategies to disrupt a criminal organization. In a natural way, I successively focused on the evaluation of the resilience of criminal networks and on the multiplex formalism, which takes into account the various relationships existing within a criminal organization.

In this context I introduce criminal network analysis tools: `LogAnalysis`, `LogViewer`, `Semantic viewer` and `Failure simulator`. I have been involved in the design, modeling, and writing of all of the works presented. In particular, I have also developed and tested all the visual tools included therein.

Finally, I introduce a framework (`CriMuxnet`) (still under development) for multilayer criminal networks analysis based on high-quality 3D visualizations of network data. `CriMuxnet` was designed to work in conjunction with a 3D computer graphics (CG) packages: Autodesk Maya[1] or Blender [2]). In my opinion `CriMuxnet` exploits 3D engine features to significantly improve both exploratory search and visualization strategy.

[1] http://www.autodesk.com/
[2] https://www.blender.org/

LIST OF PUBLICATIONS

This thesis includes an overview of the following pubblications.

JOURNAL ARTICLES

S. Agreste, S. Catanese, P. De Meo, E. Ferrara, and G. Fiumara. Network structure and resilience of Mafia syndicates, *Information Science*, 351, pp. 30-47, 10 (2016).

S. Catanese, P. De Meo, and G. Fiumara. Resilience in Criminal Networks. AAPP *Physical, Mathematical and Natural Sciences*, 94(2). pp. 15-33, (2016).

E Ferrara, P De Meo, S Catanese, and G Fiumara. Detecting criminal organizations in mobile phone networks. *Expert Systems with Applications*, 41(13), pp. 5733-5750, (2014).

S. Catanese, E. Ferrara, and G. Fiumara. Forensic analysis of phone call networks. *Social Network Analysis and Mining*, 3(1), pp. 15-33, (2013).

CONFERENCE PROCEEDINGS

S. Catanese, E. Ferrara, G. Fiumara, and F. Pagano. A framework for designing 3D virtual environments. In *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Springer Verlag, volume 78, p. 209-218 (2012).

S. Catanese, P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. Crawling Facebook for social network analysis purposes. In WIMS '11: *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, 52. ACM, (2011).

S. Catanese, E. Ferrara, G. Fiumara, and F. Pagano. Rendering of 3D dynamic virtual environments. In Simutools '11: *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, pp. 351-358, (2011).

S. Catanese and G. Fiumara. A visual tool for forensic analysis of mobile phone traffic. *Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence*, MiFor '10, ACM, Firenze, Italy, p. 71-76, (2010).

WORKSHOPS

E Ferrara, P De Meo, S. Catanese, G Fiumara. Visualizing criminal networks reconstructed from mobile phone records. *ACM Hypertext 2014 (Doctoral Consortium / Late-breaking Results / Workshops)*, (2014).

S. Catanese, Pasquale De Meo, Emilio Ferrara, and Giacomo Fiumara. Analyzing the Facebook friendship graph. In MIFI '10: *Proceedings of the 1st International Workshop on Mining the Future Internet*, volume 685, pp. 14-19, (2010).

BOOK CHAPTERS

E Ferrara, S. Catanese, G Fiumara. Uncovering criminal behavior with computational tools. Social Phenomena: From Data to Models, pp. 291-324, Springer, (2015).

S. Catanese, P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. Extraction and analysis of Facebook friendship relations. *Computational Social Networks: Mining and Visualization*, pp. 291-324. Springer (2012).

EDITORIALS

E Ferrara, S. Catanese, G. Fiumara. Criminal Network Analysis and Mining (CRIMENET2014): Introduction. *Social Informatics*, pp. 75-77, 2014.

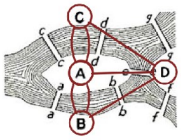
PRESS COVERAGE

1. The Dayside: Using statistics to catch cheats and criminals (*Physiscs Today*)
2. How to detect criminal gangs using mobile phone data (*MIT Technology Review*)
3. Criminal gang connections mapped via phone metadata (*New Scientist*)
4. Cell phone data analysis dials in crime networks (*Science News*)
5. New software can map criminal gang connections (*Business Standard News*)
6. Criminal Gang Connections Mapped via Phone Metadata (*Comunications of the ACM*)
7. LogAnalysis maps the structure of gangs using phone records (*Engadget*)
8. Mafia Wars: How Italy's Military Police Use Metadata To Track Organized Crime (*Fastcompany*)
9. New software can map criminal gang connections (*Free press journal*)
10. Complex networks researcher at IU fighting crime with mobile phone data (*IU Bloomington Newsroom*)
11. Matematica e pizzini: la mafia si combatte anche con l'analisi dei network per ricostruire le relazioni (*Il Sole 24 Ore*)
12. Gangster science: How police use network theory to track gang members (*The Daily Dot*)
13. Social Network Sleuths: Investigators Pursue Criminal Gangs Via Phone Chatter (*Homeland Security Today US*)
14. IU researcher helps Italian police fight crime (*Washington Times*)

INTRODUCTION

NETWORK THEORY

Networks are anywhere.

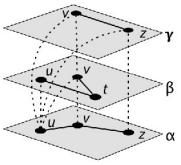


Köenigsberg bridges

Network theory is a powerful tool for modeling and analyzing the structure of real world complex systems throughout the social, biological, chemical, physical, and technological sciences. Since 1736, when Leonard Euler published the solution of the so called *Königsberg bridge problem*[3], networks studies employed the abstraction in which systems are represented as a graphs: a set of nodes and links. This approach has been used to illustrate that many real-world networks are usually *small-world* [378], show a heavy-tailed degree distribution [27], contain nodes that play central roles [291, 378], and have modular structures [156]. Graph theory was not merely limited to that mathematical development but has been used in many other scientific contexts. Social networks analysis started to develop in the early 1920s and focuses on relationships among social entities, as communication between members of a group, trades among nations, or economic transactions between corporations [53].

MULTILAYER NETWORKS

In recent years, the study of complex networks has concerned the understanding of systems composed of nodes connected by means of homogeneous relations [14, 53, 290]. Nevertheless, in many cases, a complex network is composed of subsystems caracterized by having functional independence and different structure, in which some nodes simultaneously belong to more graphs. It is the case, for example, of energetic [80], transport [181, 230], brain [81] and economic networks [394].



Multilayer network

In multilayer networks each node belongs to any subset of layers and edges are used to connect entities belonging to every possible combination of nodes and layers. We have therefore connections between nodes belonging to the same layer (intra-layer edges) and/or belonging to distinct layers (inter-layer edges) [54, 217].

MULTIPLEX NETWORKS

An important model of multilayer network is the multiplex network [37, 38, 41, 50, 54, 78, 121, 176, 216, 217, 237, 268, 282] in which the same set of nodes is connected by means of several type of links. In multiplex networks every type of connection exists in a specific layer.

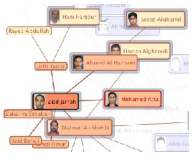
[3] The Königsberg bridge problem asks if the seven bridges of the city of Königsberg can all be traversed in a single trip without doubling back, with the additional requirement that the trip ends in the same place it began.

Multiplex networks adequately describe: *i)* social networks in which nodes represent users and every layer corresponds to a distinct type of interaction such as, for example, friendship, affinity, co-workership; *ii)* communications (e-mail, chat, phone, online and offline contacts) [183, 377]; *iii)* transport networks (streets, railways, channels, airline companies) [88, 386] and *iv)* Criminal Networks [220, 228, 279, 355].

Study of multiplex networks is one of the most interesting contemporary themes in network science together with some tightly related research area such as interacting networks [133, 239], interdependent networks [77, 80, 255], and interconnected networks [129, 249, 319, 335].

In interdependent networks, in particular, corresponding nodes in distinct layers may be connected by particular dependence relationships in which the functionality of a node in a layer depends upon the functionality of a node in another layer [80]. In these networks nodes may not exhibit correspondent nodes in every layer. In interdependent one-to-one networks, multiplex and interdependent networks are equivalent with respect to percolation [353].

CRIMINAL NETWORKS



Terrorist Network

Always more frequently law enforcement and investigative agencies all over the world active in searching efficient strategies helpful in fighting and controlling criminal and terrorist organizations exploit results from network theory and scientific research about structure and dynamics of social, biological and complex networks [292, 357].

Criminal network are a special kind of social network with emphasis on both *secrecy* and *efficiency*. Such networks are intentionally structured to ensure efficient communication between members without being detected [20, 246].

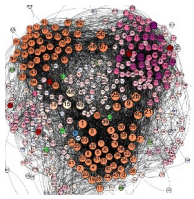
A criminal organization is a dynamic system that keeps changing over time. Many studies have applied SNA to analyze the structural properties of criminal networks, describe the changes in network members' individual characteristics and social roles (i.e., leader and gatekeeper), and capture the dynamic patterns of group membership and structure [228, 257, 320, 355]. Members can leave or join the network, can achieve apical position as well as can be hunted or killed by the organization itself. Similarly, the relations between partecipant may evolve, increasing or dissolving. Mesoscale structures may form (clans, subgroups, clusters) or split. The overall network may change from a centralized hierarchical structure to a decentralized flat (i.e., autonomous cells) [320].

RESILIENCE IN CRIMINAL NETWORKS

Social network analysis and graph theory can be used to identify key nodes in the network, which is helpful for network destabilization and disruption purposes [261]. Taking out key nodes will decrease the ability of the terrorist network to function normally.

The effectiveness of eradication strategies depends on the topology and flexibility of the network and particularly on available information data. These are often

inaccurate and fragmentary because of the secrecy concerning the diffusion of governmental information. Research on terrorist, criminal or the so-called cover networks considers destabilizing strategies [90], organizational features [279], and the methods used to identify the so-called key players [61, 355].



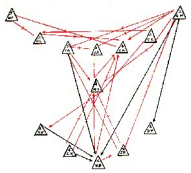
Resilient networks

It is possible to infer the structure and topology of a number of complex networks using artificial models [134, 233, 378, 379], but very few is known about the structure of relationships of cover networks and therefore on their resilience with respect to disruptive or destabilizing attacks. Cover networks rely on some topological features to organize the flow of communications among members, exercise influence and maintain secrecy at the same time.

The availability of more complete datasets in the activity of intelligence allows to deeply analyze multiple relationships (friendship, work, affinity, telephonic, etc.) existing among individuals belonging to a criminal network. Indeed, it is possible to distinctly consider heterogeneous relationships and the multilayer structure of the criminal network under study, in which each layer represents a different type of relation. This allows, in turn, to understand in a more adequate manner the dynamics of criminal networks and their resilience to attacks.

VISUALIZATION

Visualization of social networks has a rich history within the social sciences, where node-link depictions of social relations have been employed as an analytical tool since at least the 1930s [161].



Jacob Moreno's diagram

Network visualization, tools and techniques can assist intelligence analysts in various ways during a criminal networks investigation. The collected data needs to be analyzed and visualized in order to gain a deeper understanding of the criminal network.

Visual exploration can be chaotic when the network is large, due to cluttered presentations, overlapping links and illegible labels of nodes. In these situations, interactive techniques are necessary to make sense of such complex static visualizations. Interactions such as zooming, panning or filtering by inherent attributes can simplify complex visualizations [311].

Several approaches attempt to more efficiently use available display space by distorting the graph. Fisheye techniques allow users to examine a focus area in great detail, but also tend to obscure the global structure of networks. Plenty of node-link layout algorithms are used, including variants of the spring embedder [142] such as the popular Fruchterman- Reingold force-directed algorithm [165], the Prefuse gravitational N-Body approach [192], the Harel-Koren fast multi-scale algorithm [186], the high-dimensional embedding approach of Harel and Koren [187], and FM3 by Hachul and Jünger [182].

A number of existing tools support network analysis but only some of them have been developed for criminal network investigation. Related to our work we cite commercial tools like COPLINK [101, 389], Analyst's Notebook[4], Xanalysis Link

[4] i2 - Analysts Notebook. http://www-03.ibm.com/software/products/en/analysts-notebook/

Explorer[5] and Palantir Government[6]. Other related prototypes described in academic papers are Sandbox [387] and POLESTAR [314], among others.

CONTRIBUTIONS

This dissertation focuses on Criminal Network Analysis (CNA) and network visualization.

CRIMINAL NETWORK RESILIENCE AND SEMANTIC VISUALIZATION (Chapter 5) We review concepts and methods derived from Social Network Analysis and apply them to study the resilience of a criminal network. We also examine the evolution of a criminal network operating in Sicily (Italy), before and after the counter-actions of police agencies. We also show how the resilience of the network originated and the features that enable the dynamical reshaping of the criminal networks so as to continue illegal activities. In this study we tried to unveil the dynamics of resistance of a mafioso-type criminal network as an effect of two different types of interruption. According to different strategies of resilience, it has emerged that notwithstanding strong perturbations, both internal and external, the criminal network succeeded in reconstructing its structure, by reorganizing and accomplishing the necessary substitutions of the missing members.

The software tool we developed for the semantic visualization of the criminal network allows to analyze the relations among the members of the criminal network together with the relations of friendship and kinship every member maintains. This representation greatly helps in the analysis of the network when a path has been interrupted (for example, as a consequence of the removal of an element). The interaction and the filters allow to detect those links that are redundant or prone to their substitution.

CRIMINAL NETWORK ANALYSIS AND VISUALIZATION (Chapter 6) Provides a theoretical framework for the problem of detecting and characterizing criminal organizations in networks reconstructed from phone call records. We introduce an expert system to support law enforcement agencies in the task of unveiling the underlying structure of criminal networks hidden in communication data. This platform allows for statistical network analysis, community detection and visual exploration of mobile phone network data. It allows forensic investigators to deeply understand hierarchies within criminal organizations, discovering members who play central role and provide connection among sub-groups. `LogAnalysis` automatizes the import of raw phone call records data, the removal of ambiguities and redundancies in data, and the parsing and conversion to a graph format readily available for analysis and exploration. The data model is designed to improve the quality of the analysis of social relationships observed inside phone call network data through the integration of visualization and social network analysis-based statistical metrics. `LogAnalysis` implements different state-of-the-art view layouts for promoting fast and dynamic network exploration. The statistics simplify the comprehension of a sometimes chaotic visualization, al-

[5] Xanalysis (2014) - http://www.xanalys.com/products/link-explorer/
[6] Palantir government (2014) - http://www.palantir.com/solutions/

lowing users to focus on relevants nodes and edges. It introduces the possibility of analyzing the temporal evolution of the connections among individuals of the network, for example focusing on particular time windows in order to obtain further insights about the dynamics of communications before/during/after particular criminal events. Finally, it provides an unprecedented supervised community detection set of techniques that allows detectives to interact with the community detection process, incorporating expert knowledge to supervise the results and refine the unveiled community structure at different levels of granularity and resolution.

INTERACTIVE NETWORKS VISUALIZATION TECHNIQUES (Chapter 7) We employ some interactive visualization techniques to represent criminal and terrorist networks reconstructed from phone traffic data, namely foci, fisheye and geo-mapping network layouts. These methods allow the exploration of the network through animated transitions among visualization models and local enlargement techniques in order to improve the comprehension of interesting areas. By combining the features of the various visualization models it is possible to gain substantial enhancements with respect to classic visualization models, often unreadable in those cases of great complexity of the network.

A graph representation allows to overview the network structure, to identify the cliques, the groups, and the key players. The possibility of mapping the attributes of data and metrics of the network using visual properties of the nodes and edges makes this technique a powerful investigative tool. Often, however, visualization techniques become discouraging as a consequence of density and dimensions of the network. Some obstacles such as the overlap of nodes and the dense intersections of edges severely reduce the readability of the graph.

We present `LogViewer`, a next-generation Web-based criminal network analysis framework that yields advanced social network analysis functions, de facto extending `LogAnalysis` features to different types of networks, for example phone call networks and social graphs. `LogViewer` allows to study each network from three different angles: (i) static analysis, to investigate the role of nodes and edges, their centrality, and the emerging communities representing potential criminal rings; (ii) temporal analysis, to span across different temporal events and study the flow of information over time; finally, (iii) spatial analysis, embedding the network in a geographic space to determine physical closeness and locality effects on the network structure. LogViewer also allows to create multilayer spatio-temporal networks by merging different network types and to perform the above-mentioned different types of analysis on such a more complex network.

RESILIENCE OF MAFIA SYNDICATES (Chapter 8) We present the results of our study of Sicilian Mafia organizations using social network analysis. The study investigates the network structure of a Mafia syndicate, describing its evolution and highlighting its plasticity to membership-targeting interventions and its resilience to disruption caused by police operations. We analyze two different datasets dealing with Mafia gangs that were built by examining different digital trails and judicial documents that span a period of ten years. The first dataset includes the phone contacts among suspected individuals, and the second captures the relationships among individuals actively involved in various criminal offenses. Our report illustrates the limits

of traditional investigative methods like wiretapping. Criminals high up in the organization hierarchy do not occupy the most central positions in the criminal network, and oftentimes do not appear in the reconstructed criminal network at all. However, we also suggest possible strategies of intervention. We show that, although criminal networks (i.e., the network encoding mobsters and crime relationships) are extremely resilient to different kinds of attacks, contact networks (i.e., the network reporting suspects and reciprocated phone calls) are much more vulnerable, and their analysis can yield extremely valuable insights.

MULTIPLEX BREADTH-FIRST SEARCH ALGORITHM (Chapter 9) Breadth-first search (BFS) is an essential graph traversal strategy used in many real world optimization problems and computing applications. It is widely used as a basis for multiple fields: OSNs and web crawling tasks, social networking, network broadcast routing, analysis of semantic graphs, model checking (finite state machine), garbage collection, community detection and connected components algorithms.

BFS is the basic building block for other graph traversals, such as best-first search, uniform-cost search, greedy-search and the A*. Algorithm is too strongly used in Social Network Analysis to compute the maximum flow in a flow network and solve the shortest-path problem in closeness and betweenness centrality.

A variety of parallel BFS algorithms for both CPU and GPUs architectural models based on shared and distributed memory systems have since been explored. In this dissertation we present *Multiplex PBFS* (`Mx-PBFS`) a novel multi-threaded parallel Breadth-First Search algorithm for categorical and inter-layer couplings multiplex networks.

Parallel Multiplex Breadth-First Search algorithm (Mx-PBFS) systematically explores a categorical and inter-layer couplings multiplex network M consisting of L layers and V nodes per layer, to discover every vertex that is reachable from source node s to each reachable vertices of the multiplex graph. The considered topology is based on the connectivity between nodes representing the same entities (i.e. individuals) in the different layers. For instance, in Criminal Network: kinship, friendship, phone calls, instant messaging communications, email messages and bank transactions among participants. The interconnectivity pattern among layers is one-to-one between the same nodes. Mx-PBFS search provide traversal of layers with directed and undirected rela- tionships concurrently. We use a multiple adjacent list ADT, implemented by Python dictionary of dictionaries. It has vertices as keys and dictionary of nodes lists per layer where the vertex leads to.

MULTILAYER TEMPORAL ANALYSIS (Chapter 10) Criminal (and terrorist) networks are *temporal* networks "systems where connections between elements are active only for restricted periods of time". They are shaped both by the topological way in which participants are connected to each other and the temporal activity patterns of their dynamics (when and how they interact).

During the investigations, intelligence agencies collect data by looking at various sources of relational information to draw a picture of the network's structure [228]: credit files, bank accounts and the related transactions, telephone calling records, electronic mail, instant messaging, chat rooms, OSNs, court records and so on. In

these systems, each type of interaction has a given relevance and multiplex network provides a better representation of the networks.

In this context we introduce an under development time dependent framework (`CriMuxnet`) for multilayer criminal networks analysis and visualization. The library supports multilayer networks with both temporal and multiplex aspect at the same time. It provides high-quality 3D visualizations of network data inside to commercial 3D computer graphics (CG) packages Autodesk Maya[7] and open source Blender [8], exploiting 3D engine features to full the goals of exploratory search and visualization strategy.

We use 3D CG software to create visual interpretation of multilayer (and complex) network phenomena. There exist many special-purpose tools for the representation and manipulation of multiplex networks structure. These powerful tools were designed for specific analysis tasks and users, and therefore don't fulfill a complete range of visualization needs. For instance, camera, lighting, shading, and animation options are limited in even the most advanced viewing applications. However, through an integrated workflow with more sophisticated visual software packages like Maya (or Blender), users can leverage the combined power of network modeling and advanced data visualization. Moreover, the ease with which complex, dynamic systems of interacting objects can be built, animated and visualized in 3D software make it an effective tool for the rapid prototyping of network dynamics simulation.

We believe the dynamics capabilities of 3D CG application is very useful in the scientific modeling that we're interested in exploring.

INTERCONNECTED MULTILAYERED FAILURE SIMULATOR (Chapter 11) *Interdependent networks* are a system in which two or more monoplex networks are connected to each other via a *dependency edge*. A classical example is a network constituted by an electrical grid and a computer network where the proper functionality of a router in the computer network can depend on a power station and vice versa [80].

In the same way, *interconnected networks*, *interacting networks*, and *networks of networks* are sets of networks in which some of the nodes from the various networks are adjacent to each other, but the edges that connect different networks need not indicate dependency relations [217, 331]. *Multiplex networks* are the special cases where the same set of nodes appear across different layers while the links within the layers are different.

The existence of such multiple connections on different layers can be genalizated by means of multilayer interconnected networked systems, or simply *interconnected networks*. Huge literature regards two very relevant, and correlated processes: resilience of the multilayer networks' structure under random failures (and/or cascades of failures), and percolation.

We propose a real time failure simulator on interconnected networks. Our goal is to build a visualization system that end-users of complex networking analysts could use to facilitate discovery and increased awareness of their exploration (percolation process, cascading-failure, spreading processes, diffusions and random walks).

[7] http://www.autodesk.com/
[8] https://www.blender.org/

1 GRAPHS FOUNDATIONS

Network theory is a very valuable tool for describing and analyzing complex systems [14, 290, 357, 378] in social, physical, biological information and engineering sciences [14, 53, 291, 377]. Since Leonard Euler solved the so called *Königsberg bridge problem*[1] in 1736, graph theory has provided answers to a series of practical problem by abstracting away details and reduce it to a graph. The classical approach in studying networks considers, in its more general form, that a networked system may be represented by a set of entities connected by some kind of relationship. This approach has been used in the study of real networks to hsow some important properties of related graphs such as, for example, power-law distribution [27, 105] of the neighbors of a node, small-world property [315, 378] which ehibits a large density of nodes locally connected and a small average distance between pairs of nodes, related to the importance of roles and the position of nodes [377] (*centrality metrics*) and the presence of modular structures [156, 232, 316] called *communities* having dense internal connections and scarcely interconnected among them.

In this chapter we briefly introduce the basic concept of graph theory, using notation from [53]. We will start by explaining how a network is represented with a graph, its variants and basic statistical properties.

1.1 GRAPHS

A graph is a mathematical model of a network representation defined by graph theory as a set of entities called nodes (or vertices) connected by links (or edges).

Formally, a graph is an ordered pair of disjoint sets $G = (V, E)$, such that $V = \{v_1, \ldots, v_N\}$ is the set of N nodes and $E \subseteq V \times V$ is the set of edges that connect pairs of nodes [57, 291]. If there is an edge $e = (u, v) \in E$ between a pair of nodes $u, v \in V$, then those nodes are said *adjacent* to each other and we can write this relation with $u \sim v$. Edges can be undirected when $(u, v) = (v, u)$, directed if $(u, v) \neq (v, u)$, weighted if a weights function $w : E \to \mathbb{R}$ associate a real number with each edge of graph so $(u, v) = w(u, v)$. and loops if $e = (u, u)$. Consequently, a graph is undirected when edges relationships are symmetric, directed when they are not, weighted when a function associates a real number to every edge of graph, and simple when a graph admits no loop nor multiple edges between a same pair of nodes.

A directed graph may have more edges between the same pair of vertices (called *multiple edges*). Two or more edges having the same orientation between the same pair of vertices are called *parallel*. A graph having parallel edges is called *multigraph*.

Nodes and edges of the graph G are denoted respectively by $V(G)$ and $E(G)$. The order of a graph is the number of nodes $|V|$, and the size of a graph represent its number of edges $|E|$.

[1] Finding a round trip that traversed all seven bridges of the prussian city of Königsberg exactly once each.

A *subgraph* $G' = (V', E')$ of graph $G = (V, E)$ is a graph such as $V' \subseteq V$ and $E' \subseteq E$. An *induced subgraph* $G' = (V', E')$ is a subgraph of $G = (V, E)$ induced by a set of nodes such as $V' \subseteq V$ and $E' = \{(u, v) \in E, u \in V', v \in V'\}$. A *k-clique* V' is a set of nodes in $G = (V, E)$ such as the induced subgraph $G' = (V', E')$ forms a complete graph with order k. A *complete graph* $G = (V, E)$ is a graph for which every node is connected to all other such as $E = 1/2N(N-1)$ in an undirected graph or $E = N^2$ in a directed graph.

Two graphs $G = (V, E)$ and $G' = (V', E')$ are *isomorphic*, and write $G \simeq G'$, if there is a correspondence between their vertex sets that preserves adjacency. Thus G is isomorphic to G' if there is a bijection $' : V \to V'$ such that $(u, v) \in E \Leftrightarrow \varphi(u)\varphi(v) \in E'$.

A *walk* is an alternating sequence of adjacent nodes that begins with node u and ends with v. The *length* of the walk is defined as the number of edges in the sequence. A *trail* is a walk in which no edge is repeated. A *path* $u \overset{p}{\rightsquigarrow} v$ is a walk in which no node is visited more than once. The walk of minimal length between two nodes is known as *shortest path* or *geodesic*. A *cycle* is a closed walk, of at least three nodes, in which no edge is repeated.

A graph is *connected* if for every pair (u, v) of distinct vertices there is a path from u to v. A maximal connected subgraph is a component of the graph. A cutvertex is a vertex whose deletion increases the number of components.

The connectedness of an undirected graph can be evaluated by computing the largest connected component by means of an exploring algorithm: Depth-First-Search (DFS) and Breadth-First-Search (BFS) [111]. Their computational complexity is $\mathcal{O}(V + E)$ and $\Theta(V + E)$, respectively.

A directed graph is strongly connected if there exists a directed path connecting every pair of vertices. It is possible to find the strongly connected components of a directed graph by applying the optimized DFS algorithm [361].

Given a weighted graph, the weigth $\omega(p)$ of a path P is defined as the sum of the weights of the edges contained in P. It is possible to find the minimum weight path between a pair of vertices u and v, namely the shortest path with respect to the weight function ω among all possible paths between the two vertices. For non-weighted graphs, the shortest path corresponds to the path composed of the minimum number of edges. Given a graph $G = (V, E)$, the weight function $\omega : E \to \mathbb{R}$ and a source vertex $s \in V$, the problem *single-source shortest path* (SSSP) can be solved by computing all the shortest paths between s and all the vertices in G. SSSP can be computed by applying Dijkstra algorithm [130] with a computational complexity of $\mathcal{O}(E + VlogV)$, if the graph does not contain negative weighted cycles, otherwise the Bellman-Ford algorithm [111] with a computational cost $\mathcal{O}(EV)$. For an non-weighted graph, the problem can be solved by applying the BFS algorithm in $\mathcal{O}(E + V)$ [111].

Eccentricity ϵ of a vertex v is represented by the greatest geodetic distance between v and every other vertex in G. Eccentricity may vary in the interval $[V - 1]$. Diameter and other centrality measures are based on the concept of eccentricity, e.g. center and center of gravity of a graph.

The diameter $Diam(G)$ of a graph is given by the maximum value of eccentricity between all pairs of vertices. In other terms, it is the maximum distance between all pairs of vertices. Diameter may range in the interval $[1, V - 1]$. In a disconnected graph, diameter is infinite, but it is possible to find the diameter of its largest

connected component. In a similar way, it is possible to find the diameter of a subgraph. To compute the diameter, it is necessary to solve the so-called problem of *all-pairs shortest paths* (APSP). The largest value of APSP is the graph diameter. APSP problem may be solved by applying classical algorithms such as Floyd-Warshall [111] in time $\mathcal{O}(V^3)$, or by solving V times the SSSP problem, with a cost $\mathcal{O}(EV + V^2 \log V)$ or, in the case of non-weighted and undirected graphs, Seidel algorithm [340] with cost $\mathcal{O}(M(V) \log V)$, $M(V)$ being the computational cost of the multiplication of two matrices $V \times V$ containing small integers whose cost is $\mathcal{O}(V^{2.376})$.

It is often useful to consider a matrix representation of a graph. A Graph $G = (V, E)$ can be described by the *adjacency matrix* $A = (a_{uv})^{V \times V}$ in which each row or column represents a node and an entry $a_{uv} \neq 0$ represents an existing edge between nodes u and v. Element $a_{uv} = w$ when $w : E \to \mathbb{R}$ and G is a weighted graph. The diagonal entries are all equal to o when G have no loop. A is not symmetric if G is a directed graph.

1.2 TOPOLOGICAL PROPERTIES

We now present the common measures used for graph exploration and mining. Quantifying the topology gives very important insights on the network to: detecting patterns, understanding of how a graph is wired, whether its members are equally connected or not, or if every node is reachable from any other one.

1.2.1 Degree and degree distribution

The set of vertices adjacent to a vertex v in G, the *neighbourhood* of v, is denoted by $\Gamma_G(v)$. The *degree* k_i of a node i is the number of neighbours of i and is defined, if we consider a adjacency matrix of a graph, as:

$$k_i = \sum_{j=1}^{N} a_{ij} \tag{1}$$

For directed graphs we call $k_i^{\text{out}} = \sum_{j=1} a_{ij}$ the number of outgoing edges of a node i (*out-degree*) and the number of ingoing edges (*in-degree*) with $k_i^{\text{in}} = \sum_{i=1} a_{ij}$. In both cases, the sum is over all nodes of the network. We can add the in- and out-degrees to get the total number of connections of a node, or its total degree $k_i^{\text{tot}} = k_i^{\text{in}} + k_i^{\text{out}}$.

The *average degree* for an undirected network is:

$$\langle k \rangle = \frac{1}{N} \sum_{i=1}^{N} k_i = \frac{2L}{N} \tag{2}$$

and the average degree of a directed network is obtained by:

$$\left\langle k^{in} \right\rangle = \frac{1}{N} \sum_{i=1}^{N} k_i^{in}, \quad \left\langle k^{out} \right\rangle = \frac{1}{N} \sum_{i=1}^{N} k_i^{out} = \frac{L}{N} \tag{3}$$

A basic topological characterization of a graph can be obtained in terms of the *degree distribution*. The degree distribution $P(k)$, which gives the probability that a

randomly chosen node in the network has exactly k links or equivalently k nearest neighbors. In a directed graph we can consider $P(k)^{\text{out}}$ and $P(k)^{\text{in}}$ distributions. Since $P(k)$ is a probability, it must be normalized: $\sum_{k=1}^{\infty} P(k) = 1$.

For a network with N nodes the degree distribution is the normalized histogram is given by $P(k) = \frac{N_k}{N}$ where N_k is the number of degree-k nodes. Hence the number of degree-k nodes can be obtained from the degree distribution as $N_k = NP(k)$.

The n-moment of $P(k)$ is defined as:

$$\langle k_i^n \rangle = \sum_k k^n P(k). \tag{4}$$

The first moment $\langle k \rangle$ is the mean degree of G. The second moment $\langle k^2 \rangle$ measures the fluctuation of the connectivity distribution.

The degree distribution of a random graph which will be briefly discussed in section 1.3.1, is well described by a Poisson distribution law, with a peak in $P(\langle k \rangle)$. On the other hand, recent empirical results show that in the most of real-world networks the degree distribution significantly differs from a Poisson distribution. In particular, for several large-scale networks, such as the World Wide Web [13], Internet [149], metabolic networks [207], etc., the degree distribution follows a power-law:

$$P(k) \sim k^{-\lambda} \tag{5}$$

This power-law distribution falls off more gradually than an exponential one, allowing for a few nodes of very large degree to exist. Since these power-laws are free of any characteristic scale, such a network with a power-law degree distribution is called a scale-free network [27].

1.2.2 Matrix power: Walk and path

As defined before a *walk* over a graph is an alternated sequence of adjacent vertices. We can calculate a *walk* using the powers of the adjacency matrix A associated to the graph G. The component a_{ij}^p of the matrix $\mathbf{A}^p = \mathbf{A}_1 \times \cdots \times \mathbf{A}_p$ will contain the number of *walks* having length p from node v_i to node v_j. Diagonal elements of a_{ii}^p denote the number of *closed walks* starting and ending in the same node v_i.

A *path* is a particular type of *walk* in which all nodes and edges are distinct. Two nodes are reachable if does exist a path $u \overset{p}{\rightsquigarrow} v$. We can calculate the reachability of a pair of verticees in a graph G having N nodes by summing the powers of the adjacency matrix $\mathbf{A}^p$ con $1 \leq p \leq N-1$:

$$\mathbf{A}^p = \sum_{i=1}^{N-1} \mathbf{A}^i.$$

The component a_{ij}^p di $\mathbf{A}^p$ will contain the number of *walks* of any length $p \leq N-1$ starting from the vertex v_i and ending in v_j, 0 otherwise.

1.2.3 Closeness centrality

We can represent all the shortest path lengths of a graph G as a matrix $\mathcal{D}$ in which the entry d_{ij} is the length of the geodesic from node i to node j. The maximum value

of d_{ij} is the diameter $Diam(G)$ of the graph. Closeness centrality measures how close a node is to all the other nodes in the graph. It involve the computation of the *average shortest path* length, defined as the mean of geodesic lengths over all couples of nodes.

$$L = \frac{1}{N-1} \sum_{j \neq i}^{N} d_{ij} \tag{6}$$

L diverges if there are disconnected components in the graph. An indicator of the traffic capacity of a network that avoid divergence of average shortest path length is the *efficiency* of G:

$$E = \frac{1}{N(N-1)} \sum_{i,j \subset N, i \neq j} \frac{1}{d_{ij}}. \tag{7}$$

The average distance says how long it will take for information starting from node v_i to reach the whole network. Conventionally, a node with higher centrality is more important. Thus, the closeness centrality c_i of a node i is defined as a node's inverse average distance

$$c_i = \left[\frac{1}{N-1} \sum_{j \neq i}^{N} d_{ij} \right]^{-1} = \frac{N-1}{\sum_{j \neq i}^{N} d_{ij}} \tag{8}$$

1.2.4 Betweenness centrality

Node betweenness, originally introduced to quantify the importance of an individual in a social network [377], counts the number of shortest paths in a network that will pass through a node. Nodes with high betweenness play a key role in the communication within the network. The betweenness centrality of a node is defined as

$$b_i = \sum_{j,k \subset N, j \neq k} \frac{\sigma_{jk}(i)}{\sigma_{jk}}, \tag{9}$$

where σ_{jk} is the number of shortest paths between nodes j and k, and $\sigma_{jk}(i)$ is the number of shortest paths between nodes j and k that pass along node i.

The concept of betweenness can be extended also to the edges. The edge betweenness is defined as the number of shortest paths between pairs of nodes that run through that edge [67].

Betweenness centrality can also be viewed as a measure of network resilience, indicating how much effect on path length the removal of a vertex will have [199, 290].

1.2.5 Clustering coefficient

Clustering (or transitivity) is a property of acquaintance networks where two individual with a common friend are likely to know each other [37, 377].

Transitivity can be studied by means of the local clustering coefficient proposed by [378]. It is defined as the ratio between number of edges among neighbors of

node i (denoted by $\Gamma_G(i)$) and $k_i(k_i - 1)/2$ the maximum possible number of edges in neighbors Γ_i of node i

$$c_i = \frac{2\Gamma_G(i)}{k_i(k_i-1)} = \frac{\sum_{j\neq i, m\neq i}^{N} a_{ij}a_{jm}a_{mi}}{\sum_{j\neq i, m\neq i}^{N} a_{ij}a_{mi}} \tag{10}$$

If we think of three nodes i, j and m with mutual relations between i and j as well as between i and m, the clustering coefficient of i represents the likelihood that j and m are also related to each other. Averaging this quantity over all the nodes in a network gives the network clustering coefficient:

$$C = \frac{1}{N}\sum_i C_i \tag{11}$$

A similar measure which is commonly used in the social sciences, is the network transitivity [290]. It is defined as the proportion of triads (i.e. connected triples of nodes) which close into triangles:

$$T = \frac{3 \times \text{\# of triangles in G}}{\text{\# of connected triples of vertices in G}} \tag{12}$$

1.2.6 Eigenvector centrality

In degree centrality, nodes with many connections are regarded as more important. In real-word networks the more connected to important nodes a node is, the more influent.

Eigenvector centrality generalize degree centrality by incorporating the importance of the neighbors (or incoming neighbors in directed graphs). Let x_i denote the score of the i^{th} node and A the adjacency matrix of a network, the eigenvector centrality for a node i is defined as:

$$x_i = \frac{1}{\lambda}\sum_{j\in\Gamma_i} x_j = \frac{1}{\lambda}\sum_{j=1}^{N} a_{ij}x_j \tag{13}$$

where Γ_i is the neighborhood of the vertex i and $\lambda \neq 0$ is a constant. In matrix form we have $\lambda\mathbf{x} = \mathbf{x}A$. Hence the centrality vector $\mathbf{x}$ is the left-hand eigenvector of the adjacency matrix A associated with the eigenvalue λ. We choose λ as the largest eigenvalue in absolute value of matrix A. Thus the Perron-Frobenius theorem guarantees that if the A matrix is irreducible, or equivalently if the graph is (strongly) connected, then the eigenvector solution $\mathbf{x}$ is both unique and positive.

1.2.7 Katz centrality

Eigenvector centrality works well only if the graph is (strongly) connected. Real undirected networks typically have a large connected component, of size proportional to the network size. However, real directed networks do not. If a directed network is not strongly connected, only vertices that are in strongly connected components or in the out-component of such components can have non-zero eigenvector centrality [213, 291]. The other vertices, such as those in the in-components of strongly connected

components, all have, with little justification, null centrality. This happens because nodes with no incoming edges have, by definition, a null eigenvector centrality score, and so have nodes that are pointed to by only nodes with a null centrality score.

Katz centrality computes the relative influence of a node within a network by measuring the number of the immediate neighbors $\Gamma_G(i)$ and also all other nodes in the network that connect to the node under consideration through these immediate neighbors.

Let A be the adjacency matrix of a network under consideration, the powers of A indicate the presence (or absence) of links between two nodes through intermediaries. The Katz centrality of a node x_i^{Katz} can be computed as:

$$x_i^{\text{Katz}} = \sum_{k=1}^{\infty} \sum_{j=1}^{n} \alpha^k (A^k)_{ji} \tag{14}$$

A_{ij}^k is the total number of k degree connections between nodes i and j. The value of the attenuation factor α has to be chosen such that it is smaller than the reciprocal of the absolute value of the largest eigenvalue of the adjacency matrix A. In this case the following expression can be used to calculate Katz centrality:

$$x_i^{\text{Katz}} = ((I - \alpha A^T)^{-1} - I)\mathbf{u}$$

where I is the identity matrix, $\mathbf{u}$ is an identity vector of size N consisting of ones. A^T denotes the transposed matrix of A and $(I - \alpha A^T)^{-1}$ denotes matrix inversion of the term $(I - \alpha A^T)$.

1.3 NETWORKS MODELS

Complex problems from a wide range of fields can be theoretically modeled and described as *complex networks* [14, 290, 357, 378].

Concepts such as the short paths length, the clustering and the scale-free degree distribution have been recently applied to rigorously model the networks. Three main modeling paradigms exist: i) random graphs, ii) "small world" networks and, iii) power-law networks. Random graphs represent an evolution of the Erdős-Rényi model, and are widely used in several empirical studies, because of the ease of adoption. After the discovery of the clustering property, a new class of models, namely "small world" networks, has been introduced. Similarly, the power-law degree distribution definition led to the modeling of the homonym networks, which are adopted to describe scale-free behaviors, focusing on the dynamics of the network in order to explain phenomena such as the power-law tails and other non-Poisson degree distribution, empirically shown by real-world networks.

1.3.1 The Erdős-Rényi model

Erdős and Rényi [147, 148] proposed one of the first models of network, the random graph. They defined two models: the simple one consists of a graph containing n vertices connected randomly. The commonly adopted model, indeed, is defined as a graph $G_{n,p}$ in which each possible edge between two vertices may

be included in the graph with the probability p (and may not be included with the probability $(1 - p)$).

Although random graphs have been widely adopted because their properties ease the work of modeling networks (e.g. random graphs have a small diameter), they do not properly reflect the structure of real-world large-scale networks, mainly for two reasons: i) the degree distribution of random graphs follows a Poisson law, which substantially differs from the power-law distribution shown by empirical data; ii) they do not reflect the clustering phenomenon, considering all the nodes of the network with the same "weight", and reducing, de facto, the network to a giant cluster.

1.3.2 The Watts-Strogatz model

The real-world social networks are well connected and have a short average path length like random graphs, but they also have exceptionally large clustering coefficient, which is not reflected by the Erdős-Rényi model or by other random graph models. Watts and Strogatz proposed a one-parameter model that interpolates between an ordered finite dimensional lattice and a random graph. Starting from a ring lattice with n vertices and k edges per vertex, each edge is rewired at random with probability p [378]. Authors found that $L \sim n/2k \geq 1$ and $C \sim 3/4$ as $p \to 0$, while $L = L_{random} ln(n)/ln(k)$ and $C = C_{random} k/n \leq 1$ as $p \to 1$. It concludes that the Watts-Strogatz model is suitable for explaining such properties in many real-world examples.

The model has been widely studied since the details have been published. Its role is important in the study of the small-world theory. Some relevant theories, such as Kleinberg's work [218, 219], are based on this model and its variants. The disadvantage of the model, however, is that it is not able to capture the power-law degree distribution as presented in most real-world social networks.

1.3.3 The Barabási-Albert model

The two previously discussed theories observe properties of real-world networks and attempt to create models that incorporate those characteristics. However, they do not help in understanding the origin of social networks and how those properties evolve.

The Barabási-Albert model suggests that two main ingredients of self-organization of a network in a scale-free structure are *growth* and *preferential attachment*. These pinpoint to the facts that the most of networks continuously grow by the addition of new nodes which are preferentially attached to existing nodes with large numbers of connections (again, "rich gets richer"). The generation scheme of a Barabási-Albert scale-free model is as follows: (i) *Growth*: let p_k to be the fraction of nodes in the undirected network of size n with degree k, so that $\sum_k p_k = 1$ and therefore the mean degree m of the network is $\frac{1}{2} \sum_k k p_k$. Starting with a small number of nodes, at each time step, we add a new node with m edges linked to nodes already part of the system; (ii) *preferential attachment*: the probability $\prod_i$ that a new node will be connected to the node i (one of the m already existing nodes) depends on the degree k_i of the node i, in such a way that $\prod_i = k_i / \sum_j k_j$.

Models based on preferential attachment operates in the following way. Nodes are added one at a time. When a new node u has to be added to the network it creates m edges (m is a parameter and it is constant for all nodes). The edges are not placed uniformly at random but *preferentially*, i.e., probability that a new edge of u is placed to a node v of degree $d(v)$ is proportional to its degree, $p_u(v) \propto d(v)$. This simple behavior leads to power-law degree tails with exponent $\lambda = 3$. Moreover it also leads to low diameters. While the model captures the power-law tail of the degree distribution, it has other properties that may or may not agree with empirical results in real networks. Recent analytical research on average path length indicate that $\ell \sim ln(N)/lnln(N)$. Thus the model has much shorter l w.r.t. a random graph. The clustering coefficient decreases with the network size, following approximately a power-law $C \sim N^{-0.75}$. Though greater than those of random graphs, it depends on network size, which is not true for real-world social networks.

1.4 MOTIFS

A network *motif* M is a pattern of interconnection (a subgraph of n nodes) occurring in a graph G (directed and/or undirected) at numbers that are significantly higher than those in randomized graph. Milo et al. [266] introduced this concept in the study of small n motifs (n=3 and 4) in biological and other networks. Motifs are considered an important basic building block associated with specific functions within the global system. The statistical meaning of M is described by *Z-score*, defined as [266]:

$$Z_M = \frac{n_M - \langle n_M^{\text{rand}} \rangle}{\sigma_{n_M}^{\text{rand}}} \tag{15}$$

where n_M is the number of times the subgraph M appears in G, and $\langle n_M^{\text{rand}} \rangle$ and $\sigma_{n_M}^{\text{rand}}$ are, respectively, the mean and the standard deviation of the number of appearances in the randomized network ensemble [53].

2 MULTILAYER NETWORKS

Analyzing a set of entities by focusing on a single aspect may often provide approximate or incomplete information. This is particularly true when compared to the information that may be extracted as a result of an in-depth analysis which takes care of the behavior of the entities under many aspects.

Recently, thanks to the enhanced resolution in real data sets, a growing research area, thoroughly reviewed in [54, 217], is devoted to improve and generalize the instruments usually used for the analysis of real-world networks and apply them to the study of networks exhibiting more layers of connectivity. This is the case of social, transportation, metabolic and protein, neural, phone call and many other networks.

After the introduction of principal graph tools in the previews chapter, we can now focus on its natural mathematical model extension: multilayer networks. *Multilayer networks* incorporate multiple channels of connectivity in a system and embodies the natural mathematical environment to represent systems whose units are interconnected by means of different categories and interactions. Each channel (i.e. relationship, category, activity) is represented by a layer, and the same node (or entity) may have different interactions (coworkership, friendship, neighborship, etc.) with other nodes.

A particular case of multilayer network is constituted by *multiplex networks* we will discuss in the next Chapter. They provide a natural description of systems in which the same elementary units may interact through different types of relationships and in which each kind of connection is represented by a specific layer of the system.

The two principal works we refer to in this Chapter, are the reviews by Kivela et al. [217] and by Boccaletti et al. [54]. We introduce more in details, the mathematical formulation of multilayer networks based on tensor algebra developed by De Domenico et al. [121] and discuss several descriptors and diagnostics for both single-layer (*monoplex*) and multilayer networks using the tensorial framework.

2.1 LEVELS AND ASPECTS

In order to represent systems composed of networks with multiple levels or different type of relationships, we define a structure made of *levels* and *aspects* (or *dimensions*) in addition to nodes and edges. Moreover, a node may belong to any subset of levels and may exist pairs of connections between all possible combinations of nodes and layers (a node u in layer α can be connected to any node v in any layer β).

A level is intended to represent a particular type of interaction or relationship among the entities composing the system while the aspect includes multiple dimensions of connectivity that have to be considered simultaneously (different types of interactions or communication channels, subsystems, spatial locations, time dependent). For example, given a network in which an aspect is the kind of interaction

among nodes and another aspect is the time, we will represent the system by means of a set of levels which define all the possible types of interaction, and a set of levels which include the temporal dimension. The term *elementary level* refers to one of the possible elements of this set of levels, while the term *level* refers to a combination of elementary levels associated to more aspects. The aspects *kind of interaction* and *time stamp* are both examples of elementary layers while a combination of them represents a layer.

A multilayer network can have any number d of aspects. The sequence $\mathbf{L} = \{L_a\}_{a=1}^{d}$ describes the set of *elementary layers* defined by a aspects such that there is a set of elementary layers L_a for each aspect a. If $d = 0$ then the multilayer network M reduces to a single-layer network (*monoplex*). If $d = 1$, then M reduces to a single-aspect multilayer network (*multiplex*). Using sequence of elementary layers $\mathbf{L}$ it is possible to build a set of lievel in a multilayer network by considering all the possible combinations of elementary levels using the cartesian product $L_1 \times \cdots \times L_d$. To indicate if a specific node is present in a specific level we consider the set $V \times L_1 \times \cdots \times L_d$ which contains all the possible combination, and from this we define a subset $V_M \subseteq V \times L_1 \times \cdots \times L_d$ which contains only the combinations in which a node is actually present in the corresponding level. A *node-layer tuple* $(u, \alpha_1, \ldots, \alpha_d)$ represents node u on layer $(\alpha_1, \ldots, \alpha_d)$. For simplicity we denote the array of elementary layers using a bold typeface $\boldsymbol{\alpha} \equiv (\alpha_1, \ldots, \alpha_d)$ and we write tuple-layer node with $(u, \boldsymbol{\alpha}) \equiv (u, \alpha_1, \ldots, \alpha_d)$.

2.2 ADJACENCY AND INCIDENCE

As with monoplex networks, two nodes $(u, \boldsymbol{\alpha}), (v, \boldsymbol{\beta})$ are *adjacent* if there is a direct connection via an edge between a pair of node-layers. We can denote this relation with $(u, \boldsymbol{\alpha}) \sim (v, \boldsymbol{\beta})$. Two nodes are said *incident* if there is a connection between a node-layer and an edge. Edges that are incident to the same node-layer are also "incident" to each other. Multilayer network framework allow all of the possible types of edges between any pair of node-layers including ones in which a node is adjacent to a copy of itself in some other layer and ones in which a node is adjacent to some other node from another layer. The adjacencies are defined by an edge set $E_M \subseteq V_M \times V_M$ as a set of pairs of possible combinations of nodes and elementary layers.

2.3 FORMALIZATION

Summarizing the above, a *multilayer network* is formally defined as a quadruple:

$$M = (V_M, E_M, V, \mathbf{L})$$

where: $V \times L_1 \times \cdots \times L_d$ is the set of all the heterogeneous nodes in the network; $V_M \subseteq V \times L_1 \times \cdots \times L_d$ is the set of the node-layer combinations, that is the set of layers in which a node $v \in V$ is present; $E_M \subseteq V_M \times V_M$ is the edge set cointaining the set of pairs of possible combinations of nodes and elementary layers; $\mathbf{L} = \{L_a\}_{a=1}^{d}$

is the set of elementary layers defined by d aspects such that there is one elementary layer set L_a for each aspect a.

The first two elements of M constitute the tuple $G_M = (V_M, E_M)$, so a multilayer network can interpreted as a graph to whom nodes is associated the name of the corresponding layer. G_M is said underlying graph of multilayer network M.

2.4 BASICS

Now, we generalize some of the basic concepts from monoplex networks to multilayer networks.

Multilayer network is *undirected* if $((u,\boldsymbol{\alpha}),(v,\boldsymbol{\beta})) \in E_M \implies ((v,\boldsymbol{\beta}),(u,\boldsymbol{\alpha})) \in E_M$ and *weighted* if assigning weights function $w : E_M \to \mathbb{R}$ for the edges in the underlying graph $G_M = (V_M, E_M)$. We can disallow self-edges by requiring that $((u,\boldsymbol{\alpha}),(u,\boldsymbol{\alpha})) \notin E_M$. The set of edges can be partitioned by distinguishing among the inter-layers and the intra-layers: *intra-layer edges* $E_A = \{((u,\boldsymbol{\alpha}),(v,\boldsymbol{\beta})) \in E_M | \boldsymbol{\alpha} = \boldsymbol{\beta}\}$ and *inter-layer edges* $E_C = E_M \setminus E_A$. We define, moreover, *coupling edges* $E_{\tilde{C}} \subseteq E_C$ where $E_{\tilde{C}} = \{((u,\boldsymbol{\alpha}),(u,\boldsymbol{\beta})) \in E_C\}$ as edges for which the two nodes represent the same entity in different layers. From these definitions we obtain the *intra-layer graph* $G_A = (V_M, E_A)$, *inter-layer graph* $G_C = (V_M, E_C)$, and *coupling graph* $G_{\tilde{C}} = (V_M, E_{\tilde{C}})$.

Multilayer network is *node-aligned* or "fully interconnected" if all of the layers contain all nodes $V_M = V \times L_1 \times \cdots \times L_d$. It is *layer-disjoint* if each node exists in at most one layer: $(u,\boldsymbol{\alpha}),(u,\boldsymbol{\beta}) \in V_M \implies \boldsymbol{\alpha} = \boldsymbol{\beta}$. Couplings are *diagonal* if all of the inter-layer edges are between nodes and their counterparts in another layers: $E_{\tilde{C}} = E_C$. A diagonal multilayer network is *layer-coupled* if the coupling edges and their weights are independent of the nodes: if $((u,\boldsymbol{\alpha}),(u,\boldsymbol{\beta})) \in E_C$ and $(v,\boldsymbol{\alpha}),(v,\boldsymbol{\beta}) \in V_M$ then $((v,\boldsymbol{\alpha}),(v,\boldsymbol{\beta})) \in E_C$ and $w(((u,\boldsymbol{\alpha}),(u,\boldsymbol{\beta}))) = w(((v,\boldsymbol{\alpha}),(v,\boldsymbol{\beta})))$ for all $u,v,\boldsymbol{\alpha},\boldsymbol{\beta}$ The couplings are *categorical* if each node is adjacent to all of its counterparts in the other layers: $(u,\boldsymbol{\alpha}),(u,\boldsymbol{\beta}) \in V_M \implies ((u,\boldsymbol{\alpha}),(u,\boldsymbol{\beta})) \in E_M$. Couplings are *ordinal* if the layers are ordered and nodes are adjacent only to their counterparts in consecutive layers. Couplings are categorical with respect to a single aspect if each node is adjacent to all of its counterparts in layers that only differ in that aspect.

2.5 TENSOR REPRESENTATION

The main benefits of studying multilayer networks using tensor algebra gives concise mathematical representation, and it leads to natural generalisations of numerous network diagonostics from monoplex networks to multilayer networks. In this Section we present the mathematical formalization introduced by [121].

2.5.1 Indicial notation

De Domenico *et al*. mathematical formulation uses the covariant notation introduced by Ricci and Levi-Civita in Ref. [325]. In this notation, a row vector $\mathbf{a} \in \mathbb{R}^N$

is given by a covariant vector a_α (where $\alpha = 1, 2, \ldots, N$), and the corresponding contravariant vector a^α (i.e., its dual vector) is a column vector in Euclidean space.

Latin letters $i, j, \ldots$ indicate the i^{th} vector, the $(ij)^{\text{th}}$ tensor, etc. and Greek letters $\alpha, \beta, \ldots$ refer to the components of a vector or a tensor. With this terminology, $e^\alpha(i)$ is the α^{th} component of the i^{th} contravariant canonical vector $\mathbf{e}_i$ in $\mathbb{R}^N$, and $e_\alpha(j)$ is the α^{th} component of the j^{th} covariant canonical vector in $\mathbb{R}^N$.

Einstein notation is a summation convention, which is adopted to reduce the notational complexity in our tensorial equations, that is applied to repeated indices in operations that involve tensors. For example, we use this convention in the left-hand sides of the following equations:

$$A^\alpha_\alpha = \sum_{\alpha=1}^{N} A^\alpha_\alpha\,, \quad A^\alpha B_\alpha \;\; = \sum_{\alpha=1}^{N} A^\alpha B_\alpha\,, \quad A^\alpha_\beta B^\beta_\gamma = \sum_{\beta=1}^{N} A^\alpha_\beta B^\beta_\gamma\,, \quad A^\alpha_\beta B^\beta_\alpha \;\; = \sum_{\alpha=1}^{N}\sum_{\beta=1}^{N} A^\alpha_\beta B^\beta_\alpha\,,$$

It is straightforward to use this convention for the product of any number of tensors of any order. Repeated indices, such that one index is a subscript and the other is a superscript, is equivalent to perform a tensorial operation known as a *contraction*. In the following, the t-th power of rank-4 tensors, defined by multiple tensor multiplications:

$$(A^t)^{i\alpha}_{j\beta} = (A)^{i\alpha}_{j_1\beta_1}(A)^{j_1\beta_1}_{j_2\beta_2}\ldots(A)^{j_{t-1}\beta_{t-1}}_{j\beta}$$

Contracting indices reduces the order of a tensor by 2. For instance, the contraction of the 2^{nd}-order tensor A^α_β is the scalar A^α_α, and the 2^{nd}-order tensors $A^\alpha_\beta B^\beta_\delta$ and $A^\delta_\beta B^\gamma_\delta$ are obtained by contracting the 4^{th}-order tensor $A^\alpha_\beta B^\gamma_\delta$.

2.5.2 Adjacency tensor

Let $\mathfrak{I} = \{\mathbf{e}_1, \ldots, \mathbf{e}_N\}$ be a canonical base of the vectorial space $\mathbb{R}^N$ in which $\mathbf{e_i} \equiv (0, \ldots, 0, 1, 0, \ldots, 0)^T$ takes on the value 1 in the position i^{th} and 0 otherwise, we may associate to every node v_i ($i = 1, \ldots, N \in \mathbb{N}$) of a graph $G = (V, E)$ a vector $\mathbf{e_i}$ of the base $\mathfrak{I}$ and establish relations among them via the Kronecker product $\mathbb{R}^N \otimes \mathbb{R}^N = \mathbb{R}^{N\times N}$. Let define by $\mathbf{E}_{ij} = \mathbf{e}_i \otimes \mathbf{e}_j^T$ (where $i, j = 1, 2, \ldots, N$) the 2^{nd}-order canonical tensors and with w_{ij} the intensity of the relationship from node v_i to node v_j, we can write the *relationship tensor* as:

$$\mathbf{W} = \sum_{i,j=1}^{N} w_{ij}\mathbf{E}_{ij} = \sum_{i,j=1}^{N} w_{ij}\mathbf{e}_i \otimes \mathbf{e}_j^T\,, \quad \mathbf{W} \in \mathbb{R}^N \otimes \mathbb{R}^N\,. \tag{16}$$

The relationships can be undirected if $w_{ij} = w_{ji}$, directed if w_{ij} is different from w_{ji} and weighted for any value of $w_{ij} \neq 0$.

$\mathbf{W}$ corresponds to an $N \times N$ standard graph matrix representation of N nodes n_i so called *adjacency tensor*. To describe such standard networks which have only a single type of edge and distinguish it from the multilayer or multiplex network, we will use the term *monoplex networks*.

With Ricci e Levi-Civita notation, the adjacency tensor $\mathbf{W}$ can be represented as a linear combination of tensors in the canonical basis:

$$W_\beta^\alpha = \sum_{i,j=1}^{N} w_{ij} e^\alpha(i) e_\beta(j) = \sum_{i,j=1}^{N} w_{ij} E_\beta^\alpha(ij)\,, \tag{17}$$

where $E_\beta^\alpha(ij) \in \mathbb{R}^{N\times N}$ indicates the tensor in the canonical basis that corresponds to the tensor product of the canonical vectors assigned to nodes n_i and n_j (i.e., it is $\mathbf{E}_{ij}$).

In order to reduce the complexity of some tensorial equations in the applications, it is possible to avoid using canonical vectors and explicit tensors. A single-layer network represented by a rank-2 mixed adjacency tensor W_β^α can be indicated by W_j^i, where the indices i and j as nodes and W_j^i would indicate intensity of the relationship between them [131]. W_j^i represents the well-known adjacency matrix of a graph and the classical notation for the weight w_{ij} of the link between i and j corresponds to W_j^i. With this "abuse of notation" we may consider W_j^i as a rank-2 tensor.

2.5.3 Descriptors

We now introduce the most basic measures of monoplex networks using the tensorial formulation so far described.

Degree centrality

Let *1-vector* $u^\alpha = (1,\ldots,1)^{\mathrm{T}} \in \mathbb{R}^N$ whose components are all equal to 1, and let $U_\alpha^\beta = u_\alpha u^\beta$ be the 2^{nd}-order tensor whose elements are all equal to 1, the *degree centrality vector* is $k_\beta = W_\beta^\alpha u_\alpha$ in the space $\mathbb{R}^N$. The degree centrality of node n_i is obtained by projecting the degree vector onto the i^{th} canonical vector: $k(i) = k_\beta e^\beta(i)$.

In directed networks, in-degree centrality and out-degree centrality are represented using different tensor products. The *in-degree centrality vector* is $k_\beta = W_\beta^\alpha u_\alpha$, whereas the *out-degree centrality vector* is $k^\alpha = W_\beta^\alpha u^\beta$. The in-degree centrality of node n_i is $k_{\text{in}}(i) = W_\beta^\alpha u_\alpha e^\beta(i)$, the out-degree centrality of node n_i is $k_{\text{out}}(i) = W_\beta^\alpha u^\alpha e_\beta(i)$.

Mean degree and variance of the degree

The mean degree is $\langle k\rangle = (U_\rho^\rho)^{\ 1} k_\beta u^\beta$, the second moment of the degree is $\langle k^2\rangle = (U_\rho^\rho)^{-1} k_\beta k^\beta$, and the variance of the degree is $\mathrm{var}(k) = (U_\rho^\rho)^{-1} k_\beta k^\beta \quad (U_\rho^\rho)^{-2} k_\alpha k_\beta U^{\alpha\beta}$.

Clustering coefficient

In tensor notation, we may extract the number of various cycles having length m starting and ending in a same node v_i by considering the expression

$$W_{\xi_1}^\alpha W_{\xi_2}^{\xi_1} W_{\xi_3}^{\xi_2} \ldots W_{\xi_m\ 1}^{\xi_m\ 2} W_\beta^{\xi_m\ 1} e_\alpha(i) e^\beta(j)\,, \tag{18}$$

which reduces to $W_\rho^\alpha W_\sigma^\rho W_\beta^\sigma e_\alpha(i) e^\beta(i)$ for $j = i$ and $m = 3$. It is equivalent to the $i-th$ position on the diagonal of A^m, using the notation presented in the previous

Section. The local clustering coefficient can be defined by dividing the number of 3-cycles by the number of 3-cycles in a network for which the neighborhood of the node v_i is completely connected

$$c(W_\beta^\alpha, i) = \frac{W_\rho^\alpha W_\sigma^\rho W_\beta^\sigma e_\alpha(i) e^\beta(i)}{W_\rho^\alpha F_\sigma^\rho W_\beta^\sigma e_\alpha(i) e^\beta(i)}, \tag{19}$$

where $F_\sigma^\rho = U_\sigma^\rho - \delta_\sigma^\rho$ is the adjacency tensor corresponding to a network that includes all edges except for self-loops.

To calculate a global clustering coefficient of a network, we need to calculate both the total number of 3-cycles and the total number of 3-cycles that one obtains when the second step of the walk occurs in a complete network

$$c(W_\beta^\alpha) = \frac{W_\rho^\alpha W_\sigma^\rho W_\alpha^\sigma}{W_\rho^\alpha F_\sigma^\rho W_\alpha^\sigma}. \tag{20}$$

Eigenvector centrality

L'eigenvector centrality of a node in un undirected graph, represented by an adjacency matrix $\mathbf{A}$, is the solution of the equation $\mathbf{Av} = \lambda_1 \mathbf{v}$ where λ_1 be the largest ("leading") eigenvalue of $\mathbf{A}$. The components of $\mathbf{v}$, a leading eigenvector of $\mathbf{A}$, give eigenvector centralities of the nodes [58].

In tensorial formulation, the *eigenvector centrality vector* is a solution of the tensorial equation

$$W_\beta^\alpha v_\alpha = \lambda_1 v_\beta, \tag{21}$$

and $v_\alpha e^\alpha(i)$ gives the eigenvector centrality of node n_i. For directed networks, there are two leading eigenvectors, for in-going centrality and out-going centrality pertanto occorre considerare the covariant and contravariant calculations. Moreover, for directed networks, if there are nodes with only outgoing edges eigenvector centrality of those nodes is zero.

Katz centrality

The Katz centrality [213] tries to solve problems consequent to the calculation of the eigenvector centrality in directed graphs by assigning a small amount b of centrality to each node before calculation. By modifying the equation of the eigenvector centrality we may extract the leading-eigenvector $\mathbf{v} = a\mathbf{Av} + b\mathbf{1}$, where $\mathbf{1}$ is a vector in which each entry is a 1. Using tensorial notation, we obtain

$$v_\beta = \left(\delta_\beta^\alpha - aW_\beta^\alpha\right)^{-1} u_\alpha. \tag{22}$$

To calculate Katz centrality from Eq. (22), we need to calculate the tensor inverse T_β^α, which satisfies the equation $T_\beta^\alpha \left(\delta_\sigma^\beta - aW_\sigma^\beta\right) = \delta_\sigma^\alpha$.

2.6 MULTILAYER ADJACENCY TENSOR

In order to use the tensorial notation for representing multilayer systems in which heterogeneous relationships may coexist between pairs of nodes in each layer and between pairs of nodes belonging to distinct layers, it is necessary to extend the formalization of the *monoplex* case described in the previous Section. It is necessary to consider the possibility that a node v_i can be connected to any other node v_j in the same layer $\tilde{k}$ and that a node v_i from layer $\tilde{h}$ can be connected to any other node v_j in any other layer $\tilde{k}$.

We use the 2nd-order tensor $\mathbf{W}_{\beta}^{\alpha}(\tilde{k})$ (*intra-layer adjacency tensor*) for the relationships between nodes within the *same* layer $\tilde{k}$ and the 2nd-order tensor $\mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k})$ (*interlayer adjacency tensor*) to encode all relationships among nodes in multiple layers. Note that $\mathbf{C}_{\beta}^{\alpha}(\tilde{k}\tilde{k}) \equiv \mathbf{W}_{\beta}^{\alpha}(\tilde{k})$ in the case in which a pair of layers represents the same layer $\tilde{k}$.

To construct the *interlayer adjacency tensor*, in a similar way as the approach followed for the adjacency tensor in monoplex networks, we define the 2nd-order tensors

$$\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k}) = \mathbf{e}^{\tilde{\gamma}}(\tilde{h})\mathbf{e}_{\tilde{\delta}}(\tilde{k})$$

that represent the canonical basis of the space $\mathbb{R}^{L\times L}$. Vectors $\mathbf{e}^{\tilde{\gamma}}(\tilde{k})$, where $\tilde{\gamma},\tilde{k} = 1,2,\ldots,L$, are the vectors of the canonical basis in the space $\mathbb{R}^{L}$. Greek and Latin index indicates the components of the vector and the k^{th} canonical vector, respectively. The tilde symbol distinguishes these indices from the greek indices that correspond to nodes.

We can write the *multi-layer adjacency tensor* using a tensor product between the adjacency tensors $\mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k})$ and the canonical tensors $\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k})$ obtaining a 4^{th}-order (i.e., rank-4) tensor

$$\mathbf{M}_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} = \sum_{\tilde{h},\tilde{k}=1}^{L} \mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k})\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k}) \tag{23}$$

$\mathbf{M}_{j\beta}^{i\alpha}$ might be simply thought as a higher-order matrix with four indices that encoding the intensity of the relationship between a node i in layer α and a node j in layer β [131].

As we shown before the generic inter-layer tensor $\mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k}) \equiv \mathbf{W}_{\beta}^{\alpha}(\tilde{k})$ when $\tilde{h} = \tilde{k}$ can be rewritten as

$$\mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k}) = \sum_{i,j=1}^{N} w_{ij}(\tilde{h}\tilde{k})\mathbf{E}_{\alpha}^{\beta}(\tilde{h}\tilde{k}) \tag{24}$$

where $w_{ij}(\tilde{h}\tilde{k}) \in \mathbb{R}$ is the intensity of relationship between a node v_i in layer $\tilde{h}$ and node v_j in layer $\tilde{k}$. Then, by defining the 4^{th}-order (i.e., rank-4) tensors of the canonical basis in the space $\mathbb{R}^{N\times N\times L\times L}$

$$\begin{aligned}\mathcal{E}_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}(ij\tilde{h}\tilde{k}) &= \mathbf{E}_{\alpha}^{\beta}(ij)\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k}) = \\ &= \mathbf{e}^{\alpha}(i)\mathbf{e}_{\beta}(j)\mathbf{e}^{\tilde{\gamma}}(\tilde{h})\mathbf{e}_{\tilde{\delta}}(\tilde{k})\end{aligned} \tag{25}$$

and, by replacing the previous expressions in (23) we obtain the general expression of the multilayer adjacency tensor as

$$\begin{aligned}\mathbf{M}_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} &= \sum_{\tilde{h},\tilde{k}=1}^{L} \mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k})\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k}) \\ &= \sum_{\tilde{h},\tilde{k}=1}^{L} \left[\sum_{i,j=1}^{N} w_{ij}(\tilde{h}\tilde{k})\mathbf{E}_{\beta}^{\alpha}(ij)\right] \mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k}) \\ &= \sum_{\tilde{h},\tilde{k}=1}^{L}\sum_{i,j=1}^{N} w_{ij}(\tilde{h}\tilde{k})\mathcal{E}_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}(ij\tilde{h}\tilde{k})\,, \end{aligned} \tag{26}$$

It is possible to restrict the definition to the *multiplex* only by imposing that all the tensors *inter-layer* $\mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k})$ are diagonal, namely by considering only the relationships between pairs of nodes in different layers in the case in which they are one the counterpart of the other in the layers $\tilde{h}$ and $\tilde{k}$.

2.6.1 Flattening

Multilayer adjacency tensor can be represented as a special rank-2 tensor obtained by a process called *flattening* (also known as *unfolding* and *matricization*) [223]. The elements of $M_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}$, which is defined in the space $\mathbb{R}^{N\times N\times L\times L}$, can be represented as an $N^2 \times L^2$ or an $NL \times NL$ matrix. Flattening a multi-layer adjacency tensor can be very helpful when writing software implementations of computational algorithms.

2.6.2 Contraction

The contraction of a tensor is obtained by setting unlike indices (covariant and controvariant) equal and summing according to the Einstein summation convention. Contraction reduces the tensor rank by 2. We can obtain the number of nodes in a network by contracting the Kronecker tensor $N = \delta_{\alpha}^{\alpha} = \delta_1^1 + \cdots + \delta_N^N$. For unweighted networks, we can calculate the number of edges by the scalar product $\mathbf{W}_{\beta}^{\alpha}u_{\alpha}^{\beta}$ where $\mathbf{W}_{\beta}^{\alpha}$ is the adjacency tensor and the dual 1-tensor u_{α}^{β} (the rank-2 tensor with all components equal to 1).

2.6.3 Layer extraction

The operation of extracting a specific level $\tilde{\ell}$ from a multilayer networks, by using the tensorial algebra, is equivalent to project the multilayer adjacency tensor $\mathbf{M}_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}$ on the second-order canonical tensor $\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{\ell}\tilde{\ell})$ corresponding to the level $\tilde{\ell}$ one wants to extract. Since the 2$^{\text{nd}}$-order canonical tensors in $\mathbb{R}^{L\times L}$ form an orthonormal basis, the product $\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k})\mathbf{E}_{\tilde{\gamma}}^{\tilde{\delta}}(\tilde{\ell}\tilde{\ell})$ will be equal to 1 for $\tilde{h} = \tilde{k} = \tilde{\ell}$ and 0 otherwise.

Therefore, by employing equation (26) we will have

$$\begin{aligned}\mathbf{M}_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} &= \left[\sum_{\tilde{h},\tilde{k}=1}^{L} \mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k})\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k})\right]\mathbf{E}_{\tilde{\gamma}}^{\tilde{\delta}}(\tilde{\ell}\tilde{\ell}) \\ &= \sum_{\tilde{h},\tilde{k}=1}^{L} \mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k})\mathbf{E}_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k})\mathbf{E}_{\tilde{\gamma}}^{\tilde{\delta}}(\tilde{\ell}\tilde{\ell}) \\ &= \mathbf{C}_{\beta}^{\alpha}(\tilde{\ell}\tilde{\ell}) = \mathbf{W}_{\beta}^{\alpha}(\tilde{\ell})\,, \end{aligned} \tag{27}$$

which correspond to the adjacency tensor of layer $\tilde{\ell}$. In a similar way, it is possible to extract a tensor that provides inter-layer relationships. In applications may be useful, for example, extracting and comparing inter-layer tensors from a multilayer adjacency tensor to compare the strengths of the coupling between them or in the calculus of multi-layer clustering coefficient.

2.6.4 Projection

One of the most common operation on multiplex networks is the aggregation of all relations of the network in one only layer, thus building a monoplex network. The projection is accomplished by tensing (i.e. multiplying) the multilayer adjacency tensor $\mathbf{M}_{\beta\delta}^{\alpha\gamma}$ with the unitary second-order tensor, the dual 1-tensor $u_{\tilde{\gamma}}^{\tilde{\delta}}$, thus obtaining the *projected monoplex network*

$$\mathbf{P}_{\beta}^{\alpha} = \mathbf{M}_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} u_{\tilde{\gamma}}^{\tilde{\delta}} = \sum_{\tilde{h},\tilde{k}=1}^{L} \mathbf{C}_{\beta}^{\alpha}(\tilde{h}\tilde{k}) \tag{28}$$

In the projection, at variance with respect of the *projected monoplex network* which will be described in the next Subsection, both intra-layer and inter-layer relationships are considered.

2.6.5 Overlay

The *overlay monoplex network* is a structure similar to the *projected monoplex network* obtained by summing the edges of all layers per every pair of nodes. L'*overlay monoplex network* (o aggregate network) can be obtained by contracting the layer indices of the multilayer adjacency tensor

$$\mathbf{O}_{\beta}^{\alpha} = \mathbf{M}_{\beta\tilde{\gamma}}^{\alpha\tilde{\gamma}} = \sum_{\ell=1}^{L} \mathbf{W}_{\beta}^{\alpha}(\tilde{\ell})\,. \tag{29}$$

In the *overlay network*, at variance of the *projected monoplex network*, information is lost about inter-layer connections, which in some applications may be of interest. Consider, for example, multimodal transportation systems in which nodes are the places and layers represent different transportation systems. The mathematical model must take into account inter-layer transfers between the replicas of the same node at different layers.

2.6.6 Supra-adjacency matrix

A monoplex network can be represented by means of an adjacency matrix $A^{N\times N}$ whose entries a_{ij} have value 1 is does exist an edge between nodes i and j, zero otherwise. A mathematical tool able to represent complex structures such as multilayer networks in which do coexist multiple relationship (intra-layer and inter-layer) among nodes of the systems belonging to one or more layers is the *supra-adjacency matrix*. For simplicity,we will assume that the interlayer and intralayer networks are undirected, globally connected, and self-loops free.

The supra-adjacency matrix $\mathcal{A}$ of a multi-layer network $\mathcal{M}$ with N nodes and L layers can be defined in a block-matrix structure as

$$\mathcal{A} = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1L} \\ A_{21} & A_{22} & \cdots & A_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ A_{L1} & A_{L2} & \cdots & A_{LL} \end{pmatrix} \in \mathbb{R}^{NL\times NL}, \tag{30}$$

in which the diagonal matrices are the adjacency matrices of the graphs of each layer of the network, while the off-diagonal matrices represent the the connection between couple of layers (inter-layer edges). As we have said in a previously paragraph (2.6.1), the procedure of assigning a matrix to a multilayer network is called flattening, unfolding or matricization.

Similarly to the case of the supra-adjacency matrix, we may define the *supra-Laplacian* matrix of a multi-layer network G having N nodes and L layers, by considering the contributed of each layer G_A and that of all possible inter-layer graph G_C [351].

The connections of each layer α (intra-layer edges) are represented by the adjacency matrix $\mathbf{A}^\alpha \in \mathbb{R}^{N\times N}$ whose Laplacian $\mathbf{L}^\alpha$ is the usual Laplacian matrix of network. The elements of $\mathbf{L}^\alpha$ are $L^\alpha = D^\alpha - A^\alpha$, where $D^\alpha_{ij} = s^\alpha_i \delta_{ij}$ (with $s^\alpha_i = \sum_j a^\alpha_{ij}$) is the diagonal matrix of the nodes interlayer strength.

Interlayer network $\mathbf{A}^I \in \mathbb{R}^{N\times N}$ whose components represent the strength of the connection between a pair of layers, have the associated interlayer Laplacian $L^I = D^I - A^I$

Thus, the *supra-Laplacian* $\mathcal{L}$ matrix of a multi-layer network can be defined considering the two contributes that we have seen before [351]:

$$\mathcal{L} = \mathcal{L}^L + \mathcal{L}^I$$

where $\mathcal{L}^L$ is the supra-Laplacian of the independent layers and $\mathcal{L}^I$ stands for interlayer supra-Laplacian.

$\mathcal{L}^L$ is the direct sum of the intralayer Laplacians:

$$\mathcal{L}^L = \begin{pmatrix} \mathbf{L^1} & 0 & \dots & 0 \\ 0 & \mathbf{L^2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{L^L} \end{pmatrix} = \bigoplus_{\alpha=1}^{L} L^\alpha, \tag{31}$$

while the interlayer supra-Laplacian is a Kronecker product of the interlayer Laplacian and the identity matrix $I^{N\times N}$

$$\mathcal{L}^I = L^I \otimes I$$

$$\mathcal{L}^I = \begin{pmatrix} 0 & \mathbf{L}_2^1 & \dots & \mathbf{L}_L^1 \\ \mathbf{L}_1^2 & 0 & \dots & \mathbf{L}_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{L}_1^L & \mathbf{L}_2^L & \dots & 0 \end{pmatrix} = L^I \otimes I, \tag{32}$$

The decomposition above is very important for the discovery of several spectral properties of the multi-layer networks, which are shown in detail in [351].

2.6.7 Multi-layer adjacency tensor descriptors

Degree centrality

The *multi-degree centrality vector* can be introduced by performing the same projections from the case of monoplex networks using 1-tensors of an appropriate order

$$\begin{aligned} K^\alpha &= \left[M^{\alpha\tilde{\gamma}}_{\beta\tilde{\delta}} u^{\tilde{\delta}}_{\tilde{\gamma}} \right] u^\beta = \mathbf{P}^\alpha_\beta u^\beta \\ &= \left[\sum_{\tilde{h},\tilde{k}=1}^{L} C^\alpha_\beta(\tilde{h}\tilde{k}) \right] u^\beta \\ &= \sum_{\tilde{h},\tilde{k}=1}^{L} C^\alpha_\beta(\tilde{h}\tilde{k}) u^\beta \\ &= \sum_{\tilde{h},\tilde{k}=1}^{L} k^\alpha(\tilde{h}\tilde{k}), \end{aligned} \tag{33}$$

where $k^\alpha(\tilde{h}\tilde{k})$ is the degree centrality vector that corresponds to inter-layer adjacency tensor $C^\alpha_\beta(\tilde{h}\tilde{k})$ delle connections between layers $\tilde{h}$ and $\tilde{k}$. The multidegree centrality vector can be computed by taking into account that all inter-layer adjacency tensor $C^\alpha_\beta(\tilde{h}\tilde{k})$ are diagonal by definition, and that therefore $C^\alpha_\beta(\tilde{h}\tilde{k})u^\beta = diag(C^\alpha_\beta(\tilde{h}\tilde{k}))$ per $\tilde{h} \neq \tilde{k}$.

Even in the special case of multiplex networks, it is already evident that K^α differs from the degree centrality vector that one would obtain by projection of all layers of a multi-layer network onto a single weighted network.

The definitions of mean degree, second moment, and variance are analogous to the corresponding monoplex network counterparts, except that one uses K^α instead of k^α.

Clustering coefficient

Global clustering coefficient on a multi-layer adjacency tensor is defined by generalizing Eq. (20) for 4^{th}-order tensors:

$$C(M^{\alpha\tilde{\gamma}}_{\beta\tilde{\delta}}) = \mathcal{N}^{-1} \frac{M^{\alpha\tilde{\gamma}}_{\beta\tilde{\delta}} M^{\beta\tilde{\delta}}_{\epsilon\tilde{\eta}} M^{\epsilon\tilde{\eta}}_{\alpha\tilde{\gamma}}}{M^{\alpha\tilde{\gamma}}_{\beta\tilde{\delta}} F^{\beta\tilde{\delta}}_{\epsilon\tilde{\eta}} M^{\epsilon\tilde{\eta}}_{\alpha\tilde{\gamma}}}, \tag{34}$$

where $F^{\beta\tilde{\delta}}_{\epsilon\tilde{\eta}} = U^{\beta\tilde{\delta}}_{\epsilon\tilde{\eta}} - \delta^{\beta\tilde{\delta}}_{\epsilon\tilde{\eta}}$ is the adjacency tensor of a complete loop-free multi-layer network and the normalization factor $\mathcal{N}$ but ensures that Eq. (34) is well-defined for both weighted and unweighted multi-layer networks.

This contraction allows us to count the number of different triangles, including those in which the underlying walk crosses any combinations of inter- or intra-layer relationships.

Another approach proposed in [121] to calculating a global clustering coefficient of a multi-layer network is to project it onto a overlay network O^{α}_{β} defined by Eq.(29) and then calculating a clustering coefficient for the resulting network. In this case, we obtain

$$c(O^{\alpha}_{\beta}) = \mathcal{M}^{-1} \frac{M^{\alpha\tilde{\gamma}}_{\beta\tilde{\gamma}} M^{\beta\tilde{\delta}}_{\epsilon\tilde{\delta}} M^{\epsilon\tilde{\eta}}_{\alpha\tilde{\eta}}}{M^{\alpha\tilde{\gamma}}_{\beta\tilde{\gamma}} F^{\beta\tilde{\delta}}_{\epsilon\tilde{\delta}} M^{\epsilon\tilde{\eta}}_{\alpha\tilde{\eta}}}, \tag{35}$$

where $\mathcal{M} = \max_{\alpha,\beta} \{ M^{\alpha\tilde{\gamma}}_{\beta\tilde{\gamma}} \} / L$.

Eigenvector centrality

The generalization of the measure of the eigenvector centrality to the multilayer case is not immediate because of the different algebraic structure with respect to the monoplex case. Various indices have been proposed to take into account the informative complexity managed by the network. The approach presented in [352], for example, consists in calculating separately the eigenvector centrality for every layer, by constructing the vector that codes the centrality of every node in each layer. By applying a linear transformation these vectors are then aggregated in a single tensor. This formulation does not take into account the inter-layer relationships.

An approach which considers all interdependencies of the multi-layer structure is described in [121].

The *eigentensor* $\Theta_{i\alpha}$ of tensor $M^{i\alpha}_{j\beta}$ encoding the centrality of each node in each layer and can be obtained as the solution of the tensorial equation:

$$M^{i\alpha}_{j\beta} \Theta_{i\alpha} = \lambda_1 \Theta_{j\beta}, \tag{36}$$

where λ_1 is the largest eigenvalue. The eigentensor can be obtained by means of an iterative procedure, as the power method in the case of monoplexes.

3 MULTIPLEX NETWORKS

A multiplex network is a special type of multi-layer network in which the only possible types of inter-layer connections are between a node in a given layer to its counterpart nodes in the other layers. Social network and transportation system are two classical examples of a multiplex networks in which the different layers represent different types of edges (social relationships or transport modalities, respectively).

In multiplex network, the associated inter-layer adjacency tensor is diagonal and connections between a node and its counterparts can have different weights for different pairs of layers. Inter-layer connections can also be different for different entities in a network [38, 121].

When studying multiplex networks it is important to decide if inter-layer transactions (communications, flows) must be considered between the replicas of the same node at different layers. In this case, it is convenient to model the structure of the system through an 4^{th}-tensor $\mathcal{M}_{i\alpha}^{i\beta}$. If the "weight" of inter-layer relationships may be ignored, in literature exist some approaches to model a multiplex network, as reviewed in [54, 217]. In this Chapter the model and the notation described in [37, 38] will be used, which is based on a simpler formulation of order-3 tensors.

3.1 PRELIMINARIES

A multiplex network $\mathcal{M}$ with N nodes and M different types of relations can be represented by M graphs, or layers. The structure of the system is described by the set of adjacency matrices

$$\mathcal{M} \equiv \boldsymbol{A} = \{A^{[1]}, \ldots, A^{[M]}\}, \tag{37}$$

where $A^{[\alpha]} = \{a_{ij}^{[\alpha]}\}$, with $a_{ij}^{[\alpha]} = 1$ if i and j share a bond of type α and $a_{ij}^{[\alpha]} = 0$ otherwise. Similarly, a weighted multiplex networks can be described by a set of weighted adjacency matrices $\boldsymbol{W} = \{W^{[1]}, \ldots, W^{[M]}\}$, with $W^{[\alpha]} = \{w_{ij}^{[\alpha]}\}$ and $w_{ij}^{[\alpha]}$ the weight of the link between node i and j. The replicas of the same node are identified across layers. Social systems can be naturally cast within this framework, where different layers can represent for instance different interaction channels among the same nodes (e.g., phone calls, meeting communication, email exchange, online chat, etc.), but the different replicas are just a mathematical representation of the same individual in each of the M contexts.

3.2 DESCRIPTORS

To extend the measures of complex networks theory to the case of multilayer network is a complicated task. The problem requires not only the mere mathematical translation, i.e. from monoplex network to fourth-order tensor formulation. It is necessary take into account the meaning of singular structure features (centrality measures, clustering coefficient, path distance etc.) even in the multilayer model. Below, we will focus on multiplex networks descriptors.

3.2.1 Degree

The degree of a node i on a given layer is $k_i^{[\alpha]} = \sum_j a_{ij}^{[\alpha]}$, from which follows that il grado di un nodo in ciascun layer puÃš essere compreso tra $0 \leq k_i^{[\alpha]} \leq N \quad 1 \; \forall i, \forall \alpha$. The degree of node i in a multiplex network is the vector

$$\boldsymbol{k}_i = (k_i^{[1]}, ..., k_i^{[M]}), \quad i = 1, \ldots, N \tag{38}$$

The total number of edges on layer α denoted by $K^{[\alpha]}$ is given by the sum $\sum_i k_i^{[\alpha]} = 2K^{[\alpha]}$[1] [38].

3.2.2 Overlapping degree

Since the node degree in a multiplex network is a vector, there is not a clear ordering in $\mathbb{R}^M$ that could produce a node ranking according to their relevance [38]. For this reason information contained in the vector can be aggregated thus obtaining the *overlapping degree* of the node $i \in N$ as

$$o_i = \sum_{\alpha=1}^{M} k_i^{[\alpha]}, \tag{39}$$

.

3.2.3 Eigenvector centrality

To calculate eigenvector centralities in multiplex networks [352] it is necessary to consider the eigenvector centrality $\mathbf{c}_\alpha = (c_1^{[\alpha]}, \cdots, c_N^{[\alpha]})$ in each layer $1 \leq \alpha \leq M$ separately. In this way, for every node the eigenvector centrality $\mathbf{c}_i$ is another vector $\mathbf{c}_i = (c_i^{[1]}, \cdots, c_i^{[M]})$ where each coordinate is the centrality in the corresponding layer. The *independent layer* eigenvector-like centrality of $\mathcal{M}$ is then the matrix $C = \mathbf{c}_1^T \quad \mathbf{c}_2^T \mid \cdots \mid \mathbf{c}_L^T \in \mathbb{R}^{N \times L}$. As for the degree indicators, many centrality measure of each node can be obtained by using an aggregation measure $f(c_i)$ such as the sum, the maximum, or the ℓ_p-norm. The main limitation of this parameter is that it does not fully consider the multilevel interactions between layers and its influence in the centrality of each node.

[1] Authors use lowercase letters to denote node properties and capital letters for properties obtained by summing over the nodes or the edges, either at the level of single layer or at the level of the whole system.

3.2.4 Shannon entropy

It can be useful to study the distribution of degree vectors in all layers of the multiplex network: we could have nodes exhibiting a high degree only in some layers, or other nodes that have a constant degree over all layers. The information about the heterogeneity of the distribution of a node's connections across layers is provided by the Shannon entropy of the degree vector [38]

$$H_i = -\sum_{\alpha=1}^{M} \frac{k_i^{[\alpha]}}{o_i} \ln\left(\frac{k_i^{[\alpha]}}{o_i}\right). \tag{40}$$

This quantity may have a minimum zero value when all the edges incident to node v_i lay on one layer only, while assuming its maximum value when these edges are uniformly distributed over all layers of the multiplex network.

3.2.5 Partecipation coefficient

The heterogeneity of the number of neighbours of node i across the layers can be measured through a measure similar to entropy, the multiplex participation coefficient [38]

$$P_i = \frac{M}{M-1}\left[1 - \sum_{\alpha=1}^{M}\left(\frac{k_i^{[\alpha]}}{o_i}\right)^2\right], \tag{41}$$

where $P_i = 1$ when the links incident on node i are equally distributed across the layers, and $P_i = 0$ when a node is only active on one layer. It assumes values between 0 and 1, respectively when all edges incident in v_i lay on the same level, and when v_i has exactly the same number of edges in each of the M layers of the multiplex networks.

3.3 CORRELATION

Multiplex networks encode significantly more information than their single layers taken in isolation, since they include correlations between the role of the nodes in different layers and between statistical properties of the single layers [38, 54]. For this reason, a number of measures have been developed. For example, the *interlayer degree correlation* indicates if a node which is an hub (or viceversa, which has little connections) in a layers, exhibits the same features in other layers; the *overlap of links* captures if node connectivity patterns can be correlated in two or more layers like, for example, in a social network where friends communication through multiple means (phone, email, instant messaging) imply that each network has a significant overlap which others; node and layer *pairwise multiplexity* where nodes are not all active in all layers and two nodes or two layers can have correlated activity patterns.

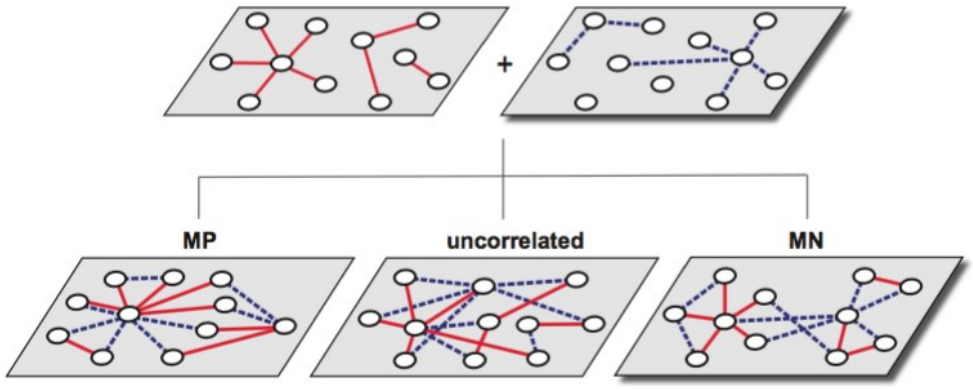


Figure 1: Illustrations of three patterns of interlayer degree-correlated multiplex networks: the maximally positive (MP), uncorrelated, and maximally negative (MN) cases of two layers. Taken from original source [236]

3.3.1 Interlayer degree correlation

To calculate the interlayer degree correlation between a layer α and a layer β we can measure the Pearson correlation coefficient

$$\rho_{\alpha\beta} = \frac{\langle k_i^{[\alpha]} k_i^{[\beta]} \rangle - \langle k_i^{[\alpha]} \rangle \langle k_i^{[\beta]} \rangle}{\sigma_\alpha \sigma_\beta}, \tag{42}$$

where k denotes the degree of the same node in different layers and σ is the standard deviation. Other interlayer correlation models such as Spearman or Kendall's have also been considered as variants [38, 298].

3.3.2 Overlap link correlation

The *total overlap* $O^{\alpha\beta}$ between two layers α and β [50] is defined as the total number of links that are in common between layer α and layer β:

$$O^{\alpha\beta} = \sum_{i<j} a_{ij}^{\alpha} a_{ij}^{\beta}, \tag{43}$$

where $\alpha \neq \beta$.

The *local overlap* $o_i^{\alpha\beta}$ between two layers α and β [50], defined as the total number of neighbors of node i that are neighbors in both layer α and layer β:

$$o_i^{\alpha\beta} = \sum_{j=1}^{N} a_{ij}^{\alpha} a_{ij}^{\beta}. \tag{44}$$

To investigate the effect of interlayer degree correlation,we can compare three specific patterns of correlated coupling, the maximally positive (MP) coupling, random (uncorrelated) coupling, and the maximally-negative (MN) coupling [237]. Given two layers, the MP-coupled multiplex is build by connecting the nodes of the same degree-rank from each layer (see Figure 1). Conversely, the MN coupling is constructed

connecting nodes with dissimilar degree. The uncorrelated coupling version is obtained by randomly connecting nodes from different layers without taking into account their degree [236].

3.4 LAYER PROPERTIES

The activity of a layer α of a multiplex network is correlated to the patterns of node activities at that layer. It is represented by the *layer-activity vector* [298]

$$\boldsymbol{d}^{[\alpha]} = \{b_1^{[\alpha]}, \ldots, b_N^{[\alpha]}\}, \tag{45}$$

where $b_i^{[\alpha]} = 1$ if $k_i^{[\alpha]} > 0$, and $b_i^{[\alpha]} = 0$ otherwise. For each layer α, the total layer activity $N^{[\alpha]} = \sum_{i=1}^{N} b_i^{[\alpha]}$ describes the total number of nodes with at least one connection in layer α, with $0 \leq N^{[\alpha]} \leq N$ [38].

Another measure to quantify the relative overlap between two layers at the level of node activity is the normalized Hamming distance between the two corresponding layer-activity vectors [298]:

$$H^{[\alpha,\beta]} = \frac{\sum_i b_i^{[\alpha]}(1 - b_i^{[\beta]}) + b_i^{[\beta]}(1 - b_i^{[\alpha]})}{\min(N^{[\alpha]} + N^{[\beta]}, N)}. \tag{46}$$

where $H^{[\alpha,\beta]} = 0$ if $\boldsymbol{d}^{[\alpha]} = \boldsymbol{d}^{[\beta]}$ and $H^{[\alpha,\beta]} = 1$ if all active nodes are active in no more than one layer. It has been suggested that real multiplex networks are normally characterized by heterogeneous distributions of layer activity and of pairwise multiplexity [298].

3.5 TRIADIC RELATIONS

To describe the multiplex clustering coefficient [31, 37, 75, 79] we refer to triadic relations defined by Emanuele Cozzo et al. in [116].

The *local clustering coefficient* C_u for node u is the number of three-cycles $t_u = (\mathbf{A}^3)_{uu}$ that start and end at the focal node u divided by the number of three-cycles $d_u = (\mathbf{AFA})_{uu}$ such that the second step of the cycle occurs in a complete graph; $\mathbf{A}$ is the adjacency matrix of the graph and $\mathbf{F}$ is the adjacency matrix of a complete graph with no self-edges:

$$C_u = \frac{t_u}{d_u} \tag{47}$$

Analogously, the global clustering coefficient for a monoplex network can be calculated either by averaging C_u over all nodes or by computing

$$C = \frac{\sum_u t_u}{\sum_u d_u} \tag{48}$$

In addition to three-cycles (i.e., triads) that occur within a single layer, multiplex networks also contain cycles that incorporate more than one layer but still have

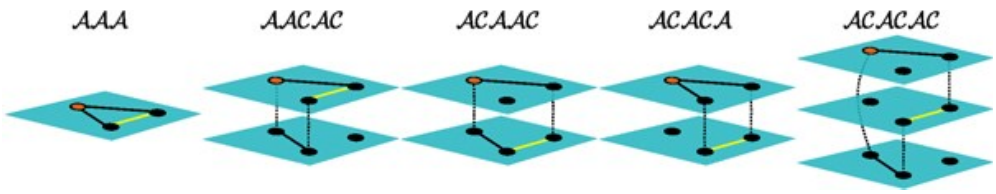


Figure 2: Illustrations of elementary cycles $\mathcal{AAA}$, $\mathcal{AACAC}$, $\mathcal{ACAAC}$, $\mathcal{ACACA}$, and $\mathcal{ACACAC}$. The orange node is the starting point of the cycle. The intra-layer edges are the solid lines, and the intra-layer edges are the dotted curves. In each case, the yellow line represents the second intra-layer step. Taken from [116].

three intra-layer steps. Moreover it is important to consider three-cycles that traverse different numbers of layers, so one needs to take them into account when defining a multiplex clustering coefficient.

Let $\bar{\mathcal{A}} = \mathcal{A} + \mathcal{C}$ the supra-adjacency matrices of an undirected multiplex network where $\mathcal{A} = \oplus_\alpha \mathbf{A}^{(\alpha)}$, $\mathbf{A}^{(\alpha)}$ is the adjacency matrix of layer α (i.e., the adjacency matrix associated to G^α). We consider undirected networks, so $\mathcal{A} = \mathcal{A}^T$. The coupling supra-graph matrix $\mathcal{C} = \mathcal{C}^T$ indicates the connections between corresponding nodes (i.e., between the same entity) on different layers.

A *supra-walk* is a walk in which, either before or after each intra-layer step, a walk can either continue on the same layer or change to an adjacent layer. It can be represented by matrix:

$$\hat{\mathcal{C}} = \beta\mathcal{I} + \gamma\mathcal{C} \tag{49}$$

where $\mathcal{I}$ is the $|V| \times |V|$ identity matrix, the parameter β is a weight that accounts for the walk staying in the current layer, and γ is a weight that accounts for the walk stepping to another layer.

In a supra-walk, a *supra-step* consists either of only a single intra-layer step or of a step that includes both an intra-layer step and an inter-layer step, in which one changes from one layer to another. The latter type of supra-step disallow two consecutive inter-layer steps. The number of three-cycles for node i is then:

$$t_{M,i} = \left[(\mathcal{A}\hat{\mathcal{C}})^3 + (\hat{\mathcal{C}}\mathcal{A})^3\right]_{ii} \tag{50}$$

where the first term corresponds to cycles in which the inter-layer step is taken after an intra-layer one and the second term corresponds to cycles in which the inter-layer step is taken before an intra-layer one. Subscript M refers to the particular way that we define a supra-walk. Due to the symmetry of both $\mathcal{A}$ and $\hat{\mathcal{C}}$ the equation can be simplify in:

$$t_{M,i} = 2\left[(\mathcal{A}\hat{\mathcal{C}})^3\right]_{ii}. \tag{51}$$

It may be useful to decompose multiplex clustering coefficients that are defined in terms of multilayer cycles into so-called elementary cycles by expanding Eq.(51) and writing it in terms of the matrices $\mathcal{A}$ and $\mathcal{C}$. That is, we write $t_{M,i} = \sum_{\mathcal{E}\in\mathcal{E}} w_{\mathcal{E}}(\mathcal{E})_{ii}$, where $\mathcal{E}$ denotes the set of elementary cycles and $w_{\mathcal{E}}$ are weights of different elementary cycles. All of the elementary cycles can be express in a standard form with terms from the set $\mathcal{E} = \{\mathcal{AAA}, \mathcal{AACAC}, \mathcal{ACAAC}, \mathcal{ACACA}, \mathcal{ACACAC}\}$ (see Figure 2).

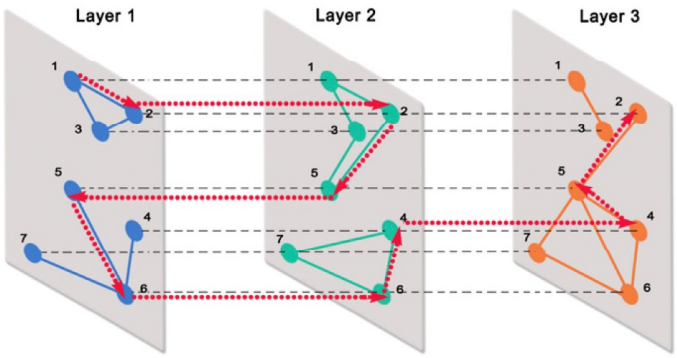


Figure 3: Illustrations of a path (dotted red line) between two nodes in a three layer multiplex (from node 1 in layer 1 to node 2 in layer 3). The length of this path is 10 if we count inter-layer edges or 6 if not. Image taken from [122]

The above formulation allows us to define local and global clustering coefficients for multiplex networks analogously to their definition in monoplex networks as:

$$c_{M,i} = \frac{t_{M,i}}{d_{M,i}} \tag{52}$$

$$C_{M,u} = \frac{\sum_{i \in l(u)}}{\sum_{i \in l(u)} d_{M,i}} \tag{53}$$

where $l(u) = \{(u, \alpha) \in V | \alpha \in L\}$ is the set of supra-adjacency matrix indices that correspond to node u.

3.6 PATH

The length of the path is an important topic in graph theory and a basis to calculate measures such as graph distance, connected components, betweenness centralities, random walks, community detection and many others. As we mentioned earlier, one of the decisions that we must consider in the translation of the concepts of path and distance to the case of multiplex network is if we have to count the interlayer links. The choice depends on the particular network we are representing. For instance, in multiplex transportation network (train, subway, bus, planes) the interlayer links for transferring in the same location may be associated at a cost. In this case we can represent it by multilayer model [121, 217]. In social network multilayer systems, instead, nodes are the same individuals represented in each layer so it may not be necessary to account interlayer edges (see Figure 3). In this paragraph and in the next one, we define the shortest path and the betweenness centrality referring to mathematical formulation described in [350]. A path $s_\alpha \overset{p}{\rightsquigarrow} t_\beta \in \mathcal{P}_{s_\alpha \rightsquigarrow t_\beta}$ on a multiplex network with L layers and N nodes per layer, is an ordered sequences of a nodes which starts form node s on layer α and ends in node t on layer β with restriction that an edge exists between every pair of consecutive nodes in p. $\mathcal{P}_{s_\alpha \rightsquigarrow t_\beta}$ is the set of all possible paths between s_α and t_β. For each path $s_\alpha \overset{p}{\rightsquigarrow} t_\beta$ we define a distance

function $d(s_\alpha \overset{p}{\rightsquigarrow} t_\beta)$ that counts the number of edges traversed, as for monoplex networks. Even in this case, holds the same condition for walks, namely that the edge representing the inter-layer relationship between two nodes is part of the path. Hence, the set of shortest-paths $P^*_{s\rightsquigarrow t}$, from node s to node t, in the multiplex is defined as the set of path which minimize the distance function betweemn the two vertices

$$P^*_{s\rightsquigarrow t} = \underset{\substack{s_\alpha \overset{p}{\rightsquigarrow} t_\beta \in P_{[s_\alpha \rightsquigarrow t_\beta]} \\ \alpha,\beta \in \{1,\dots,L\}}}{\operatorname{argmin}} \quad d(s_\alpha \overset{p}{\rightsquigarrow} t_\beta) \tag{54}$$

Therefore, a geodetic between two generic nodes v_i e v_j in a multiplex network is the shortest path starting from the node v_i laying in any layer and ending in nodes v_j in any level. This definition is coherent with the definition of multiplex network, considered that a node represents the same entity in any level of the network.

3.7 BETWEENNESS CENTRALITY

Let $P^*_{s\rightsquigarrow t}$ in Eq.(54), the shortest path betweenness a node v on layer ℓ, betweenness centrality $g(v_\ell)$ is defined as the sum, for every couple of vertices (s,t), of the fraction of times that node v on layer ℓ, belongs to a path in $P^*_{s\rightsquigarrow t}$

$$g(v_\ell) = \sum_{\substack{s,t=1 \\ s\neq t\neq v}} \frac{\sigma_{s,t}(v_\ell)}{\sigma_{s,t}}, \tag{55}$$

where $\sigma_{s,t} = |P^*_{[s\rightsquigarrow t]}|$ is the number of shortest-paths from s to t and $\sigma_{s,t}(v_\ell)$ is the number of times node v_ℓ is in a shortest-path from s to t.

It is important to consider that with Eq.(54) the shortest-path degeneration increase. That is $P_{[s_\alpha \rightsquigarrow t_\beta]}$ may contains classical shortest path degeneration and multiplex shortest path degeneration: several shortest paths between s and t in same layer together with shortest paths that start and end in the same node but in different layers.

Eventually, the shortest-path betweenness of a node, in a multiplex network, can be obtained by:

$$g(v) = \sum_{\ell=1}^{L} g(v_\ell). \tag{56}$$

4 NETWORK VISUALIZATION

Graphs can be used to represent relational data in many fields, ranging from physics, biology, chemistry, artificial intelligence to software engineering, social networks and transportation systems. For example, in social networks people may represent the vertices of a graph where the different relations among them are represented by a set of edges; in biology and chemistry graphs are widely used to represent molecular and genetic maps, as well as protein production paths; in software engineering, graphs are used to represent the structure of complex software systems or the internal states of compilers; in transportation system nodes are locations (town, place, road intersections) and edges represent different transport modalities (bus, underground, trains, etc) or multiple kind of connection between locations (roads, rail lines etc).

Applications require that graphs may be visualized and manipulated so that the information they represent can be better comprehended and utilized by humans. Thus, graph visualization plays an important role in the research field of *Information Visualization*. The studies in this field reveal that the way data is represented can affect our ability to use it more efficiently. However, we have to take into account a great challenge when visualizing networks, first of all the development of a drawing algorithm of the graph itself [36].

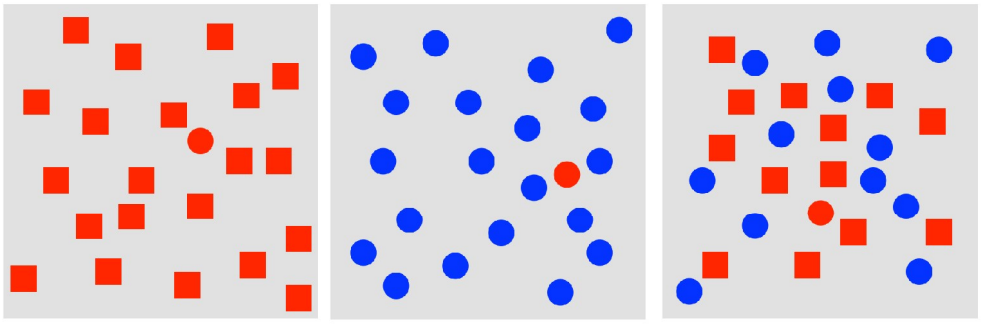
We start with a brief background on *Information Visualization* followed by an overview of networks and graphs visualization layout algorithms. Throughout the overview, special attention is given to layout and interaction techniques that we use in Criminal Network Analysis.

4.1 INFORMATION VISUALIZATION

Information Visualization often called InfoVis is a research area that focuses on the use of computer-supported, interactive, visual representations of abstract data to amplify cognition [87]. The basic method is to generate interactive visual representations of the information that exploit the perceptual capabilities of the human visual system and the interactive capabilities of the cognitive problem-solving loop [376].

4.1.1 Pre-attentive processing

Some different features of visual scenes are unconsciously captured by the brain even before any other attention or cognition process - with retinal rods and cones responding under 200 *ms* [190]. A limited set of visual properties are detected very rapidly and accurately by the low-level visual system. These properties were called *pre-attentive*, since their detection seemed to precede focused attention [368].



(a) (b) (c)

Figure 4: Examples of different features pre-attentively percived based on shape, color and conjunction, adapted from [189].

Form	Colour	Motion	Spatial
line orientation	hue	flicker	2D position
line width	intensity	direction	stereoscopic depth
line lenght	lighting diretion		convex
size			concave
curvature			grouping
blur			
added marks			
numerosity			

Table 1: Visual features that can be perceived in under 200 *ms* [190, 376].

An example of a pre-attentive task is illustrated in Figure 4. In the left panel (Figure 4a) is shown a search task for which the target is a red circle. A viewer can rapidly determine whether the target is present or not. Here, the visual system identifies the target through a difference in form. Figure 4b illustrates a search task for which *hue* is a pre-attentive feature in a sea of blue *distractors* [189]. On the right panel (Figure 4c), there is an example of conjunction search. The visual system has two visual features to search when trying to locate the target: *hue* (red) and *shape* (circle). In this case, target cannot be detected preattentively.

Healey *et al.*, 1996 [190] and Ware, 2000 [376] compiled a non exhaustive list of features that are recognized at glance by pre-attentive processing (see Table 1). Feature can also be combined to group homogeneous items, avoiding many of the cognitive tasks by users.

Pre-attentive processing is a powerful tool in draw attention onto elements of interest in visual representation.

4.1.2 Gestalt theory

Gestalt theory, founded by Max Wertheimer in 1912, is a branch of philosophy and psychology based on a belief that the brain is capable of understanding an entire system more than just a collection of isolated features when certain principles are

applied. The Gestalt laws presented by Ware (2004) [376] can be applied to studies of visual perception. They are based on six principles that bring us understanding of an image:

CONTINUITY Smooth and continuous lines will be perceived before discontinuous ones.

SIMILARITY objects of similar shape or color tend to be grouped together.

PROXIMITY Objects near one another are seen as a unit.

CLOSURE Contours with gaps will be perceptually closed.

SYMMETRY Symmetrical pairs are perceived more strongly than parallel pairs.

RELATIVE SIZE Smaller regions in a pattern are perceived as objects, larger regions as the background.

Gestalt laws have been extensively applied in user interface design, graphic design, and information visualization. Several graph drawing conventions and aesthetics seem to rely on Gestalt principles [36, 45, 184].

4.1.3 Mantra of Visual Information Seeking

"Visual Information-Seeking Mantra" is a model of an InfoVis process proposed by Shneiderman in 1996 [344]. He insightfully summarized basic visual design principles as starting point in trying to characterize Information Visualization: "Overview first, zoom and filter, then details-on-demand". Shneiderman breaks down the overall area of information visualization into a data type by task taxonomy, consisting of seven basic data types (include 1D linear, 2D map, 3D world, multidimensional, temporal, tree, and network) and seven basic tasks (overview, zoom, filter, details-on-demand, relate, history, and extract):

OVERVIEW FIRST The visualization first provides an overview of the entire collection of data set displaying high-level features and allows the analyst to successively specify a region of interest.

ZOOM Zoom functionality allows the analyst to target a region of interest. Zooming on details can be geometric (for example, make regions of the display larger in order to read small words or differentiate between nearby nodes), or semantic (for example, provide more specific theme words or break clusters to show individual authors).

FILTER Filtering can take several forms: (1) remove the context from the display, leaving only items of specific interest, (2) provide more detail on a focal region, abstract and display surrounding data (focus + context), or (3) show detail in a new window, highlight region of enlargement on the overview display (overview + detail).

DETAILS-ON-DEMAND Select an item or group and get details when needed.

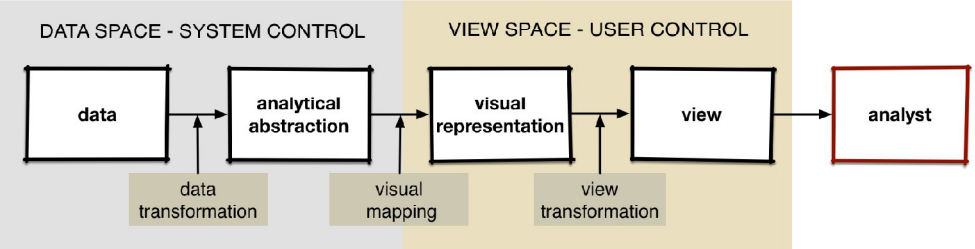


Figure 5: The information visualization pipeline including system space and user control space, from Chi and Riedl (1998) [102].

RELATE View relations hips among items. A task that truly leverages visual displays, relating items or groups of items have a broad range of variations and choices, including proximity, containment, connectors, colors, and highlighting.

HISTORY Keep a history of actions to support undo, replay, and progressive refinement.

EXTRACT Allow extraction of sub-collections and of the query parameters.

4.1.4 The Pipeline

The visualization pipeline is the computational process of converting information into a visual form that users can interact with [87]. In literature there are many models of visualization that use composed series of steps which implement data transformation. Chi and Riedl, in 1998 [102], proposed a model which describes the data transformations performed by the system and the visual transformations resulting from analyst interaction with a visualization, as shown in Figure 5.

A modified visualization pipeline without a separation between system and user analyst control has been described by Card *et al.* [87]. In the pipeline (shown in Figure 6) there are three ways in which data is encoded:

DATA TRANSFORMATION The first step transforms raw information into well-formed data format. Dataset typically contains a set of entity-attribute value pairs.

VISUAL MAPPING The second step is the main process of visualization pipeline: maps the dataset into visual form. The visual form contains visual symbol that correspond to the relative dataset entities such a graph.

VIEW TRANFORMATION The third step embeds visual form into views, which display the visual form on screen and provide various view transformations (such as navigation, zoom and panning etc.). Visualization abstraction defines what to be displayed since visual transformations decides how to display. The view is then rendered to the screen. Users interpret the view to reconstruct the underlying information and can interact with any of the steps in the pipeline to alter the resulting visualization, and make further interpretations.

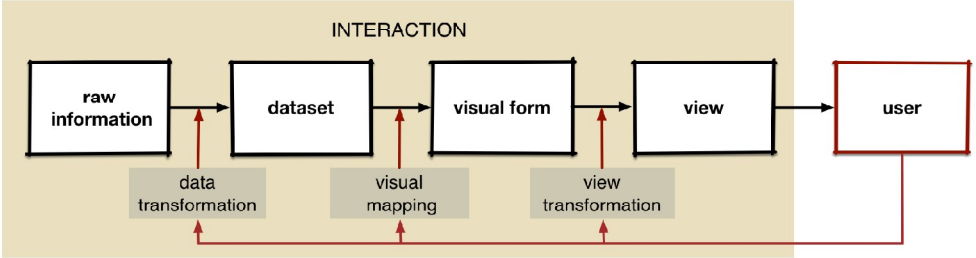


Figure 6: The information visualization pipeline without separation between system and user analysis control, proposed by Card *et al.* (1999) [87]

4.2 GRAPH LAYOUT

In graph visualization the size of a network to view is a key issue. Large number of elements can compromise performance or even reach the limits of the screen space. Displaying an entire large graph may give an indication of the overall structure or a location within it, but makes it difficult to comprehend. There are many layout algorithms that attempt to draw a graph minimizing link crossings and adhere to aesthetic principles. These algorithms, however, fall short when the large number of overlapping links makes it impossible to distinguish between nodes and edges.

A survey of layout and interaction techniques for information visualizations of graph is provided by Herman *et al.* [194]. They illustrate the limitations of many graph layout algorithms, methods to aid the navigation of large graphs, and strategy of reducing visual complexity.

The main goals of graph visualization techniques are: (1) increase the comprehension level of the data by providing intuitive, intelligible layouts; (2) providing a suitable technique helping the user interacting and navigating through the data.

We start with a brief background on graph drawing, which is followed by overviews of 2D graph layouts and visualization techniques.

4.2.1 Graph Drawing

The basic graph drawing problem can be put simply: given a graph, calculate the position of the nodes and the curve to be drawn for each edge [194].

Graph visualizations algorithms take into account different aesthetic criteria (minimize the bends along the edges, distribute nodes and edges evenly, avoid edge crossing, display isomorphic substructures in the same manner) that the generated drawing must adhere to, and properties layouts according to the type of graphs to which they can be applied. In-depth technical details on algorithmic techniques for graph drawings are described in Di Battista *et al.* book [36].

An important property of a layout algorithm is the *predictability*: two different executions of the same algorithm with the same or similar data inputs, should not lead to radically different visual representations. This property is also referred in literature as "preserving the mental map" of the user.

A special kind of layout problem is *planarity*, that raises the question whether it is possible to draw a graph on the plane with no edge crossings.

Another important issue related to aesthetic rules is the *time complexity*. Any visualization system needs to provide near real-time interaction, however, it is quite impossible to apply all rules at the same time. Some of them conflict with each other and are very computationally expensive. Thus, graphical layouts are usually the results of compromise among various aesthetic issues.

4.2.2 Node-Link Layout

In the literature, node-link diagrams were supporting social network analysis as early as Moreno, 1934 [273]. It intuitively depicts relationships with nodes represented as glyphs, and edges as lines.

Spring Layout

Most popular network visualization tools rely on the popular force-directed layout. Such algorithms calculate the layout of a graph using only information contained within the structure of the graph itself, rather than relying on domain-specific knowledge. Graphs drawn with these algorithms tend to be aesthetically pleasing, exhibit symmetries, and tend to produce crossing-free layouts for planar graphs [36].

Spring layout method was originally proposed by Eades in 1984 [142]. His spring embedder algorithm finds a pleasing layout for small graphs by balancing a constant repulsive force between all nodes against a logarithmic attractive force between nodes connected by an edge. The method proceeds iteratively, each iteration moving nodes by a small amount according to a force vector calculated by summing the attractive and repulsive forces.

Kamada and Kawai [211] model uses spring forces proportional to the graph theoretic distances determined by the lengths of shortest paths between the nodes.

This particular layout arrangement has the advantage of grouping vertices in clusters which can be identified according to the heightened connectivity. The Barnes-Hut algorithm [29] associated to this layout simulates a repulsive N-body system in order to continuously update the position of the elements.

The algorithm of Fruchterman and Reingold in 1991 [165] modified Eades' model to more closely approximate the physical analogy of electrostatic repulsive forces between all nodes, but also attractive forces between nodes that are adjacent.

The attraction force f_a is applied to the neighbors node which are connected by a spring, while the repulsive force f_r is applied to other nodes of a graph. These forces are defined as follows:

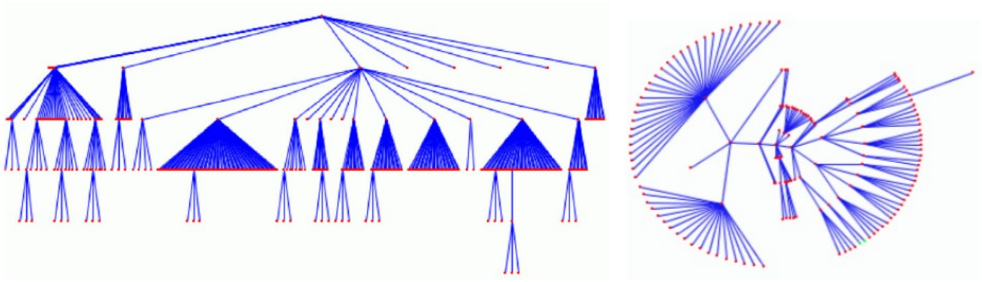
$$f_a(d) = \frac{d^2}{k}, \quad f_r = \frac{-k^2}{d},$$

where d is the distance between two vertices and k is the optimal distance between vertices defined as:

$$k = C\sqrt{\frac{areas}{|V|}}$$

.

The model represents the structure of the graph on the same foot as a physical system, in which nodes' coordinates (and therefore the layout itself) derive from



(a) (b)

Figure 7: (a) Reingold-Tilford tree layout [322] and (b) radial layout proposed by Eades [143].

the search of an equilibrium configuration of the physical system modeled by the algorithm [66].

Frick et al. [162] introduced a few refinements to this basic model. They added an extra force applied to all nodes attracting them to the center of the visual space, thereby preventing disconnected components from repelling each other out of the viewable area.

Force-directed layout is not *predictable* as the algorithm produces different results each time it is executed. The main drawback of spring embedders algorithms is that they are relatively expensive in terms of computational resources, which gives rise to scalability problems even for graphs with a few thousands vertices. Their time complexity exceeds $\mathcal{O}(V^3)$. To overcome this limit, sophisticated variants of force-directed algorithms have been proposed: they include hierarchical space partitioning, multidimensional scaling, stress-majorization, and multi-level techniques [32, 170, 182, 222].

Tree Layout

Node-link layouts use links between nodes to indicate the parent-child relationships. Reingold *et al.* 1981 [322] proposed one of the early methods that produces simple, fast and predictable 2D trees representation layout. Its implementation is straightforward, however, is not declared space-efficient because of its preference for one of two dominating growth directions, i.e., horizontal growth or vertical growth.

Some compact tree layout algorithms have been implemented to produce more clear appearance [36, 194, 195] (see Figure 7a). Node-link layout algorithm proposed by Eades, called radial layout, recursively positions children of a sub-tree into a circular wedge shape according to their depths in the tree (see Figure 7b). Radial views, including its variations [143, 383, 395], generally place the focus node at the center of the layout, and the other nodes radiate outward on separated circles. The circle to which that node belongs to depends on distance from the focused one.

4.2.3 Space-Nested Layout

Treemaps layout (see Figure 8[1]) is one of the most well known techniques in the field of Information Visualization. A Treemap layout algorithm divides the

[1] Drawed with `Treemap` 4.1, developed by Human-Computer Interaction Lab - University of Maryland.

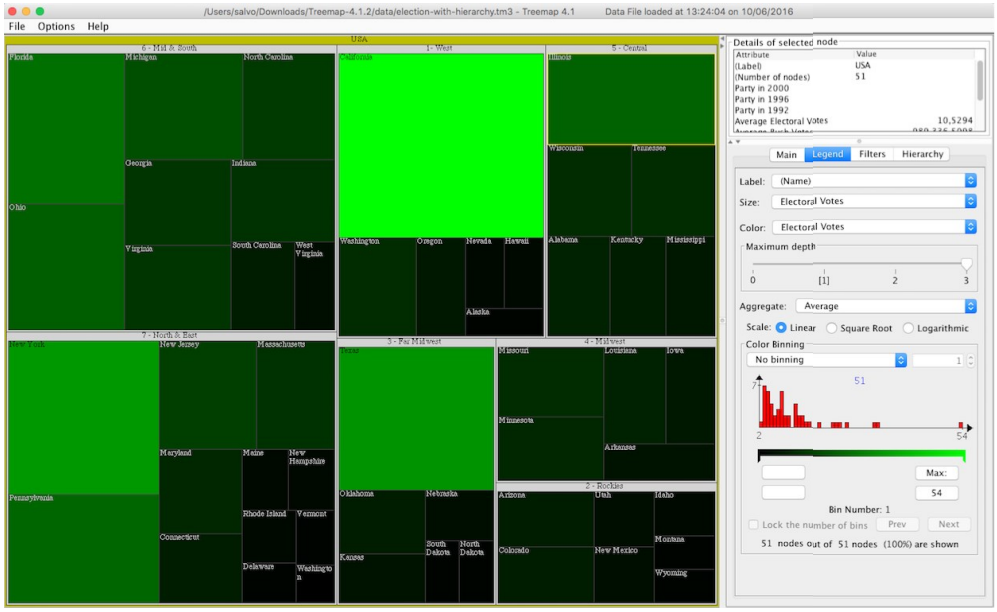
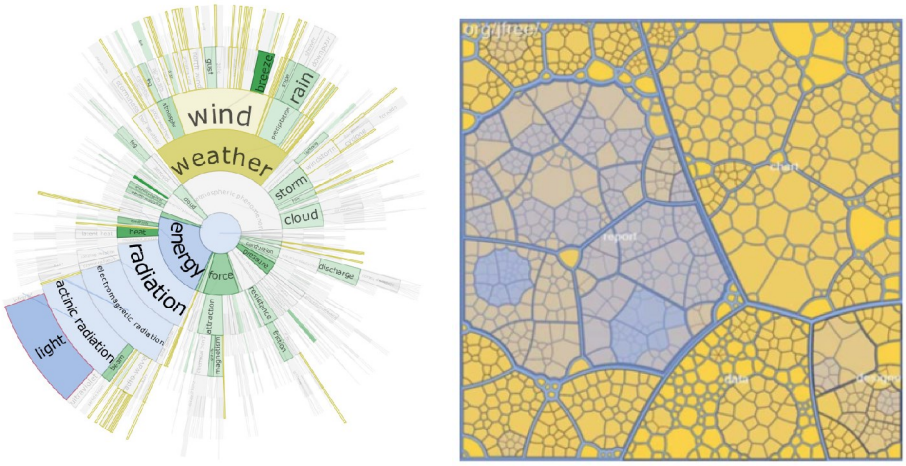


Figure 8: Treemaps view of USA 2000 election results by regional division. Each rectangle (node) correspond to a State. The dimension and the color intensity of a rectangle is proportional to the number of electoral votes per State.

display area into a nested sequence of rectangles whose areas correspond to an attribute of the data set. The original Treemap layout algorithm was introduced by Shneiderman and Johnson in 1991, [209, 345] and originally served for the display of hierarchical structured data from software modules or file system. This technique is popular because it uses the screen-space efficiently, and it shows the size of the leaves in a tree. Treemaps basic layout algorithm has two main drawbacks: (1) long thin rectangles easily appear and they can lead to a interaction problems; (2) the hierarchical structure is harder to discern than in classical node-link tree layout.

Squarified algorithm proposed by Bruls *et al.* [76] is a good solution and can be computed in short time. It does not consider the subdivision for all levels simultaneously, instead, it tries to recursively produce square-like rectangles for a set of siblings, given the rectangle where they are to fit in. The startpoint for each next level will then be a square-like rectangle, which gives good opportunities for a good subdivision. Furthermore, algorithm replaces the straigthforward subdivision process for a set of siblings of the standard treemap technique by a process that is similar to the hierarchical subdivision process.

Several other subdivision algorithms are proposed to generate better aspect ratios. Voronoi Treemaps [25, 362] keeps the size ratios while leaving gaps between siblings. The algorithm use arbitrary polygons to give more meaningful visual structures (see Figure 9b).



(a) (b)

Figure 9: (a) DocuBurst, a SunBurst layout of hyponymy, from C. Collins *et al.* 2009 [107]. (b) Voronoi Treemap layout, from M. Balzer *et al.* [25]

4.2.4 Space-Division Layout

In space division layouts, the parent-child relationship is indicated by attaching child nodes to their parent. The *SunBurst* technique is a space-filling visualization that uses a radial layout. In SunBurst, items in a hierarchy are laid out radially, with the top of the hierarchy at the center and deeper levels farther away from the center. The angle swept out by an item and its color correspond to some attribute of the data. For instance, in a visualization of a file system, the angle may correspond to the file/directory size and the color may correspond to the file type. An example Sunburst display is shown in Figure 9a. One issue of this layout is that it is difficult to distinguish between the child-parent relationships and the sibling relationships, because both of them are expressed using adjacency. Moreover, the node sizes are difficult to control and the final layout might occupy a large space for node, which has many children.

4.2.5 Matrix View

A network can be represented by using an adjacency matrix A in which each element a_{ij} represents the edge existing between the vertex i and the vertex j. The natural visualization technique associated to this two-dimensional representation of the graph is the matrix layout. Instead of showing numbers in rows and columns, they can encode relationships with colours or intensity. The advantage of matrix representation with respect to the node-link layout is the non-overlapping display of graph edges, and the readability of the graph for larger and denser graphs. Furthermore, a matrix can be adapted in different order to show grouping (clusters) and connections among groups (bridges) can be easily identified. Although it does not show the issue of edge crossing such as in a node-link diagram, matrix layout makes not easily identifiable the paths connecting the vertices. *Multiple synchronized*

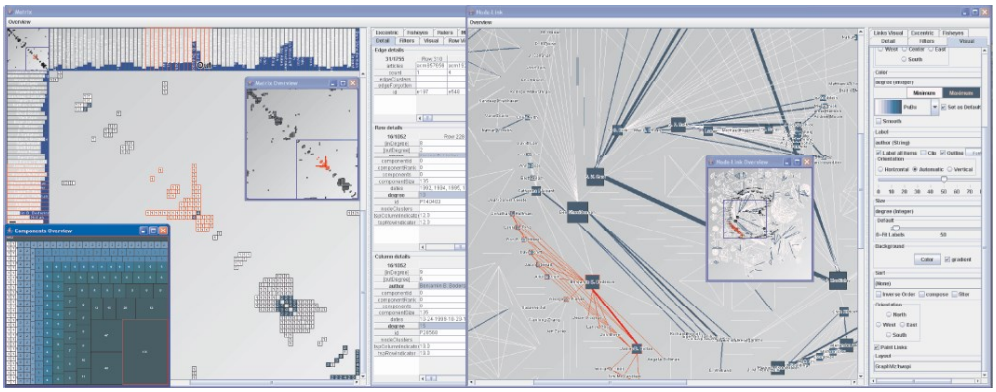


Figure 10: `MatrixExplorer` from Henry and Fekete, 2006 [193]. On the left panel is shown the matrix layout, while on the right is the synchronized node-link view with the same data set.

views is one of many techniques in literature that uses a combination of the node-link and matrix layout for overcoming their limitations. `MatrixExplorer` [193] uses this approach and shows the same data with synchronized view during exploration (see Figure 10).

4.2.6 3D Layout

A major difficulty for 2D visualization tools is dealing with increasing complexity as the number of entities increases. After a certain number of nodes and edges the graph becomes incomprehensible. One possible solution to this problem is to visualize networks using also the third dimension. 3D visualization offer more flexibility for placing vertices and edges of the graph. Furthermore, the extensive navigational features such as zooming, translation and rotation effectively employ the display space.

In the last years 3D graph drawing with a variety of aesthetics and edge representations have been extensively studied.

Force-directed methods are almost always described in dimension independent terms, which allows them to be generalized to 3D [118, 119].

Rekimoto implements Information Cube [323], a 3D version of space nested layout. He puts the information cubes inside their parent cubes to represent parent-child relationships. It displays textural labels on semi-transparent cube surfaces. The platform provides a 3D environment that the user can navigate by zooming, shifting and rotation.

The cone tree [326] is one of the best known 3D graph (in this case, tree) layout techniques in information visualization. In contrast to the previous examples, cone trees have been developed directly for 3D, instead of generalizing another 2D algorithm.

Hyperbolic layout H3 by Tamara Munzner's [284] provides a distorted view tree with the fish-eye future in 3D hyperbolic space. It allows for the visualization of much larger structures than the traditional techniques. The algorithm compute layout according to the hyperbolic metric, which increases exponentially opposed to

geometric ones of Euclidean space. It expands the cone tree layout for 3D hyperbolic space by placing children on a hemisphere around the cone base instead of its perimeter. The effect is a fisheye-like focus+context view of the graph, which allows the visualization of thousands of nodes with minimal screen clutter.

Although 3D graph visualization techniques overcome many kind of problems of 2D graph drawing techniques, they have introduced many other issues with object occlusion and interaction in 3D space. Thus achieving good 3D visualization remains a challenging problem.

4.2.7 2.5D Graph Drawings

2.5D term was introduced by Marr [252] to describe the imperfect human cognitive process of spatial perception. Human detect edges around shapes and then infer surface that stay inside based on depth perception. Human is able to resolve horizontal and vertical distances better then depths through stereo and motion viewing. The design principle behind the model is derived from Ware's guideline: a 2.5D design attitude that uses 3D depth selectively and pays special attention to 2D layout may provide the best match with the limited 3D capabilities of the human visual system [375].

2.5D visualisation is a 3D visualisation in which the third dimension is treated in a fundamentally different way with respect to the other two. It is defined more formally as a mapping from an information space to two separate visual channels: (1) a channel maps the information space to position in 2D and visual attributes such as glyphs, colour, texture and so on; (2) the other channel is a direct and discrete mapping to position in the third dimension from an independent variable in the information space.

Recently, a number of researchers have suggested a more understandable model for visualisation of large and complex networks in 2.5D layout [39, 68, 201, 202].

Algorithms for 2.5D layout of directed graphs are presented in [202]. Methods for drawing clustered graphs in 2.5D are described in [196]. Among them we cite y25. The main goal of y25, a project y25[2] by Werner Jainek[3], is an extension of a 2D graph drawing library with 2.5D functionality interface. y25 provides a framework for viewing and navigating 2.5D graphs.

In his work the author proposes a set of constraints that drastically restrict the use of the third dimension and lead to "more understandable" three-dimensional structures of graphs:

1. A graph is composed of multiple layers.
2. Each layer lives on a 2D plane inside the 3D space. All layer planes are parallel and have a specific distance to each other.
3. Nodes can only be positioned inside a layer.
4. Intra-layer edges connect two nodes from the same layer.
5. Inter-layer edges connect nodes laying in different layers.

[2] http://jainek.de/projects/y25/y25.html (2006)
[3] From University of Tübingen (Germany)

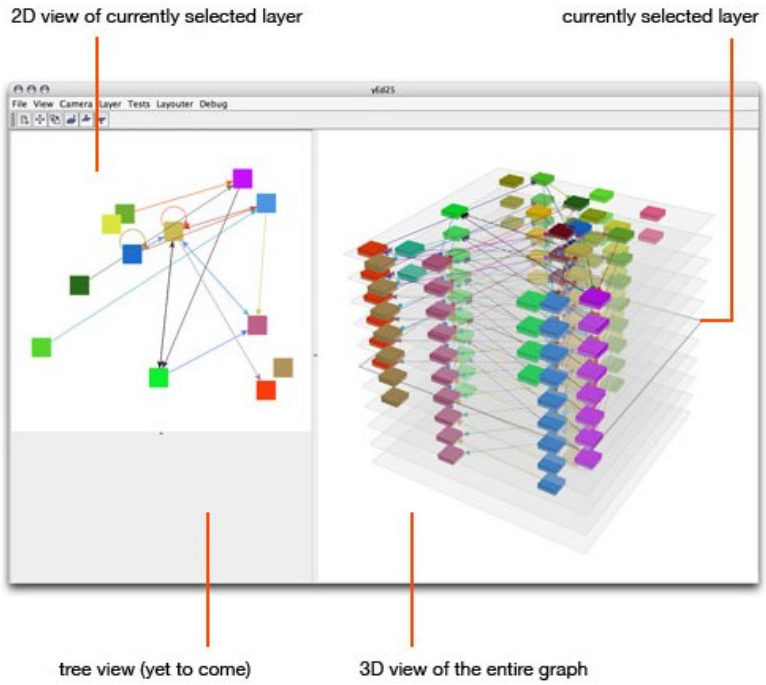


Figure 11: y25 editor.

Layers have the full freedom of three-dimensional coordinates. Nodes and edges are embedded in the 2D plane determined by the layer. 2.5D layouts of y25 (see Figure11) can be used, i.e. with time dependent graph or when a graph is hierarchical in nature. In the former case the entire process can be displayed simultaneously by using one layer for each time step and assigning the graph at the current time to that layer. In the latter, i.e. when multiple nodes can be successively grouped to form a more "high-level" representation of the graph, we can use one layer for each level inside the hierarchy.

2.5D hierarchical layout has been introduced by Hong and Nikolov [202], as an extension to the classical 2D hierarchical layout (also well-known as the Sugiyama method) for drawing directed graphs [358]. The vertex set is partitioned into layers, then each layer is divided into k parallel walls each containing a 2D layered drawing. The authors outline an efficient way of using the third dimension for reducing the visual complexity and minimizing occlusion.

A framework `MultiPlane` methods for drawing general graphs in 2.5D was presented in [200]. It uses a divide and conquer approach. More specifically, the algorithm divides the graph into a set of subgraphs, and then draws each subgraph in a plane using well-known 2D drawing algorithms. Finally, a 2.5D drawing of the whole graph is constructed by combining each drawing in a plane satisfying defined criteria. `ViENA` [151] is a visual analytics approach for analyzing dynamic networks. `ViENA` integrates: a dynamic layout; three temporal views based on different combinations of node-link diagrams (layer superimposition, layer juxtaposition, and 2.5D view); the visualization of social network analysis metrics; and specific interaction techniques for tracking node trajectories and node connectivity over time. In such a

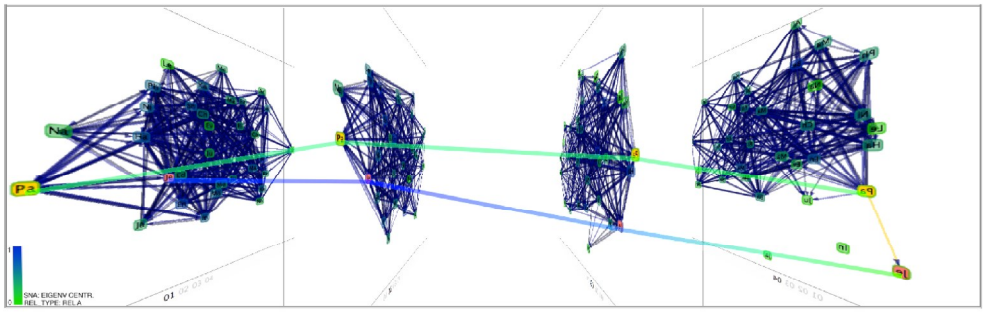


Figure 12: ViENA. A 2.5D view of four time slices. Trajectories of selected nodes are visualized as polygonal chains. SNA centrality metric is mapped to the color of nodes and trajectories [151].

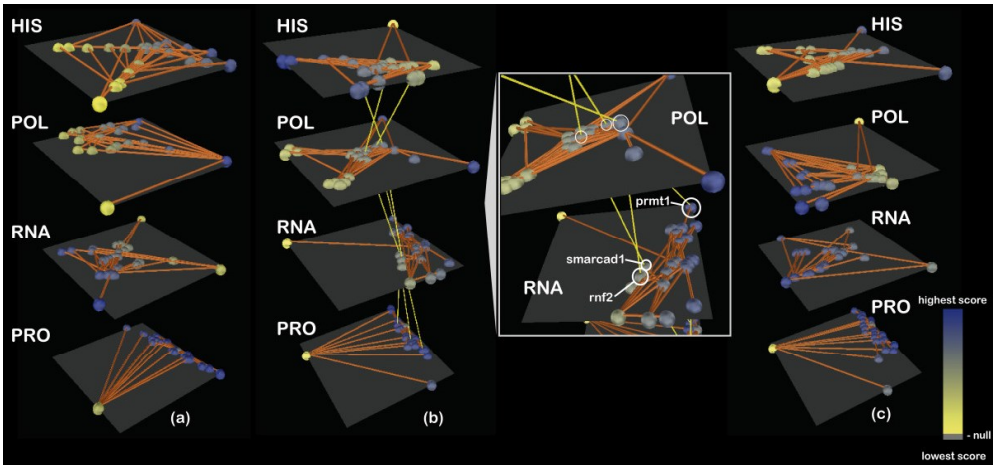


Figure 13: Arena3D. Dynamic clustering of layered biological profiles [339].

2.5D view, ViENA draw diagrams for each time slice on separate transparent planes, stacked along the horizontal time axis. The spatial dimension offers the opportunity to include additional information (see Figure 12). The disadvantage is that diagrams are distorted and occlusion may occur between planes.

Arena3D [310] introduces a new concept of staggered layers in 3D space (see Figure 13). Proteins, chemicals, or pathways data can be grouped onto separate layers and arranged via layout algorithms. Data on a layer can be clustered via k-means, affinity propagation, Markov clustering, neighbor joining, tree clustering, or UPGMA ('unweighted pair-group method with arithmetic mean'). Version 2.0 [339] introduces features that allow handling time course data in a phenotypic context.

Ahmed et al. have developed a software called GEOMI (GEOmetry for Maximum Insight), a tool for 3D visualization and analysis of large, complex networks [7]. Such visual analytic tools involve taking advantage of the graphic capabilities of computers to support the analysis of the network structure. GEOMI Temporal Network Plug-In (see Figure 14) use 2.5D method to visualize a graph snapshot at a particular time placed on a 2D plane in which a layout algorithm can be applied. Then, a series of

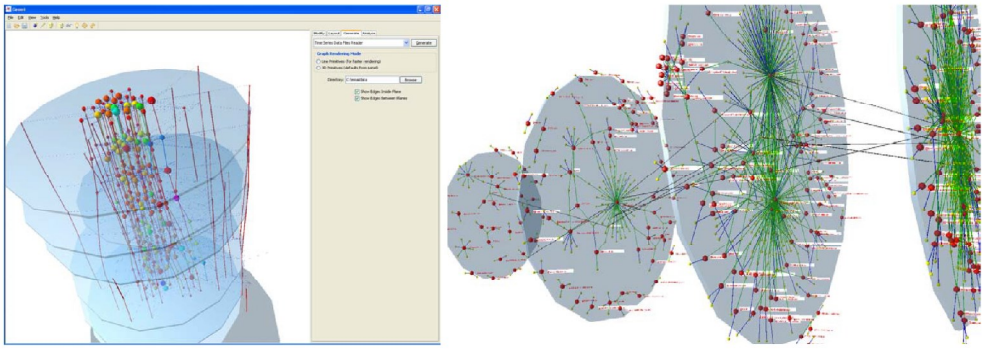


Figure 14: GEOMI. On the left, email connections of a research group represented in time series[7]. On the right panel, Email virus propagation [342].

such planes are stacked together following time order to show the changes. In order to identify a particular node in different time plane, same nodes in different planes are connected by edges. GEOMI provides users the ability of tracing the change of each individual node's relationship to others and also can evaluate the evolution of the whole network in general.

muxViz[4] developed by M. De Domenico et al. [120], is a open source platform[5] for the visualization and the analysis of interconnected multilayer networks (see Figure 15). It allows an interactive visualization and exploration of multilayer networks. It is suitable for the analysis of social networks exhibiting relationships of different type (e.g., family, work, etc) or interactions on different platforms (Twitter, Facebook, etc), biological networks characterized by different type of interactions (e.g., electric, chemical, etc, or allelic, non-allelic, etc), transportation networks consisting of different means of transport (e.g., trains, bus, etc) and many others.

muxViz supports different types of multilayer data analysis (multilayer correlation analysis, multilayer centrality analysis and annular representation, multilayer community structure detection, multilayer structural reducibility, and multilayer motifs analysis), visualization (edge-colored network, interconnected multiplex, interdependent network, general multilayer), and many layer layouts (one-line and multi-line layered, force directed and matrix). In muxViz, are considered two different types of inter-layer connectivity: ordinal and categorical. In ordinal multilayer networks, inter-layer edges exist only between layers that are adjacent to each other with respect to some criterion (e.g., temporal ordering). By contrast, categorical multilayer networks include inter-layer edges between counterpart nodes from every pair of layers.

Pymnet (Multilayer networks library for Python[6]), developed by M. Kivelä[7], contains data structures for representing multilayer and multiplex networks, and some functions for analyzing them. The interface is based on mathematical frameworks for multiplex and multilayer networks presented in the review article [217]. The open

[4] http://muxviz.net/
[5] https://github.com/manlius/muxViz
[6] http://people.maths.ox.ac.uk/kivela/mln_library/
[7] http://www.mkivela.com/

Figure 15: `muxViz`. Multiplex networks visualization examples [120]. Network of European airports, where each layer represents a different airline (on the left). Multiplex network community (right image).

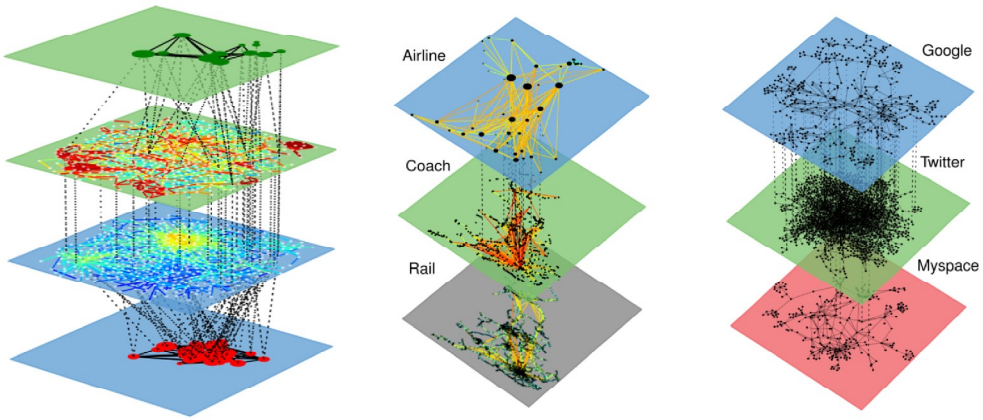


Figure 16: Multiplex networks visualization layout with `Pymnet`.

source library[8] supports even more general types of networks with multiple aspects (i.e. can have both temporal and multiplex aspect at the same time).

4.3 INTERACTION

Interaction and navigation are essential tools that can help to overcome the tasks related to exploration of a large graph and to increase the comprehension level of the given data. Yi *et al.* [396] summarize popular interaction techniques based on the purpose:

SELECT help user to highlight something as interesting.

EXPLORE is used to change view point to another part of data in the same layout representation.

RECONFIGURE to show a different arrangement with the same representation scheme.

[8] https://bitbucket.org/bolozna/multilayer-networks-library

ENCODE changing layout representation of the same data set.

ABSTRACT/ELABORATE show more or less detail of abstraction of a data representation, such as zooming and clustering.

FILTER reduce the amount of data being displayed.

CONNECT used to highlight the connection between items or the items which are relevant to the focus item.

In the following we will focus on standard interaction techniques commonly used in graph visualization such as zooming, panning, filtering, focus+context and animation.

4.3.1 Zoom and Pan

Zooming and panning are two fundamental and traditional tools in visualization. Panning allows to navigate in any direction across the scene. Though zooming, change between abstract or detailed in the view. Graphs is usually displayed by simple graphical component: lines and shapes. Thus zoom can be performed by simply adjusting screen transformation and by redrawing the contents. There are two main form of zooming: *geometric* and *semantic*. Geometric zooming just blows up the graph content and can fail when graph are very dense. Semantic zooming changes the information content of a selected area as with zoom level, thus more details are shown. When zooming out, detailed information is hidden, and only abstract information is drawn. The most technical difficulty with semantic zoom is with assigning an appropriate level of detail to subgraphs.

4.3.2 Filtering

Interactive filtering enables users to dynamically reduce information quantity in the display, and focus in on information of interest. It can decrease the complexity of the visualization by removing data that aren't relevant to the task.

Filtering can use algorithm for the selection of the nodes/edges to be removed based on node/edge attributes, or on topological values such as centrality measure or other graph properties.

A useful visual filtering interface should provide various visual browsing tools. Dynamic queries and range sliders should be used to give users freedom to see how the filtering process affects their data. Magic Lenses [51], including distortion techniques change the representation or allocate more space to items in focused areas and thereby improve the readability of the data of interest. They are used both for node-link and space filling graph visualization techniques.

4.3.3 Focus and context

The number of edges within a network usually grows faster than the number of nodes. As a consequence, the network layout would necessarily contain groups of nodes in which some local details would easily become unreadable because of

density and overlap of the edges. As the size and complexity of the network grow, eventually nodes and edges become indistinguishable. This problem is known as visual overload [17]. A commonly used technique to work around visual overload consists of employing a zoom-in function able to enlarge the part of the graph of interest. The drawback of this operation is the detriment of the visualization of the global structure which, during the zooming, would not be displayed.

Focus and context is an interactive visualization technique [243] that allows the user to focus on one or more areas of interest (the *focus*) in greater detail, (the *context*). These techniques do not replace zoom and pan, but rather complement them.

The *fisheye distortion* is a particular focus and context visualization technique which has been applied to visualize self-organizing maps in the Web surfing [392]. It was first proposed by Furnas [166] and successively enriched by Brown *et al.* [333]. It is known as a visualization technique that introduces distortion in the displayed information. The fisheye layout is a local linear enlargement technique that, without modifying the size of the visualization canvas, allows to enhance the region surrounding the focus, while compressing the remote neighboring regions. The overall structure of the network is nevertheless maintained. The fisheye technique is independent of the layout algorithm and is defined as a separate processing step on the graphical layout of the graph.

Brown *et al.* [333] suggest a different function that magnifies continuously so as to avoid local minification and demonstrate applying the distortion to each dimension separately, resulting in Cartesian distortion. This technique, straight lines parallel to the x or y axis remain straight even after distortion.

4.3.4 Animation

In many applications graphs are dynamic. They change their structure and layout according to user and application actions. Preserving the mental map during these changes has been identified to be crucial for the usability of a system [36]. Main approaches to the problem consist in developing graph drawing algorithms that minimize changes or visualize them with an animation.

Friedrich and Eades [163] present an approach that transform a drawing of a graph into another without any restriction to the class of graphs or type of layout algorithm. Animation is as a sequence of frames characterized by subtle but highly structured changes between consecutive frames over space and over time [163]. For a graph animation, the frames are drawings of graphs. The changes in the drawings are changes in the positions of the nodes and edges. Authors define in particular, a set criteria for good animation as the following:

1. Minimize temporary edge crossing
2. Maintain a minimal distance between nodes which do not move uniformly
3. Maximize uniform movement
4. Maximize symmetry
5. Maximize movement interpreted as movement of a rigid object
6. Minimize the length of the path of a node

7. Provide smooth transition and adequate speed

Based on the previous principles for 2D and 3D graph layout, Hong and Ahmed [6] define criteria for navigating graphs drawn in 2.5 dimension. In particular they design and implement methods for trees, clustered graphs and hierarchical graphs in 2.5D based on primitive 3D operations such as translation, rotation, scaling, and shearing.

5 CRIMINAL NETWORKS

In the fight against the racketeering and terrorism, knowledge about the network structure and the organization behind is of fundamental importance for both the investigations and the development of efficient strategies to prevent and restrain crimes.

Intelligence agencies exploit information obtained from the analysis of large amounts of heterogeneous data deriving from various informative sources to acquire knowledge about criminal networks and initiate accurate and destabilizing actions, including: the phone calling records, the social networks, surveillance data, interview data, e-mail, instant messaging, chat rooms and web site visits, court records, business, payroll and tax records, vehicle sale, credit files, bank accounts and the related transaction.

In criminology and research on terrorism, Social Network Analysis (SNA) has been proved a powerful tool to learn the structure of a criminal organization. It allows analysts to understand the structural relevance of single actors and the relations among members, when regarded as individuals or members of (one or more) subgroup(s). SNA defines the key concepts to characterize network structure and roles, such as centrality [159], node and edge betweenness [65, 159], and structural similarity [248]. The understanding of network structure derived from these concepts would not be possible otherwise [377].

Various research streams focus on finding structural properties of criminal networks, including phone call communication networks [257]. Understanding network properties such as the communities present in the network, or the roles that network members play, can help network analysts to unveil vulnerabilities and identify potential opportunities to take destabilizing actions to fight criminal organizations.

The above-mentioned structural properties are heavily employed to visually represent social and criminal networks as a support decision-making processes. The most common graphical layouts have historically been the node-link and the matrix representations [161]. Nevertheless, the utility of visualization tools may become limited when the dimension and the complexity of the system under analysis grow beyond certain terms.

Resilience identifies the ability of criminal networks to face pressures from law enforcement agencies and rapidly reorganize after perturbations or destabilizing attacks. Apart from environmental considerations, this concept is strongly tied to the topology of criminal networks which, unlike social networks, can be configured as hierarchical, cellular (or modular), flat or, even more frequently, as a combination of them. Resilience has implications in the techniques of investigation of law enforcement agencies especially during the phases of information gathering or planning of police actions.

In this Chapter we provide a background on social network analysis, and we survey existing literature in criminal network analysis. We focus on work about communities and communication dynamics, discovering patterns of interaction,

identifying central individuals, criminal networks resilience and uncovering network organization and structure.

5.1 DEFINITIONS

There is no universally accepted definition of organized crime group[1]. It may be defined differently depending on the scope (legal, government, academia etc). The criteria needed to classify an organization or group as organized crime vary between jurisdiction, agencies, states and countries. Definitions are crucial because they provide information needed for effective laws, investigations, and prosecutions.

Albanese [10] discusses the consensus of international perspective on the common features of organized crime: (1) planned criminal activity for profit; (2) a conspiracy of a continuing enterprise formed around social, ethnic, or business relationships, or around a certain product or opportunity; (3) use of violence, threats, and intimidation to achieve goals; (4) the use of corruption to protect its interest and avoid arrest and prosecution. Other approaches divide organized crime groups into traditional organized crime, like Italian Mafia, American La Cosa Nostra, Yakuza, drug-specific organized crime (i.e. South American cartels), and entrepreneurial organized crime as, for instance Russian, Ukrainian, West African etc.

Transnational Organized Crime Threat Assessment, produced by United Nation Office on Drugs and Crime[2] (UNODC), focuses on trafficking flows, connects the dots between regions, and gives a global overview of illicit markets (see Figure 17). It reports about the ways and means international mafias have grown into an international problem [136].

Criminal networks often emerge as a consequence of radicalization, either of individuals or groups. For instance the violent Islamist radicalization of individuals toward forming or joining terrorist networks. Such as the criminal organization, terrorist groups fall in the category of *dark networks*, where the network achievements come at the cost of other individuals, groups or societies and, in addition, their activities are both 'covert and illegal' [318].

The two types of dark networks are different. Organized crime network wants to remain invisible and operate without the scrutiny of law enforcement. Terrorism, conversely, occurs through highly visible acts. A terror group uses awful and violent attacks to draw attention to their political goal, while maintaining secrecy about its membership, location, organization and finances. Attacks responsibility are publicly claimed by group to draw the nexus between the attack and their campaign.

Terrorist networks are complex adaptive systems composed of dynamic autonomous covert cells which are widely dispersed. Given that individuals play different roles in their cells, the illegal activities of the terrorists are split among them. Therefore, isolation of terrorist cells requires the identification of important actors and respectively their different roles [185].

Intelligence and law enforcement agency investigators adopt the term *criminal network* in reference to criminal organizations made up by some individuals intercon-

[1] See http://www.organized-crime.de/organizedcrimedefinitions.htm for a collection of 180 definitions of organized crime.

[2] https://www.unodc.org/unodc/data-and-analysis/TOC-threat-assessments.html

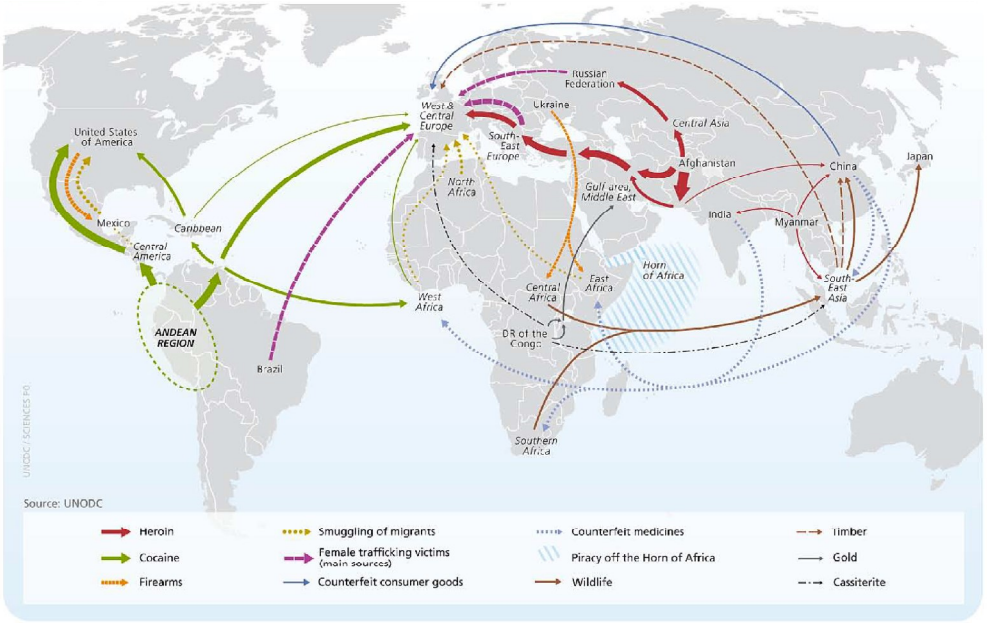


Figure 17: The main global transnational organized crime flow (TOCTA report 2010), ranging from trafficking in persons, to smuggling of migrants, to cocaine and heroin trafficking, trafficking in firearms, smuggling of natural resources, to the illicit trade in counterfeit goods and maritime piracy. Taken from [136].

nected and whose common goal is the execution of crimes or offences established in order to obtain, directly or indirectly, a financial or other material benefit. In a broader sense, a criminal organization consists of various entities: individuals, locations and places, vehicles and weapons, societies and real estates, checking accounts and financial transactions, etc. An in-depth comprehension of the relations and interactions among these entities is of crucial importance in order to uncover and fight criminal activities.

What distinguishes individual crime from crimes committed by groups of people is the term *organized* or *organization*, which emphasizes the cooperation among groups of people to accomplish illegal objectives. Organized crime incorporate many characteristic principles such as: a unit of command, authority and responsibility, the span of control, the purpose of the organization, contacts insulation by personnel and their boss and peers, structure (hierarchical, cells, etc.), job function specialization, secrecy rules, and so on.

Even if there is a strong difference between the locutions *criminal organization* and *criminal network*, in the following we shall use both as if they were interchangeable. Having in mind the analysis of networks, in fact, the criminal network obtained from investigations about a criminal organization is the only network representation of the organization we can work on and, as a matter of fact, *represents* the criminal organization.

Criminal networks are different from other networks in a number of ways. Morselli [279] (2009) discusses the criminal network perspective and emphasizes that

while the crime is a social phenomenon, criminal networks and general criminal behavior do have distinctive features from noncriminal counterparts.

Criminal organizations dynamically change as a consequence of several factors: pressure of competing groups, repressive actions of law enforcement agencies, new business opportunities [276, 355]. To survive and prosper, criminal organizations must be sufficiently resilient and adapt to changes deriving from competing illicit activities, legislative interventions, controls of law enforcement agencies all of which may lead to the collapse, the stagnation or the adaptation of the network [276], to the expansion or contraction of criminal traffic [19].

Changes in a criminal network may also originate from internal conflicts: the organization may split up, merge with other groups or undergo a reorganization. The variable effects of the interruption of the network may therefore be understood only by studying its dynamics, such as adaptive complex systems [348].

Criminal networks differ from the social networks in the counterbalance between secrecy and efficiency [365]. On the one hand, illegal activities have to remain concealed to governmental authorities and rival criminal organizations. As a consequence, communications among its members must be reduced to the minimum. On the other hand, to limit the risk of being uncovered during an illegal activity, it is necessary to ensure that communications among its members are highly efficient and trustworthy [280]. Therefore, criminal organizations are constantly trying to keep an equilibrium between efficiency and secrecy with respect to their illegal interests [43].

5.2 FORMS AND STRUCTURES

Knowledge about the structure and organization of criminal networks is important for both investigation and development of effective strategies to prevent terrorist attacks or criminal organization offenses.

Criminal networks evolve over time. The *a priori* knowledge of the organizational model is critical to the enforcement activities but it will be necessary not to lose sight over the network dynamics.

Criminal networks can be classified according to global and local structural properties. At global level, they may include hierarchical structures, cohesive subgroups (cells) connected by bridges, and flat structures where individual entities are distributed in a random manner. Locally, they also shows a smaller sub-structural components: cliques, bridges, hub, triad and motifs.

A United Nation research [250, 373] delineated some "ideal" types of criminal organization ranging from the most traditional forms of organized crime to modern organized networks, as follow[3]:

5.2.1 Standard and regional hierarchy

A standard hierarchy is a single organized crime group, usually led by an single powerful individual. These organizations have clearly defined roles, a readily identified chain of command, and a hierarchy that is designed to provide a strong system

[3] Not all criminal organization will conform precisely to a specific type, but most will have the predominant characteristics of one one of these types.

of internal discipline. Standard hierarchies usually have a name by which they are known and often have a strong ethnic or social identity. For example, members usually come from the same ethnic background (e.g., Albanians, Russians, Italians, etc.) or a similar background experience (e.g., prison gangs). Violence is an integral tool of both legal and illegal businesses and these groups usually operate in clearly defined geographical areas.

Regional hierarchies are also tightly controlled groups with strong systems of internal discipline and clearly defined roles and lines of authority. The major difference between these groups and standard hierarchies is that considerable autonomy and independence are granted to local organizations operating within the criminal organization. They have a single leadership structure and a clear line of command. They tend to be regional in their geographic scope and engaged in multiple illegal activities. Like standard hierarchies, regional hierarchies have a strong social or ethnic identity and employ violence as a primary means of maintaining discipline and resolving disputes.

5.2.2 Clustered hierarchy

A clustered hierarchy is an organized crime group that involves a number of smaller organized crime groups that coordinate their activities and enterprises. Clustered hierarchies consist of a number of criminal groups that have established an arrangement for managing their respective activities in a coordinated manner. As the organization develops, the cluster develops a stronger identity for members than the smaller groups in which they are actual participants. The strength of the organization is found in the fact that it is virtually impervious to law enforcement activities. The arrest of any leader would have no impact on the organization.

5.2.3 Core group

One of the most important emerging forms of organized crime readily adapted to conducting enterprise in a global economy is the core group. A core group is an unstructured group of organized criminals surrounded by a larger network of individuals engaged in serious criminal activity. Unlike hierarchies, a core group has a flat organizational structure in which power is shared by all participants. It consists of a small number of individuals, which makes it much easier to avoid law enforcement interference and maintain internal security. Group identity is maintained through their illegal activities, but no strong social or ethnic identities are associated with core group organizations. Rarely such an organization is known by a specific name or, for that matter, known to the general public or law enforcement at all.

5.2.4 Criminal network

Criminal networks are loosely organized, highly adaptable, very fluid networks of individual participants who organize themselves around an ongoing criminal enterprise. The membership, shape, and organization of a network is defined by those individuals who take part in it at any given time. Individual attributes, such as specific skills, financial resources, political connections, and the like, determine the

importance of network participants. Networks are created, re-formed, and initiated around a series of continuing criminal projects. Individuals come and go from the network, so the organization is constantly re-forming itself from project to project. Criminal networks maintain a very low public profile and almost never identify themselves by any name or attribution other than the participation of the individuals in the network itself.

5.2.5 Terrorist networks

Terrorism involves crimes designed to intimidate or coerce civilians or a government in order to achieve political or social objectives [1, 10]. The main differences between terrorist networks and other criminal networks are: they have certain ideologies; they spread ideology by propaganda units, books, web sites, radio and TV channels. Terrorist networks have financiers in order to supply resources for terrorist activities and they take great care about their secrecy during meetings and interactions in order to hide themselves from surveillance of the police and other legal institutions [306].

5.2.6 Flat

In terrorist network, ties between participants are usually strong, but not transparent and visible in every day routine. Relations are long-term. Participation in a terrorist plot requires a high level of trust in the network. Terrorist networks are often "sleeping"; they are prepared, but remain inactive. This way they are more difficult to uncover. Figure 18 shows an example of the final relatively *flat* structure of September 11 terrorist network structure [228]. Black border nodes correspond to the hijackers of American Airlines 11 crashed into Word Trade Center (north). Mohamed Atta was the ring leader of this conspiracy (he has the greatest degree and betweenness centrality in the network). Mohamed Abdi has the lowest network degree centrality. It has only one direct link with terrorist Nawaf Alhazmi and eight second level relationship with highlighted yellow nodes. Hs has not connections with Mohamed Atta and the hijackers in his group.

5.2.7 Cellular

From social network perspective, a cellular network is a single-component and undirected network of actors and their relationships, strictly consisting entirely of actors who are members of a specific cell; therefore, it is a network in which all actors are a member of a cell [158]. In criminal networks, cell structures (see Figure 19 are often embedded in top-down command-and-control hierarchy. But, a cellular network may evolve towards a decentralized, fragmented structure, which is only loosely coordinated, or not at all. Broken up under increased pressures from counter terror attacks, the remnant cells may regroup and launch independent operations.

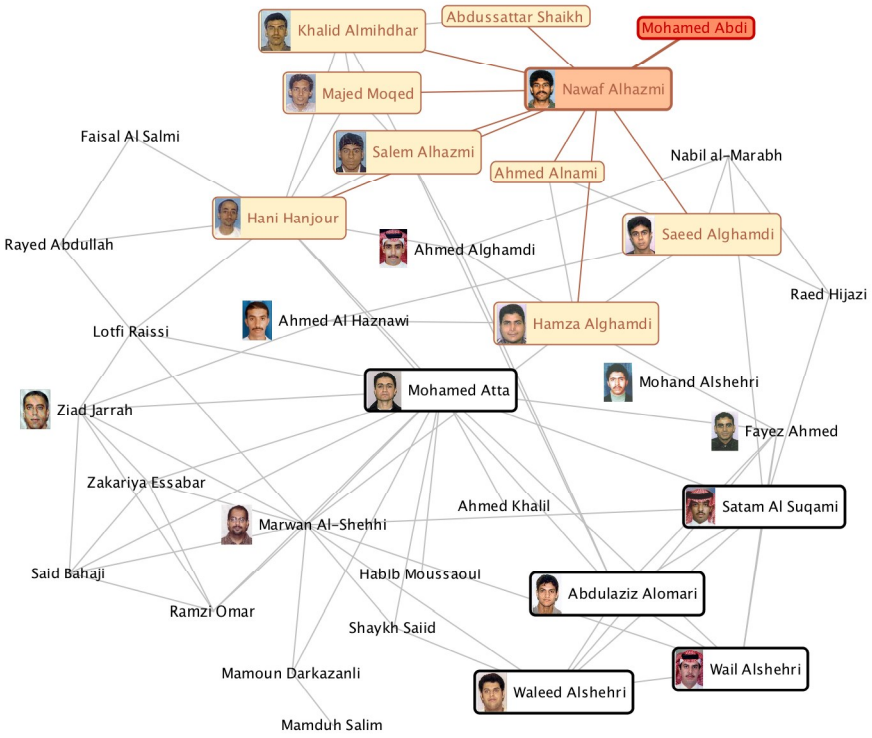


Figure 18: Terrorist network of the September 11 hijackers and associated [228].

5.3 NETWORK ANALYSIS

Network Analysis is an empirical tool that can be employed to identify, measure, visualize and analyze connections among people (friendship, kinship, etc.), groups and organizations [338]. It keeps track of relations among individuals or entities by representing them as nodes and showing the connections among them with lines (edges). Lines may be represented in different manners to show features such as the frequency or the type of relation. Nodes and edges form a network which describes relations among its members and the roles of the nodes: this is, for example, the case of gatekeepers (nodes which control the network), liaisons, core and peripheral members [355]. This way, hidden models of interaction are often discovered and the structure of the underlying connections can emerge [117]. Graphical representations allow to explicitly analyze, although in an empirical way, the topology of the network, identify the weak nodes and suggest proper interventions. Link analysis is focused on methods of constructing criminal networks from database records or textual documents.

In addition to its visual contribution, mathematical models, along with network representations and metrics, make it possible to gain deeper insights into the actual texture and operation of these types of networks. Social Network Analysis can be used to examine the resilience of a network by analyzing its vulnerability through the identification of central nodes, the availability of alternate nodesreplace lost central

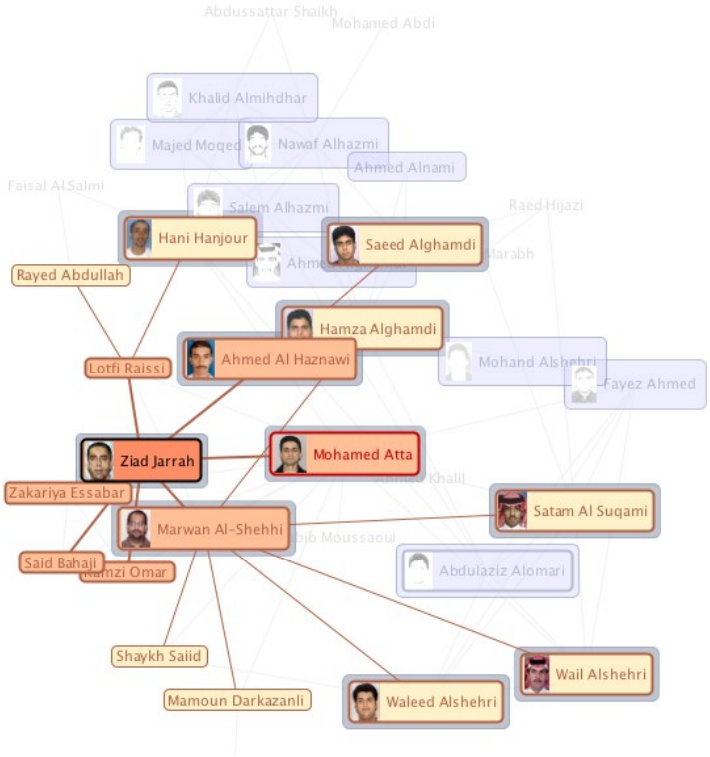


Figure 19: Cell inside terroristic network. Hijacker pilot (Ziad Jarrah) of United Airlines Flight 93 crashed in Pennsylvania, and his highlighted cell.

nodes, and less-central but bridging nodes tying together remote sections of the network [73].

One of the primary uses of SNA is the identification of the most central or most well-connected actors in a network. Actor centrality is based on the concept that "actors, who are the most important or the most prominent, are usually located in strategic locations within the network" [377]. Centrality measures such as degree, betweenness, closeness, and eigenvector centrality indicate the important insights into the structural properties of dark networks.

DENSITY Network density is a characteristic of a network as a whole. Formally, it is a measure, ranging from 0 to 1, of the number of actual connections compared to the total number of possible connections. The higher the density score, the higher the level of cohesion within a network. A clique network will have maximal density because all actors will be connected with the other actors.

Centrality indicates how concentrated a network is: a high concentration shows that a small number of people control the flow of resources.

AVERAGE PATH DISTANCE is an indication of how quick it is to navigate around the network. This measure provides insights into how close or remote certain actors are and, as a consequence, their level of knowledge.

The two most common centrality measures that relate to strategic positions are *degree centrality* and *betweenness centrality* [377].

DEGREE CENTRALITY is defined as the number of direct links a node has (see Section 1.2.1). A node with a high degree can be seen as a *hub*, an active node and an important communication channel. Hubs have great influence on the overall structure of the network, and the networks which mainly gravitate around some influential nodes are defined as scale-free networks.

Scale-free networks are more resilient against random attacks because the majority of nodes is poorly connected [378]. For networks with central hubs the removal of peripheral nodes is less significant in terms of its survival. Viceversa, decentralized networks are more influenced by random attacks in which the loss of each member is more important for the rest of the network. Under focused attacks, scale-free and random networks exhibit opposite resilience and vulnerability than in the case of random attacks: scale-free networks are very sensitive to focused attacks [11], whereas random networks are less vulnerable. As central members are more likely to be attacked, centralized networks are more vulnerable to targeted attacks than decentralized ones. The knowledge of the structural features of a network is therefore of crucial importance to fully understand the effects of each intervention.

BETWEENNESS CENTRALITY measures the extent to which a particular actor lies between other nodes in a network. The betweenness of a node is defined as the number of geodesics passing through it (see Section 1.2.4). Betweenness measures information flows through an individual.

Actors that control information within a network will have much higher betweenness values than those who appear on the fringes. Highly central actors, like intermediaries, may yield strategic control and influence on other members of the network. An individual with a high betweenness may be a gatekeeper in the network. A gatekeeper criminal should often be targeted for removal because the removal may destabilize a criminal network or even cause it to fall apart [90].

Closely related to the betweenness centrality is another centrality index, called *closeness centrality*. It is a measure of the proximity that an actor has to all other actors in the network, and is related to the flow of information within a network.

CLOSENESS CENTRALITY is the average distance of a vertex from every other vertex in the network. This definition has some known issues when the network has more than one component. Formally, is the inverse of the sum of the shortest paths (geodesics) connecting a particular node to all other nodes in a network (see Section 1.2.3). The idea is that an actor is central if it can quickly interact with all the others, not only with its first neighbors [287].

In the context of criminal networks, this measure highlights entities with the minimum distance from the others, allowing them to send and receive information more quickly than anyone else in the organization. For this reason, the adoption of closeness centrality is crucial to highlight those individuals that are closer to others (in terms of communication paths). High values of closeness centrality in this type of communication networks are usually regarded as an indicator of the ability of the given actor to quickly spread information to all other actors of the network.

EIGENVECTOR CENTRALITY is another way to assign the centrality to an actor of the network based on the idea that if a node has many central neighbors, it should be central as well (see Section 1.2.6).

This measure establishes that the importance of a node is determined by the importance of its neighbors. In the context of criminal networks communications, eigenvector centrality is usually regarded as the measure of influence of a given node. High values of eigenvector centrality are achieved by actors who are connected with high-scoring neighbors, which in turn, inherited such an influence from their high-scoring neighbors and so on. This measure well reflects an intuitive important feature of communication networks that is the influence diffusion.

TRANSITIVITY measures the density of connections among one's friends[4]. It can be measured by *clustering coefficient*, the number of connections between one's friends over the total number of possible connections among them (see Section 1.2.5).

It is well-known from the literature [377] that communication networks show high values of clustering coefficient since they reflect the underlying social structure of contacts among friends and acquaintances. Moreover, high values of local clustering coefficient are considered a reliable indicator of nodes whose neighbors are very well connected and among which a substantial amount of information may flow.

Strategies of network disruption based on centrality considerations can be efficient to dismantle centralized or decentralized networks, but the application of this approach to criminal networks has some exceptions [276]. In these types of networks, more central members could be at the same time more visible and therefore more detectable. As a consequence, central nodes are vulnerable [313]. Furthermore, the most central node is not necessarily the member who holds the leadership. In criminal networks, leadership and centrality are usually detained by different actors; focusing on the central node does not necessarily imply a disruption of the network and the substitution of the leader with a more central member [90, 258].

Studies show that, even if the approach through the centrality measures is useful to identify the potentially critical actors to disrupt the criminal network, a qualitative evaluation at the individual level is essential to understand the effects of the disruption of the network.

5.3.1 Cohesive Subgroups

A major focus of the social network analysis is to identify dense clusters of actors among whom there are relatively strong, direct, intense, and/or positive ties [377]. It is known as *community detection* (also known as assortative mixing or homophily [296]).

COMMUNITY STRUCTURE is defined intuitively as groups of nodes that are more tightly connected to each other than they are to the rest of the network [295, 316].

[4] Friends of a friend are likely to be friends as well.

Several studies have been conducted in order to investigate the community structure. A comprehensive overview of community-detection can be found in Fortunato [156]. Community detection is important for many reasons, such as node classification which entails homogeneous groups, group leaders or crucial group connectors.

The problem of finding communities in a network is often formalized as a clustering problem. The most popular family of methods involves the optimization of a quality function known as *modularity* [295]. To optimize modularity, one compares the actual network structure to some *null model*, which quantifies what it means for a pair of nodes to be randomly connected to. The standard null model for modularity optimization is the Newman-Girvan null model [295].

NETWORK MODULARITY Let consider a network, represented by means of a graph $G = (V, E)$, an expression for modularity is:

$$Q(C) = \frac{1}{2m} \sum_{i=1} \sum_{j} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(C_i, C_j) \tag{57}$$

assuming m the number of edges in the network, A is the adjacency matrix whose entries $A_{ij} = 1$ if there is a link between i and j and zero otherwise, k_i is the degree of node i, C_i refers to the community to which node i belongs and $\delta(C_i, C_j)$ (Kronecker delta function) is equal to 1 if nodes i and j are in the same community and zero otherwise.

The generalization formula to weighted graph is:

$$Q(C) = \frac{1}{2w} \sum_{i=1} \sum_{j} \left(W_{ij} - \frac{w_i w_j}{2w} \right) \delta(C_i, C_j) \tag{58}$$

where w is the sum the weights of the weighted adjacency matrix W_{ij} and w_i is the strength of node i.

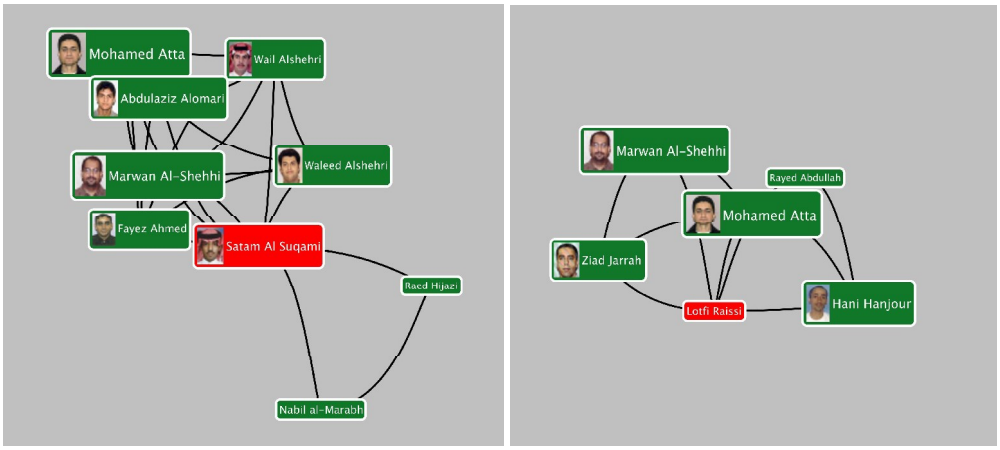
High values of Q imply high values of links number for each discovered community, yielding to communities internally densely connected and weakly coupled among each other.

The network modularity is therefore used as fitness function to solve an optimization problem: several methods exist, including the Girvan-Newman algorithm and its optimized variants [52, 172, 294, 295]. The goal of such strategies is that of producing a network clustering that exhibits a high network modularity. Although such methods are usually efficient, two limitations exist: first, the modularity function carries a resolution limit [157] preventing the detection of communities smaller than an intrinsic scale determined by the network size and its inter-connectivity; moreover, such techniques produce hard partitioning of the networks thus assigning each node to one and only one community.

Strategies to work around both limitations exist, and recently some approaches have been proposed to discover overlapping communities [308, 388], in order to allow nodes to belong to different communities.

5.3.2 Ego Networks

An ego-centered approach focuses on the person - typically termed ego - and the set of contacts (i.e., alters) who have ties to the person and measurements on the ties



(a) **(b)**

Figure 20: Example of two ego networks of September 11 hijackers and associated: (a) Satam Al Suqami (red node) hijaker; (b) Lofti Raissi (red node) terrorist associated.

among these alters. This yields a data structure similar to that displayed in Figure 20. A common use of ego-network data is to estimate the size of peoples' core networks to see if it has changed over time or is correlated with certain types of behavior.

5.4 CRIMINAL NETWORK ANALYSIS

Networks are seen as a more suitable structure for organized crime because they facilitate the flow of information, can adapt to changes in law enforcement responses and have the flexibility to deal with the associated risks inherent in all organized crime activities. Within an organized crime network, SNA techniques can identify network members that control information and how the removal of one or more members can inhibit the flow of information or alter the network's ability to adapt or perform at its best [90]. This type of analysis is essential in destabilizing networks. Thus, the utility of SNA as an analytical framework is evident, given that the structure of organized crime groups are shifting towards more flexible networks and that the nature of information provided by SNA can potentially disrupt organized crime activity [231].

In the latest years, the academic community working on the application of social network analysis to intelligence and study of criminal organizations has been constantly growing. One of the main contributions in this field is due to Sparrow [355], who focused on the application of SNA in order to identify the vulnerabilities of different types of criminal organizations. He highlighted three key aspects of Criminal Network Analysis (CNA), namely: i) the importance of SNA in order to analyze information; ii) the potential of intelligence when applied to the analysis of the networks; and, iii) the common results obtained from the collaboration of the two sectors. Sparrow also introduced the following definitions: i) dimension — the Criminal Networks (CNs) may have up to thousands elements; ii) incompleteness — criminal or terroristic networks are inevitably incomplete due to the fragmentary or

erroneous information available; iii) undefined borders — it is difficult to determine all the relations of each member; and, iv) dynamism — new connections necessarily imply an evolution of the structure of the network.

Starting from Sparrow's work, several authors tried to augment the superposition between the two fields by analyzing the CNs with the instruments typical of SNA: this is the case, for example, of the analysis Baker and Faulkner [22] carried out on the nature of white collar crime networks and drew conclusions about the nature of how such a network must operate, of Klerks' study [220] of criminal organizations in Netherlands, and the network analysis of Iranian government carried out by Deckro and Renfro [324]. In 2001, Silke [347] and Brannan et al. [70] examined the state of research in the field of terrorism and documented some cases in which it was lacking and empiric.

Arquilla and Ronfeldt [16] summarized the preceding work and introduced the concept of *NetWar* and its applicability to terrorism. In particular, they drew attention to the differences existing between the analysis of social and criminal networks, and highlighted the usefulness of research in these fields in order to understand the nature of criminal organizations. Notwithstanding the fact that the framework proposed by Arquilla and Ronfeldt provided a novel method to conceive network analysis, these authors received disapprovals because their approach was considered purely theoretical. Before 2001-09-11 one of major criticisms came from Carley, Reminga e Kamneva [90], concerning the initiatives for destabilizing the dynamic terroristic networks.

The published studies of terrorist groups are far less extensive than those of organized crime. This may be a result of the relative inaccessibility of data. However, the publicity attached to terrorism is much greater than that for organized crime and this has allowed some researchers to construct their own datasets of terror networks. In 2006, Krebs [228] applied network theories to the analysis of the Al Qaeda cell responsible of the 2001-09-11 attack. That work started a series of academic papers in which SNA has been directly applied to real cases, differently from previous research which was applied to artificial data or networks. Krebs' paper is still one of the most cited works in the field of the application of SNA to criminal networks and inspired a number of SNA applications used by intelligence agencies for the counter-terrorism war.

Article published by Koschade [224] analyzing Jemaah Islamiyah terrorist networks using SNA techniques to make significant findings about the strengths and weaknesses of the group. Mullins and Dolnik [283] studied other Islamic terrorist groups. Memon et al. [262] found small world characteristics in terror networks.

Nicosia et al. [37] introduce a set of metrics to characterize the structural properties of multiplex networks, tested and validated on a genuine multiplex data set of Top Noordin Terrorist Network. In particular, authors focus on the quantification of the participation of single nodes to the structure of each layer, and of the importance of each node for the overall efficiency of the multiplex network, in terms of node reachability and triadic closure.

Many recent studies have shown great interest to model criminal networks using the tools from statistical mechanics, complex networks, partial differential equations, and game theory [53, 92, 134]. D'Orsogna et al. [253] model the hierarchical evolution of an organized criminal network via antagonistic recruitment and pursuit

processes. They shows results in the context of dark network disruption and their implications on possible law enforcement strategies. In [346] is presented a model to study the emergence, dynamics, and steady-state properties of crime hotspots, by empirical observations of spatio-temporal clusters of crime across a wide variety of urban settings. Latora and Marchiori [233] discuss how *complex networks* can be successfully exploited to elaborate good strategies against terrorist attacks and terrorist organizations. In particular, they show how the information on the topology of a network can be used to spot the *critical components* of the network (i.e. the most important components for the efficient functioning of the network).

While analyzing criminal networks, detectives must focus on the features of the structure of the organization in order to answer the following questions [257, 355]: Who is central in the network? Which are the subgroups? Which are the models of interaction among subgroups? How does the overall structure of the network look like? The removal of which member(s) would perturb the network the most? How does information flow? The knowledge of these structural features may greatly help in detecting the vulnerabilities of criminal networks and has interesting implications in a criminal investigation.

The removal of central members may effectively dislocate the organization and interrupt the continuation of a criminal activity. Detectives should pay particular attention to subgroups or teams of criminal networks, since each of them may be responsible of specific tasks. Members of the group have to interact and cooperate to accomplish their illicit activities. Therefore, the detection of subgroups whose members are tightly inter-related may increase the comprehension of the CN organization. Groups may interact among them according to some given schemes. For example, members of a clan could frequently interact with those of another clan while seldom interact with the remaining members of the network. The detection of interaction dynamics and relations among subgroups often uncovers crucial information on the overall structure of the criminal network.

Reliable data and refined analysis techniques are crucial to fight criminal networks. Law enforcement and intelligence agencies often have to face the problem of handling large amounts of raw data gathered from various sources, including phone call logs, bank transactions, selling of cars and car registrations, etc. [86, 355].

5.5 RESILIENCE

Reducing the resilience of a network increases its vulnerability [19], since resilience is far more important than describing the vulnerability of central nodes or their features. As a matter of fact, the removal of key subjects, such as the most central nodes, does not necessarily imply the damaging or destruction of the network [72]. Resilient networks are flexible and adapt themselves for survival. The adaptation may acquire various forms among which we enumerate the substitution of lost nodes and the reshaping of the network.

Criminal networks develop the capacity of absorbing and tolerating inconveniences and adapting themselves to changes as a consequence of destabilizing or destroying attacks [138]. According to various authors, resilience consists of two aspects: the capacity of absorbing and tolerating perturbations, and the ability of

adapting, if necessary, to changes deriving from those interruptions [19, 63]. The ability of absorbing perturbations depends on the level of redundancy of the criminal network, in the sense of the diversity of the relations among its actors. Redundancy allows network members to fulfill the tasks previously appointed to those members no more belonging to the criminal network as a consequence of action of the law enforcement agencies. Notwithstanding the fact that some relations were broken, the diversity of connections allows the network to keep operating.

Redundancy is also associated to strong ties among the members of the network able to pledge reliable alternative relations [276]. Reliable substitutions can often be found within kinship connections, friendship or love affairs. This implies that substitutions often can be found at short distance from the cohesive nucleus. If actors with an essential expertise must be replaced, and if their role is uncommon within the network, it is necessary to find the substitutes outside the criminal network. In this case, non redundant connections become important for finding new associates who will be at greater distance from the reliable criminal nucleus. This principle, according to which non redundant ties within the network offer access to opportunities, resources and information to new members, is called the "strength of weak ties" [180].

Even if weak ties could turn into new business opportunities, the quest for substitutes exploiting these connections implies serious risks in terms of network security. In fact, criminal networks searching for competent and reliable substitutes could need to cooperate with individuals whose reliability is uncertain. Moreover, this research requires smarter internal and external communication methods, whereas an increased volume of information flow implies a certain exposure to security risks. In other words, the increase of information flow amplifies the risk of exposing the network as a whole, to contrast action of law enforcement agencies [246].

The capacity of adapting to new risks is the second important aspect of resilience. To adapt and protect the network from this kind of attacks, the flow of information is often managed by dividing the competencies into different subgroups. This means that important information is contained within various clans or organizational cells. This strategy prevents the exposure of the entire network in case a group is discovered and disrupted [382].

These features make the resilience of criminal networks a paradoxical concept. On the one hand, it depends on redundancy, which is an essential ingredient to find reliable substitutes after the losses due to interruptions or external interventions. On the other hand, it depends on the non-redundancy, in the case of the partition of the flow of information in order to prevent further scans [138]. The conclusion is that the resilience of a criminal network is a dynamic concept evolving along the trade-off between efficiency and security which consequently reshapes the structure of the network.

Related work

The resilience of a network which undergoes the deletion of nodes and/or edges has been studied in a number of scientific areas. The term resilience was first used in physics to illustrate the ability of certain materials to resume their original shape after external strain actions [302].

Albert et al. [11] studied the effect of node deletion in two example networks: a 6,000-node network representing the topology of the Internet at the level of autonomous systems and a 326,000 - page subset of the World Wide Web. Both the Internet and the World Wide Web have been observed to have degree distributions that have an approximate power-law form [12, 149]. The authors measured average node-node distance as a function of the number of nodes removed, both for random removal and for progressive removal of the nodes with the highest degrees. In the case of both networks, they found that distance was almost entirely unaffected by random node removal; that is, the networks were highly resilient to this type of removal. On the other hand, the removal of the highest degree nodes had a devastating effect. In this case, average node-node distance increases very sharply with the fraction of nodes removed, and typically only a few percent of nodes need to be removed before destroying almost all communication paths in the network. Albert et al. [11] expressed their results in terms of the failure or sabotage of network nodes. The Internet (and the World Wide Web), they suggest, is highly resilient against the random failure of nodes in the network but highly vulnerable to a deliberate attack on its highest-degree nodes.

Following these studies, many authors have investigated the question of resilience for other networks. In general, the results seem to be consistent with that seen for the Internet and the World Wide Web. Most networks are robust against random node removal but considerably less robust to targeted removal of the highest-degree nodes. Jeong et al. [206] looked at metabolic networks, Dunne et al. [140, 141] investigated food webs, Newman et al. explored email networks, and a variety of authors studied the resilience of model networks [84, 106].

A particularly comprehensive study of the resilience of both real-world and artificially generated networks has been conducted by Holme et al. [199], who investigated not only node removal but also edges removal. The authors also considered additional strategies for selecting edges and nodes. They compared the impact of node and edge removal on the size of the giant component for four removal strategies: initial degree, initial betweenness, recalculated degree, and recalculated betweenness. In the case of the two former strategies, nodes and edges were removed in order of decreasing initial degree and betweenness. In the case of the two latter strategies, nodes and edges were removed from the network in decreasing order of degree and betweenness, where these two quantities were recalculated after each removal.

Bouchard [63] used environmental studies of resilience to develop a 3-point list of characteristics which are useful in determining network resilience: *(i) vulnerability*, referred to the likelihood of damage from a specific type of attack; *(ii) elasticity*, the systems ability to return to its original state after taking damage; and, *(iii)* the *adaptive capacity*, the network's ability to change to reduce its vulnerability.

Milward and Raab [72] identified three alternative criteria of resilience: *(i)* the members need to have characteristic traits that support the network; *(ii)* the members have to be able to trust each other; and, *(iii)* the network is more resilient if it has connectivity robustness —the ability to respond and recover from losses of critical nodes.

Ayling [19] explored the possible sources of resilience of criminal organizations, with particular emphasis on institutionalized gangs. According to this study, the

reduction of resistance increases the vulnerability. This can be achieved by shrinking the stability domain of the gang. For example, community support for a gang can be reduced by effectively improving financial and social conditions of the community.

Kenney [214] presented a comparative study of Colombian drug-smuggling enterprises, terrorist networks, including al Qaeda, and the law enforcement agencies that seek to dismantle them. The analysis revealed that the resilience of the Colombian drug trade and Islamist extremism in wars on drugs and terrorism stems partly from the ability of illicit enterprises to change their activities in response to practical experience and technical information, store this knowledge in practices and procedures, select and retain routines that produce satisfactory results. Traffickers and terrorists learn, building skills, improving practices, and becoming increasingly difficult for state authorities to eliminate.

Recently Duijn et al. [138] studied the resilience of criminal networks involved in organized cannabis cultivation.

In criminal networks, internal efficiency is somewhat hindered by secrecy. Therefore, even if a network becomes stronger after a targeted attack, the shortening of the chain of command causes a decrease of secrecy. Interventions of law enforcement agencies during the re-organization of a criminal network have a high chance to provoke a lasting disruption.

An interesting feature of fight against criminal and terrorist network consists in the identification of key players as introduced in [61]. They are defined as those nodes whose removal *"would maximally disrupt communication among the remaining nodes"*. The problem of resilience of terrorist network has also been addressed in [356]. The main idea is that the disruption of a network is efficient only if the key players are targeted and removed. To this purpose the authors developed STONE, a software platform which identifies the key players and suggests their removal. STONE is based on three algorithms which help in identifying: i) the successor of a terrorist, ii) the shape the new network will have after the removal of a group of terrorists, and iii) a set of new terrorists to be removed from the new network.

Criminal and terrorist networks are known for their ability to regenerate after targeted attacks. In [83] a new approach is introduced: the resilience of a network is reduced by increasing its network-wide centrality (first introduced by Freeman [160]), namely making it a more centralized organization. The authors introduce the term *shaping* to refer to the modifications that security or law-enforcement agencies have to induce in a network in order to increase its network-wide centrality and therefore make it more fragile to targeted attacks.

5.6 NETWORK VISUALIZATION

A number of visual representation methods has been introduced in the previous Chapter. The overwhelming majority of these metaphors are variations of sociograms, in which network components are shown as graphic elements and their relations as connection lines [377]. This representation allows an easy comprehension and provides detailed information about real relations emerging from data.

Graphical representation adopted in SNA since the origin [161] is a natural and fast method to highlight links among individuals. Some problems, however, arise

relative to the optimization of graphical representations [36]. While some algorithm tries to minimize crossings among edges while keeping a certain readibility, their efficiency decreases as the number of elements to be represented increases. This feature has negative consequences for analysis and evaluation of networks. Two of the most popular tools used by sociologists for social network analysis are UCINET [60] and Pajek [35]; they focus on statistical analysis and feature-limited interaction in their visualizations.

Pajek was built to overcome UCINET's limitation on network size, and it describes itself as a "program for large network analysis". Its interface resembles UCINET, inasmuch as it organizes its analysis methods in deep, hierarchical menus and outputs all analysis to a textual report screen. Unlike UCINET, Pajek has a built-in graph visualizer. It supports recursive factorization of such big graphs into several smaller networks, which are then visualized. In addition to standard layout algorithms such as force-directed and hierarchical algorithms in two and three dimensions, Pajek contains eigenvector methods, block matrix representations, and supports user constraints.

Other software applications focus on improving the interactive exploration of a social graph. One of them is GUESS [3, 4]. Ghoneim et al. [169] presented a graphical representation based on a matrix layout rather than usual graph diagrams. SocialAction [312] is a system that uses attribute ranking and coordinated views to help users systematically examine numerous SNA measures. NodeXL [349] is an extensible toolkit for network overview, discovery and exploration implemented as an add-on to the Microsoft Excel spreadsheet. JUNG [210, 303] and Prefuse [192] are Java toolkits that give the programmers the possibility to create analysis tools for social networks. visone [68] is a tool for the analysis and visualization of social networks. The visone software is an attempt to integrate analysis and visualization of social networks and is intended to be used in research and teaching. Methods are based on force-directed, spectral, layered, and radial layout methods. Tulip [18] is a huge graphs visualization framework, and here this means graphs with up to 1.000.000 elements. This software focuses on clustering in two and three dimensions, metrics algorithms, and visual attribute mapping algorithms for the purpose of information visualization. Special emphasis is put on the running time and the memory consumption of the algorithms. igraph[5] is a free software package for creating and manipulating graphs. It also implements algorithms for some recent network analysis methods. NetworkX[6] is a Python language software package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks. Some of these tools have been designed to analyze a generic graph. They often combine the node-link layout with standard statistical schemas, such as dispersion graphics and histograms. Although structural visualization are still used, technology succeeded in making graphic effects more and more refined and able to cover other dimensions beyond the structure of the network graph, such as the semantic and temporal dimension, necessary to comprehend social dynamics which allow user to assert hypotheses and validate theoretical and visual inferences [112]. In other words, analysis and visualization converge together with the interaction. The traditional analysis preceded by data processing and the visualization as a

[5] http://igraph.org/redirect.html
[6] https://networkx.github.io/

presentation instrument have been replaced by an interactive approach, in which visualization comprehends raw data and metrics derived from automatic analysis. Force-directed layout, extensively used in our work, assimilates the structure of the graph to a physical system, in which nodes are seen as material points subject to forces of various kinds; the coordinates of nodes (and therefore the layout) are calculated to obtain an equilibrium configuration of the modeled physical system [66]. A force-directed strategy consists of two main phases: i) *modeling*: starting from the choice of the features to highlight in the layout, a physical model is studied, by assigning attractive/repulsive forces to all pairs of nodes and, eventually force fields independent from nodes; ii) *research of equilibrium configuration*: given the system of forces and starting from an initial configuration in which positions are approximately or randomly fixed, various iterations are needed in order to find a configuration which minimizes the total energy of the system.

5.7 A CASE STUDY

Datasets used in this Section have been built starting from publicly available judiciary documents relative to legal prosecution against a mafioso criminal organization active in Sicily. For privacy sake, in the following details about places and/or people will be omitted, since final sentences against some members of the organization have not yet been emitted. Information about interpersonal relationships have been integrated with data extracted from Facebook and a certain number of newspaper articles available on the Web.

The criminal organization under consideration, originally composed of more than 500 members, has a pyramidal scheme, the boss being on top. The boss takes advantage of the collaboration of a limited number of counselors, who in turn are in charge of specific tasks. Decisions are taken during secret meetings limited to the members of the directive council. The initial configuration of the network is shown in Figure 21. It is a multiplex representation of a Criminal Network (CN) in which the layers correspond to an interaction network of criminal associates, money transfer relationship, phone calls connections, online social network relationships (facebook). In Table 2 we report the properties of the layers of this network.

Table 2: Criminal Networks datasets

Layer	N	M	$\langle k^{[\alpha]} \rangle$	APL	d	WCC
Phone calls	400	882	4.41	3.70	7	1
Facebook	325	2514	15.52	3.75	11	1
Crimenet	104	2596	49.92	1.53	3	1
Money T.	79	69	1.75	3.93	11	12
Aggregated	400	5774	28.87	2.40	5	1

The Criminal multiplex network consists of four layers, one for each kind of relatinship. For each layer α we report the number of nodes N, the number of edges M, the average degree $\langle k^{[\alpha]} \rangle$, average path legth, diameter and weakly connected component. We also report the number of nodes, the number of edges, the average

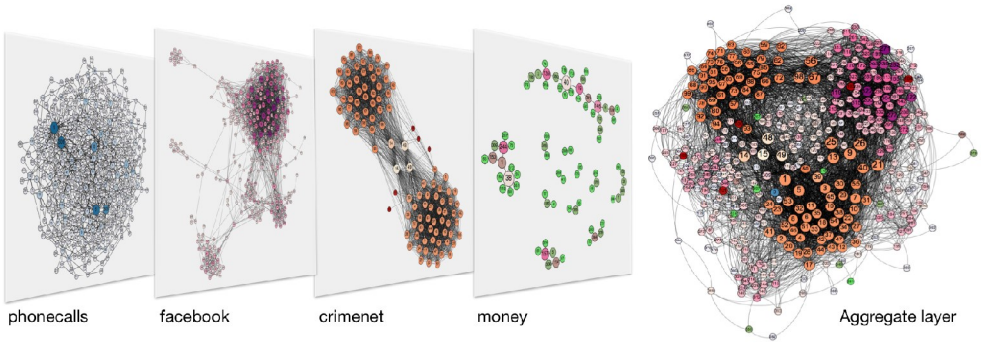


Figure 21: Multilayer representations of a Criminal Network in which the layers correspond to an interaction network of criminal associates, money transfer relationship, phone calls connections, online social network relationship (facebook). In the fifth layer, we show an aggregated network. On the aggregate network we mantein the nodes colours is equal to those the same of the nodes on crimenet and money layers, considered the most 'strategic' for the resilience of the criminal network. This representation was used in the experiment described in this Section to highlight the key features that make it resistant to attack criminal networks SF.

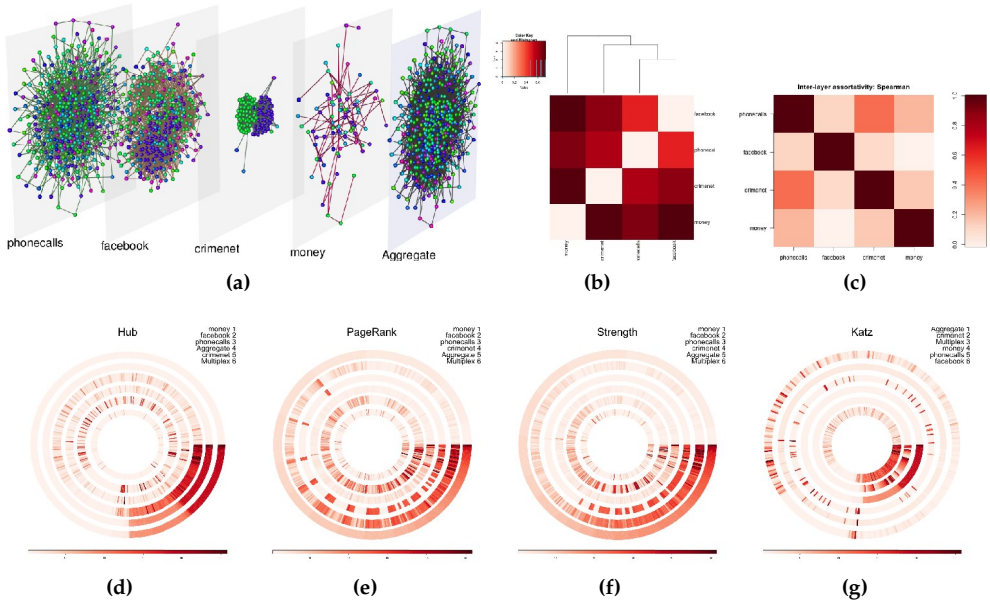
degree, average path legth, diameter and weakly connected component of the single-layer network obtained by aggregating all the layers.

We show a "reducibility dendrogram" in Figure 22b. A reducibility dendrogram merges a set of layers in a Criminal network step by step, and calculate a quality function based on the relative Von Neumann entropy to estimate information gain (or loss) at each step [131]. To obtain a reduced version of the original multilayer network, we stop the merging procedure at the level of the hierarchy that maximizes the relative entropy. In Figure 22c, we show degree-degree Spearman correlation coefficients between layers to quantify the tendency of nodes to be hubs in different layers simultaneously.

To summarize all of the information obtained from multilayer-network calculations in a compact figure, we include an annular visualization from `muxViz` that facilitates the ability to capture patterns and deduce qualitative information about multilayer data. Figure 22d, 22e, 22g and 22f show the annular visualization of some centrality diagnostic on the multiplex criminal network. Each ring refer to a metric. The angle indicates node identity (regardless of the layer or layers in which occur). The color of each bin encodes its value.

A common feature of criminal organizations is represented by the attractiveness of apical positions, in particular of the boss position. This is particularly evident during interregna, when strong competing instances emerge before a member is appointed as the new boss. Usually, the need of turnover derives from an objective inadequacy of the boss and his action when situations change and may threaten economic incomes.

In the criminal organization whose network we studied, the internal power struggles were bloody but not devastating. Finally, clans recognized the supremacy of the winning part which, in turn, accepted new rules regarding the division of spheres of influence and the obedience to the boss of bosses.



(a) (b) (c)

(d) (e) (f) (g)

Figure 22: Multilayer analysis. **(a)** Multilayer representation and the corresponding aggregated network of Criminal Network. **(b)** Distance matrix, based on quantum Jensen-Shannon divergence between each pair of layers and the corresponding reducibility dendrogram, which indicates the order in which pairs of layers are combined in hierarchical clustering [123]. **(c)** Degree-degree correlations quantified by pairwise Spearman coefficients between layers. **(d, e, f, g)** Annular visualization of Hub, Page Rank, Strength and Katz centrality: rings represent the layers or the multilayer network. The labels on the upper right from the inner ring (top) to the outer ring (bottom).

In Figure 23 is shown the criminal network during the climax of the struggle for separation. The two groups show some significant structural differences. Although the group on the left is less numerous than the other, it is denser than the opposite group and it is composed of members more specialized in committing crimes of different types. On the other hand, the group on the right part of Figure 23 is characterized by a lesser number of interactions and it shows less connection and amalgam. Moreover, members of this group are specialized in committing only two type of crimes. Key players shown in the middle of the graph will be subject to a destabilizing attack by the law enforcement agencies.

The intense internal struggle did not completely destroy the criminal network which maintained the control of illicit activities in the territory, even if a new structural configuration was implemented. The areas of competence and the coexistence relations have been strengthen.

The second phase of the study deals with the resilience of the organization to the strategies of attack conducted by the law enforcement agencies starting from the network structure shown in Figure 23.

The investigative phase focused on the so-called key players as they emerged from the transcripts of the court proceedings. Key players are the most important nodes: their rank has been attributed according to their position in the network rather than popularity. Nodes having the highest betweenness centrality have been focused. This

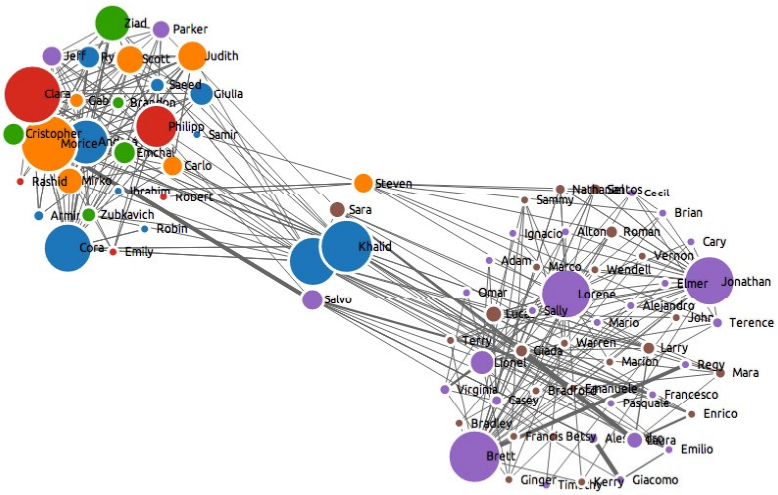


Figure 23: A criminal network during an internal struggle for a change at the top of the organization. The visualization layout applies diverging forces to nodes according to the group they belong to, thus resulting in the configuration depicted here. Dimension of nodes is proportional to their degree; color illustrates the criminal environment.

strategy is associated to the potential control activity of key players that could be related to the promoters of the association or involve central and bridge actors.

The ability of a criminal network of restoring its structure after a disruptive strategy is based on the retrieval of those connections which were destroyed. It is necessary to find substitutes that may fill the gap left by the removed member (for example because he has been arrested) and that interrupted or destroyed the paths of the network he belonged to. The substitute must be endowed with the same competencies and knowledge of his predecessor. After the replacement of a member with another who is in charge of the same responsibilities, the essential connections are restored and the paths winding through the missing actor are active again. As already described in Section 5.5, these substitutes are often redundant contacts of the network. An example can clarify how this process takes place. The software tool we developed for the semantic visualization of the criminal network allows to analyze the relations among the members of the criminal network together with the relations of friendship and kinship every member maintains. In Figure 24 is shown the core of the criminal network. Edges of different colors have been used to represent different relations: i) black for friendship, ii) green for kinship and, iii) red for membership. The type of relation is also visible thanks to labels on the edges. This representation greatly helps in the analysis of the network when a path has been interrupted (for example, as a consequence of the removal of an element). The interaction and the filters allow to detect those links that are redundant or prone to their substitution, as shown in Figure 25.

If, for example, 'Sistha' were arrested, the structure of the organization could be significantly damaged as a consequence of the prestigious position she has. In this case, a possible and fast substitution could be represented by her kin 'Micaela'

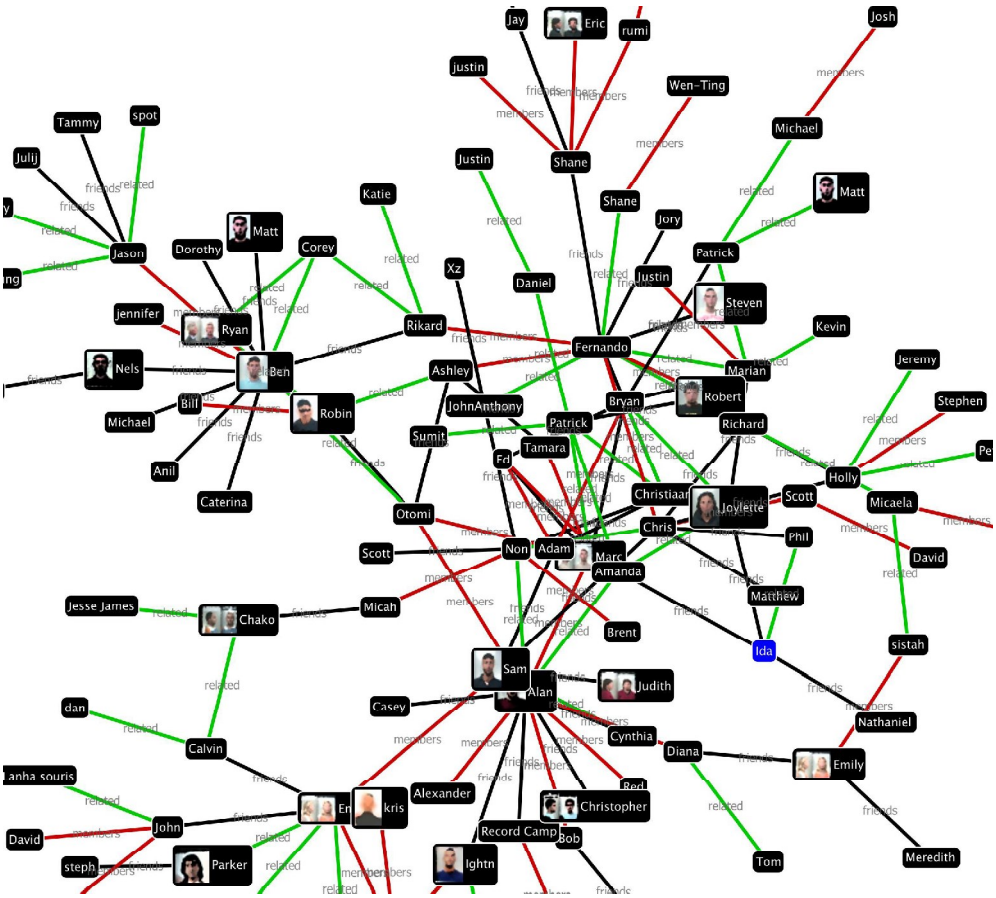


Figure 24: Criminal network visualization using a semantic layout. A detail of the central part (core) of the network.

(who already belongs to the criminal organization). Although the two nodes were already connected through a relation of membership to the criminal network, the redundant links suggest more indications as to the dynamics of structural variations and therefore are very useful for the evaluation of the resilience and, from an investigative point of view, the weak members of the network on which an attack should be concentrated.

5.8 CONCLUSIONS

In this study we tried to unveil the dynamics of resistance of a mafioso-type criminal network as an effect of two different types of interruption. According to different strategies of resilience, it has emerged that notwithstanding strong perturbations, both internal and external, the criminal network succeeded in reconstructing its structure, by reorganizing and accomplishing the necessary substitutions of the missing members.

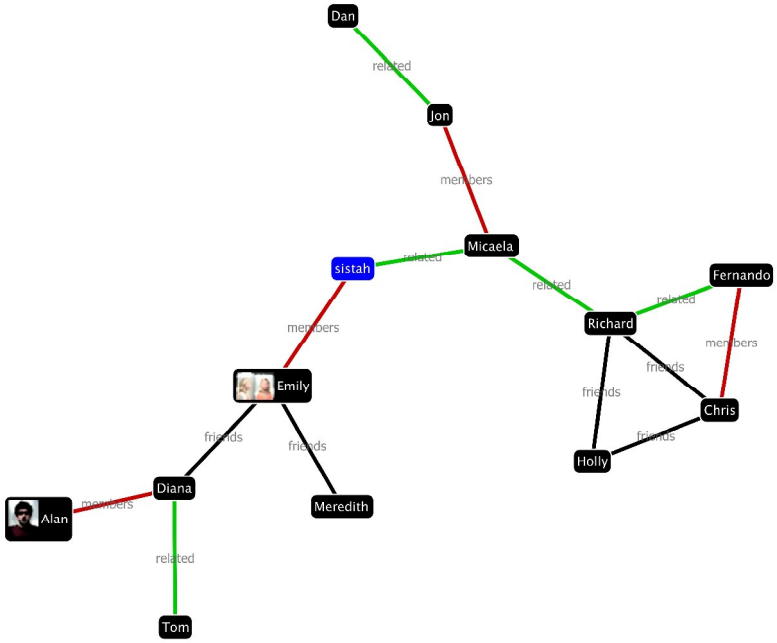


Figure 25: Filtered semantic layout

Large economic incomes, deriving from the criminal activities, sustain criminal networks during the transformation processes giving them the possibility to resist even in situations of large pressure.

Even after important removals, the efficiency of the network seems not to suffer significant effects. On the contrary, it increases over time thanks to strategies of restoring and/or building new paths and reducing the overall dimension of the structure. Results do confirm the powerful organizational structure of mafioso-like criminal associations which are flexible, adaptive, and highly resistant against the most incisive interruptions.

The ability of criminal networks of secretly reorganizing after an attack depends on its flexibility. It is necessary when trustable substitutions are needed within the network. Moreover, non-redundancy may be essential to find substitutions at larger distances, so to maintain secret both roles and information and avoiding to expose the overall organization to an abrupt stop.

Thanks to these strategies of resilience, the network does not expose in the long term, thus avoiding significant perturbations.

Another conclusion that can be drawn from this study is the temporal evolution of a network and its consequences on the study of its resilience. A future study will be necessary that focuses on the resilience along a timeline rather than analyzing a few static snapshots of the network.

6 DETECTING CRIMINAL ORGANIZATIONS IN MOBILE PHONE NETWORKS

The study of criminal networks using traces from heterogeneous communication media is acquiring increasing importance in nowadays society. The usage of communication media such as phone calls and online social networks leaves digital traces in the form of metadata that can be used for this type of analysis. Data sets on mobile telephone calls have many advantages over other sources for studying social and criminal network behavior. First, mobile telephones are ubiquitous and used by all age groups and in all social strata, whereas the user base of, say, Twitter cannot yet be considered as representative of the general population. Second, a phone call needs to be picked up before its details are recorded as CDRs by the operator (caller, callee, time, duration). Hence, CDRs are records of verified, time-stamped one-to-one communication. This greatly facilitates constructing social networks from the data, and especially allows for temporal analysis of communication patterns, conversely emails where recipient lists may be long and where there is no guarantee when or if an email has been actually read [332].

The goal of this work is twofold: first we provide a theoretical framework for the problem of detecting and characterizing criminal organizations in networks reconstructed from phone call records. Then, we introduce an expert system to support law enforcement agencies in the task of unveiling the underlying structure of criminal networks hidden in communication data. This platform allows for statistical network analysis, community detection and visual exploration of mobile phone network data. It allows forensic investigators to deeply understand hierarchies within criminal organizations, discovering members who play central role and provide connection among sub-groups. Our work concludes illustrating the adoption of our computational framework for a real-word criminal investigation.

6.1 INTRODUCTION

We live in a society where ubiquitous connectivity allows millions of users to communicate and enjoy the services provided by the Internet and other communication technologies, now even in mobility, by the technical and commercial success of handheld devices (smartphones, tablets, etc.). Such type of human communication activities produces a deluge of metadata and digital traces that have been studied to understand inter-connectivity and mobility patterns at scale [43, 85, 145, 146, 304, 305]. Online social network services such as Facebook and Twitter further increase the amount of information available to describe users' interests, activities and behaviors [8, 44, 95, 96, 109, 110]. Powerful technologies are although prone to abuse: mobile phone networks and online social media are constantly used to perform or coordinate criminal activities [277, 390]. Mobile phone networks can be

used to connect individuals involved in criminal activities in real time, often during real-world criminal events, from simple robberies to terror attacks. Online social media, instead, can be exploited to carry out illicit activities such as frauds, identity thefts or to access classified information.

Criminal network analysis is pivotal when applied to the investigation of organized crime like terrorism, narcotics trafficking, fraud, etc. [390]. Criminal organizations are established based on the collaboration of criminals who usually form groups with different roles. The analysis of a criminal network is thus aimed at uncover the structural schemes of the organization, its operations and, even more importantly, the flow of communications among its members. In this respect, law enforcement agencies and intelligence agencies often deal with large amounts of raw data gathered from various sources, including phone records and online communication, in order to unveil the network of relations among suspects. In modern investigative techniques the analysis of phone records represents a first approach that precedes a more refined scrutiny covering financial transactions and interpersonal relations. For these reasons a structured approach is needed.

The goal of this work is twofold. First, we provide a computational framework based on theoretical foundations and principles from network science, forensic science and statistical analysis to detect and characterize criminal organizations in networks reconstructed from phone communication records. Then, we propose an expert system, called *LogAnalysis*, that implements such framework.

The problem of detecting communities in criminal networks is here formalized as a two-step process: the first step aims at unveiling such communities hidden in larger networks of organic communication involving potentially many individuals, over different time scales; once such criminal organizations are clearly identified, the second step involves the study of the relations existing among the members of the criminal gangs, their communication dynamics, the reconstruction of the their hierarchical relations, to infer the structure of the entire organization and the roles they play therein.

Our expert system implements this computational framework encoding the entire work flow discussed above. `LogAnalysis` for example automatizes the import of raw phone call records data, the removal of ambiguities and redundancies in data, and the parsing and conversion to a graph format readily available for analysis and exploration. The data model is designed to improve the quality of the analysis of social relationships observed inside phone call network data through the integration of visualization and social network analysis-based statistical metrics. `LogAnalysis` implements different state-of-the-art view layouts for promoting fast and dynamic network exploration. It introduces the possibility of analyzing the temporal evolution of the connections among individuals of the network, for example focusing on particular time windows in order to obtain further insights about the dynamics of communications before/during/after particular criminal events. Finally, it provides an unprecedented supervised community detection set of techniques that allows detectives to interact with the community detection process, incorporating expert knowledge to supervise the results and refine the unveiled community structure at different levels of granularity and resolution.

A number of existing tools support network analysis but only some of them have been developed for criminal network investigation. Related to our work we

cite commercial tools like COPLINK [101, 389], Analyst's Notebook[1], Xanalysis Link Explorer[2] and Palantir Government[3]. Other related prototypes described in academic papers are Sandbox [387] and POLESTAR [314]. *LogAnalysis* represents the next-generation criminal investigation expert system in that it introduces significant improvements over these tools, and it provides specific support to detect criminal organizations in network data reconstructed from phone records.

The strength of *LogAnalysis* consists in the adoption of several statistical and interactive visualization layout techniques that improve network analysis while highlighting different aspects and features of the considered network and identifying and visualizing community structures.

6.2 *loganalysis*: MAIN FEATURES

In this section we summarize the main features of *LogAnalysis* including metrics and visualization layouts.

6.2.1 Network metrics

Members of criminal networks dynamically modify their relations with other members of the network thus resulting in a change of their role and importance. A series of centrality measures typical of the Social Network Analysis can help in capturing these changes.

Plenty of statistics are used to filter the network view based on specific node value and highlight their position inside the network: degree centrality, betweenness centrality, closeness centrality, eigenvector centrality and clustering coefficient.

6.2.2 Network layouts

LogAnalysis has been developed as a tool to help forensic detectives in the analysis of phone log records by means of a network representation. We adopted different state-of-the-art view layouts for promoting fast exploration and discovery of the analyzed networks.

It allows to analyze the relational structure of a criminal networks and to unveil the mechanisms of communications among its members. Various types of relations can be categorized by identifying those members who occupy central positions, those who play a key position in the communication flows among various groups (clans), etc. The application also enables to study the temporal evolution of the criminal network and to highlight some crucial information regarding the dynamics of the links in concurrence with criminal events.

[1] i2 - Analysts Notebook. http://www-03.ibm.com/software/products/en/analysts-notebook/
[2] Xanalysis (2014) - http://www.xanalys.com/products/link-explorer/
[3] Palantir government (2014) - http://www.palantir.com/solutions/

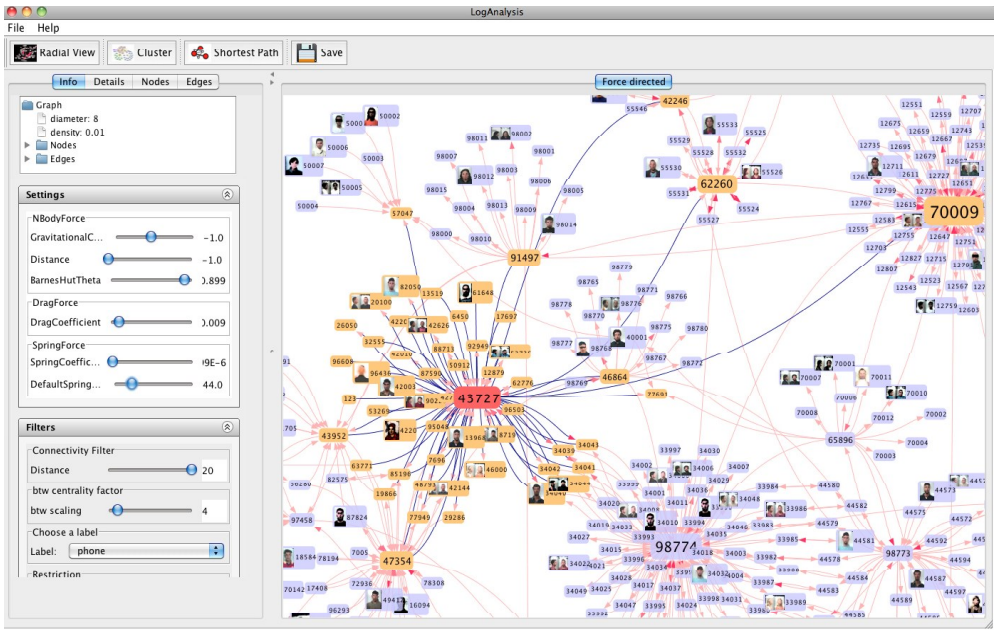


Figure 26: *Log Analysis* interface and force-directed layout. This figure shows the criminal network resulting from a case study of 543 nodes and 1229 edges. The node labeled in rad has been selected by the user. The nodes labeled in yellow are those at distance 1 from the selected node.

Node-link

Phone calls logs infer a social network. The tool mainly employs the node-link representation in order to visualize networks in which node was created for each unique cell phone, and an edge was created for each phone call. This results in a social network as shown in Figure 26.

To increase the readability of the network, when the mouse is passed over a node, first order connections of the node are highlighted. Moreover, it is possible to set the distance-based filter in order to represent only the nodes which fall within a given distance from the selected node. *LogAnalysis* also includes panning and zooming, and it implements the search by means of textual keys with the subsequent highlighting of nodes matching the query criteria.

Radial Tree

As shown in Figure 27 Radial tree layout allocates the elements of a graph in radial positions and defines several levels upon concentric circles with progressively increasing radii. The algorithm [395] also puts nodes in radial positions but gives the possibility of varying positions while preserving both orientation and order.

According to that technique, a selected element is placed at the center of the canvas and all the other nodes are subsequently placed upon concentric circles with radii increasing outwards. This visualization strategy is instrumental in the context of the forensic analysis because it allows to focus the attention of detectives on a suspect, and to have a close look to its connections.

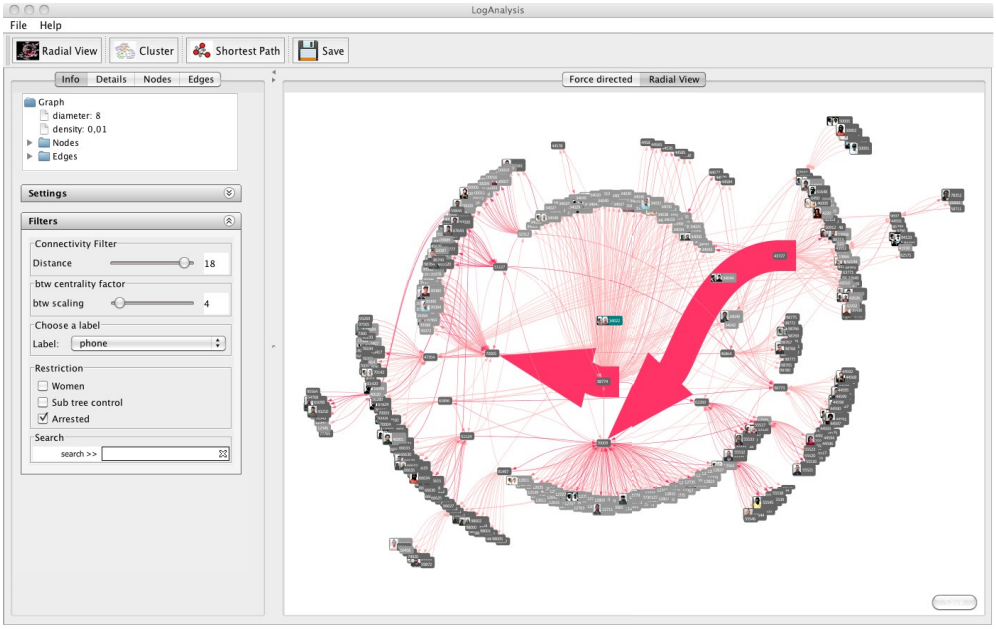


Figure 27: Example of Radial View layout. The node selected by the analyst is central in this visualization. The thickness of the edges connecting pairs of nodes is proportional to the amount of communication flowing between those pairs.

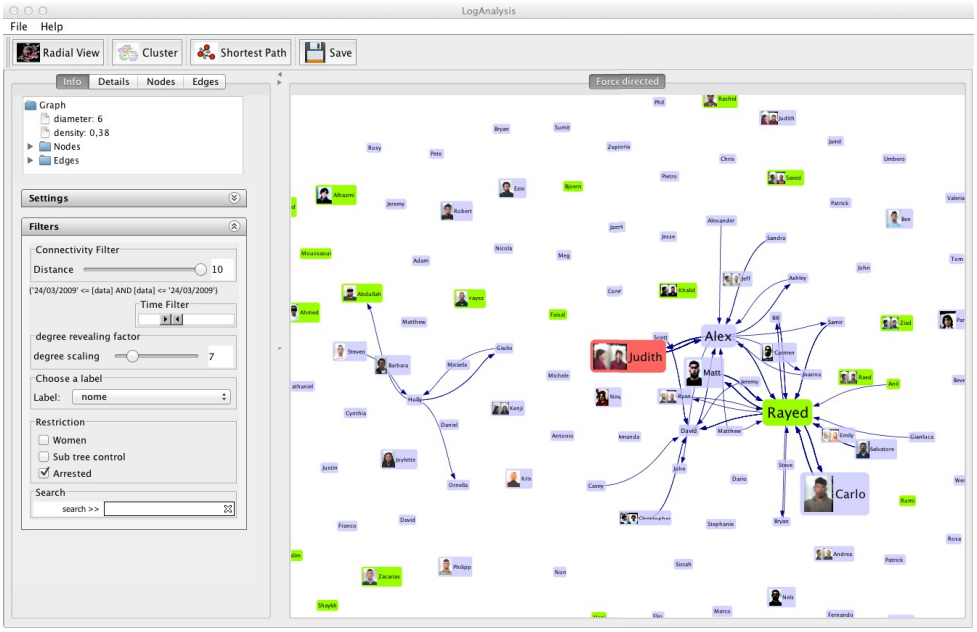
Nodes laying on the circumference of concentric circles, centered on that node, could be also progressively displaced from the selected one. Moreover, edges are visualized by using different thickness, calculated with respect to the number of calls among the given connected nodes.

Dinamics analysis

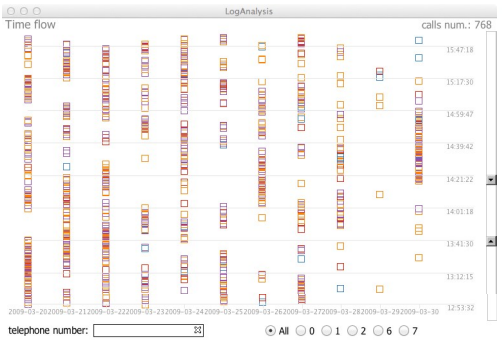
Phone call networks are not static and the structure of the network could change over time. So it is crucial for investigators "filtering" an analyzing the the social dynamics of the network with respect to specific temporal constraints. Our tool provide three temporal analysis features, shown in Figure 28, that heightens these capability:

i) *Time filter*. It is possible to select a time slice by using a slider. The structure of the network is filtered accordingly, removing all the edges representing connections (i.e., phone calls) which did not take place in that specific time window, and insulating (or hiding) those nodes not involved in the network at that given time. Modiying the time interval, nodes involved are automatically "engaged" or detached and are attracted or rejected inside/outside the network.

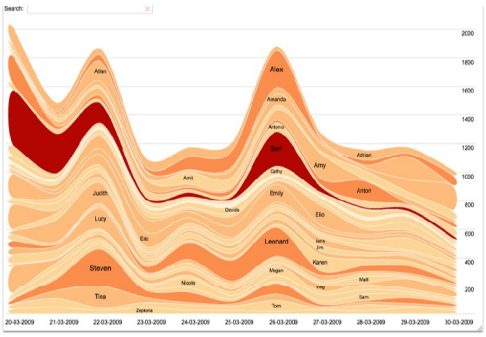
ii) *Time Flow analyzer* considers each single phone call as an *event* in a scatterplot. The days are on the x-axis and the hours on the y-axis. The colors of nodes are determined by the type of communication (i.e., sent/received calls and SMS and other type of communications, etc.). User can zoom in/out the time interval using a range slider to obtain additional insights about connections of events and query the data about specific key world.



(a) *Time filter.*



(b) *Time flow analyzer.*



(c) *Stacked histogram.*

Figure 28: (a) The Time Filter feature allows to investigate the network structure evolution. Nodes are dynamically engaged or detached according to the time range slider. (b) The Time Flow scatterplot is helpful to consider the time-dependence of events (i.e., phone calls) in a specific time window and it is crucial to highlight phone call cascades during criminal events. (c) The Stacked Histogram is helpful to visually summarize the communications among actors elapsed in a temporal interval.

iii) *Stacked histogram*. In this visualization each node in the network is assigned a stack. The thickness of each stack is according to the nodes degree at the time on the horizontal axis. This feature is helpful to get a picture of the phone call activity of the set of suspected elapsed during a specific time window ed in particular before, during and after criminal commission event. It is helpful to some interesting discoveries. For instance, why after the peak cell phones not contact each other? Any why, did the activity increase in a specific date?

6.3 CRIMINAL NETWORK COMMUNITY DETECTION

A criminal network is a special kind of social network with emphasis on both secrecy and efficiency. Such networks are intentionally structured to ensure efficient communication among members without being detected [381]. Knowledge about the criminal network structure is crucial to the investigators in order to reveal the functional or operational nature of the organization.

Typical criminal network information structures that emerge during investigations include hierarchical structure [329], cellular structure [364] comprised of cohesive subgroups connected by bridges, and flat structure [228]. These structures are emergent and evolving as the criminal network is modeled incrementally.

One of the most relevant features of graphs representing real systems like criminal networks is the community structure, or clustering.

The main goal of community detection in criminal networks (in particular, in phone call networks) is the identification of groups (or, clans) and their structures thanks to information coded in the topology of the corresponded graph.

In this section we will discuss how we approached the problem algorithmically and in terms visualization layout. To detect the community structure of the criminal network discusses as case study in this paper we use our tool *LogAnalysis*. This framework includes two strategies to detect and explore communities: i) Girvan and Newman algorithm [172] (in the following, GN) and ii) a variant based on modularity optimization, known as Newman's algorithm [293] fast enough to support interactive real-time adjustments.

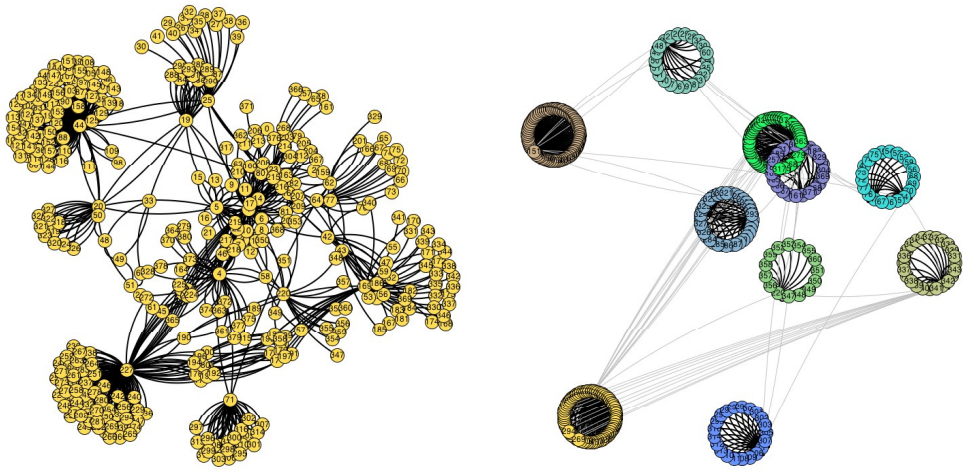
The simple idea behind the GN algorithm is to identify those edges that interconnect nodes belonging to different clusters and progressively remove them, so that the clusters are disconnected and the community structure emerges. The identification of bridge edges can be obtained by various means. In the case of GN, the algorithm adopts the edge betweenness centrality.

The following steps describe in detail how the GN algorithm works: *(i)* the edge betweenness of all the edges is computed; *(ii)* the edge with the highest value of edge betweenness is removed; *(iii)* the edge betweenness is computed for the new configuration and, *(iv)* the algorithm is repeated going back to step *(ii)*.

The edge betweenness centrality is computed in a $O(mn)$ time, m being the number of the edges and n the number of nodes. It has to be repeated m times, so the worst computational cost is $O(m^2n)$, or $O(n^3)$ for sparse graphs. Note that, although for large networks such high computational cost makes this solution often unfeasible, for criminal networks constituted (most often) by hundreds or at most thousands of nodes, this algorithm works well.

LogAnalysis visually presents the communities identified by GN via a force-directed node-link layout [165]. The deletion of an edge affects the structure of the network, iteration after iteration, and the network is represented accordingly: deleted edges are depicted as transparent. The number of the edges to be deleted can be chosen interactively. Finally, nodes are colored according to the cluster they belong to.

Figure 29(a) illustrates the typical structure of a network representing the phone calls network of 148 nodes and 210 edges, according to the node-link layout. Figure 29(b) shows the network after 46 iterations of the GN algorithm: 10 communities



(a) *Phone call network of 148 nodes and 210 edges.* **(b)** *Clusters detected after 46 edges deleted.*

Figure 29: Community detection using the Girvan Newman algorithm and the Fruchterman-Reingold layout. The sequence shown: (a) a phone call networks of 148 nodes and 210 edges (b) clustered view after 46 edges deleted. In this configuration modified force-directed algorithm visually present communities in circular layout.

have been detected. This configuration is a modified force-directed layout in which community members are visualized using a circular layout.

This characteristic of the GN algorithm is particularly well suited for the analysis of criminal networks: when the most central edges are progressively deleted, intermediate structures emerge, and an appropriate level of clustering can be determined.

The second method used in *LogAnalysis*, the Newman's fast algorithm, is a variant of GN aiming to maximize the network modularity function as described in Section 5.3.1 by means of a greedy strategy. It is a hierarchical clustering method in which groups of nodes are progressively aggregated in order to form larger communities whose modularity increases after the aggregation. At the first step, n clusters are considered, each composed of a single node. Edges are added one by one during the procedure. The modularity of the partitions is computed by taking into account the complete topology of the network. By adding the first edge to the set of disconnected nodes, the number of groups is decreased to n 1 so that a new partition is obtained. At each step of the algorithm the edges to be added are chosen so that the partition obtained results in an increase, or at least the minimal decrease, of the modularity with respect to the previous configuration. At each iteration, the variation ΔQ of modularity is to be computed as a result of the fusion of two any communities belonging to the running partition so to allow to choose the best resulting partition. The algorithm requires $n-1$ iterations, therefore its computational complexity is $O((m+n)n)$, or $O(n^2)$ in the case of a sparse graph. As a consequence, the community detection is feasible in the case of networks larger than those which can be tackled using the GN algorithm.

To visually present the Newman's algorithm results in *LogAnalysis*, community are shown within "convex hulls" (like in Vizster [191]). Additional forces separate the communities avoiding their overlapping. Besides the visualization of communities

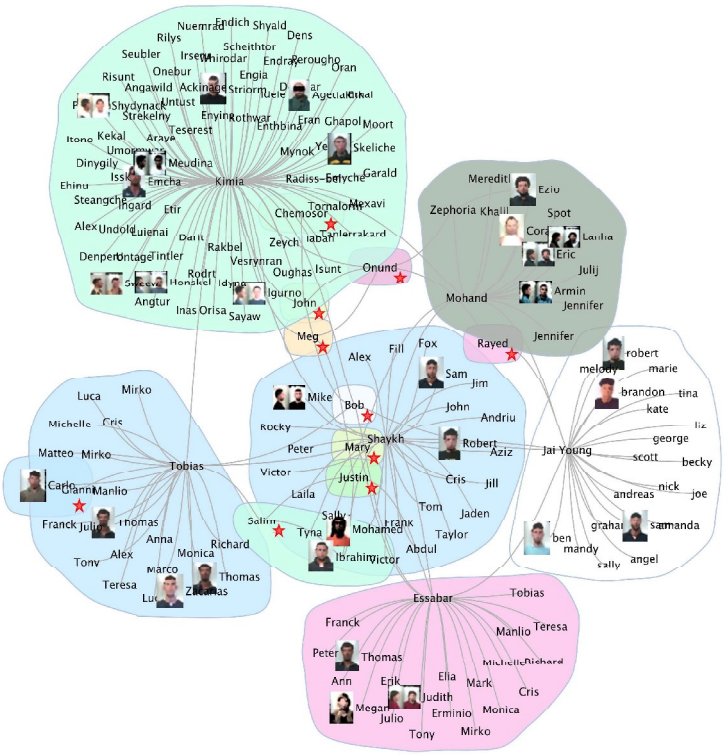


Figure 30: Community layout: Newman's fast algorithm [293]. The algorithm finds fourteen communities, eight of which are collapsed into a single node (For privacy reasons, photos have been anonymized).

inside the hulls, it is possible to filter and navigate the network by compressing the clusters around their most representative (i.e, central) nodes. Figure 30 shows a Newman community detection on a 223-node network.

Generally speaking, community detection methods based on modularity optimization are imperfect: some detected clusters can be larger with respect to the clans really existing in the network. This effect can be related to the resolution limit [157] mentioned in Section 5.3.1. To overcome this problem, GN and Newman algorithms are combined with a parameter so that users can tune the state of clustering at an given granularity. Analysts can split/merge the communities into smaller/bigger groups and can choose configurations that make more sense for their analysis. This capability is especially useful with dynamic networks such as telephone call ones, in which interpersonal relationships and the organization structure may changes over time.

Another feature of *LogAnalysis* is the possibility of interactively analyze the communities detected by the Newman algorithm. One example of this type of investigation is illustrated in Figure 31. In this small network of 18 nodes and 30 edges, the algorithm detected 5 communities (see Figure 31(a)). By setting to zero the parameter that tunes the number of inter-community hops, and selecting the convex hull of a given cluster (for example, the one containing four nodes, like in Figure 34(b)), the graph is filtered and collapsed accordingly: Figure 34(c) shows

how the specific cluster connects to the others (note that collapsed communities are identified by a red star and labeled with the id of the node(s) distant one hop from the selected node in the selected cluster). When the number of inter-community hops is set to 2, some of the clusters are automatically exploded (see Figure 34(d)) indicating that such communities are reachable in one hop from the selected one, while some others remain collapsed because their members are farther away from the selected community.

The set of techniques presented above simplify greatly the analysis of criminal networks inferred from (possibly large) phone call data. Our system allows to achieve a trade-off between granularity of the information presented in the visual interface, and the ability for the analyst to explore large networks and the criminal communities therein exposed. In the next section we focus on the characteristics of criminal network by means of a case study reconstructed from a real criminal investigation supported by our expert system.

6.4 A CASE STUDY

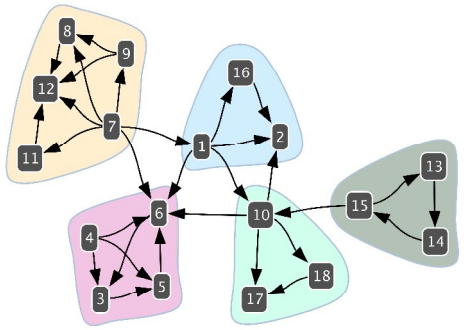
Real police investigations have been successfully carried out supported by *LogAnalysis*. In this Section we report a case study whose results were obtained during a forensic investigation. Although for sake of privacy protection some information is obfuscated, in the following we will drive the reader through all steps of the criminal investigation carried out by means of our framework.

6.4.1 The initial configuration

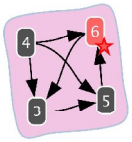
In this case, some people allegedly belonged to a criminal network. Among the available data about the structure of the criminal organization, phone logs undoubtedly convey the most important information that detectives use in order to verify the existence of interpersonal relationships and the communication flow.

The initial configuration of the network representing the phone call connections is shown in Figure 32. The network has been obtained from the processing of the log files containing the phone call traffic during a period of fifteen days among some people allegedly belonging to a criminal association responsible of a series of robberies, extortions and drug illicit trafficking. *LogAnalysis* may also automatically expand metadata on actors of the network, whether available: in this case (obfuscated) mugshots, and other metadata (e.g., criminal records, etc.), are autonomously extracted by consulting other internal police databases. In addition, for anonymization purpose, phone numbers are here replaced by numerical IDs. Information concerning the relational structure and some important statistical metrics are shown in Figure 32.

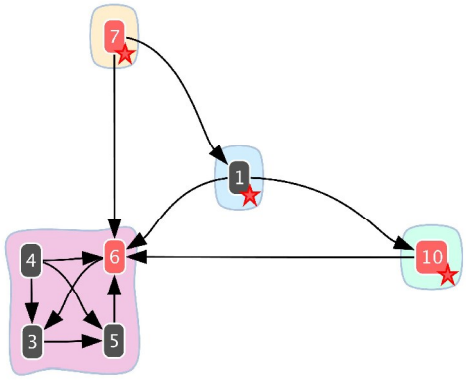
From the analysis of phone contacts among some people in a given time interval it is also possible to unveil the most important links in terms of frequencies of relations and flow of information. Links do not refer to the same type of relations and therefore it is important to improve the analysis starting from the community detection. Crucial is the ability to gain as much information as possible from the topology of the network and then ascertain the details.



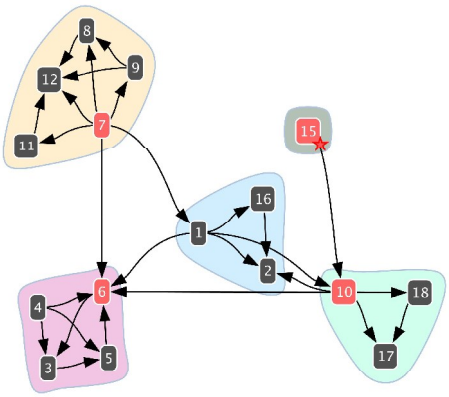
(a) *Sample network of 18 nodes and 30 edges where 5 communities have been detected.*



(b) *Selected community with 4 nodes. The node labeled with a red star indicates a connection to one or more communities currently collapsed.*



(c) *Configuration obtained setting to 1 the inter-community hop parameter. Red stars indicate convex hulls representing collapsed communities.*



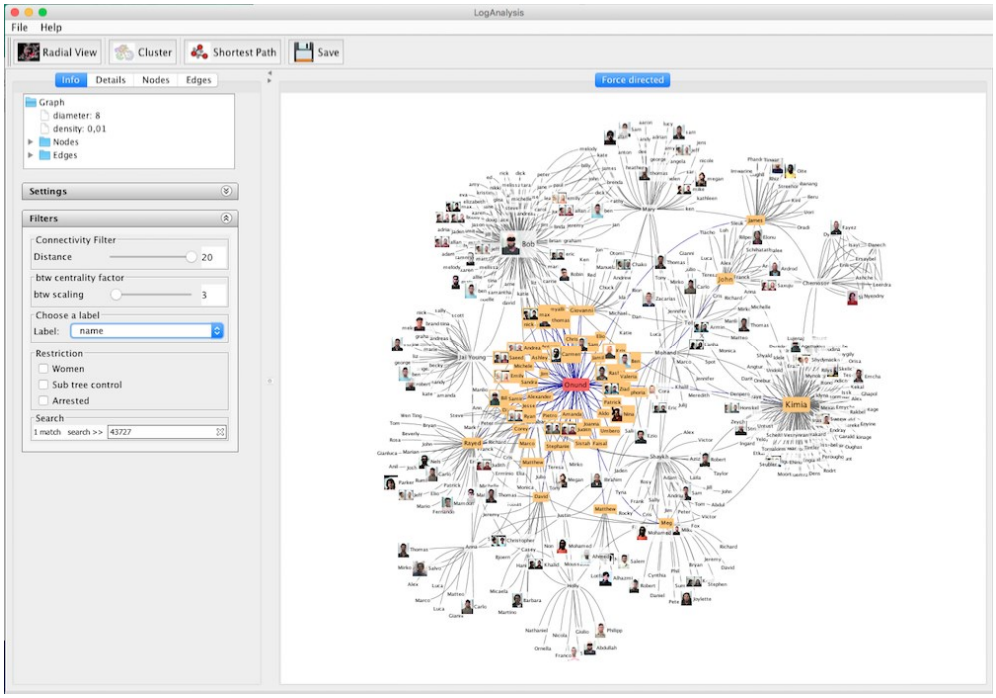
(d) *By setting the inter-community hop to 2, some communities are automatically exploded, while one remains collapsed.*

Figure 31: Example of community detection with the Newman algorithm, visualization and interactive exploration.

6.4.2 Finding subgroups

In Figure 33(a) we show the case study network after the GN algorithm has been executed and 16 communities have been detected. The assignment of each node to a community is visually encoded by the different colors used to depict the nodes. To improve the clarity of the network visualization, we exploit the clustered view as shown in Figure 33(b). This configuration adopts a modified force-directed layout in which nodes of the same community (same colors) form macro-nodes visualized with a circular layout. In such a way, inter-connectivity among communities is captured better. The macro-nodes can be further exposed to reveal intra-community relationships (see Figure 33(c)).

In this case we were mainly interested not only to those nodes which occupy prominent positions. Rather we focused on those edges whose deletion during the execution of the algorithm unveils new structural configurations which in turn can



(a)

Figure 32: Network visualization in node-link layout of the entire cell phone log data set composed by 381 nodes and 428 in 15 days of activities. Each node is a unique cell phone, and each edge is a relationship (calls, SMS, MMS, etc.) between them. Graph generated by `LogAnalysis`.

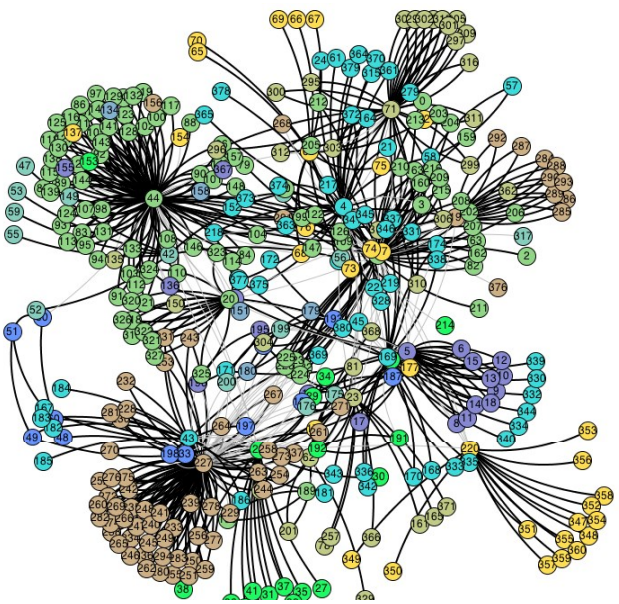
(a) Overall Metrics.

Network Metric	Value
Network type	directed
Vertices	381
Edges	428
Connected components	1
Self-Loop	0
Maximum Geodesic Distance	7
Average Geodesic Distance	3.898
Network Density	0.006
RecordSet	15,845
SMS	6342
MMS	133
Voice calls	7,334
Internet	1,180
Others	856

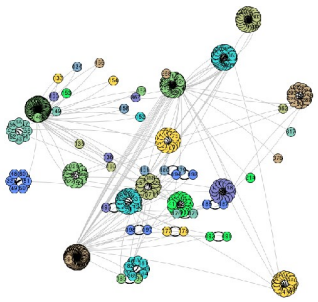
(b) Centrality measures of top 15 vertices in.

vertex	degree	btw centr.	page rank
134	845	5741.467	6.941
32	710	5870.000	5,718
18	532	13130.767	6.150
31	358	14245.000	14.183
91	349	12647.173	12.756
94	220	31622.172	19.896
106	211	19505.405	8.407
37	188	16559.613	7.458
102	163	28694.821	35.803
128	157	8293.357	9.812
124	152	5515.127	5.397
289	133	21104.667	29.204
25	130	4637.467	5.598
16	110	3224.286	4.498
69	105	2742.500	4.117

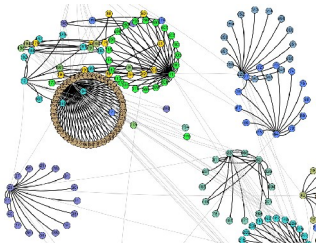
Table 3: (a) Overall metrics and (b) centrality measures of the top 15 vertices of the case study network.



(a) *Case study network after the GN algorithm. 16 communities have been detected.*



(b) *Clustered view. Nodes of the same community form macro-nodes visualized with a circular layout.*



(c) *Macro-nodes zoom reveals intra-community relationships.*

Figure 33: Girvan Newman community detection on case study network.

be investigated using other information available to police detectives. This analysis will result of fundamental importance for the successful outcome of the investigation.

Thus, the analysis of a criminal network can be accomplished using *LogAnalysis* as follows: *(i)* data extracted from heterogeneous sources must be parsed; *(ii)* a mathematical model in form of a network is derived; *(iii)* a node-link layout for the visual representation is chosen; *(iv)* communities are detected and visualized; *(v)* the member of each cluster is analyzed in depth and, *(vi)* step *(iv)* is refined using the results of step *(v)*.

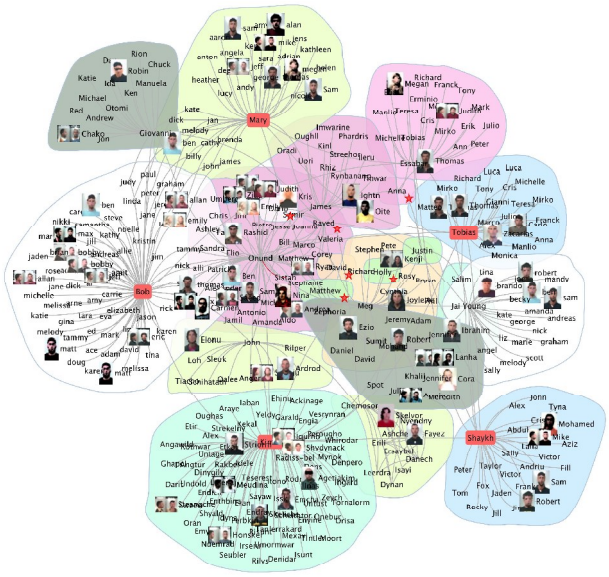
The choice of the best level of granularity during the clustering is not automatic. In this case study it was derived from Table 4, where are shown the edges which were deleted and the number of clusters which were obtained step by step until the best configuration was obtained. In Table 4 are shown the edges and the nodes through which information can flow towards all the members of the network or, at least, a large part of it. A detailed analysis demonstrated, however, that the more central edges are not always responsible of driving the majority of the information. They are, of course, important edges from a topological point of view and "lethal" when regarded as members of a criminal network, but only on a theoretical viewpoint. An important consideration follows: the algorithms of clustering, when used to analyze a criminal network help to detect the most close groups of the network, but the nature of the relations must be carefully evaluated using information which can not be directly drawn from the mathematical model or its graphical representation.

N. Clusters	N.	Edges	Vertices
2 Clusters	1	634	25⟷1
	2	576	64⟷44
	3	635	25⟷33
	4	679	5⟷1
	5	651	1⟷19
	6	617	25⟷42
	7	614	43⟷23
	8	615	25⟷43
3 Clusters	9	254	220⟷227
	10	301	220⟷169
	11	381	169⟷64
	12	681	1⟷4
	13	567	71⟷4
	14	559	1⟷77
	15	616	1⟷42
4 Clusters	16	610	20⟷44
	17	612	44⟷20
5 Clusters	18	638	1⟷23
	19	17	368⟷1
	20	306	1⟷219
	21	104	304⟷1
	22	300	220⟷4
	23	299	220⟷1
6 Clusters	24	639	23⟷4
	25	687	81⟷4
	26	16	368⟷4
	27	304	219⟷23
	28	641	22⟷23
7 Clusters	29	611	44⟷25
	30	601	50⟷44
8 Clusters	31	275	222⟷227
	32	255	227⟷226
	33	274	223⟷227
	34	276	221⟷227
	35	273	224⟷227
	36	272	225⟷227
9 Clusters	37	569	64⟷71
10 Clusters	38	341	169⟷199
	39	350	169⟷193
	40	347	169⟷195
	41	344	169⟷197
	42	372	169⟷177
	43	375	169⟷175
	44	369	169⟷179
	45	356	169⟷189
	46	359	169⟷187

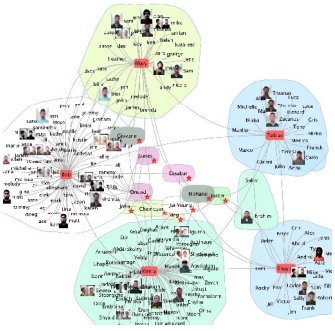
Table 4: Results of the application of the GN algorithm to the case study. Are shown the edges which were deleted at each iteration of the Edge Betweenness Clusterer algorithm along with the incident nodes. Are also shown the edges through which information can still flow towards all the members of the network or, at least, a large part of it.

The Social Network Analysis applied to our case study for example shows that the node with the highest degree (i.e., the highest number of phone calls) has a lower betweenness centrality if compared to other nodes. In fact, criminal networks heavily employ secrecy to escape investigations and, in particular, a policy of internal communications according to which the most important members issue orders to a very limited number of members which in turn make them known to an increasing number of less important members until the leaves of the network are informed.

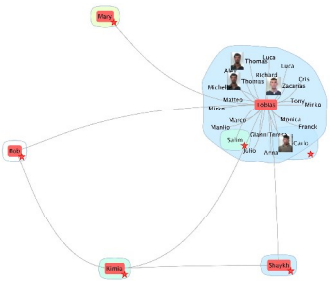
In our case study, the nodes having the highest number of communications (i.e., the highest degree) represent the lieutenants of the criminal organization and not necessarily the boss of the clan, while the edges traversed by the highest number of



(a) *Case study network after applying the Newman algorithm. 19 communities have been detected, 5 of which are visualized, centered on the selected node ("Tobias", in red).*



(b) *Filtered communities with inter-community hops 2 from Tobias node.*



(c) *Tobias egonet (intecommunity hops = 1).*

Figure 34: Girvan Newman community detection on case study network.

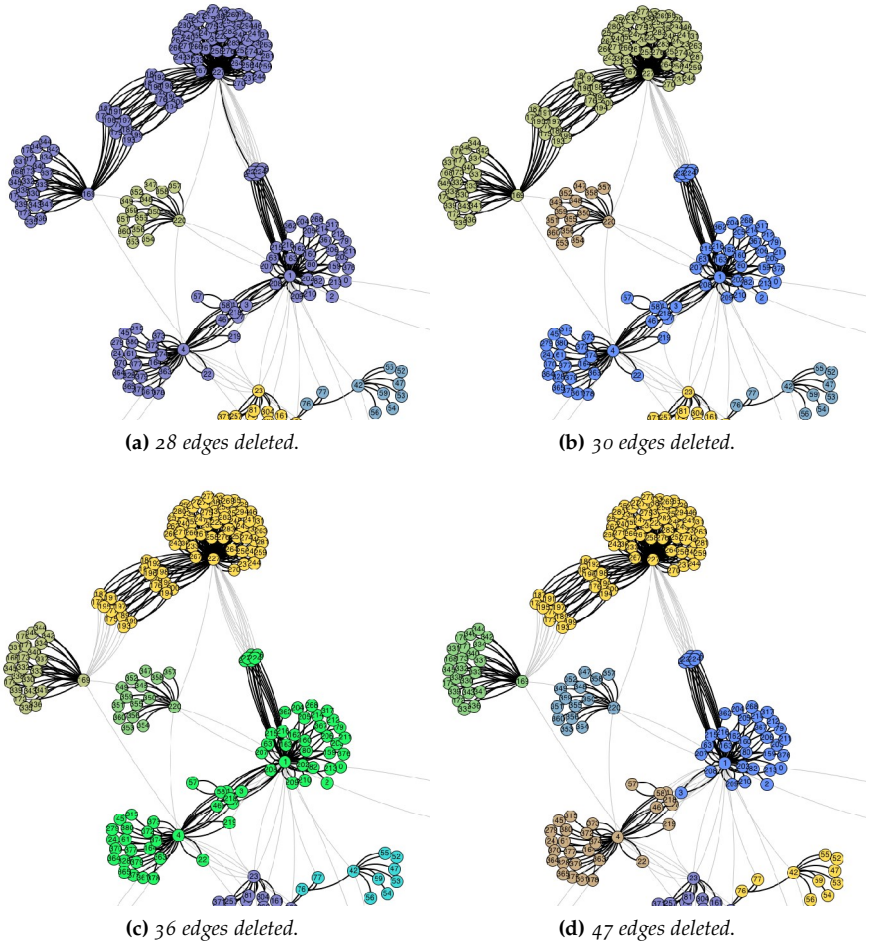
shortest paths (i.e., having the highest betweenness centrality) represent the most important links among the various groups.

Moreover, the granularity of the clustering allows to identify the members and the edges which represent the ideal target when trying to hinder the criminal activities of the clan.

The next step of analysis is carried out by using the Newman algorithm. Figure 34(a) shows communities embedded in convex hulls. Since the visualization might be cluttered and compromise the interpretation of the results, we here exploited the community compression techniques described above to improve the quality of the representation. For example, by setting the inter-community hop filter to a value of 2, Figure 34(b) shows the communities, and the respective members, that can be reached from the selected nodes at most in two hops. Figure 34(c) represents the egonet of the selected user, and the summary of communities connected in one hop.

6.4.3 Overlapping communities

As already discussed in Section 6.3 and in Section 5.3.1, an important aspect in the analysis of communities is represented by the potential overlap of communities. Both the algorithms implemented in *LogAnalysis* actually perform a partition of the



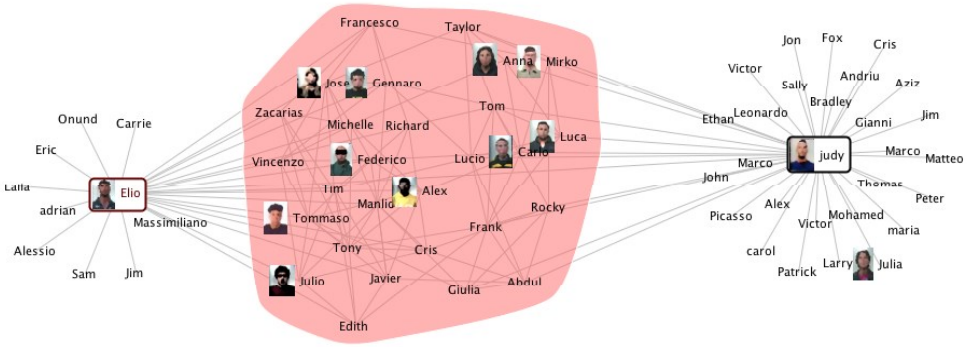
(a) *28 edges deleted.* **(b)** *30 edges deleted.*

(c) *36 edges deleted.* **(d)** *47 edges deleted.*

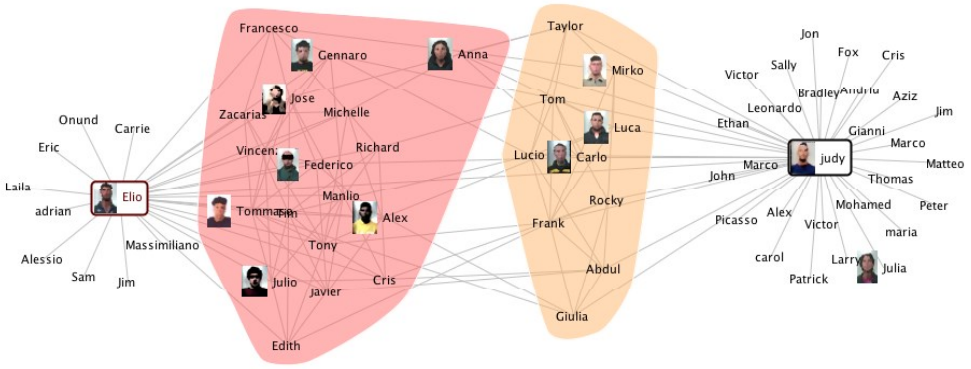
Figure 35: Non coherent examples of clustering produced applying the GN clustering algorithm [172].

network, thus assigning each of the the nodes to exactly one cluster. Often this is not a correct representation, at least on a semantic basis, of the network. In a specific case such ours, even the algorithmic approaches described in [308, 359] may produce questionable results because of the multiplicity of meanings which can be given to any edge of the network. For this reasons, we decided to implement *LogAnalysis* in such a way which allows the user to choice the level of clustering in order to approximate the results. The network shown in Figure 29 is an example of the level of clustering we believed to be the most appropriate according to the aforesaid criteria.

Some examples follow. Figure 35 shows a situation in which the GN algorithm can produce a series of results in which the outcoming partition does not correspond to the results of the studied network. In the example, only a portion of the entire network is shown. After the deletion of 28 edges (see Figure 35(a)), a community is obtained (colored in violet) which is composed of more groups. The deletion of two more edges leads to the configuration shown in Figure 35(b). Some of the



(a) Q_{max} *modularity.*

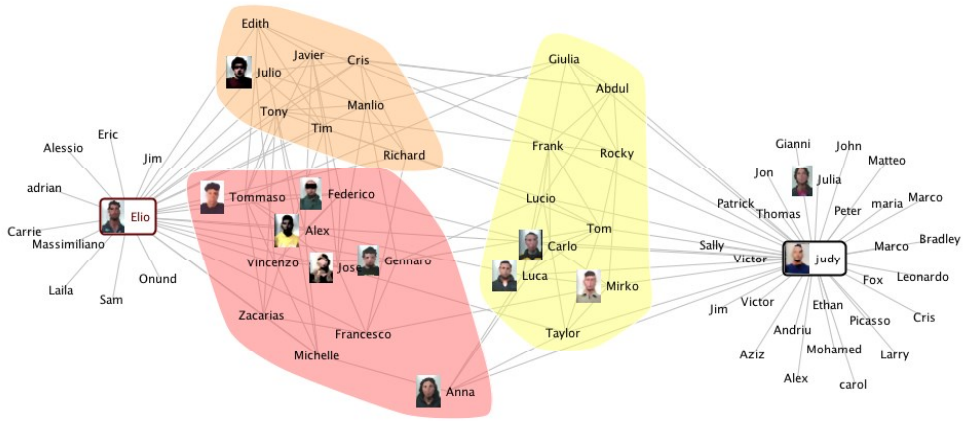


(b) $Q_{max} - 1$ *modularity.*

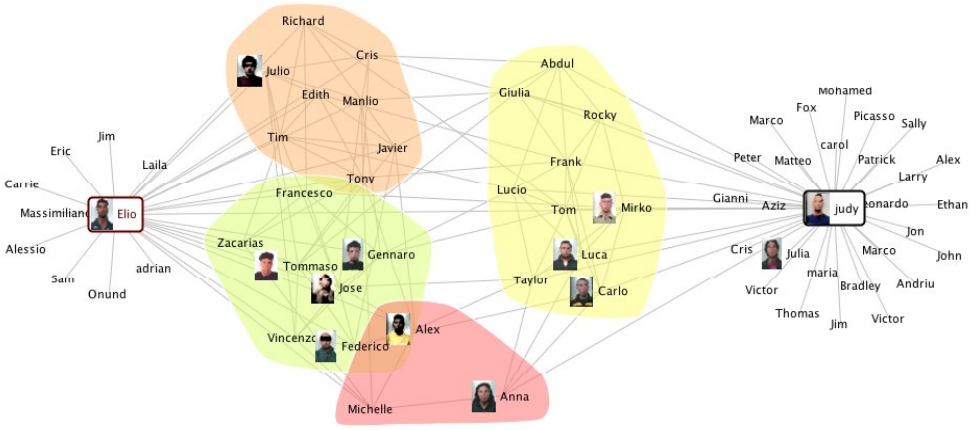
Figure 36: An example of community detection using the Newman algortihm [293]. The convex-hull layout has been adopted for the visualization of the communities.

nodes belonging to the blue cluster should belong also to the green cluster. In this example, even if the clustering obtained through the application of the edge betweenness centrality is undoubtedly correct from the computational point of view, nonetheless is debatable from the semantic point of view. Our conclusion is that in such situations an automatic computation should be supported by the assistance of the analyst. Other examples are shown in Figures 35(c) and 35(d). In these cases we have the same results when partitioning the nodes among yellow and green cluster (see Figure 35(c)) and when partitioning the nodes among blue and brown cluster (see Figure 35(d)). In each of the aforesaid examples, the interconnected nodes could belong to one or the other group or both, or more simply they could belong to a group of its own which has very few links to other groups.

Also the interconnections among various communities have been analyzed using the Newman community detection algorithm [293]. Figure 36 shows the initial phase of the execution of the algorithm. In Figure 36(a) only one cluster has been detected which is composed of the nodes interconnected among the external clusters represented by the nodes "Elio" and "Judy", while in Figure 36(b), Q_{max} has been in-



(a) *The criminal network at time t_1.*



(b) *The criminal network at time t_2.*

Figure 37: Community detection of a time-varying criminal network.

teractively decreased to a previous lower value. As a consequence, the interconnected nodes are subdivided and new communities emerge.

The in-depth analysis carried out on the members of the clusters interconnected shown in Figure 36 and the temporal analysis accomplished with *LogAnalysis* allowed the investigators to discover that some clans belonging to the criminal network had worked with a certain degree of autonomy and were responsible of some murders. It turned out from the investigations that these clans had the task of committing murders. In Figure 37 are shown the clans at times t_1 and t_2.

Some additional remarks follow. Applying Newman community detection algorithm with an automatic clustering produces a partition according to which the criminal network is composed of 14 clusters. This can be seen from the dendrogram shown in Figure 39. The maximum partition density is 0.014 and the largest community is composed of 84 nodes. As already seen, this clustering is not coherent with the real structural subdivision of the criminal network as it emerged from the super-

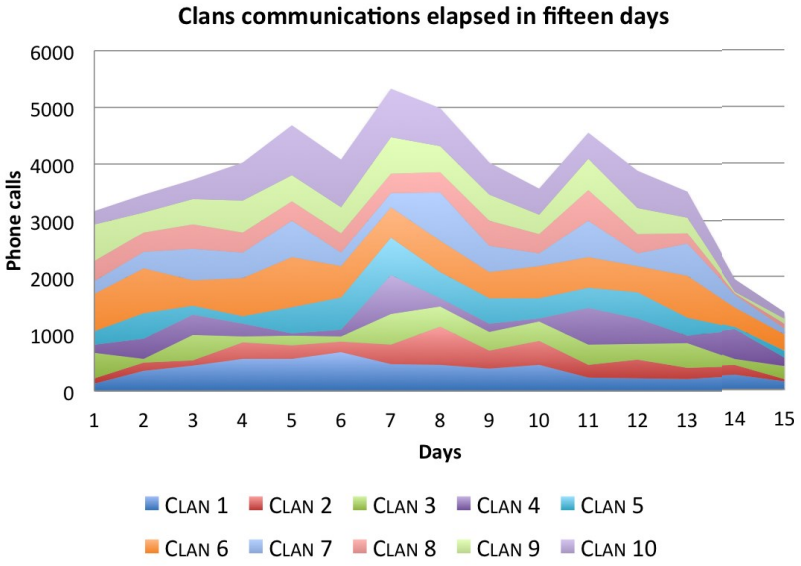


Figure 38: Stacked histogram showing the phone call traffic carried out by each community in the time interval of 15 days.

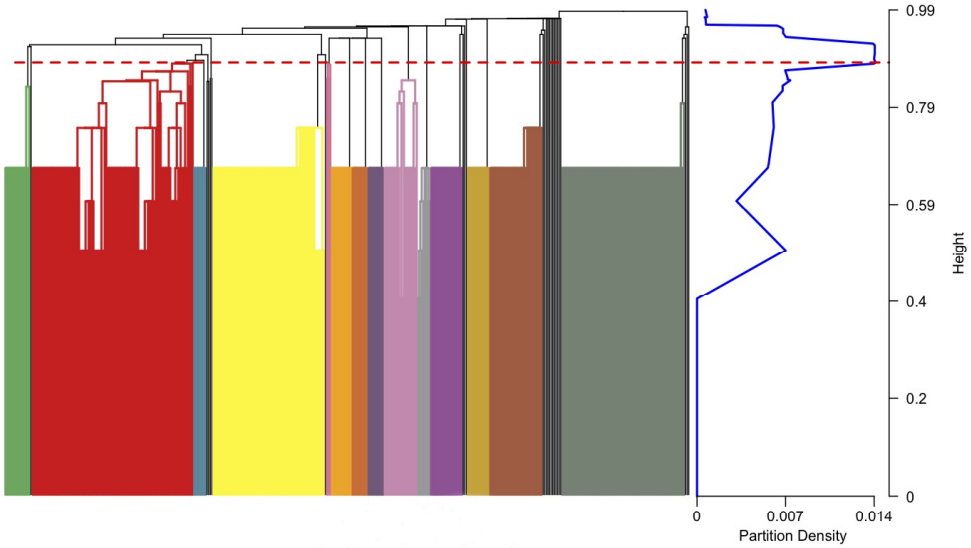
vised interactive community detection combined with additional comparisons and in-depth examinations obtained from other informative sources. Nevertheless this result was very interesting in that important information regarding some members of the network emerged.

In particular, from the analysis of the different levels of clustering interactively selected, and from the observation of the relative variations in the obtained configurations, we identified which elements of the network were affected mostly.

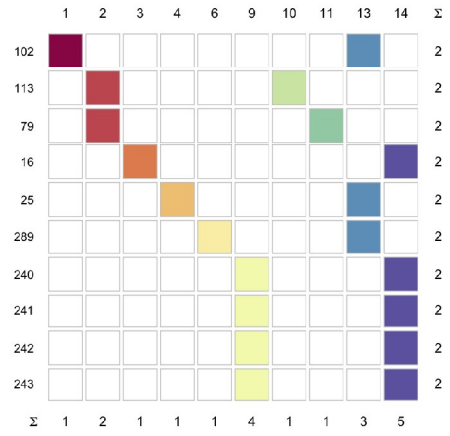
This is shown in Figure 39(b), with respect to the most connected nodes and how they belong to the 14 different communities. For example, node 102 belongs to clusters 1 and 13, while the most important nodes belonging to cluster 14 are 16, 240, 241, 242 and 243.

We also computed the modularity of the various communities, that is a measure of how dense are the connections among the nodes within the clusters in respect to the connections between nodes in different clusters. Figure 39(c) shows the modularity of each cluster of the criminal network.

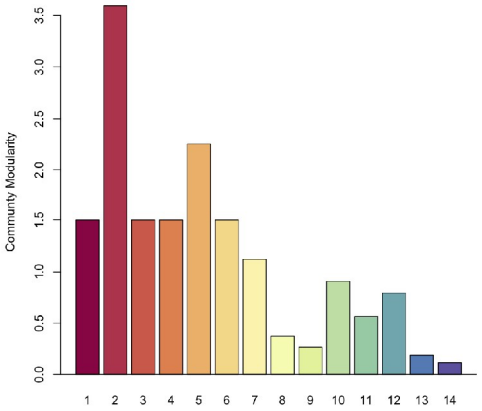
The analysis of the distribution of phone calls carried out by each "clan" is a method generally very useful it must be decided if a good level of clustering has been obtained after the execution of the community detection algorithm (see Figure 38). The goal of this analysis is twofold: first, it identifies the groups among which the largest number of phone calls, texts, MMS took place, second, it highlights the peaks of the stream of communications related not to single users but rather to each cluster as a whole, on the occasion of a crime.



(a) *Dendrogram: 428 edges, 381 nodes, 14 clusters, 84 nodes in the largest cluster.*



(b) *Community membership matrix for the most connected nodes. Colors indicate community-specific membership.*



(c) *Modularity of various communities.*

Figure 39: The figure shows: (a) the dendrogram resulting from the community detection; (b) the community membership matrix for the most connected nodes; (c) the distribution of modularity for the clustering resulting from the dendrogram cut of subfigure (a). Colors in the membership matrix correspond to those of the histogram in subfigure (c).

6.5 CONCLUSIONS

In the latest decade or so there has been an active involvement of academic researchers in the study of terrorist and criminal networks to improve public safety. In this respect, Social Network Analysis has proved a valuable tool in order to

ascertain the central members of criminal networks, the existence of subgroups, the interactions among individuals and subgroups, the flow of information in the network, the sensitive members and/or relations whose removal could eventually lead to the destruction of the network.

In this context, the analysis of phone call networks is crucial to gain fundamental information about inter-connectivity and communication among criminals, and to progress fruitfully the investigations. The study of information flow allows to identify those individuals who play a key role inside the criminal organization, or connect different subgroups. Statistical approaches also provide remarkable insights, for example if one considers the quantity of information and their temporal distribution with reference to a given criminal act. Moreover, the spatial distribution of such events can be taken into account, since it gives insights with respect to the identification of suspected individuals and their most frequent locations.

In this work we presented *LogAnalysis*, an expert system that allows for semi-supervised detection of criminal communities in networks reconstructed from phone call records. We discussed some of its features describing how they are instrumental to study criminal networks, presenting a case study inspired by a real criminal investigation. This allowed us to unveil few primary characteristics of criminal communities in real world phone call networks.

The analysis of criminal networks cannot be reduced, however, to the study of the relations established by means of phone calls. We must take into due account a larger amount of data, possibly originated from various sources. This is the case, for example, of physical meetings and financial transactions. Also time plays an important role, in that relations and transactions usually may or may not happen simultaneously.

To analyze such types of data a radical extension of the capabilities of *LogAnalysis* is necessary. For this reason we are designing a natural successor of *LogAnalysis*, conceived to study multiplex and temporal criminal networks. Such expert system will integrate and deal with multiple data sources including online social network data and financial records, and it will be integrated with other law enforcement databases to infer and learn new associations in a fully unsupervised way.

7 VISUALIZING CRIMINAL NETWORKS

In this Chapter we show how we employed some interactive visualization techniques to represent criminal and terrorist networks reconstructed from phone traffic data, namely foci, fisheye and geo-mapping network layouts. These methods allow the exploration of the network through animated transitions among visualization models and local enlargement techniques in order to improve the comprehension of interesting areas. By combining the features of the various visualization models it is possible to gain substantial enhancements with respect to classic visualization models, often unreadable in those cases of great complexity of the network.

7.1 INTRODUCTION

The pervasive diffusion of technologically-mediated communication channels pushed to unprecedented frontiers the ability of individuals to interconnect and exchange information. Mobile phone networks, social networking and media platforms like Facebook and Twitter, and over-IP messaging systems like Skype and WhatsUp, represent some examples of the multitude of communication media broadly adopted in nowadays society. These phenomena generated lot of interest in the research community. Several aspects of socio-technical systems have been studied [372]: from macroscopic characteristics, like network structure [128, 153, 229, 269], to network dynamics, like information diffusion [23, 155, 285, 328], from microscopic behaviors, like how individuals address their attention [238, 380] and what topics they discuss [103, 108], to social issues, like how people organize and mobilize using technology [110, 179] and what effects technological media have at the societal level [178, 271].

One aspect that has vast societal impact is the improper usage of such platforms. Technologies have been long exploited for criminal activities: for example, various studies showed how the Internet has been exploited for cybercrime, terror and militancy purposes [16, 91, 208]. In terms of abuse, mobile communication networks and social media have been mostly studied as vectors for the diffusion of computer viruses and malware [204, 234]. On the other hand, the possibility that such communication channels can be exploited by criminals to organize and coordinate their illicit activities in the physical world has been recently found very real [278, 279]. The ability to detect criminal behavior across different communication media is of paramount importance to avoid abuse and fight crime. For this reason, computational tools and models have been recently proposed to study criminal behavior in online platforms [389–391], social media [374], and mobile phone networks [56, 152]. Usually, such models and techniques are limited to one or few specific use-cases. For example, we recently proposed a tool called *LogAnalysis* that allows an investigator to reconstruct and visualize networks from mobile phone call data [93].

A graph representation allows to overview the network structure, to identify the cliques, the groups, and the key players. The possibility of mapping the attributes of data and metrics of the network using visual properties of the nodes and edges makes this technique a powerful investigative tool. Often, however, visualization techniques become discouraging as a consequence of density and dimensions of the network. Some obstacles such as the overlap of nodes and the dense intersections of edges severely reduce the readability of the graph. In other words, there is a limit to the number of elements which can be distinctly viewed from the human eye. An influential theory about the improvement of the quality of network visualization has been suggested by Shneiderman in [343], where the so-called "Network Nirvana" is described. According to this theory, some demanding targets must be pursued: i) the visibility of each node; ii) the possibility of counting the degree of each node; iii) the possibility of following each edge from the source to the destination nodes and, iv) the possibility of identifying the clusters. Although it can be challenging, or even impossible, to satisfy all these conditions at the same time as the network grows in size and complexity, an effective network analysis strategy should try to optimize the visualization methods in order to incorporate these guidelines.

Here we present *LogViewer*, a next-generation Web-based criminal network analysis framework that yields advanced social network analysis functions, *de facto* extending *LogAnalysis* features to different types of networks, for example phone call networks and social graphs. *LogViewer* allows to study each network from three different angles: *(i)* static analysis, to investigate the role of nodes and edges, their centrality, and the emerging communities representing potential criminal rings; *(ii)* temporal analysis, to span across different temporal events and study the flow of information over time; finally, *(iii)* spatial analysis, embedding the network in a geographic space to determine physical closeness and locality effects on the network structure. *LogViewer* also allows to create multilayer spatio-temporal networks by merging different network types and to perform the above-mentioned different types of analysis on such a more complex network.

Our framework inherits different visualization layouts and algorithms from *LogAnalysis*: some of them are discussed in details in our previous work [93]. Here we first give an overview of the basic concepts borrowed by social network analysis and their meaning in criminal network analysis; this includes network centrality measure to identify roles in criminal networks, and community detection to unveil criminal gangs hidden within the network. After that, we present the new features provided by our criminal network analysis framework, especially *ad hoc* visualization methods that we devised keeping in mind the needs of law enforcement agencies, analysts and investigators. We illustrate these advanced criminal network analysis features by presenting examples or use cases inspired by real investigations, carried out by Italian law agencies, that benefited from the adoption of *LogViewer*.

7.2 RELATED WORK

In the latest thirty years academic research related to the application of social network analysis to intelligence and study of criminal organizations has constantly grown. One of the most important studies is due to Malcolm Sparrow [355], related

to the application of techniques of network analysis, and the study of network vulnerabilities, for intelligence scopes. He underlined three key aspects of so-called *criminal network analysis* (CNA): i) the importance of *social network analysis* (SNA) for the analysis of criminal data; ii) the potential of added intelligence from network analysis and, iii) the results deriving from the collaboration between the two sectors.

Sparrow defined four features peculiar of criminal networks (CNs): i) limited dimension — CNs are often composed of at most few thousand nodes; ii) information incompleteness — criminal or terrorist networks are unavoidably incomplete due to fragmentary available information and erroneous information; iii) undefined borders — it is difficult to determine all the relations of a node; and, iv) dynamics — new connections imply a constant evolution of the structure of the network.

Thanks to Sparrow's work, other authors tried to study criminal networks using the tools of SNA. For example, Baker and Faulkner [22] studied illegal networks in the field of electric plants and Klerks [220] focused on criminal organizations in The Netherlands. In 2001, Silke [347] and Brennan et al. [70] acknowledged a slow growth in the fight against terrorism, and examined the state of the art in the field of criminal network analysis.

Arquilla and Ronfeldt [16] summarize prior research by introducing the concept of Netwar and its applicability to terrorism. They illustrate the difference between social networks and CNs, demonstrating the great utility of network models to understand the nature of criminal organizations. Their work shed light on strategies, methods and systems of information flow for intelligence purpose. The framework proposed by Arquilla and Ronfeldt provided new ground for conceiving network analysis. Nevertheless, they received some criticism due to their theoretical approach. Before 2001-09-11, some criticism can be found in the work of Carley, Reminga and Kamneva [258], devoted to destabilizing initiatives of dynamic terrorist networks.

All these early studies somehow neglected the importance of network visualization, stressing aspects related more to statistical network characterization, or interpretation of individuals' roles rooted in social theory. However, in 2006, a popular work by Valdis Krebs [228] applied graph analysis in conjunction with network visualization theory to analyze the Al Qaeda cell responsible of the 2001-09-11 terrorist attacks in the USA. This work represents a starting point of a series of academic papers in which social network analysis methods become applied to a real-world cases, differently from previous work where mostly toy models and fictitious networks were used. Krebs' paper is one of the more cited papers in the field of application of social network analysis to Criminal Networks and it inspired further research in network visualization for the design and development of better SNA tools applications to support intelligence agencies in the fight against terror, and law enforcement agencies in their quest of fighting crime.

In criminology and research on terrorism, SNA has been proved a powerful tool to learn the structure of a criminal organization. It allows analysts to understand the structural relevance of single actors and the relations among members, when regarded as individuals or members of (one or more) subgroup(s). SNA defines the key concepts to characterize network structure and roles, such as centrality [159], node and edge betweenness [65, 159], and structural similarity [248]. The understanding of network structure derived from these concepts would not be possible otherwise

[377]. The above-mentioned structural properties are heavily employed to visually represent social and criminal networks as a support decision-making processes.

SNA provides key techniques including the possibility to detect clusters, identify the most important actors and their roles and unveil interactions through various graphical representation methodologies [393]. Some of these methods are explicitly designed to identify groups within the network, while others have been developed to show social positions of group members. The most common graphical layouts have historically been the node-link and the matrix representations [161].

Visualization has become increasingly important to gain information about the structure and the dynamics of social networks: since the introduction of sociograms, it appeared clear that a deep understanding of a social network was not achievable only through some statistical network characterization [377]. For all these reasons, a number of different challenges in network visualization have been proposed [336]. The study of network visualization focuses on the solution of the problems related to clarity and scalability of the methods of automatic representation. The development of a visualization system exploits various technologies and faces some fundamental aspects such as: i) the choice of the layout; ii) the exploration dynamics; and, iii) the interactivity modes introduced to reduce the visual complexity.

Recent studies tried to improve the exploration of networks by adding views, user interface techniques and modes of interaction more advanced than the conventional node-link and force-directed [165] layouts. For example, in *SocialAction* [311] users are able to classify and filter the nodes of the network according to the values of their statistical properties. In *MatrixExplorer* [193] the node-link layout is integrated with the matrix layout. Nonetheless, these visualization systems have not been explicitly developed with the aim of the exhaustive comprehension of all properties of the network. Users need to synthesize the results coming from some views and assemble metrics with the overall structure of the network.

Therefore, we believe that an efficient method to enhance the comprehension and the study of social networks, and in particular of criminal networks, is to provide a more explicit and effective node-link layout algorithm. This way, important insights could be obtained from a unique layout rather than from the synthesis derived from some different layouts.

We recently presented a framework, called *LogAnalysis* [93, 152], that incorporates various features of social network analysis tools, but explicitly designed to handle criminal networks reconstructed from phone call interactions. This framework allows to visualize and analyze the phone traffic of a criminal network by integrating the node-link layout representation together with the navigation techniques of zooming and focusing and contextualizing. The reduction of the visual complexity is obtained by using hierarchical clustering algorithms. In this Chapter we discuss three new network layout methods that have been recently introduced in *LogViewer*, namely fisheye, foci and geo-mapping, and we explain how these methods help investigators and law enforcement agents in their quest to fight crime.

It's worth noting that various tools to support network analysis exist. However, only few of them have been developed specifically for criminal network investigations. We mention, among others, commercial tools like COPLINK [101, 389], Analyst's

Notebook[1], Xanalysis Link Explorer[2] and Palantir Government[3]. Other prototypes described in academic papers include Sandbox [387] and POLESTAR [314]. Some of these tools show similar features to *LogViewer*, but, to the best of our knowledge, none of them yields the same effective and scalable network visualization with support to criminal networks reconstructed from phone call records.

7.3 ASPECTS OF STRUCTURAL ANALYSIS

A central node of a criminal network may play a key role by acting as a leader, issuing orders, providing regulations or by effectively assuring the flow of information through the various components of the CN. The removal of these central nodes may efficiently fragment the organization and interrupt the prosecution of a criminal activity.

Apart from studying the roles of various members, investigative officers must pay particular attention to subgroups or gangs each of which may be in charge of specific tasks. Members of the organization must interact and cooperate in order to accomplish their illicit activities. Therefore, the detection of subgroups whose members are tightly interrelated may increase the comprehension of the organization of the CN. Moreover, groups may interact according to certain schemes. For example, the members of a clan could frequently interact with the members of another and seldom with the remaining members of the network. The detection of interaction models and the relations among the subgroups highlights information particularly useful about the overall structure of the network.

A significant aspect of the analysis of criminal networks is that it requires, differently from other networks, the ability of integrating information deriving from other sources in order to precisely understand its structure, operation and flow of information. A typical process employed by an investigator is to start from one, or a few, known entities; after analyzing the associations these entities have with others, if any interesting association emerges, one may follow such a lead and keep expanding the associations until any significant link is uncovered between seemingly unrelated entities.

Mobile phone networks and online platforms are constantly used to perform or coordinate criminal activities [277, 390]. Phone networks can be used to connect individuals involved in criminal activities in real time, often during real-world criminal events, from simple robberies to terrorist attacks. Online platforms, instead, can be exploited to carry out illicit activities such as frauds, identity thefts or to access classified information.

The analysis of a criminal network is thus aimed at uncover the structural schemes of the organization, its operations and, even more importantly, the flow of communications among its members. In modern investigative techniques the analysis of phone records represents a first approach that precedes a more refined scrutiny covering financial transactions and interpersonal relations. For these reasons a structured approach is needed.

[1] ibm.com/software/products/analysts-notebook/
[2] http://www.xanalys.com/products/link-explorer/
[3] http://www.palantir.com/solutions/

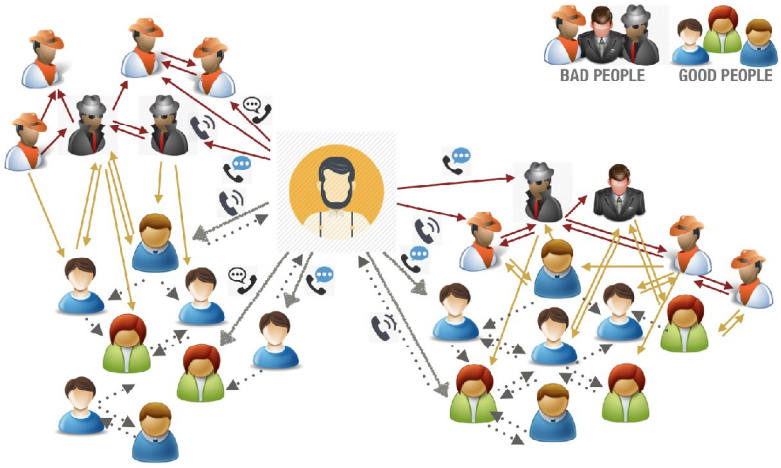


Figure 40: Phone calls network of a suspected. Investigators start from some known entities, analyze the associations they have with others and expand the associations until some significant link is uncovered. Here are highlighted personal interactions (gray arrows), links between criminal and personal connections of the suspect (yellow) and connections between members of the organization (in red).

Figure 40 shows a stylized representation of a criminal network reconstructed from phone call records. We show the flow of phone communications of an individual subject of investigation, and we highlight various kind of phone interactions among individuals belonging to that person's social circles, and those belonging to the same criminal organization the individual is part of.

In the following we discuss three techniques that allow to efficiently and scalably inspect criminal networks reconstructed from phone interactions.

7.4 *logviewer* PIPELINE

7.4.1 Architecture and workflow

LogViewer is a Web-based framework that allows advanced network analysis on criminal networks reconstructed from various data sources, including (mobile) phone data and online social network data. It supports spatio-temporal analysis and it extends, *de facto*, the horizon of possibilities provided by *LogAnalysis* [93].

This framework implements various techniques of network generation, statistical measurement, partitioning (or clustering), and visualization that rely on powerful open-sources tools; the list includes GraphML for data storage, Python network libraries for data import, normalization and network representation like NetworkX[4] and iGraph, the Stanford Network Analysis Project (SNAP) library[5] to efficiently

[4] http://networkx.github.com/
[5] http://snap.stanford.edu/

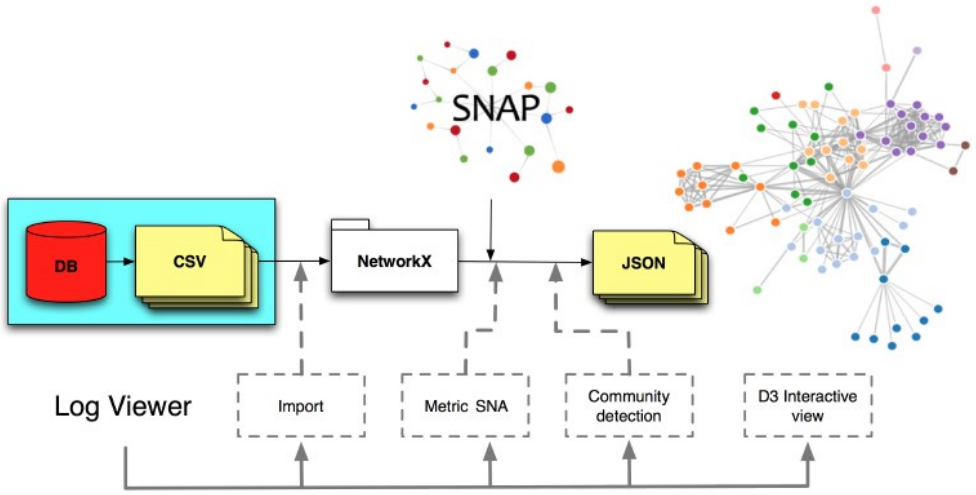


Figure 41: Architecture of LogViewer.

compute network statistics, the Louvain method for network clustering [52], and the Javascript D3.js[6] library for interactive network visualization and exploration.

The architecture of LogViewer is represented in Figure 41. In the following we illustrate the typical workflow to bootstrap a criminal investigation using LogViewer. Let us use the example of data representing a mobile phone call network —the analysis of other sources, such as social network data, follows straightforwardly.

During an investigation, the agency in charge of it will obtain, usually through court warrants, raw data from a Telecommunication Service Provider related to the phone call interactions of a (possibly large) set of suspects involved in a certain criminal activity. Such data are generally provided in different formats: LogViewer allows some degree of standardization, supporting different formats adopted by various European service providers, e.g., Vodafone, Orange, and others.

The analyst can import one (or more) datasets into LogViewer, which will take care of appropriately reconstruct a network representation of such data, where each node corresponds to a given entity (generally speaking, in the mobile phone cases, the framework assumes a 1-to-1 mapping from phone to person, but it also supports the assignment of multiple phone numbers to the same entity, whereas such information is provided). Interconnections among entities, representing phone calls, are imported as links of this network. Duration and frequency of the calls are encoded in the network representation by means of different weighting systems that can be adopted by the analysts. For example, the raw number of calls between a pair of entities, or the average or total duration, among others, are available metrics that can be used for this purpose. This yields the possibility of performing dynamic network representation and temporal analysis.

In addition, each phone interaction reports geo-referenced data about the location of the caller and the called nodes (e.g., extrapolated from the GPS sensors on the mobile device, or approximated by the telephone cell corresponding to the physical location of the individuals at the time of the call); such information is attached

[6] http://d3js.org/

Table 5: An example of the structure of a phone log file.

Field	Description
`IMEI`	IMEI code MS
`called`	called user
`calling`	calling user
`date/time start`	date/time start calling (GMT)
`date/time end`	date/time end calling (GMT)
`type`	sms, mms, voice, data etc.
`IMSI`	calling or called SIM card
`CGI`	Lat. long. BTS company

to each event, to allow for spatial analysis. Once the data import procedure is completed, static representation (and spatio-temporal representation when meta-data are available) becomes available through LogViewer's visualization interface.

In the following, let us provide a bit more details about the type of data commonly processed by LogViewer for criminal network analysis purposes.

7.4.2 Data and network representation

Mobile phone data

In the context of real-world investigations, mobile phone service providers, upon request by judiciary authorities, release data logs, normally in textual file format, with space or tab separation (CSV format). A typical log file contains, at least, the values shown in Table 5.

Similarly, information about owners of SIM cards, dealers of SIM cards and operations like activation, deactivation, number portability are provided by the service providers as additional material to ease and support the investigation activities. Log file formats produced by different companies are heterogeneous. *LogViewer*, first of all, parses these files and converts data into GraphML format. It is an XML valid and well-formed format, containing all nodes and weighted edges, each weight representing the various weighting strategies (e.g., the frequency of phone calls) used to represent the interactions between two connected nodes. GraphML has been adopted both because of its extensibility and ease of import from different SNA toolkits and graph drawing utilities.

Social graph data

Another rich source of information that is increasingly becoming adopted during criminal investigation is represented by Online Social Network data. Such types of datasets are provided by the Service Providers (like Facebook or Google) through court warrants to the law enforcement agency, similarly to mobile phone records.

Generally speaking, the datasets obtained by OSN service providers provide user meta-data related to the set of accounts of interest for the criminal investigation, including registration details (e.g., personal information, dates of account creation/deletion, etc.) along with the IP addresses corresponding to the devices used for connection (and/or the GPS coordinates of the mobile device, in case any

connection is performed in mobility). Logs include, among other data, the entire history of wall posts and comments, pictures and photographs, check-in events in specific physical locations, the chronology of incoming and outgoing friendship requests, the list of friends (on Facebook) or contacts (followers and followees on Twitter and similar platforms). Some platforms, like Facebook and Twitter, can provide detailed logs of personal interactions, such as chat or personal messages. Possibly, the same set of information is provided about any number of friends/contacts of the given individual target of the criminal investigation, if deemed relevant for the investigation by the judiciary authorities. Such data about the target's neighbors help enriching the amount of information available to LogViewer to perform its analysis.

LogViewer processes these datasets and extracts the information that can be put in form of network representation. For example, when reconstructing a social network, link weighting schemes represent the interactions (e.g., number of wall comments, frequency of chatting, etc.) between a pair of individuals. Although our framework does not yet provide advanced content analysis, such additional information is often adopted by the analysts by using external tools for traditional corpora analysis.

It's worth noting that, in the context of a criminal investigation, the analysts will study social network information with different lenses, say in respect to the perspective of phone interactions. This is clearly due to the fact that online friendship, say on Facebook, has a very different meaning if compared to phone interactions. On the other hand, the possibility of performing further analysis on textual content produced by personal interactions (e.g., chat) eases the analysis, say with respect to phone calls monitoring and analysis (which might not be possible whereas recordings are not readily available for investigation purposes or need additional warrants to be accessed).

7.4.3 Data normalization and cleaning

Data clean-up usually means the deletion of redundant edges and nodes. This step is very importante since datasets often contain redundant information, that crowds graph visualization and biases statistical measures. In these circumstances, redundant edges between the same two nodes are collapsed and a coefficient – i.e., a edge weight – is attached, which expresses the number of calls. Our tool normalizes data after reading and parsing log files whichever format they have been provided among the standard formats (i.e., *fixed width text*, *delimited*, CSV, and more) used by mobile service providers.

7.5 STATIC ANALYSIS OF CRIMINAL NETWORKS

LogViewer takes into account the concept of *centrality measure* to highlight actors that cover relevant roles inside the analyzed network [277]. Several notions of centrality have been proposed during the latest years in the context of Social Network Analysis.

There are two fundamentally different class of centrality measures in communication networks. The first class of measures evaluates the centrality of each node/edge in a network and is called point centrality measure. The second type is called graph

centrality measure because it assigns a centrality value to the whole network. These techniques are particularly suited to study phone traffic and criminal networks.

In detail, in *LogViewer* we adopted four point centrality measures (i.e., *degree*, *betweenness*, *closeness* and *eigenvector* centrality), to inspect the importance of each node of the network.

The set of measures provided in our tool is a selection of those provided by Social Network Analysis [377]. It could be not sufficient to solve any possible task in phone call network analysis. In fact, for particular assignments it could yet be necessary to use additional tools in support to *LogViewer* and in further evolutions we plan to incorporate new centrality measures.

For each centrality measure, the tool gives the possibility, to rank the nodes/edges of the network according to the chosen criterion. Moreover, *LogViewer* allows to select those nodes that are central, according to the specified ranking, highlighting them and putting into evidence their relationships, by exploiting the node-link layout techniques (discussed in the following). This approach makes it possible to focus the attention of the analysts on specific nodes of interest, putting into evidence their position and their role inside the network, with respect to the others.

They represent the centrality as an indicator of the activity of the nodes (degree centrality), of the control on other nodes (betweenness centrality), of the proximity to other nodes (closeness centrality) and of the influence of a node (eigenvector centrality).

7.6 VISUALIZATION TECHNIQUES

Typical network visualization tools rely on the popular force-directed layout [165]. This particular layout arrangement has the advantage of grouping users in clusters which can be identified according to the heightened connectivity.

To optimize the visualization, it is possible to interactively modify the parameters relative to the tension of the springs (edges). Nodes with low degree are associated a small tension and the elements are located in peripheral positions with respect to high degree nodes. Other parameters can be tuned, such as spring tension, gravitational force and viscosity. Our goal, in the following, is to suggest two methods to improve force-directed based layouts. As we will show, these techniques are especially well suited for criminal network analysis; however, they could potentially be generalized for broader usage in other domains of network analysis — for example, for applications in social and political sciences.

7.6.1 Focus and context based visualization

During an investigation, it is crucial to narrow down the analysis to the relevant suspects, to efficiently employ human and computational resources. Police officers typically draw some hypotheses about an individual suspect of being part of a criminal organization, or of being involved (or about to) in some crime; they concentrate the initial investigation on this individual, and on that person's social circles, as a ground to build the social network object of analysis. The main role of visual analysis lies in allowing the detection of unknown relations, on the base of the available

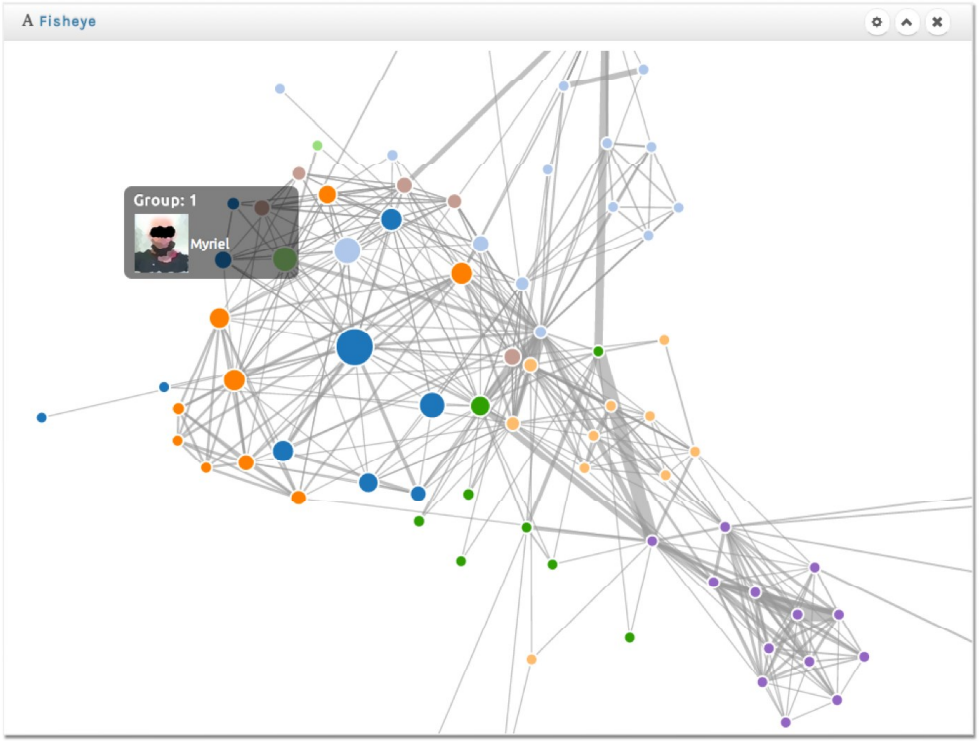


Figure 42: The picture shows a force-directed layout of a criminal network with a fisheye distortion.

limited information. A typical procedure starts from known entities, to analyze the relations with other subjects and continue to expand the network inspecting first the edges appearing the most between individuals apparently unrelated. During this procedure, only some nodes are relevant and it is important to focus on them rather than on the network as a whole.

Nevertheless, a spring embedded layout (including force-directed ones) does not provide any support to this kind of focus and analysis. In these situations, *focus and context* visualization techniques are needed in order to help a user to explore a specific part of a complex network. To this purpose, we here introduce the fisheye and the foci layouts.

7.6.2 Fisheye layout

The *fisheye view* is a particular focus and context visualization technique which has been applied to visualize self-organizing maps in the Web surfing [392]. It was first proposed by Furnas [166] and successively enriched by Brown et al. [333]. It is known as a visualization technique that introduces distortion in the displayed information.

The fisheye layout is a local linear enlargement technique that, without modifying the size of the visualization canvas, allows to enhance the region surrounding the

Figure 43: Matrix layout and clustering.

focus, while compressing the remote neighboring regions. The overall structure of the network is nevertheless maintained.

An example of application of this technique is show in Figure 42. The picture shows a moderately small criminal network reconstructed from phone call interactions of about 75 individuals. The layout on the left panel is obtained by using a force-directed method implemented in our framework, *LogAnalysis*. The analyst can inspect the nodes of the network, which contains known criminals, suspects, and their social circles. When the focus is applied on a given node, the visualization transitions to the fisheye layout (see the right panel). A tool-tip with additional information about the node appears when the node is selected — it shows the phone number, personal details, address, photo, etc. The layout causes edges among remote nodes to experience stronger distortions than local nodes. The upside of the presented method is the possibility to achieve the three recommendations of Network Nirvana [336] when focusing on a given node: all the nodes' neighbors are clearly visible, the node degree is easily countable, and the edges incident on that node can be identified and followed.

Note that fisheye and force-directed layouts can be used in a complementary way. By combining the two methods, our framework efficiently yields focus and context views.

Matrix layout

A network can be represented by using an adjacency matrix in which each cell ij represents the edge existing between the vertex i and the vertex j.

In our case, the vertices represent the phone numbers of the users (the caller and the called), and the edges represent their contacts.

The natural visualization technique associated to this two-dimensional representation of the graph is the matrix layout. Nevertheless, the efficiency of a matrix diagram strongly depends upon the order of rows and columns: if the nodes that are

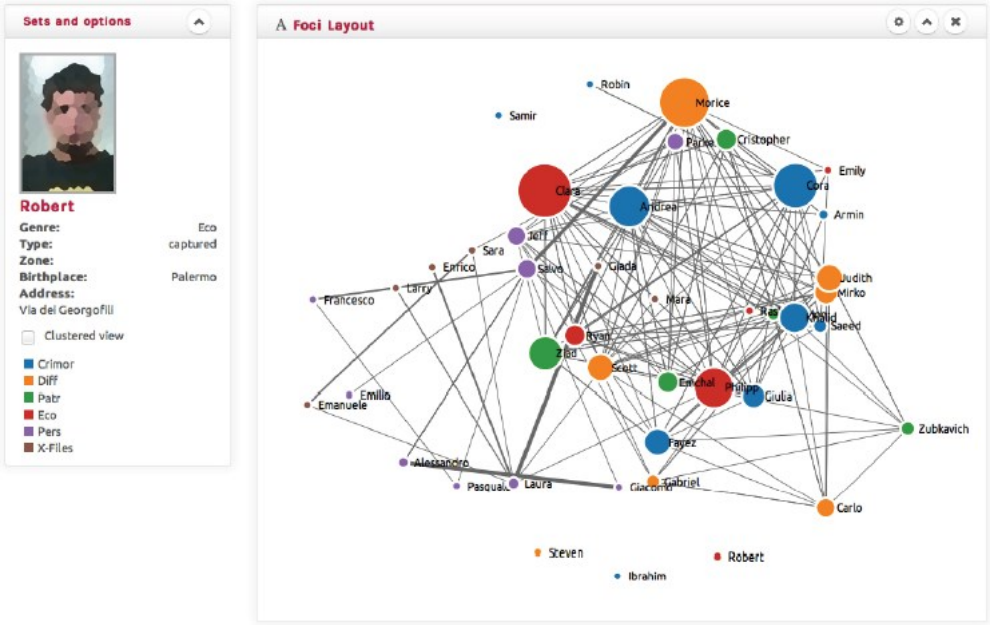


Figure 44: Foci layout.

connected are placed in order, then clusters and connections among communities can be easily identified.

As shown in Figure 43, matrix cells can be coded to show additional information: in this case different colors represent different clusters.

On the contrary of node-link diagrams, matrix layout makes not easily identifiable the paths connecting the vertices. On the other hand, when dealing with highly connected networks, the node-link layout rapidly becomes unreadable as a consequence of the large superposition of nodes and edges.

7.6.3 Foci layout

The *foci layout* implements three network visualization models: force-directed, semantic and clustered layouts. The latter is based on the Louvain community detection algorithm [52, 126]. Future implementations will explore other methods [125, 127]. Our model supports multilayer analysis of the network through interactive transitions from the force-directed layout, with a single gravitational center, to the clustered one with more force centers placed in predetermined distinct areas. This layout allows to analyze the network on various layering levels depending on specified node attributes. Figure 44 shows the phone traffic network of some clans the previous criminal network, in which the color of the nodes denotes the type of crime committed by the members.

In this example, the clustering truthfully reflects the known territorial division among the groups belonging to the organization. In Figure 44 the focus is on a specific node. Using this layout it is possible to contextually analyze the community structure, the type of committed crime in respect to the members of the clan, and

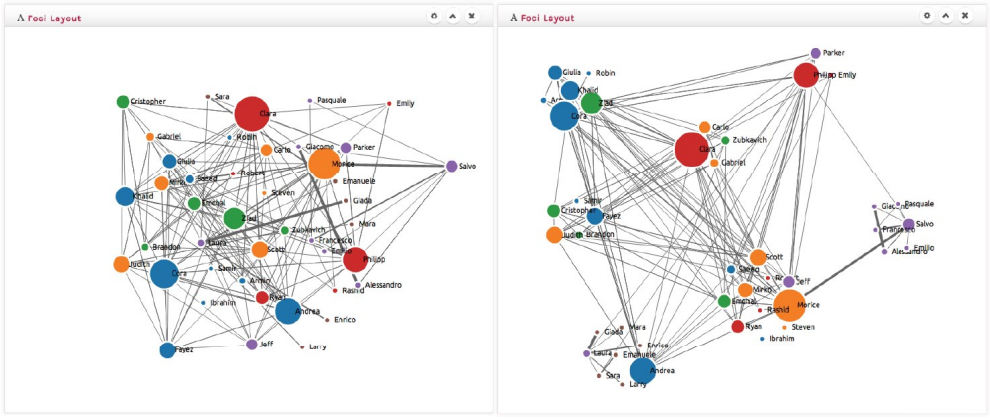


Figure 45: Multi-foci layout.

the direct relations of each single individual. This layout integrates also the forth Network Nirvana recommendation, namely the possibility to identify clusters and to highlight the community structure.

7.7 SPATIO-TEMPORAL CRIMINAL NETWORKS ANALYSIS

7.7.1 Temporal network analysis

Phone call records and online social network data come with temporal information attached to many events. For example, the time and duration of a call or a chat session, or the timestamp associated with the creation of a given phone contract or account on a social platform, are common meta-data available for investigation.

LogViewer provides extensive support to encode and exploit temporal information, when available, to perform network dynamic and temporal pattern analysis. One example is provided in Figure 47, where we display LogViewer's interface reporting aggregate temporal statistics related to the activity ongoing on a mobile phone network under investigation.

In this example three types of information are displayed: on top, a time series reports the volume of calls per day during the investigation period. It's possible to see how heterogeneous is this traffic, with a strong attenuation toward the end of the observation period, after a spike coinciding with an actual criminal event in the real world. The analyst has the possibility of zooming in the time series, to select different sub-intervals, to display different types of statistics over time (e.g., total volume of calls, or total duration, etc.) and to filter according to different types of constraints (e.g., showing only the information related to a subset of users, for example a particular cluster). The applied filters are also reported underneath, for example as pie charts that show specific statistics per day of the week, per type of event (e.g., phone calls, texts, video calls, etc.) and per geographic area. Better resolution is provided by histograms that bin the given statistics, say number of calls, per hour of the day.

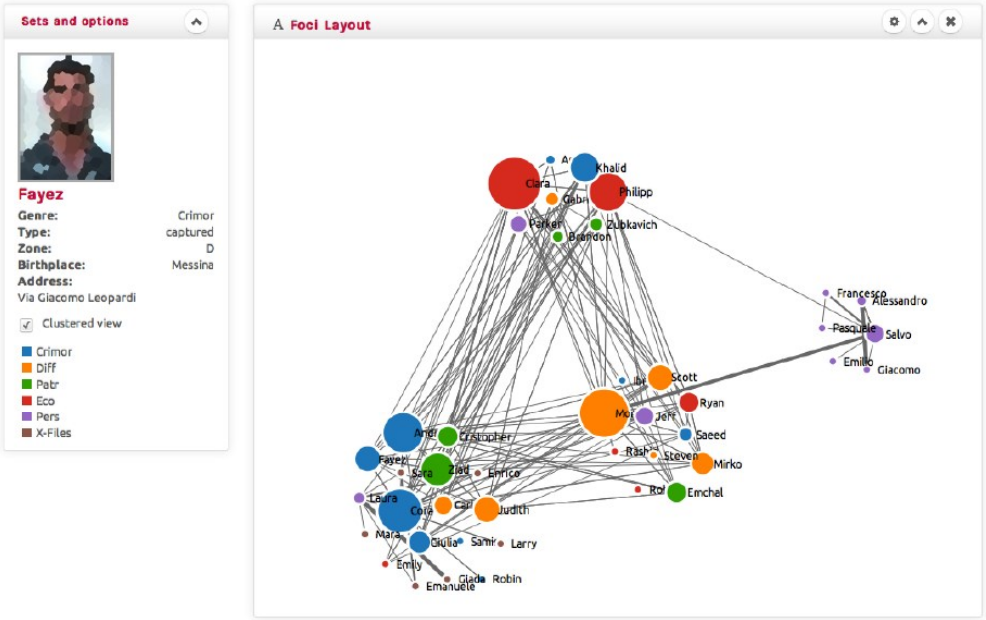


Figure 46: Filtered and clustered multi-foci layout.

Another example is provided in Figure 48 that shows a *stream graph* adopted to visualize a sequence of temporal events on an aggregate basis. Stream graphs show the potential of tools that provide dynamics and interactive data exploration. The x axis of the stream graph represents time, whereas the y axis reports an arbitrary metric, say for the example in Figure 48 the total volume of phone interactions, subdivided by type (e.g., calls, texts, Internet sessions, etc.), each displayed using a different color. The stream is proportional to the number of events of each type per unit of time (one bin here is one hour). LogViewer also implements stacked graphs. Stream and stacked graphs represent especially helpful tools when the analysts want to visually compare extensive metrics that depend on the volume of events in a predetermined period.

By selecting the various temporal analysis tools and filters available, the analyst can dissect the dataset under analysis to obtain granular temporal information or to highlight and let emerge specific patterns of interactions among particular groups of individuals. This, in conjunction with spatial filters that are discussed in the next section, yields the ability to determine when (and where) information flows, and to identify the peeks and lows of interaction activity among the members of a criminal organization, to narrow down investigations towards specific periods of interest (that might concur with events in the real world).

7.7.2 Network geo-mapping

It is possible to extend the phone traffic analysis to include the phone logs recorded by the BTS (Base Transceiver Station), in which the GPS coordinates of the cell are reported. All base stations are provided with directional antennas and

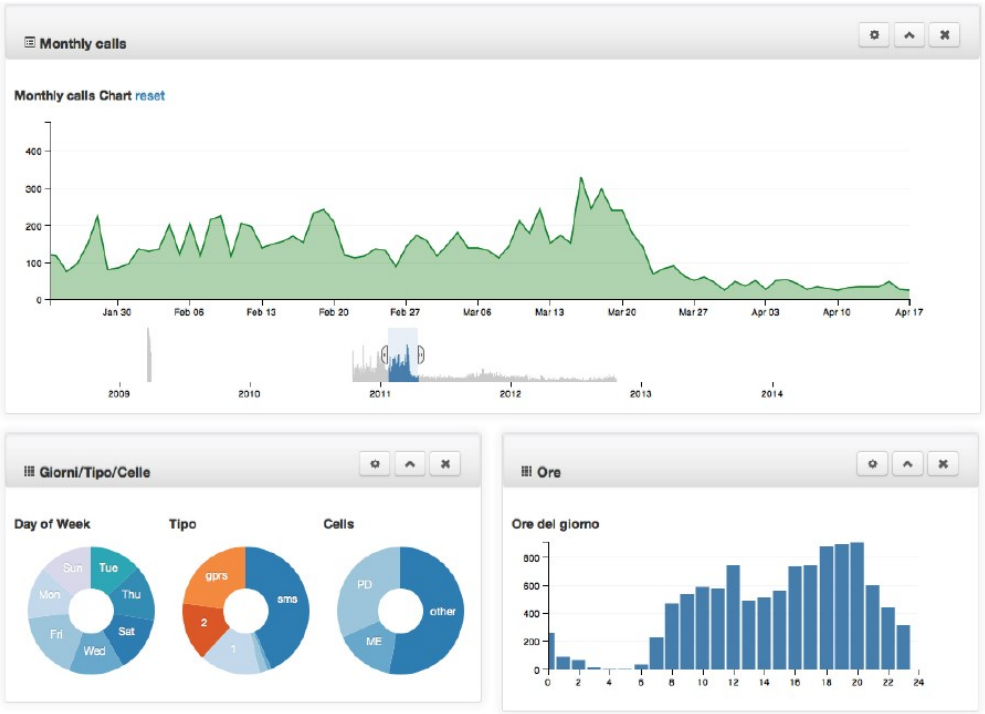


Figure 47: Temporal analysis of a criminal network.

each cell has two or more sectors. For each cell it is known the azimuth (direction) corresponding to the central axis of each sector, together with the width of the beam of each antenna, which determines the coverage angle of the sector. These data do not allow to localize the geo-referenced position of the phones involved in the events recorded in the logs. Nevertheless, it is possible, within a certain approximation, to localize the users falling within the coverage area.

Zang et al. [398] described a technique based on Bayesian inference to localize mobile phones using additional information, such as the round-trip-time of data transmission packets and the measure of SINR (Signal to Interference plus Noise Ratio). The parameters obtained experimentally have been compared with the records of phone calls and the corresponding GPS entries to ascertain their distribution. This localization technique produces satisfactory results with a reduction of the error amounting to a 20% with respect to the *blind approach*. Traag et al. in [366] used Bayesian inference to deduce, starting from phone traffic data, profiles about the places and the proximity of a given social event.

Our framework provides network geo-mapping by using this type of techniques to infer the spatial origin of each call. Here we describe the network geo-mapping visualization method adopted in *LogAnalysis*. This layout allows to simultaneously carry out spatial and temporal relational analysis of phone call logs. It places nodes of the network on a map, in correspondence of the coordinates of the cells linked during the events recorded in the logs. Nodes are connected by links related to displacements. Contacts falling within the sectors of a given zone are represented with nodes of the same color. Information about displacements, routines and areas

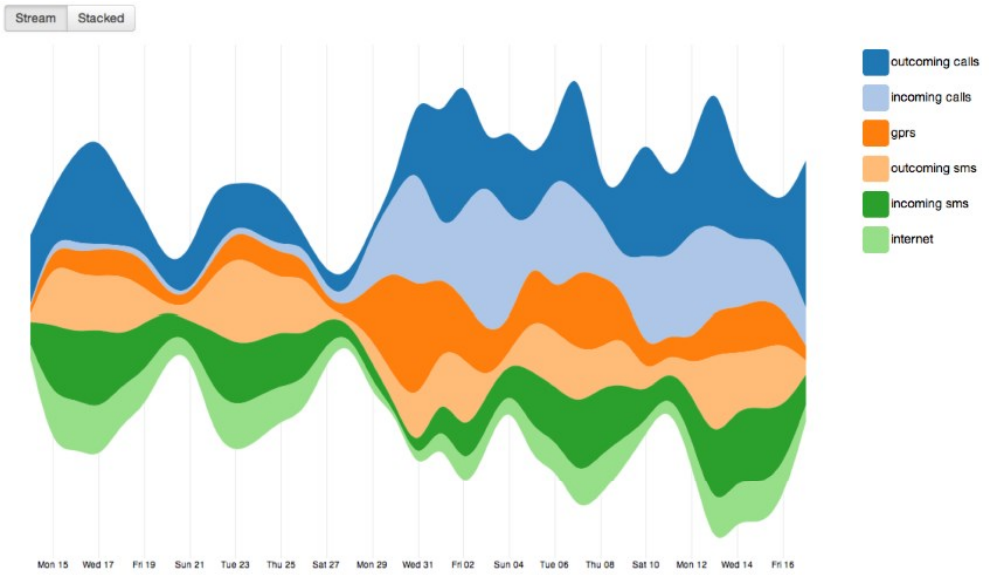


Figure 48: Stream layout of temporal dynamics in a criminal network.

of interest for the investigation are displayed. The adoption of network geo-mapping has proved extremely useful during real investigations. Figure 49 shows, as an example, a case study in which larger nodes identify zones in which, in the time period of the investigation, a high number of contacts has been recorded among some members of the CN. Unsurprisingly, the inspection by police officers of such high-profile locations provided crucial insights on the investigation. Unfolding the temporal evolution of the geo-mapped phone traffic network also allows to reproduce individuals' movements and communication dynamics during specific criminal events embedded in space and time, like robberies, assaults, or homicides.

7.8 CONCLUSIONS

Criminal network analysis benefits from visualization methods used to support the investigations, especially when dealing with networks reconstructed from heterogeneous data sources, characterized by increasing size and complexity. In this paper we integrated the spring embedded algorithm with the fisheye and foci layouts to allow interactive exploration of criminal networks through our network analysis framework. The combination of these techniques proved helpful to support investigators in the extraction of useful information and critical insights, to identify key members in terrorist groups, and to discover specific paths of interaction among members of criminal organizations. Experimental results show that the combination of force-directed layouts, distortion techniques and multi-force systems yield better performance in terms of both efficiency and efficacy.

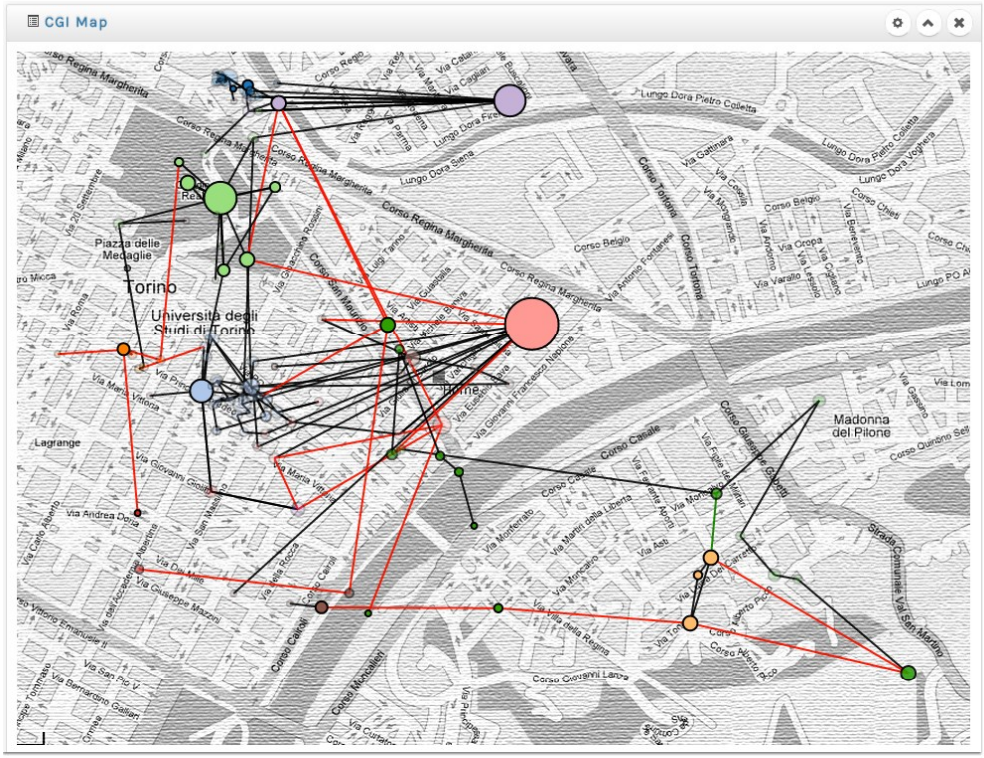


Figure 49: Geo-mapping layout.

8 RESILIENCE OF MAFIA SYNDICATES

In this Chapter we present the results of our study of Sicilian Mafia organizations using *social network analysis*. The study investigates the network structure of a Mafia syndicate, describing its evolution and highlighting its plasticity to membership-targeting interventions and its resilience to disruption caused by police operations. We analyze two different datasets dealing with Mafia gangs that were built by examining different digital trails and judicial documents that span a period of ten years. The first dataset includes the phone contacts among suspected individuals, while the second captures the relationships among individuals actively involved in various criminal offenses. Our report illustrates the limits of traditional investigative methods like wiretapping. Criminals high up in the organization hierarchy do not occupy the most central positions in the criminal network, and oftentimes do not appear in the reconstructed criminal network at all. However, we also suggest possible strategies of intervention. We show that, although criminal networks (i.e., the network encoding mobsters and crime relationships) are extremely resilient to different kinds of attacks, contact networks (i.e., the network reporting suspects and reciprocated phone calls) are much more vulnerable, and their analysis can yield extremely valuable insights.

8.1 INTRODUCTION

The Sicilian Mafia (also known as *Cosa Nostra*) is a criminal organization which originated in Sicily, and, after decades of immigration waves, has now spread worldwide [256, 259, 354].

Police investigations revealed that the Mafia is a loose confederation of smaller syndicates (called "cosche," "clans," or "families") such that each syndicate takes control of a specific territory (usually a town or a part of it) by organizing and overseeing illegal activities. Members of a Mafia syndicate can be both mobsters and associates (i.e., people like drug-dealers, hitmen, or even corrupted politicians who are not part of the syndicate but act as collaborators or bystanders of its illicit activities). Mafia syndicates show a strong hierarchical organization [256]: on top of the organization there is a "boss," who is aided by an "underboss" and by various "lieutenants" who head branches of the Mafia syndicate. The boss also commands a crew of "soldiers" (often known as *picciotti*) who commit acts of violence that include intimidation, threats and murders.

Due to its normative structure, as well as strong ties with finance, entrepreneurs and politicians, the Mafia has risen to prominence as a worldwide criminal organization controlling illegal activities, including the trade of drugs, money laundering, and military weapon trafficking [98].

Understanding the structure of Mafia syndicates, unveiling the functional role of each of the members, and quantifying the ability of a syndicate to react to the detention of its members, are all crucial steps to effectively fight and dismantle these syndicates. In recent years, various researchers [152, 256, 275] illustrated the benefits of using *social network analysis* [291] to study the structure of criminal organizations.

The adoption of methods from social network analysis in the study of criminal organizations has strong theoretical and practical motivations. Studies from sociological literature (known as *social facilitation models* [256]) point out that the membership of an individual in a crime gang enormously amplifies her/his tendency to criminal behavior [363]. Destroying the network structure associated with a criminal organization is central to preventing individuals from committing crimes and also results in lower delinquency rates.

The first step in analyzing Mafia syndicates by means of social network analysis tools is to collect a sufficiently large data sample describing the various units composing the syndicate and their operations. Interactions among mobsters materialize under various forms. For instance, two mobsters can be tied if they committed the same crime together or if they have been seen together in the same setting. A powerful and well-known investigative method is *wiretapping*, i.e., the procedure of recording information flow among suspected criminals which has been sent using any type of electronic media, like phone calls (from both land lines and mobiles), emails, SMS messages and private communications over social media platforms. Wiretapping has proven effective for preventing and solving many crimes, such as terrorism, drug trafficking, kidnapping and political corruption.

Wiretapping has been extensively employed in Mafia-related investigations, but, if used alone, it may fail to reliably capture the structure of a Mafia syndicate. Newspapers, for instance, report that Mafia bosses often reveal their whereabouts to just few gang members and, in many cases, they issue orders and communications through handwritten notes known as *pizzini*.[1]

A promising investigation strategy requires supplementing information collected by wiretapping with data generated by other methods of investigation—like video surveillance, use of informants and under-cover agents, interviews of subjects, analysis of bank transactions, and so on. By gluing together these pieces of information, investigators can capture a more detailed picture of the structure of a Mafia syndicate. Unfortunately, the type of information cited above is the outcome of a long, expensive and often dangerous investigation process which likely spans years, or, in certain cases, even decades.

After examining several types of judicial documents spanning a ten- year period (including judicial documents, verdicts, depositions, interrogations, etc.) we built two datasets of information about Mafia gangs operating in the north of Sicily (Italy).

Law enforcement collected data about phone calls among suspected individuals; this dataset allowed us to build a network called *contact network* N_{con} in which each individual was associated with a vertex, and an edge between two vertices denoted the existence of at least one reciprocated phone call between the two individuals. The network N_{con} contains 1716 vertices and 8481 edges.

Further investigation allowed us to identify *crime relationships*. We say that a crime relationship exists between two individuals if they took part in the same criminal

[1] See http://news.bbc.co.uk/2/hi/europe/4899512.stm

offense or if they have been seen together in the same setting. Criminal relationships were then mapped on a second network called *criminal network* N_{cri}. The network N_{cri} contained only 104 vertices and 2596 edges; all but *six* individuals in the criminal network were also present in the contact network. This means that the original dataset contained almost all mobsters, but there were mobsters who were part of the Mafia syndicate but never used mobile or land lines to communicate.

The availability of these datasets occurred under a collaboration framework with law enforcement. This offered us the unprecedented opportunity to understand the actual structure of a Mafia syndicate, and to quantify how syndicates are able to react to police operations leading to the detention of some of their members. In the first stage of our research, we studied the structural properties of N_{cri} and N_{con}. Our primary goal was to understand whether meaningful differences arise between the structural features of the two networks.

Subsequently, we investigated and compared the robustness of N_{con} and N_{cri}. We simulated a police operation leading to the arrest of a fraction f of individuals from the two networks, and we studied how these perturbations impacted the structure of both N_{con} and N_{cri}. Individuals were selected either randomly or on the basis of their centrality in the network. To this end, we used three different centrality metrics, namely *degree centrality* (*DC*), *betweenness centrality* (*BC*), and *closeness centrality* (*CC*). We considered two types of operations, namely: *(i) parallel attack*, i.e., we assumed that a fraction f of individuals were *simultaneously* deleted from the network along with their connections, and *(ii) sequential attack*. i.e., we were supposed to iteratively neutralize individuals along with their connections from the network until a fraction f of individuals had been neutralized. To measure the effectiveness of each operation, we computed two parameters—the size of the strongly connected component (SCC) of each network and the average path length (APL), defined as the mean of shortest path lengths in the network.

The main findings of our analysis can be summarized as follows:

1. The degree distribution in N_{con} followed a power law $k^{-\alpha}$ with $\alpha = 2.5$. By contrast, the degree distribution in N_{cri} was almost uniform, and about 76.92% of N_{cri} affiliates had a degree ranging between 15 and 85. With the help of police officers, we observed that leaders in the Mafia syndicate were the individuals in N_{cri} who exhibited the lowest degrees. Therefore, the top elements in a Mafia syndicate often do not occupy the most central positions in the criminal network.

2. In both N_{cri} and N_{con} we empirically observed that if two vertices v and w were both connected to a third vertex i, there was a high chance that v and w were connected too. We used a popular metric from network science—the *average clustering coefficient* of a vertex i—to measure the amount of triangles in a network containing i (see Section 8.5.1 for a formal definition). We computed the average clustering coefficient ACC_i of each vertex i as a function of its degree, and we found that ACC_i was always bigger than 0.6 (that is, more than five times the value measured on social networks like Facebook [96, 369]). The large values of ACC_i are likely to depend on the policies adopted by Mafia gangs to recruit new members. According to the Mafia's normative structure, an individual—say i—can not spontaneously join a gang, but rather she/he

has to be introduced by an intermediary—say v—who is already affiliated with the gang. After entering the gang, i is likely to start interacting with persons who are familiar to v, thus forming triangles. A large number of triangles in a network explain large values of the average custering coefficient.

3. In the case of a parallel police operations, we observe that targeted attacks are able to quickly destroy the strongly connected component (SCC) of N_{con}. In particular, *DC* has the most disruptive effect on SCC. In contrast, N_{cri} showed an exceptional degree of robustness independent of the adopted centrality index.

4. In the case of sequential police operations, the *CC* has the most disruptive effect on SCC, and this happens both in N_{con} and in N_{cri}. *CC* is still the best option if the goal is to increase APL in N_{con}; by contrast, *DC* yields the largest increase in APL if applied on N_{cri}.

The organization of this paper is as follows: in Section 8.2 we discuss the related literature; in Section 8.3 we provide some basic definitions which will be largely used throughout the paper; Section 8.4 details the steps we followed to build N_{con} and N_{cri}; Section 8.5 describes the main structural features of both the contact and the criminal network; Section 8.6 is devoted to investigating the resilience of the contact and criminal networks under parallel and sequential attacks; and finally, in Section 5.8 we draw our conclusions and illustrate our future research plans.

8.2 RELATED LITERATURE

In this section we review the scientific literature related to our approach. We begin by discussing how social network analysis techniques have been applied to study Mafia-related organizations (Section 8.2.1). We then illustrate approaches that concentrate on how the power of a criminal organization depends on the social relationships among its members (Section 8.2.2). One of the major contributions of our study, in fact, consists of exploring the effects associated with the dismissal of one (or more) members of a criminal gang.

8.2.1 Social Network Analysis and Mafia Syndicates

One of the early reports on the structure of Mafia syndicates dates back to 1876. Its creation is attributed to the Italian deputy Leopoldo Franchetti [354] who depicted the Mafia as a criminal organization deeply rooted in Sicilian society. Franchetti argued that the Mafia was impossible to destroy unless there was a deep change in the Sicilian social institutions.

Such a study has deeply influenced prosecuting magistrates, politicians, criminologists and sociologists committed to fighting the Mafia. Mafia syndicates are organized according to rigid normative structures—the most popular code of conduct perhaps being the *Mafia Decalogue*. According to that Decalogue, mobsters must respect each other. For instance, it is forbidden to appropriate money if it belongs to other members of the same syndicate or to other "families." Ties among mobsters belonging to the same syndicate are very strong: in some cases the members are

related by blood and, in any case, the syndicate members come first—even before their own birth family.

Because of the rich and strong web of relationships among mobsters, the analysis of the social structure of a Mafia syndicate is of great scientific interest, and it well explains why social network analysis methods have been extensively applied to the study of those syndicates.

For instance, Morselli [275] studied the connections within a New York-based family (the Gambino family). The study focused on the career of one of its members, Saul Gravano. One of the main findings is that Gravano's ability to build and extend over time his personal network of contacts was a key factor to climbing the ranks of the Gambino family organization. Natarajan [286] studied a dataset consisting of 2408 wiretap conversations gathered during the prosecution of a heroin-dealing Mafia syndicate in New York. Starting from available data, the author built a network of phone calls, which revealed a group of 294 individuals forming the core of the criminal organization. Natarajan showed that most of the group members had very limited contacts with others in the group.

Other relevant studies are reported by Sarnecki [334] (who applied social network analysis to study co-offending behaviors among Swedish teenagers) and by McGloin [259] (who analyzed the network structure of street gangs in Newark, New Jersey).

Social network analysis is not only a tool to describe the structure and functioning of criminal organizations, but it has been largely employed in the construction of crime prevention systems [100]. For instance, Xu and Chen [389] jointly applied social network analysis with hierarchical clustering algorithms. The proposed approach worked in two stages: first, a criminal network was partitioned into subgroups by means of a clustering algorithm. Then, block modeling techniques were used to extract interaction patterns between these subgroups. A further application of social network analysis to crime detection reported by Drezewski *et al.* focused on money laundering [135].

Social network analysis tools were finally employed to identify leaders within a criminal organization. For instance, Mastrobuoni and Patacchini [256] used a dataset containing criminal profiles of 800 Mafia members active in the United States from the 1950s to 1960s to investigate the structure of criminal ties among mobsters. Various features were considered to predict the criminal rank of mobsters, including family relationships and legal and illegal activities.

In our previous work we focused on the joint application of social network analysis tools and advanced data visualization techniques [93, 152]. We described a software system able to extract criminal organizations from a network of recorded mobile phone calls, and we combined statistical network analysis, community detection, and visual exploration to unveil the structure of criminal networks hidden in communication data.

This paper introduces many novelties with respect to the approaches cited above. In fact, our analysis focuses on two datasets which capture interactions and events occurring during the same time interval and referring to the same geographical area. The first dataset records phone calls, while the second one is about crime relationships. And, as will become clear in the following, the networks extracted from each dataset show deep differences from a structural viewpoint. Our work highlights the limits of wiretapping as an investigative method to fight Mafia gangs,

and it shows that the most prominent criminals do not occupy the most central positions in the criminal network. The procedure we followed to build the datasets in this paper essentially relies on the analysis of judicial documents like verdicts or depositions. An approach to collecting crime-related data—but orthogonal to ours—is described by Furtado *et al.* [167]. In that paper the authors describe *WikiCrime*, a Web application that enables its users to directly report crimes occurring in a specific geographical area and temporal window, or to search for a specific crime recorded in the past. One of the core features of WikiCrime is its ability to give more transparency and diffusion to criminal information, and with this, to prevent crimes. As claimed by the authors, WikiCrime is also effective in reducing *under-reporting*, i.e., the fact that some crimes are not reported to law enforcement authorities. WikiCrime integrates a reputation module to verify the credibility of generated information. On the one hand, the collaboration of large masses of users enables us to quickly and cheaply collect vast amounts of data. On the other, the source of available data is often unknown, and therefore it is hard to determine if the information is credible and accurate.

Concluding, we point the interested reader to an informative and comprehensive review recently compiled by D'Orsogna and Perc [134] that summarizes current efforts in computational modeling of crime from a statistical physics perspective.

8.2.2 The Power of Criminal Organizations and Social Interactions

Many researchers have tried to determine the best policies for fighting (and hopefully dismantling) a criminal organization. Most of these studies highlight the importance of social relationships as a multiplier of the aptitude of single individuals to commit crimes. For instance, one of the early contributions was made by Sah [330], who proposed a crime model based on social interactions. The key point of that study is that the severity of punishment perceived by an individual as a consequence of illicit behavior depends on her/his social setting. As a result some individuals (under the influence of their peers, the social environment they live in, and the institutions with whom they interact) may feel that the punishment is not severe, and this is more conducive to criminal actions. An interesting study by Gleaser *et al.* [174] classified individuals in a criminal network as *conformist* (if they simply imitate the behaviors of their peers) and *non-conformist* (if they decide on their own to commit/not commit crimes). The studies reported above highlight that the structure of social ties among members of the same community, as well as the culture individuals have been exposed to, may have a crucial impact on their tendency to commit crimes. The main question derived from these studies is how to perturb a criminal network to reduce the potential of its members to commit crime.

Ballester and collaborators [24] suggested a *key-player* policy that aims to remove the criminal who is most responsible for the level of criminality in a gang. They propose that such a policy is more effective than traditional punishment policies. Borgatti [62] defined a different approach for key-players based on qualitative features of vertices rather than a mere quantitative evaluation of their centralities. Unfortunately, such an approach requires access to further information that might not be readily available to the investigators, or might be dangerous to collect in the context of a criminal investigation.

Liu et al. [247] analyzed networks of delinquent adolescents in the United States with the goal of detecting the criminals whose removal would generate the highest possible reduction in the aggregate crime level. They found that, in delinquent adolescent networks, key players have less educated parents, are more likely to be male, are less attached to religion, and feel more socially excluded. In this paper we consider a similar problem; i.e., we focus on finding what police strategy has the most disruptive effect on the structure of a Mafia gang. Our analysis highlights that in Mafia syndicates, the key players are not the best connected mobsters. In fact, the criminals occupying leadership roles in Mafia gangs often prefer not to use phones to communicate. In addition, judicial documents at our disposal revealed that bosses were uniformly spread in the criminal network, and this encodes the fact that bosses often belong to different family units. As a consequence, the task of arresting bosses is extremely difficult and dangerous, and the criminal network (i.e., the network encoding mobsters and crime relationships) is extremely resilient to different kinds of attacks. In contrast, the contact network (i.e., the network encoding suspected individuals, reciprocated phone calls, etc.) rests on a few highly central individuals. The main implication is that the network of phone contacts, for example, can be easily divided into small clusters by selectively deleting some of its vertices.

8.3 BACKGROUND

In this section we briefly introduce centrality scores (Section 8.3.1), and then illustrate the concept of network robustness (Section 8.3.2).

We define a network $N = \langle V, E \rangle$ as a pair in which V is the set of vertices and E is the set of edges. The symbol $\langle i, j \rangle$ denotes an edge in E between vertices i and j.

A network N can be represented through its *adjacency matrix* $\mathbf{A}$, which is defined as follows: $\mathbf{A}_{ij} = 1$ if (and only if) there is an edge going from vertex i to vertex j; otherwise it is 0. In the following, we suppose that $\mathbf{A}$ is *symmetric*, i.e., if an edge $\langle i, j \rangle$ belongs to E, then the edge $\langle j, i \rangle$ belongs to E too.

8.3.1 Centrality in Networks

The centrality of an individual, represented by a vertex i, in a network N, is a measure of the importance of i in N. A large number of centrality indices have been proposed in the literature (see [291] for an excellent review). In this study we adopted three recognizable and widely used ones: *(i) degree centrality* (*DC*), *(ii) betweenness centrality* (*BC*), and *(iii) closeness centrality* (*CC*).

We chose these indices because they have a clear geometrical interpretation, and they convey easy to understand concepts of node importance. These indices rely on complementary philosophies: *DC* is based only on the local connectivity of a vertex, and it only needs to know the number of neighbors in a vertex. More formally, given a vertex i, its $DC(i)$ is defined as the number of edges incident onto i. The *BC* and *CC* indices are based on the concept of *shortest path* (also known as *geodesic path*) in a network. Given an unweighted and undirected network and a pair of vertices i and j, the shortest path connecting i and j is the path consisting of the fewest number of edges. According to the literature [159], shortest paths are preferential pathways to

convey and spread messages in a broad range of networks like biological or social networks.

Some authors [9, 124, 125, 287] argue that the assumption that information travels along geodesic paths may not hold true in real scenarios; for instance, in the case of large online social networks like Facebook, users are agnostic about the whole network topology, and therefore they are not able to find shortest paths and use them to convey messages. In addition, the computation of shortest paths is computationally infeasible, even on moderately large networks.

In the case of criminal networks, however, we guess that geodesic paths are preferred to randomly generated paths (i.e., random walks). Criminal networks are much smaller than other types of social networks, and we can afford to compute geodesic paths. In addition, to ensure secrecy in the transmission of information, shortest paths are preferred to longer ones. It is known that criminals systematically try to expose sensitive information to a minimal number of trusted others.

On the basis of these considerations, we claim that the importance of a vertex i depends on the fraction of shortest paths passing through i, because this means that i is able to intercept a relevant portion of the information flowing through the network. This intuition naturally leads to introducing the betweenness centrality $BC(i)$ of a vertex i which can be formally defined as follows: let i, u and w be any three distinct vertices in a network, and let σ_{uw} be the number of shortest paths from u to w; finally, let $\sigma_{uw}(i)$ be the number of the shortest paths from u to w passing through i. We define $BC(i)$ as:

$$BC(i) = \sum_{i \neq u \neq w \in V} \frac{\sigma_{uw}(i)}{\sigma_{uw}} \tag{59}$$

Alternatively, we may classify i as important if its "distance" from other vertices in the network is small because this certifies the ability of i to communicate with other vertices and contribute to the information spreading. There are, of course, various possible definitions of distance between network vertices. The simplest one perhaps consists of measuring the distance between two vertices i and j as the length $\mathtt{SP}(i,j)$ of the shortest path connecting them. Bearing in mind such a notion of distance, we define the closeness centrality $CC(i)$ of i as the reciprocal of the sum of all geodesic distances from i to all other vertices in the network [40]:

$$CC(i) = \frac{1}{\sum_{u \in V} \mathtt{SP}(u,i)} \tag{60}$$

Some experiments devoted to studing collaboration in social groups show that individuals perceived as leaders are those who generally feature high closeness values [40].

8.3.2 Network Robustness

The study of network robustness (i.e., the ability of a network to react to the failure of some of its components [11, 291]) is strongly linked to studies about the reliability of many biological and artificial systems.

A system S, in fact, can often be modeled as a network $N = \langle V, E \rangle$ such that each vertex in V identifies one of the components of S while edges describe interactions

among components. The system S is said to be *robust* if it can maintain its functions even if some of its components fail or stop interacting [11].

The robustness of S greatly depends on the topological structure of N and on the existence of multiple paths connecting two vertices in N. For example, let us refer to a communication network whose devices exchange messages by means of suitable physical links. In an extreme case, suppose that the network presents a star topology; if we removed the center of the star (along with the edges coming out from it), we would disconnect the whole network. Another extreme case occurs if we consider a clique. The removal of an arbitrary vertex would have no impact on the network functioning (any other vertex would remain connected to all others). Between these extreme cases, we observe that the malfunctioning of one or more components (or physical links) may not prevent a source component from correctly interacting with a target component because the source component could find alternative paths. This observation legitimizes a popular approach to studying network robustness; we analyze the fragmentation processes taking place in the network by progressively deleting vertices from N along with their connections [11, 74]. Real networks often include a large, strongly connected component (hereafter, SCC) retaining most of the network vertices. After deleting some vertices, along with their incident links, other vertices could detach from SCC to form small clusters (or even remain isolated). Because of this fragmentation process, we expect the network to become less and less connected, and this implies that the size of the SCC decreases. In a complementary fashion, the network diameter and the average path length APL (i.e., the mean of pairwise shortest-path lengths) should increase, thus making communication between vertices more difficult. According to the literature [369], APL is a more robust parameter than diameter. In fact, the existence of a long shortest-path in the network would imply a large network diameter even if vertices are, on average, only few hops away.

Albert and collaborators [11] focused on the robustness of two classes of networks, namely: *(i) homogeneous networks* in which the probability $P(k)$ that an arbitrary vertex has degree k exponentially decays for large values of k, and *(ii) heterogeneous networks* in which $P(k)$ follows a power law distribution. Examples of homogeneous networks are the Erdös-Rényi random graph or the small-world model by Watts and Strogatz [291]. Examples of heterogeneous networks include the Internet [149], the World Wide Web [27], and in general most (large-scale) social [92], and techno-social systems [274]. In their experiments, the authors considered both artificial and real networks (i.e., a sample of Web pages and hyperlinks connecting them) and empirically measured the size of the SCC and the diameter of the network if an increasingly larger fraction f of vertices was removed [11].

Two vertex removal strategies were considered: in the first strategy, vertices were randomly selected, while in the second strategy, the most connected vertices were progressively deleted from the network one by one.

In the case of homogeneous networks, no substantial variation in network diameter emerged if vertices were selected at random or in a decreasing order of connectivity.

A completely different behavior was observed for heterogeneous networks. The random removal of vertices had no effect on the diameter, while if the most connected vertices were deleted, then the diameter would quickly increase.

An analogous study was later proposed by Broder *et al.* [74], which took a sample of the World Wide Web graph and removed Web pages on the basis of the number of their outgoing hyperlinks. Conforming to the previous study [11], the authors found that it was sufficient to remove Web pages referred to by at least five other pages to destroy the Web connectivity [74].

8.4 FROM JUDICIAL DOCUMENTS TO CRIMINAL NETWORKS

In this section we describe the procedure we followed to extract the contact and the criminal networks from judicial documents.

Our dataset refers to a Mafia syndicate operating in the north of Sicily. To build such a dataset, we collected and manually analyzed hard-copy judicial documents covering a time period of 10 years. According to the Italian penal code, all evidence collected during investigation phases must be made public during the criminal trial.

The Italian penal code refers to the Mafia as a kind of criminal organization with a long history of violence and intimidation, as well as a deep-rooted code of silence. In the effort to defeat the economic and criminal power of Mafia gangs, many investigative tools are currently in use by law enforcement agencies. A widely used tool is electronic surveillance—namely, phone wiretapping as well as interception of on-site conversations. In the Italian legal system, if sufficient clues (*indicia*) exist, law enforcement agencies can request a judge to issue a warrant to carry on investigations by electronic surveillance. Despite the fact that general regulation of electronic surveillance is very strict, and wiretapping can be granted for no longer than forty days, law enforcement finds it an effective means for collecting evidence against Mafia gangs. In particular, on-site interceptions were acknowledged as the most effective electronic surveillance methods.

Collaborators of justice are an excellent investigation tool. In the Italian legal system, no immunity can be granted to an accused, but if that individual helps investigators in the discovery of decisive evidence against a criminal organization, she/he can be granted a protection program, which may be extended to family members and may include a change of personal details and assistance in the organization of a new life. Collaborators of justice play a key role in the fight against Mafia gangs, and they have contributed to the unveiling and understanding of the hierarchical structure of Mafia syndicates, as well as the role of each individual in the organization, the criminal strategies pursued, the collusion between politicians and the gang, etc.

Other popular investigative methods are stakeout, evidence from witnesses, phone records, and investigations of suspicious financial flows and bank transactions.

According to the Italian legal system, all evidence collected at the crime scene by one or more of the investigation tools must be reported in the judicial documents.

By means of data collected during phone wiretapping and phone records, we were able to generate a dataset consisting of individuals who appear in the phone record or whose phone calls have been recorded. Such a set is, in general, large but not necessarily revealing. It could contain phone calls made by mobsters, along with friends and family members, figureheads and, sometimes, victims of Mafia crimes (e.g., extortion rackets). Usually, police forces begin by intercepting phone calls made by a small group of suspected individuals. Phone calls allow them to discover new

	No. super-target	No. target	No. phone calls	No. useful links
Phone wiretapping	286	37	931 480	621
Phone record	188	21	293 280	6 954

Table 6: We report some statistics about phone wiretapping and phone records in our judicial documents. We consider the super-target (i.e., the overall number of phone lines subjected to wiretapping or included in the phone record), and the target (i.e., the number of phone lines subjected to wiretapping or included in the phone record which lead to the discovery of mobsters). We also report the overall number of phone calls subject to wiretapping and the overall number of phone records collected in the investigation. Finally, we report the number of phone conversations subjected to wiretapping or recorded in phone records that were used as evidence in the crime trial (*useful links*).

individuals (who sometimes are without a criminal record) as well as new social ties which are not necessarily crime ties. Police forces monitor phone conversations and decide whether it is or is not worthwhile to continue the investigation on a certain individual; this implies that the number of tapped people is much larger than the actual number of those that are accused (and those who are actually mobsters). The same procedure is repeated in the case of phone records.

By means of phone wiretapping, police authorities focused on a group of 286 telephone lines (called *super-target*), and they intercepted 931,480 phone calls ascribable to those lines. At the end of this phase of investigation, they determined that only 37 individuals (called *target*) were classified as suspected, and only 621 phone conversations were used as evidence in a crime trial. As for phone records, the super-target was formed by 188 telephone lines which generated 293,280 phone calls. At the end of the investigation, the target included 21 individuals, and 6,954 calls were used in the crime trial as evidence (see Table 6).

The *contact network* $N_{\text{con}} = \langle V_{con}, E_{con} \rangle$ is the network built on top of phone wiretapping and phone records. Here $V_{con} = \{v_1^{[con]}, v_2^{[con]} \dots, v_n^{[con]}\}$ is the set of individuals (vertices) subject to wiretapping or found in the phone call logs by law enforcement agencies, and $E_{con} \subseteq X_{con} \times X_{con}$ is the set of edges connecting a pair of vertices. The set of edges E_{con} is composed of all phone relationships (voice calls, SMS, MMS, etc.). In this work the edges E_{con} are considered undirected and unweighted, thus disregarding the orientation and the number of contacts between two any vertices.

We call $\mathbf{N}^{[con]} = (n_{ij}^{[con]}) \in \mathbb{N}^{n \times n}$ the adjacency matrix of $\mathrm{N}^{[con]}$ given by:

$$n_{ij}^{[con]} = \begin{cases} 1 & \text{if } \langle v_i^{[con]}, v_j^{[con]} \rangle \in E_{[con]}, \\ 0 & \textit{otherwise}, \end{cases} \tag{61}$$

We then refined N_{con} to obtain a new network, called N_{cri} which included all individuals arrested and tried; various sources were used to infer ties between members of N_{cri}. Among them, we cite on-site interceptions/phone wiretapping taken as evidence, phone records, witness statements of collaborators of justice, and bank transactions. All these details were digitally recorded in the judicial documents and extracted for our analysis. In Table 7, we report each source along with the information it yields and the vertices/edges it produces.

Source	Information	Vertices	Edges
Remedial custody document	Crimes, perpetrators and their roles roles	Perpetrators	Complicity
Phone wiretapping	Phone calls among defendants	Calling and called party	Phone conversations
Phone records	Phone numbers of defendants	Calling and called party	Phone contacts
Evidence from collaborators of justice	Information about the structure of a Mafia gang, mobsters and the crime offense they were involved to	Suspected	Crime ties
Stakeout	Meetings among suspected	Suspected	Acquaintance
Bank	Bank transactions	Bank account holder	Financial operation
Minutes of hearings	Witnesses	Accused	Crime ties
Judicial documents	Crimes committed by accused	Perpetrators	Crime ties

Table 7: We report the sources exploited to build N_{cri}, the information associated with each source, and the semantics of vertices and edges that such information yields.

By leveraging the sources described in Table 7 we were able to build a criminal network $N_{\text{cri}} = \langle V_{cri}, E_{cri} \rangle$ such that $X_{cri} = \{v_1^{[cri]}, v_2^{[cri]} \ldots, v_m^{[cri]}\}$ is the set of individuals subject to deepened investigations by law enforcement agencies by taking into account other kinds of relationships among them. The set $E_{cri} \subseteq V_{cri} \times V_{cri}$ comprises relationships among components of N_{cri} that are not telephone-based—such as joint bank transactions, complicity in a crime, and so on.

The matrix $\mathbf{N}^{[cri]} = (n_{ij}^{[cri]}) \in \mathbb{N}^{m \times m}$ is the adjacency matrix of $\mathrm{N}^{[cri]}$; it is defined as:

$$n_{ij}^{[cri]} = \begin{cases} 1 & \text{if } \langle x_i^{[cri]}, x_j^{[cri]} \rangle \in E_{[cri]}, \\ 0 & \textit{otherwise}, \end{cases} \tag{62}$$

Now we define the aggregate network $\mathrm{A}^{[aggr]} = \langle V_{aggr}, E_{aggr} \rangle$ where $V_{aggr} = X_{con} \bigcup V_{cri}$, $E_{aggr} = E_{con} \bigcup E_{cri}$ and the aggregated topological *adjacency matrix* associated $\mathbf{A}^{[aggr]} = (a_{ij}^{[aggr]}) \in \mathbb{N}^{|X_{aggr}| \times |X_{aggr}|}$ given by:

$$a_{ij}^{[aggr]} = \begin{cases} 1 & \text{if } n_{ij}^{[con]} \vee n_{ij}^{[cri]} = 1, \\ 0 & \textit{otherwise}, \end{cases} \tag{63}$$

of the unweighted network obtained from N_{con} and N_{cri} by joining all pairs of vertices i and j which are connected by an edge in at least one network and neglecting the possible existence of multi-ties between a pair of nodes and the nature of each tie as well [37].

The construction of N_{con} and N_{cri} is graphically reported in Figure 50.

8.5 THE STRUCTURE OF CONTACT AND CRIMINAL NETWORKS

In this section we analyze the main structural features of our contact and criminal networks. In Table 8 we report some statistics about N_{con} and N_{cri}. For each network we indicate the number of vertices ($|V|$) and the number of edges ($|E|$). We observe that N_{cri} contains only 104 vertices while N_{con} has 1,716 vertices. However, N_{cri}

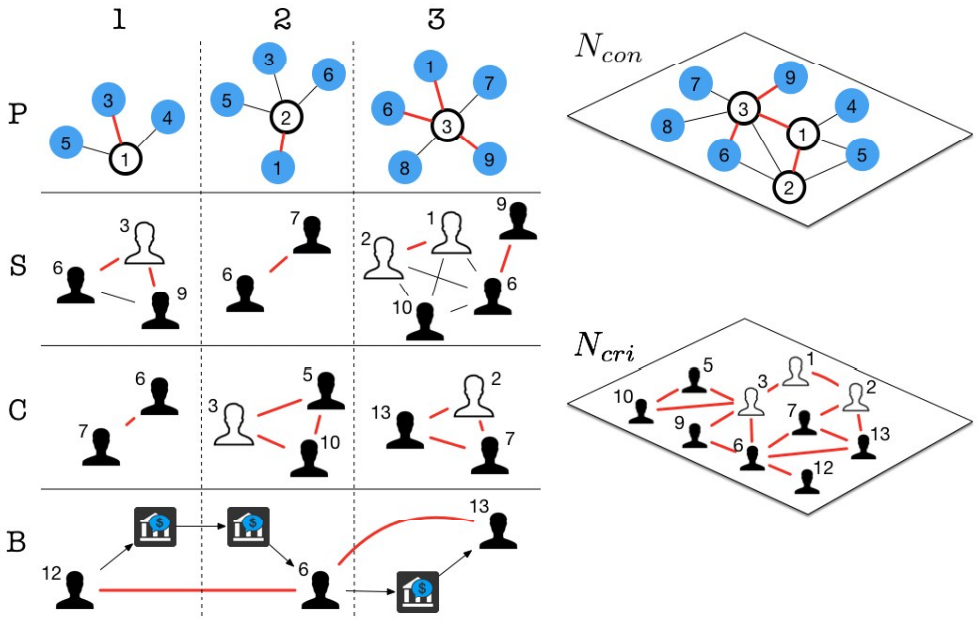


Figure 50: We graphically describe the construction of N_{con} and N_{cri}. In $P_{\{1,\dots,3\}}$ we report three graphs. In each of them a vertex identifies a phone line subjected to wiretapping (white vertices are target vertices and in blue we report the calling/called party). Red edges identify phone calls that proved useful to investigations. Therefore, N_{con} is generated by merging the three graphs P_1, P_2 and P_3. In $S_{\{1,\dots,3\}}$ we describe the output of a stakeout. Each vertex represents a suspect and an edge specifies that two suspects were seen together. Red edges identify meetings that were used as evidence in the criminal trial. White vertices correspond to targets intercepted in $P_{\{1,\dots,3\}}$. Graphs labeled $C_{\{1,\dots,3\}}$ describe crime ties obtained from the deposition of collaborators of justice or witnesses of complicity in crimes. In $B_{\{1,\dots,3\}}$ we report crime ties involving vertices $\{b_{12}, b_6, b_{13}\}$ which were inferred from the analysis of bank transactions. In both C_i and B_i graphs, red edges are those edges identifying relationships between pairs of individuals that were classified as interesting from prosecutors and were used as evidence in the criminal trial. Consequently, N_{cri} contains the vertices of graphs P_i, S_i, C_i, B_i with $i = \{1, \dots, 3\}$; its edges correspond to the red edges in each of these graphs, i.e., $N_{cri} = P_i^{[red]} \bigcup S_i^{[red]} \bigcup B_i^{[red]} \bigcup C_i^{[red]}$.

contains 2,596 edges, and therefore, it is much denser than N_{con} which contains only 8,481 edges. For this reason the average number of edges per vertex is only 9.88 in the case of N_{con}, and it amounts to 49.92 in the case of N_{cri}.

In Figure 51a we provide a graphical representation of N_{con}. The size of each vertex is proportional to its degree. We used different colors to pinpoint the role of mobsters in the Mafia syndicate. In yellow we report the leaders of the organization (the so-called "boss"). Green vertices represent *lieutenants*, i.e., the head of a branch of a Mafia syndicate who commands a crew of soldiers (known as *picciotti*) and who reports directly to the boss. Blue vertices represent actual mobsters, i.e., individuals who are known to be members of the syndicate. Blue vertices in the phone traffic

Network	$\|V\|$	$\|E\|$	$\langle k \rangle$	APL	Diameter	SCC
Contact Network (N_{con})	1716	8481	9.88	2,75	6	1
Crime Network (N_{cri})	104	2596	49.92	1.53	3	1
Aggregated (A_{aggr})	1722	11070	12.86	2.73	6	1

Table 8: Some statistics about N_{con} and N_{cri}. For each network we report the number of vertices ($V|$), the number of edges ($|E|$), the average degree ($\langle k \rangle$), the average path length (APL), the diameter and the size of the strongly connected component (SCC). We also report the same statistics for the aggregate networks A_{aggr} obtained from N_{con} and N_{cri} by joining all pairs of nodes i and j which are connected by an edge in at least one network.

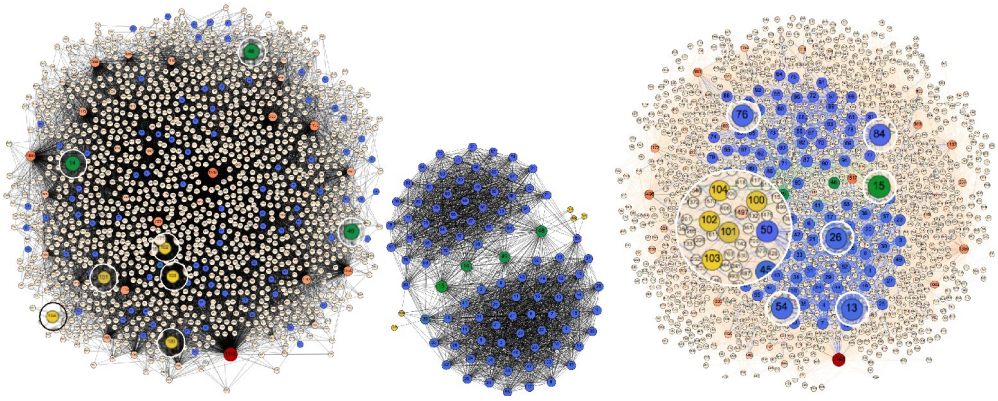
network are not key network actors as they are spread all over the network, often in peripheral positions. In fact, both their position and ranking are often not prominent.

Figure 51b shows the criminal network of dataset N_{cri}. It is composed of 2,590 links referring to relationships other than telephone-based contacts among the vertices of the network (examples include, but are not limited to, complicity in a crime, acquaintances, police inspections, bank transactions, etc.) found by the prosecutors during the investigations. N_{cri} includes the subset $I_{cri} = \{13, 26, 84, 15, 76, 54\}$ whose members are not present in N_{con}. They are mobsters that were never tapped during the investigations. The structure is characterized by two clusters (clans) tied together by the subset $L_{cri} = \{14, 15, 48, 49\} \subset N_{cri}$ whose members are the so-called lieutenants (in green). As expected, the bosses (yellow vertices) of subset $B_{cri} = \{100, 101, 102, 103, 104\}$ are situated between the two clusters, have a small number of links and at first look are not marginal.

Figure 51c represents the aggregated network A_{aggr} which comprises the overall structure of the two networks in our study. We highlighted the vertices representing the bosses B_{cri} (in yellow) together with the members of the subset $I_{cri} \subset N_{cri}$ not belonging to N_{con}. The density of connections among the elements of N_{cri} changes the structure of network N_{aggr}. Indeed, the two clusters of N_{cri} appear in the center, so that a core is formed.

From Table 8, which summarizes information about the datasets, and from the graphical representations shown in Figure 51, we can conclude that almost every member of the criminal network N_{cri} under arrest is also a member of network N_{con}. Interestingly, telephone-based relationships among associates are rare in N_{con}. Indeed, only seven links were found, namely $C_{cri} = \{(90, 2), (104, 87), (50, 87), (28, 87), (33, 87), (19, 87), (23, 93)\}$.

It is known that mob associates are aware of investigative techniques and are inclined to minimize direct telephone-based communications. Indirect communications are accomplished by intermediaries without a criminal record, and are above suspicion and unknown to law enforcement agencies. This feature is clearly illustrated in Figure 52a in which all telephone-based connections are shown among vertices V_{cri} and a subset capturing the most central vertices belonging to N_{con}. In our opinion this is one of the most strategic elements to assuring the resilience of a network. In this way a criminal network is not exposed to destabilizing attacks of law enforcement agencies because it manages a very limited number of telephone-based



(a) (b) (c)

Figure 51: Left panel: We report a graphical representation of N_{con}. Here, a vertex is associated with a suspected mobster while an edge indicates that the two suspects called each other at least once. Yellow vertices correspond to bosses, green vertices identify lieutenants, and blue vertices identify associates that were later arrested by law enforcement agencies. The size of each vertex is proportional to its degree, and the same holds for the color coding: light yellow is associated with nodes having the minimum degree, and red is used for nodes having the maximum degree. **Center panel**: Graphical representation of N_{cri}, namely mobsters and crime relationships between them (e.g., complicity in a crime, acquaintance, police inspections, bank transactions, etc.) **Right panel**: we show the aggregate network N_{aggr} where we highlighted vertices corresponding to bosses of the criminal organization (yellow) together with vertices $13, 15, 26, 54, 76 \in N_{\text{cri}}$ not belonging to N_{con}, corresponding to mobsters that were never tapped during investigations.

contacts among the members of the network. Nevertheless communications are still spread via elements that are not directly attributable to the network.

Figure 52b shows the connections of network N_{cri} and those among the members of B_{cri} and N_{con}. In this case some of the most central vertices of the network N_{con} are implicated. The egonets of bosses $\{102, 103, 104\}$ are shown in Figure 52d. Finally, Figure 52c shows the subgraph N_{aggr} which comprises all of the criminal and telephone-based connections of N_{cri} where we highlighted the vertices of the subset I_{cri}.

The analysis of vertices of bosses and their position within the structure of the network yields an important insight in the study of the resilience of a criminal network. As we can see from Figure 53a, bosses of the organization do not occupy important positions in the network A_{aggr}. Nevertheless, they are connected to the most important vertices in terms of degree. Even in this case, relations among the most authoritative members of the organization are limited in time and amount. They manage the overall network indirectly via trusted people that do not necessarily belong to the criminal network. In Figure 53b this concept is even clearer. Two subgraphs are shown that are obtained from the union of the egonets of the bosses of the set B_{cri}, precisely of the subgraph $B_{\text{ego1}} = \{101_{ego}\} \bigcup \{103_{ego}\}$ e $B_{\text{ego2}} = \{100_{ego}\} \bigcup \{102_{ego}\} \bigcup \{104_{ego}\}$ in which the bosses are connected to very few strategic vertices to guarantee communications and flow of orders towards all the members of

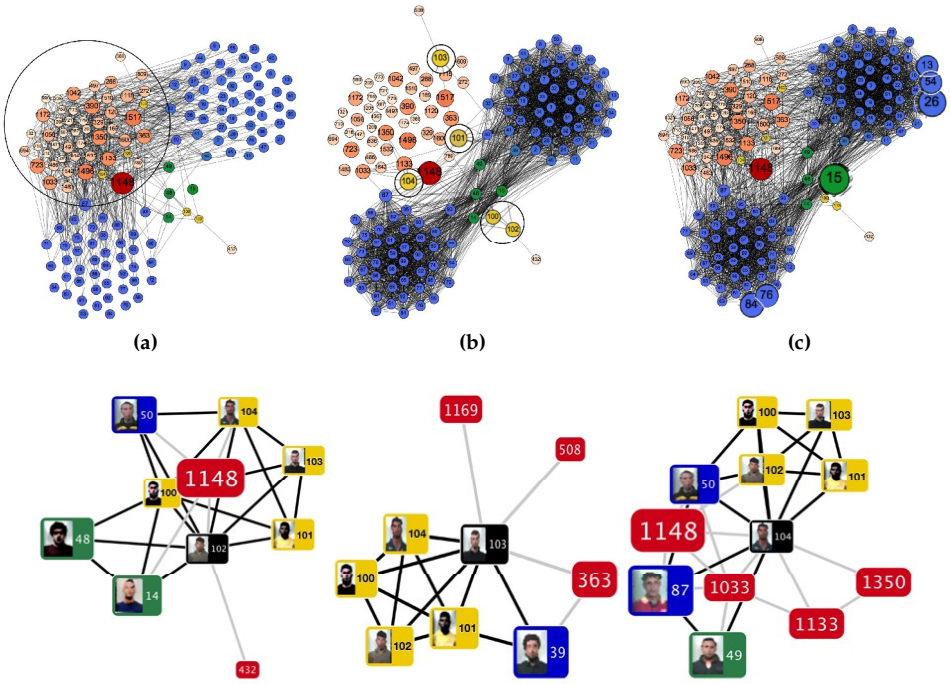


Figure 52: Panel (a): We show all connections among the vertices belonging to X_{cri} together with a subset of N_{con} having a high value of degree. **Panel (b):** We show the edges of the network N_{cri} together with the edges connecting the elements of B_{cri} and N_{con}. **Panel (c):** We show all criminal and telephone-based connections of network N_{cri} and highlight (zoom) vertices of subset I_{cri}. **Panel (d):** We shown the egonets of bosses $\{102, 103, 104\}$ filtered via the tool *LogAnalysis* [93]. The black lines represent edges of set E_{crim}, the grey lines represent edges of set E_{con}. Color codes: yellow vertices represent bosses, green vertices represent lieutenants, blue vertices represent associates, and red vertices denote members of the telephone-based network N_{con}.

the criminal network. Direct connections are lacking between the bosses of the set B_{ego1} and the set B_{ego2} in the network A_{aggr}. The bosses of the two groups never had telephone-based communications (or they had phone contacts that somehow escaped investigation), were never charged with the same crime, never left evidence of bank transactions, etc. This is another element of the resilience of criminal networks: the removal of a vertex from a subgroup has no consequences on the other subgroups. Nevertheless, the bosses are tightly tied and occupy the uppermost position in the criminal organization. This is why, in Figure 52d,we decided to include even missing relations (not present in the datasets) in order to increase the meaning of the visualization.

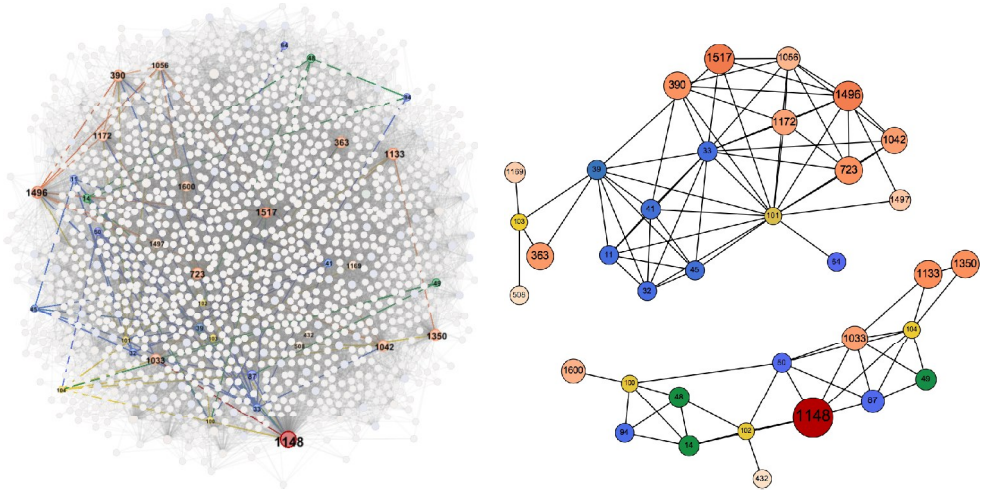


Figure 53: Left panel: Network A_{aggr} in which the egonets of the bosses B_{cri} of the criminal network are highllighted. **Right panel**: Subgraphs $B_{\text{ego1}} \subset A_{\text{aggr}}$ and $B_{\text{ego2}} \subset A_{\text{aggr}}$ of the egonets of the bosses obtained as the union of the egonets of every vertex of B_{cri}.

8.5.1 Analysis of the Structural Properties of Contact and Criminal Networks

The next step of our analysis consists of studying the structural properties of N_{cri} and N_{con}. To perform our analysis we considered two main parameters:

Degree distribution. Given a vertex i in N_{con} (resp., N_{cri}), we compute its degree k_i in N_{con} (resp., N_{cri}). From the analysis of the degree distribution it is possible to check whether there are vertices in N_{con} (resp., N_{cri}) which are much more connected than others, or vice versa, if the number of connections of each individual is roughly the same.

Average clustering coefficient. Given a vertex i in N_{con} (resp., N_{cri}), we define the *neighborhood of level 1* $\mathcal{N}(i)$ associated with i as the set of vertices which are adjacent to i in N_{con} (resp., N_{cri}).[2] The *average clustering coefficient* $\texttt{ACC}_i$ of i is defined as follows:

$$ACC_i = \frac{2 \times |\{\langle v, w\rangle : v \in \mathcal{N}(i), w \in \mathcal{N}(i)\}|}{k_i \cdot (k_i - 1)}$$

Here, $\{\langle v, w\rangle : v \in \mathcal{N}(i), w \in \mathcal{N}(i)\}$ is the set of pairs of vertices v and w which are connected to each other and are both simultaneously connected to i. The triplet formed by vertices i, v and w is also called *closed triplet*, and therefore ACC_i measures the number of closed triplets having a vertex in i out of the total number of triplets of vertices that contain i. The ACC_i ranges in $[0, 1]$, and if it is nearly equal to 1, then the neighbors of i tend to form a large number of triangles which favors the spreading of information.

[2] The vertex i is not considered in $\mathcal{N}(i)$.

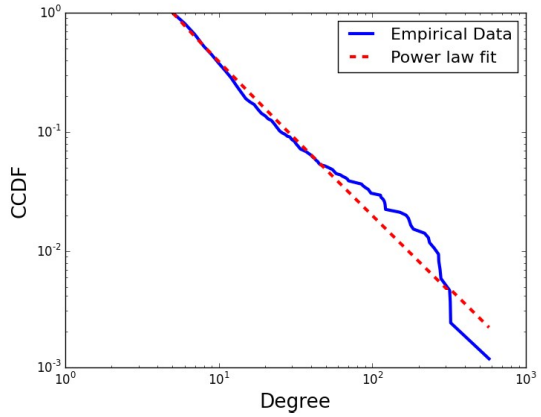


Figure 54: The CCDF associated with the degree distribution k_i in N_{con}. We used a log-log scale and, in the same plot, we report the power law distribution best fitting the experimentally observed data.

We begin our study by discussing the vertex degree distribution in both N_{con} and N_{cri}. As for N_{con}, we plotted the cumulative complementary distribution function (CCDF) which specifies, for a fixed threshold $\bar{k}$, the probability that a randomly selected vertex has degree greater than $\bar{k}$. We displayed the CCDF in Figure 54 on a log-log scale.

Similar to many other socio-technical systems, contacts among individuals in N_{con} are rather sparse and unevenly distributed, with a few vertices capturing most of the edges in N_{con}. We used the statistical tool described in [15], and we found that the degree distribution followed a power law with $\alpha = -2.5$ (p-value $< 10^{-5}$).

Since N_{con} is quite dense, we adopted a different graphical procedure to investigate vertex degree distribution. We ranked vertices in N_{cri} on the basis of their degree; in this way, the ℓ-th ranked vertex is the vertex showing the ℓ-th largest degree (see Figure 55).

We noticed a few individuals who were connected to almost all other individuals in N_{con}, but there was also a small number of individuals who were connected to few individuals and just one of them was isolated (i.e., there was a vertex with degree 0). The vast majority of vertices in N_{con} had a degree ranging from 15 to 85.

In N_{con} we found a few individuals who were well connected with many other individuals, but the largest part of vertices shows a low degree. In contrast, in N_{cri} there were only 15 individuals with degree less than 40 and only 17 individuals with degree larger than 55. This suggests the possibility for partitioning individuals in N_{con} in three disjoint classes on the basis of their degree—namely: *(i) Class A*, if $k_i \leq 15$, *(ii) Class B*, if $15 < k_i \leq 85$ and *(i) Class C*, if $k_i > 85$. With the aid of police officers, we observed that individuals belonging to *Class A* did not have leadership roles in the gang, but they often acted as intermediaries. Surprisingly enough, all the leaders of the gang were members of *Class C*; this relates to the fact that leaders in N_{cri} are aware of risks and are therefore in touch with just an handful of gang associates. The density of N_{cri} is likely to depend on the nature of many crimes; activities like drug trafficking or gambling encourage mobsters to organize into

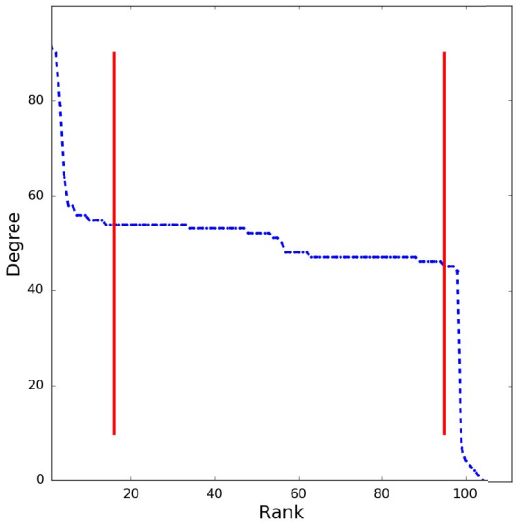


Figure 55: We report the degree of each vertex vs. its rank. The vertex with rank ℓ is the vertex having the ℓ-th largest degree. We split vertices on the basis of their degree, and we obtained three classes—namely *Group A* ($0\ k_i \leq 15$), *Group B* ($15 < k_i \leq 85$) and *Group C* ($k_i > 85$).

gangs and coordinate their actions. This implies that the resulting network must be dense, and it well explains why the average degree is roughly 50; i.e., any mobster is connected with half of the members of N_{cri}.

We continue our analysis by focusing on the structure of the social relationships of a given individual in both N_{con} an N_{cri}. Figure 56 shows the values of the average clustering coefficient as a function of vertex degree for both contact and criminal networks. In the case of N_{cri}, ACC features generally low values and is monotonically decreasing with the degree k_i. We notice that ACC in N_{cri} is always bigger than 0.6, which is a surprisingly large value. In fact, any socio-technical systems and Web platforms like Facebook or MSN Messenger feature a value of ACC in the range of $0.01 - 0.14$ [96, 369]. In addition, ACC achieves its peak for *Class B* individuals. Such a result can be paired up with our previous discussion: in N_{cri} there is a large faction (that accounts for roughly 90% of the whole population) of individuals with degrees ranging from 40 to 60 and, at the same time, the contacts of these individuals are themselves well-connected to each other.

The abundance of triangles in N_{cri} depends on the normative structure governing syndicates. In fact, past testimony, [251] as well as documents found during the arrest of Mafia boss Salvatore Lo Piccolo, showed that one of the first rules in the Mafia Decalogue was as follows: "No one can present himself directly to another of our friends. There must be a third person to do it."[3]

[3] See http://news.bbc.co.uk/2/hi/europe/7086716.stm

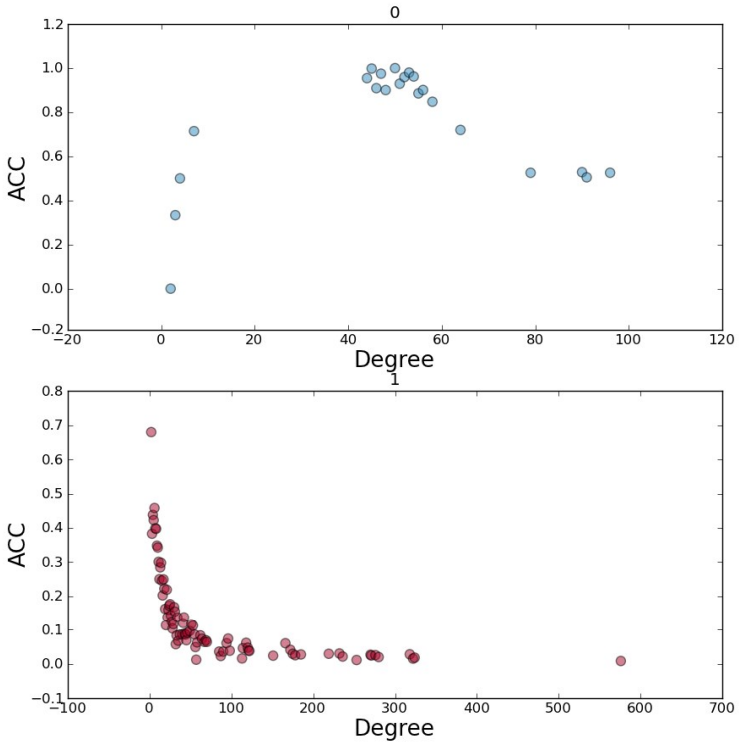


Figure 56: Top: Average clustering coefficient as function of k_i in N_{con}. **Bottom**: Average clustering coefficient as function of k_i in N_{cri}.

8.6 RANDOM AND TARGETED ATTACKS

In this section we aim at studying the resilience of the contact and criminal networks at our disposal to both random and targeted attacks. The resilience of a contact/criminal network is a crucial parameter to quantify the ability of a Mafia syndicate to react to the arrest of some of its members. More generally, Mafia syndicates tend to structure themselves in an way that is resilient to police operations aimed at hindering or temporarily/permanently inhibiting the functions of any specific member of the organization. We first introduce metrics applied to measure network robustness (Section 8.6.1), and then we discuss strategies to remove vertices from contact and criminal networks (Section 8.6.2). In Section 8.6.3 we describe how parallel and sequential police operations are performed; finally, Sections 8.6.4 and 8.6.5 summarize the main findings of our experimental analysis.

8.6.1 Metrics to Assess Network Robustness

To assess network robustness in presence of the dismissal of some of the network members, we rely on previous studies [11, 74] (see Section 8.3.2). We consider two parameters: *(i)* the size of the largest strongly connected component (`SCC`) and *(ii)* the average path length (`APL`).

A large value of SCC implies that a pair of arbitrarily selected individuals in N_{con} (resp., N_{cri}) are able to find a path (going through other individuals) along which a message can be routed. Relatively small values of APL imply that, on average, a person has to go through a short chain of intermediaries to get in touch with any other individual and, in the crime context, a quick flow of information is a crucial parameter to establish the survival of the organization itself.

8.6.2 Vertex Removal Strategies

For our network disruption analyses, we considered two attack strategies:

- *Random attack strategy*. We selected, uniformly at random, a fraction f of vertices from N_{con} (N_{cri}) and removed them along with their incident connections. We then measured the corresponding variation of SCC and APL. To produce statistically robust results, we ran the procedure described here 100 times and computed the average of SCC and APL. In our experiment, f varied from 1% to 25%.

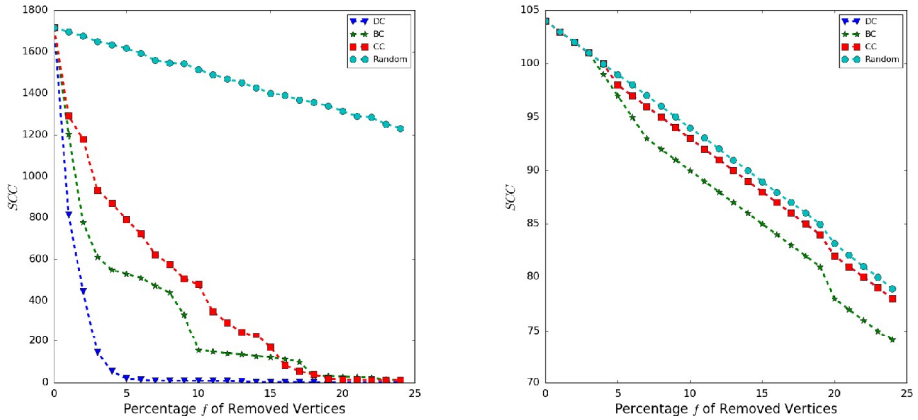
- *Targeted attacks informed by DC, BC and CC strategies*. We computed the centrality of each vertex by applying one of the three centrality indices introduced in Section 8.3.1: degree centrality (DC), betweenness centrality (BC), and closeness centrality (CC). Vertices having the higher centrality were deleted before vertices with lower centralities, according to the procedures described in Section 8.6.3.

8.6.3 Parallel and Sequential Police Operations

We are now in the position of introducing parallel and sequential police operations. For parallel police operations, let us consider a particular centrality index c and let $c(v)$ be the centrality of v according to c. Vertices and their centralities were stored in an hashmap $\mathcal{M}$ such that keys correspond to vertices and values correspond to centralities. The hashmap $\mathcal{M}$ was sorted by values; let $V' = \{v'_1, v'_2, \ldots, v'_n\}$ be the set of vertices sorted in decreasing order of centrality.

In parallel police operations we fixed $f \in (0,1)$, and we took the first $\lceil f \times n \rceil$ vertices from V'. For instance, if our network would consist of $n = 100$ individuals and $f = 0.05$, we should pick the first $\lceil f \times n \rceil = \lceil 0.05 \times 100 \rceil = 5$ individuals. We will denote the set of selected vertices as $V'(f)$. We deleted all vertices from $V'(f)$ along with their connections, and we measured SCC and APL. Such a procedure was repeated by varying f from 1% to 25%.

In a sequential police operation, we wanted to study how a criminal organization is able to reorganize itself when one (or more) of its members is neutralized. We suppose that mobsters are arrested one-by-one, and we measure how each arrest affects SCC and APL. This leads us to design the following experimental procedure: *(i)* We select a vertex according to the three centrality indices DC, BC and CC. *(ii)* We neutralize the selected vertex by deleting it along with its incident connections. *(iii)* We calculate SCC and APL on the network obtained at the end of Step *(ii)*. Steps *(i)-(iii)* are repeated until the top 30% vertices of N_{con} (resp., N_{cri}) are neutralized.



(a) (b)

Figure 57: Left panel: SCC vs. the fraction f of removed vertices in N_{con} in the case of parallel police operation. **Right panel**: SCC vs. the fraction f of removed vertices in N_{cri} in the case of parallel police operations.

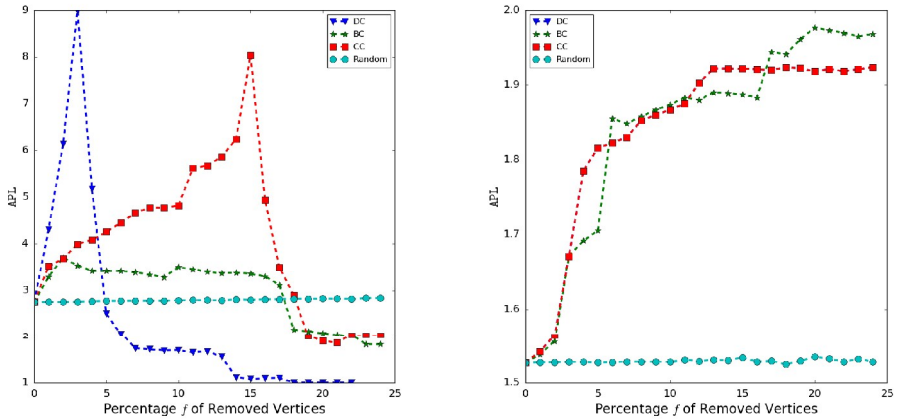
Observe that parallel and sequential police operations generally select different vertices and therefore have a different impact on SCC and APL. For illustration, let us focus on a specific centrality measure c and consider the set of network vertices $V' = \{v'_1, v'_2, \ldots, v'_n\}$, sorted in decreasing order of their centrality scores $c(v'_i)$, for $v'_i \in V'$. Suppose we now delete the vertex v'_1 along with edges incident onto it. We have no evidence that v'_2 will be the vertex having the largest centrality after the deletion of v_1, and we need to recalculate vertex centralities to detect the next vertex to delete.

8.6.4 Experimental Findings: Parallel Police Operations

We begin our analysis by examining the outcomes of our experiments when a parallel police operation is simulated (see Figures 57a and 57b).

As for SCC, we observe that targeted attacks are able to quickly destroy the strongly connected component in the case of N_{con}. In particular, from Figure 57a, *DC* has the most disruptive effect on SCC, and the removal of less than 5% of the most central vertices is enough to completely destroy the largest connected component. Random attacks yield a linear decrease in SCC and if f shifts from 5% to 25%, then SCC decreases about 24%. Here, *BC* and *CC* are respectively the second and third most effective strategies; however, they require, on average, a removal of between 15% and 20% of vertices to effectively disrupt SCC.

Different conclusions can be drawn if we focus on N_{cri} (see Figure 57b). Such a network displays an exceptional degree of robustness and, independent of the centrality index we decide to adopt, SCC always decreases in a linear fashion. Here, *BC* yields the largest decrease in SCC, and the lines associated with *DC* and *CC* mostly overlap. This result illustrates that there are no obvious targeted strategies that effectively disrupt a criminal network, at least by using parallel police operations.



(a) (b)

Figure 58: Left panel: APL vs. the fraction f of removed vertices in N_{con} in case of parallel police operation. **Right panel**: APL vs. the fraction f of removed vertices in N_{cri} in case of parallel police operation.

As a further experiment, we studied the variation of APL in N_{con} and N_{cri} when an increasing fraction of vertices was deleted from these two graphs under both random and targeted attacks (see Figures 58a and 58b).

We observe that random attacks are ineffective for increasing the value of APL in the case of N_{con}. Indeed, in N_{con} the most effective strategy is, once again, *DC*. We need to neutralize only the top 2% vertices from N_{con} to nearly triple the APL. Yet using *DC*, if $f > 4\%$, the contact network breaks into separate components. Analogous observations hold if we use *BC* and *CC* to score vertices even if the breaking point roughly occurs again with $f = 15 - 20\%$. Random attacks are, *de facto*, ineffective in augmenting APL in N_{cri}.

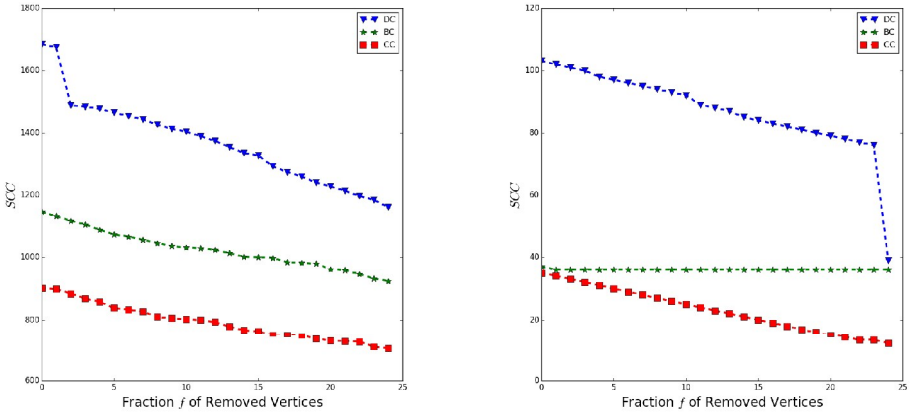
Our experiments suggest that in N_{con}, it is sufficient to neutralize a small fraction of vertices (around 5 - 7% if the *DC* strategy is adopted) to significantly reduce SCC and, simultaneously, increase APL. In contrast, due to its high density of crime ties, N_{cri} is much more resilient, and therefore it is able to effectively react to targeted attacks.

8.6.5 Experimental Findings: Sequential Police Operations

We conclude our experimental study by analyzing the resilience of both contact and criminal networks in a sequential police operation.

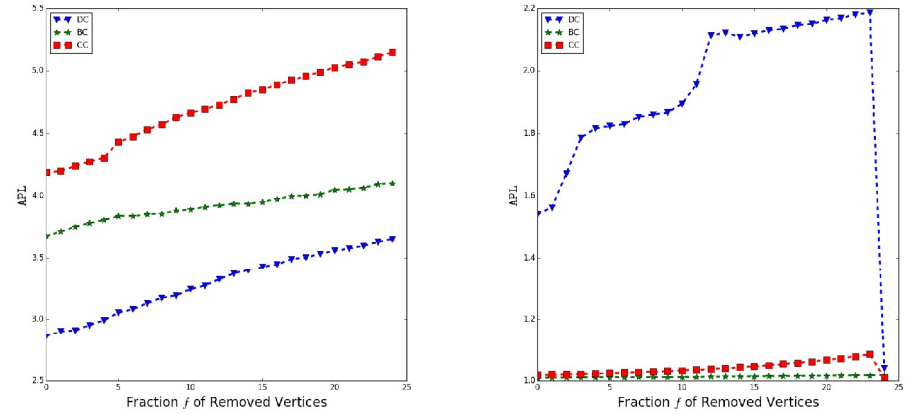
In Figures 59a and 59b we plot the variation of SCC. We observe that *CC* has the most disruptive effect on the reduction of SCC, and this happens both in N_{con} and in N_{cri}. In Figures 60a and 60b we plot the variation of APL when an increasing fraction f of vertices is neutralized from N_{con} and N_{cri}.

From these figures we observe that *CC* remains the best option to increase APL in N_{con}; however, the application of *DC* yields the largest increase in APL if applied on N_{cri}.



(a) **(b)**

Figure 59: Left panel: SCC vs. the fraction f of removed vertices in N_{con} in the case of a sequential police operation. **Right panel**: SCC vs. the fraction f of removed vertices in N_{cri} in the case of a sequential police operation.



(a) **(b)**

Figure 60: Left panel: APL vs. the fraction f of removed vertices in N_{con} in the case of a sequential police operation. **Right panel**: APL vs. the fraction f of removed vertices N_{cri} in the case of a sequential police operation.

To compare the effectiveness of parallel and sequential police operations, we focus only on the results we achieved on N_{cri}.

From our results, it seems that a sequential police operation has to be preferred to a parallel one. In fact, if we remove the top 5% mobsters in a sequential police operation we would reduce the size of SCC up to 65%. In contrast, in the case of a parallel police operation, we obtain a modest decrease in SCC of about 4.76%. Similar results hold for APL. From this discussion, it seems that sequential strategies should be preferred to parallel ones but, in practice, the identification of the best strategy to

dismantle a Mafia gang is a really hard task. In fact, police investigations last many years because of the need to collect a large amount of evidence prior to arresting individuals. In most cases, law enforcement agencies raid Mafia meetings held to discuss crime plans, and thus the end effect of this operation is that some high-caliber mobsters are captured. Therefore, parallel police operations are more realistic (and occur much more frequently) than sequential ones. In addition, sequential police operations will achieve the best results if the Mafia syndicate reacts slowly. In real cases, there are many events which may require a Mafia network to reorganize, not only police operations, but also *feuds*, i.e., conflicts between opposite gangs that often culminate in the killing of some gang members.

Mafia syndicates, as it emerges from judicial documents, are able to *instantaneously* adapt themselves to external events both at the group level (i.e., the Mafia gang may reform their internal organization by electing new bosses) and at the individual level (i.e., criminals may change their behavior or temporarily suspend illicit actions to avoid being targeted by law enforcement).

8.7 CONCLUSIONS

In this work we presented an experimental analysis of the network structure and resilience of Mafia syndicates. Thanks to collaborations with law enforcement, we were able to collect a precious dataset of digital trails and judicial documents that span a ten-year period of investigations of real crimes committed by Mafia gangs in the north of Sicily (Italy). The framework we presented here consists of reconstructing two types of networks—a contact network and a criminal network. The former was constructed from phone-based communications involving suspected individuals, while the latter is based on much stronger evidence of crimes involving individuals connected to Mafia syndicates. This includes evidence from stakeouts, sightings, bank transactions, etc. The sets of actors greatly overlap, yet our work highlights the presence of a small number of high-end criminals who do not appear in the contact network. This suggests that prominent bosses in Mafia syndicates are not adopting technology to help them remain off the radar during police investigations. This shows the limits of traditional investigation techniques like wiretapping, and calls for the adoption of complementary methods that help shed light where data cannot reach.

Given the unprecedented opportunity to adopt real data for our study, we focused here on investigating the resilience properties of contact and criminal networks. We found that criminal networks exhibit an exceptional robustness to targeted attacks, yet contact networks are much more vulnerable. We showed that various targeted strategies yield different effects of disruption with different performances; however, we provided quantitative evidence that sequential police operations should be preferred to parallel operations, although the latter are much more common and secure in that they expose investigators and the police force to fewer risks and violent encounters.

8.7.1 Limitations of this study and future work

Our study refers to one of the most relevant and powerful Mafia syndicate operating in Sicily. A comparative study on judicial documents associated to other Mafia-related trials may shed light on the differences between structure and business objectives of Mafia syndicates based in different parts of Italy and of the world. The techniques proposed in this paper are easy to generalize to datasets obtained from trials and other types of evidence: therefore, our methodology can be applied to study other Mafia syndicates and even other types of criminal enterprises.

Decisions taken by the investigators may introduce important biases in the data. The networks describing Mafia syndicates are a byproduct of the strategies adopted during the investigation phase: for example, if the investigators only employed wiretapping techniques (because other investigation strategies were too dangerous or expensive), other types of interactions will escape observation and analysis. Analogously, if some mobsters were never wiretapped due to some investigation decisions, the networks under analysis will be partial and the findings may be incomplete or inaccurate. Our study, therefore, pinpoints to the importance of deploying several investigation tools to get a more detailed portrait of Mafia syndicates and their activities.

Only few data sources are rich enough to allow reconstructing directed and/or weighted networks: for instance, in the case of phone calls, one can leverage the frequency or duration of contacts between suspects to create a directed and weighted network. Other such examples include financial transactions, exchanges of goods, or face to face interactions.

Concluding, our results specify under what conditions it is possible to reduce the ability of mobsters to communicate through shortest chains of intermediaries (which should ensure a higher level of secrecy) or decrease the number of contacts that a mobster can exploit to spread messages. Because the strategies adopted by Mafia gangs to react to raids and subsequently reorganize are largely unknown, we are not able to quantify to what extent the reduction in the ability of communicate actually impacts on the criminal power of a Mafia gang.

Our future research will focus on envisioning strategies of intervention that successfully complement the insights we obtained from this analysis. From a computational perspective, we aim at defining new methods to identify and predict crimes perpetrated by Mafia syndicates.

9 MULTIPLEX BFS

Breadth-first search (BFS) is an essential graph traversal strategy used in many real world optimization problems and computing applications. It is widely used as a basis for multiple fields: OSNs[1] and web crawling tasks [96, 97, 173, 269, 384], social networking, network broadcast routing, analysis of semantic graphs [244], model checking (finite state machine), garbage collection, community detection [104, 137, 289, 293] and connected components algorithms [295]. BFS is the basic building block for other graph traversals, such as best-first search, uniform-cost search, greedy-search and the A*. Algorithm is too strongly used in Social Network Analysis to compute the maximum flow in a flow network and solve the shortest-path problem in closeness and betweenness centrality [160].

A variety of parallel BFS algorithms for both CPU and GPUs architectural models based on shared and distributed memory systems have been explored for monoplex networks [5, 21, 82, 111, 240, 399].

In this Chapter we introduce a novel parallel Breadth-First Search algorithm (Mx-PBFS) developed for categorical and inter-layer couplings multiplex networks. The purpose is not to measure the performance of the algorithm design or to analyze in depth the theoretical issues related to architecture of the underlying computational model, that we remand to a future work, rather to give a simple parallel breadth-first search algorithm for multiplex network, by dynamic multithreaded programming.

The remainder of this Chapter is structured as follows. In Section 9.1 we review breadth-first search algorithm for monoplex graph, while in Section 9.2 we propose a novel parallel Breadth-First Search algorithm (Mx-PBFS) for categorical and inter-layer couplings multiplex networks. In Section 9.3 we will show the Mx-PBFS implementation details and will illustrate its serial and parallel execution steps.

9.1 SERIAL BFS FOR MONOPLEX NETWORKS

The first breadth-first search algorithm was discovered by Moore [272] while studying the problem of finding paths through mazes. Lee [235] independently discovered the same algorithm in the context of routing wires on circuit boards. Breadth-first search is one of the simplest algorithms for searching a graph and the archetype for many important graph algorithms. Prim's minimum-spanning-tree algorithm [317] and Dijkstra's single-source shortest-paths algorithm [130] use similar ideas.

Given a graph $G = (V, E)$ and a distinguished source vertex s, BFS systematically explores the edges of G to discover every vertex that is reachable from s. It computes the shortest path from s to each reachable vertex. It also produces a *breadth-first tree* with root s that contains all the reachable vertices. All vertices at a distance d (or

[1] Online Social Networks.

Algorithm 1: *BFS* monoxplex graphs algorithm [111].

Input: A monoplex graph $G = (V, E)$ and a source vertex s
Output: all the reachable vertices from s in G

```
1  for each u ∈ G.V − {s} do
2      u.color = WHITE
3      u.d = ∞
4      u.π = NIL
5  s.color = GRAY
6  s.d = 0
7  s.π = NIL
8  Q = ∅
9  ENQUEUE(Q, s);
10 while Q ≠ ∅ do
11     u = DEQUEUE(Q)
12     for each v ∈ G.Adj[u] do
13         if v.color == WHITE then
14             v.color = GRAY
15             v.d = u.d + 1
16             v.π = u
17             ENQUEUE(Q, v);
18     u.color = BLACK
```

level d) are first visited, before discovering any vertices at distance $d + 1$. The BFS *frontier* is defined as the set of vertices in the current level [21, 111]. Breadth-First Search works on both undirected and directed graphs. A queue-based sequential algorithm runs in optimal $\mathcal{O}(V + E)$ time.

Algorithm 1 gives a standard serial breadth-first search approach operating on a monoplex graph G. It assumes that the input graph is represented using adjacency lists and employs a FIFO queue Q as an auxiliary data structure to manage the set of *discovered* vertices. BFS stores the color of each vertex $u \in V$ in the attribute $u.color$, initially setting to white, and the predecessor of u in the attribute $u.\pi$. If u has no predecessor then $u.\pi$ = NIL. The attribute $u.d$ holds the distance from the source s to vertex u computed by the algorithm.

At the beginning of the algorithm, the distance of each vertex is set to infinity: it means that a node has not been reached yet, and therefore it has no distance from the source vertex. The parent attribute of each vertex is useful to access the nodes in a shortest path by backtracking from the destination node up to the starting node. NIL value represents the absence of a parent node. The **while** loop of lines 10-18 iterates as long as there remain gray vertices in the frontier, which are discovered vertices that have not yet had their adjacency lists fully examined. Once the **for** loop of lines 12-17 has examined all the u's adjacency list vertices, it blackens u in line 18.

Enqueuing and dequeuing operations take $\mathcal{O}(1)$ time, and so the total time devoted to queue operations is $\mathcal{O}(V)$. The scan of the adjacency list of each vertex is computed only when it is dequeued. Since the sum of the lengths of all the adjacency lists is $\Theta(V)$, the total time spent in scanning adjacency lists is $\mathcal{O}(E)$. The overhead for initialization is $\mathcal{O}(V)$, and thus the total running time of the BFS procedure in the RAM model is $\mathcal{O}(V + E)$.

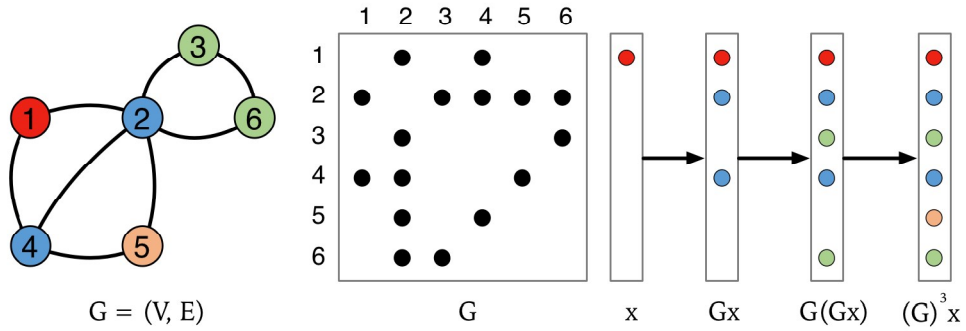


Figure 61: Breadth-first search implemented with matrix-times-vector multiplication on sample graph $\mathbf{G}$ with $N = 6$ nodes and 8 edges. A sparse vector $\mathbf{x}(i) = 1$ corresponds to the source node i. Repeated multiplication yields multiple breadth-first steps on the graph. Representation inspired at [171].

ADJACENCY MATRIX DUALITY

Breadth-first search can be also performed using linear algebra by multiplying the adjacency matrix $\mathbf{G}$ of a graph with a sparse vector $\mathbf{x}(i) = 1$, where i is the source vertex and all other elements zero [215]. The product $\mathbf{y} = \mathbf{Gx}$ extract column i of $\mathbf{G}$. If $\mathbf{G}$ is directed this produces the in-neighbors $\Gamma^{in}_{G}(i)$ of vertex i. To obtain the $\Gamma^{out}_{G}(i)$ neighbors we would compute $\mathbf{x}^T\mathbf{G}$ or $\mathbf{G}^T\mathbf{x}$ [171].

Each matrix-time-vector operation corresponds to a breadth-first step, and each result vector is the representation of the visited nodes in the graph after step k (See Figure 61).

We can make use of the matrix-times-vector operation on the Boolean semiring to perform BFS [370], by replacing the regular multiply and add operations with the Boolean AND and OR operators, respectively.

Multiplying adjacency matrix $\mathbf{G}$ with a matrix $\mathbf{X}$ with one column for each starting node, we can perform several independent breadth-first searches simultaneously. Column of matrix $\mathbf{Y} = \mathbf{GX}$, contains the result of a breadth-first search step from the node (or nodes) specified by column j of $\mathbf{X}$. If we add identity matrix $\mathbf{I}$ to graph $\mathbf{G}$ we have that the selecting all nodes at distance at most k on the k^{th} step. Breadth-first search work with sparse matrix multiplication is the same as that obtained via other efficient graph data structures implementation.

9.2 PARALLEL MULTIPLEX BREADTH-FIRST SEARCH

A variety of efficient[2] parallel BFS algorithms have been explored in literature for *monoplex* networks [5, 21, 48, 49, 82] based on commodity processors, for large distributed memory systems with the message passing programming model or special purpose hardware (GPU's with a different parallel programming model).

[2] The total number of operations performed by a parallel algorithm is linearly comparable to serial version.

A growing number of concurrency platforms support dynamic multithreading, including Cilk [164], Cilk++ [205], OpenMP [99], and Threading Building Blocks [321]. Our parallel BFS algorithm for multiplex graphs is implemented following the linguistic model for multithreaded pseudocode in [111, 164] and in MIT Cilk++ [205]. It augments the serial programming model with the three keywords: **parallel**, **spawn**, and **sync**. The keyword **spawn** creates parallel work of a function. It is semantically different from a C or C++ function call because the parent may execute in parallel with the child. A **sync** statement grants that a function safely uses the values returned by its children. It suspends the function until all of its spawned children return. Spawned function are managed by the scheduler on the individual processor cores by synchronizing their returns according to the fork-join logic provided by the **spawn** and **sync** keywords. Loops can be parallelized by the keyword **parallel**. Parallel loops can be implemented recursively using **spawn** and **sync**. Cilk++ also provides **reducer hyperobject** construct useful for no contention concurrent updates of a shared variable or data structure. The **serialization** of a multithreaded algorithm is the serial algorithm that results from deleting the multithreaded keywords: **spawn**, **sync**, and **parallel**. Therefore, Mx-BFS multithreaded pseudocode has the property that its **serialization** is always ordinary serial pseudocode to solve the same problem.

The parallel multiplex Breadth-First Search algorithm systematically explores a categorical and inter-layer coupled multiplex network M consisting of L layers and V nodes per layer, to discover every vertex of the multiplex graph that is reachable from a source node s. The considered topology is based on the connectivity between nodes representing the same entities (i.e. individuals) in the different layers. For instance, a Criminal Network with multiple level of relationships among participants (kinship, friendship, phone calls, instant messaging communications, email messages and bank transactions) where the interconnectivity pattern among layers is one-to-one.

Multiplex network definitions of distance function $d(s_\alpha \overset{p}{\rightsquigarrow} t_\beta)$, path $s_\alpha \overset{p}{\rightsquigarrow} t_\beta \in \mathcal{P}_{s_\alpha \rightsquigarrow t_\beta}$ and shortest path $P^*_{s \rightsquigarrow t}$, (Equation 54) used in this section have been introduced in Section 3.6.

In some studies that we have mentioned earlier in Chapter 2, each layer is treated independently and distances are evaluated on each of them. In the others, metrics are computed over an aggregated network obtained by projection of all layers into a single layer network. In the proposed approach, we do not perform any aggregation keeping the inherent structure of the interconnected layers in the multiplex.

Mx-PBFS search provides traversal of layers with directed and undirected relationships concurrently. Algorithm has been developed as a function of multilayer time dependent Criminal Network framework (`CriMuxnet`), a library for temporal multiplex networks analysis and visualization, that will be shown in Chapter 10. We use a multiple adjacent list ADT, implemented by Python dictionary of dictionaries. It has vertices as keys and a dictionary of nodes lists per layer where the vertex leads to.

To describe the implemented Mx-PBFS algorithm, we will use the sample multiplex network shown in Figure 62a. It is composed of three layers with the same sets of nodes in all layers. In Figure 62b is illustrated the same multiplex network as edge-coloured multigraph. Edge-coloured multigraph is a triple $G_e = (V, E, C)$, where V is the node set, C is the colour set used for labelling the type of edge and $E \subseteq V \times V \times C$ is the edge set [217]. There is a bijective mapping between

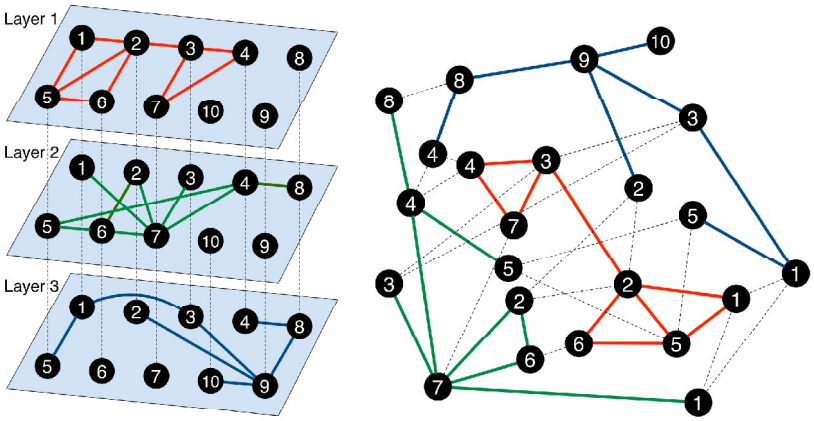


Figure 62: (a) A multiplex network sample with $L = 3$ layers and $N = 10$ nodes per layer. (b) Representation of the equivalent edge-coloured multigraph.

edge-coloured multigraphs structure and the equivalent sequences of graphs. This type of mapping gives only the intra-layer edges. However, we can assume implicitly that nodes are coupled to their counterparts in all layers. For an edge-coloured multigraph, each edge colour corresponds to a layer in a multilayer network.

Mapping multiplex network to edge-coloured multigraphs allows us to treat multiplex as an ordinary graph when explore network and computes the shortest path from source vertex to each reachable ones. Thus, we could use existing BFS "software" tools. Figure 63 shows classical BFS operation on the sample multiplex network. However, in this way the shortest paths are computed (a) considering the destination nodes in the different layers correspond to different entities, and (b) not distinguishing the interlayer connections from those intra-layers. The BFS serial algorithm steps (and running time) on edge-coloured multigraphs roughly increases of factor L (as many as the layers). For these reasons, we follow a different approach described in the next section.

9.3 MX-PBFS ALGORITHM

Mx-PBFS is based on the principles of classical breadth-first search we have seen before, and implemented in the similar way to that shown in *Introduction to Algorithms* course[3], by professor Erik Demaine[4] of MIT. It is outlined in Algorithm 2 for the simple directed and undirected cases.

The parallel random access machine (PRAM) approach to multiplex BFS is a straightforward extension of the serial algorithm presented in Section 9.1. The graph traversal loops (lines 10 and 12) are executed in parallel by multiple processing elements, and the distance update and stack push steps (lines 14-17) are atomic. A

[3] https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-videos

[4] http://erikdemaine.org/

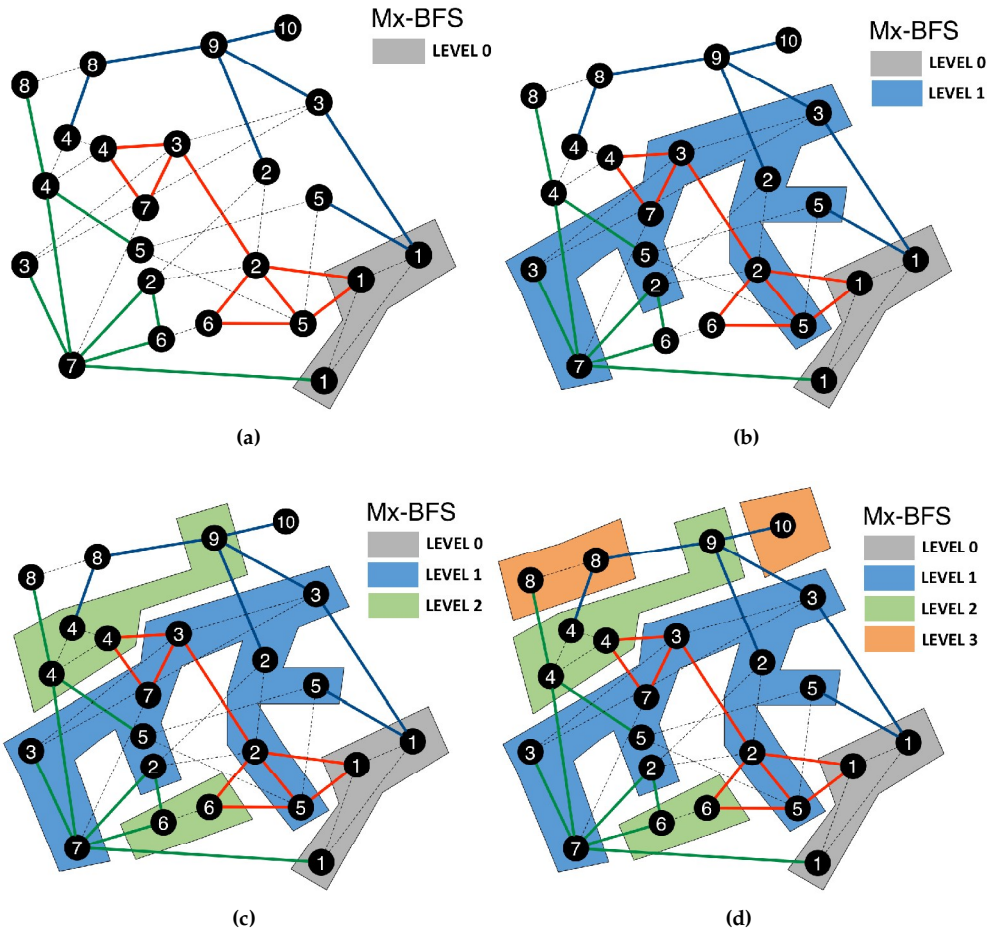


Figure 63: Illustrating classical Breadth-First Search on edge-colored multiplex network. The algorithm explores a graph level by level starting from the source vertex s (in this case the $s = 1$ in each layer). All vertices at a distance d (or level d) are first visited, before discovering vertices at distance $d + 1$. The BFS frontier is defined as the set of vertices in the current level.

barrier synchronization step occurs, once for each level, and thus the execution time in the PRAM model is $\mathcal{O}(D)$, where the D is the diameter of the multiplex graph. Since the PRAM model does not weigh in synchronization costs, the asymptotic complexity of work performed is identical to the serial algorithm [82]. Parallel implementation developed for Mx-PBFS follows the general structure of this *level-synchronized* algorithm [21].

Let $s \in V$ be the source vertex of multiplex network $M = (V_M, E_M, \mathbf{L})$, *level* d is the set $V_d \subseteq V$ of vertices at distance d from s in each *layer* $\ell \in L$ and their counterparts (as shown in Figure 65). Thus, we have that $V_0 = \{s^\ell, s \in V_M, \ell = \{1, \ldots, L\}\}$. Each iteration process all the vertices on a single level d at the same time by checking all the neighbors of vertices in V_d for those that should be added to V_{d+1}.

As we said before, level-synchronized parallel algorithm exploits concurrency at two key steps in Mx-BFS:

Algorithm 2: Parallel BFS algorithm for multiplex networks.

Input: A multiplex network graph $M = (V_M, E_M, L)$, and a source vertex s
Output: Visit all the nodes in M reachable from given node s

```
1  level = {s : 0}
2  parent = {s : None}
3  Q ← ∅
4  ENQUEUE(Q, s)
5  while Q ≠ ∅ do
6      for each v^ℓ ∈ Γ_M(u) in parallel do
7          u ← DEQUEUE(Q)
8          Γ_M(u) = γ(u) ⋃ λ(u)
           /* where γ(u) = {v^β ∈ V_M | (u^α, v^β) ∈ E_A, α = β ⊂ {1,...,L}} and
                λ(u) = {v^β ∈ V_M | (v^α, v^β) ∈ E_C, v^α ∈ γ(u), α ≠ β ∈ {1,...,L}}   */
9          for each v^β ∈ Γ_M(u) in parallel do
10             if v^β not in parent then
11                 if v not in level then
12                     parent[v^β] = (u, β)
13                     level[v] = level[u] + 1
14                     if v not in Q then
15                         ENQUEUE(Q, v);
16                 else
17                     if level[v] > level[u] then
18                         parent[v^β] = (u, β)
19                     ENQUEUE(Q, (v, β));
20             else
21                 if level[v] > level[u] then
22                     tmp =[ ]
23                     tmp.append(parent[v])
24                     tmp.append(u, β)
25                     parent[v] = tmp
26 return parent, level;
```

LEVELS All vertices in the queue Q at a given *level* in the graph can be processed simultaneously (line 7 in Algorithm 2)

NEIGHBORS The neighbors of each vertex can be inspected in parallel (line 9 in Algorithm 2).

It is possible, therefore, that when the algorithm processing in parallel two vertices u and v of the same level d, will both find the same vertex neighbor z at level $(d+1)$. The update of *level*[z], and also to set *parent*[z] to each of them at the same time creates a "benign data race" – it does not affect the correctness of the algorithm. In the former case, the same distance value is twice assigned to *level*[z], while in the latter it doesn't matter whether *parent*[z] turns out to be u or v. A more difficult problem about the BFS parallel implementation occurs due to parallel insertions of new level-$(d+1)$ vertices into FIFO queue at the same time.

The adjacency-list representation of a monoplex graph $G = (V, E)$ consists of an array *Adj* of V linked lists, where for each vertex $u \in V$, *Adj*[u] stores the neighbors of u. With the multiplex network $M = (V_M, E_M, L)$ where V_M is the set of nodes nodes ($|V|$ nodes per layer), $E_M \subseteq V_M \times V_M$ is the edge set containing the set of pairs

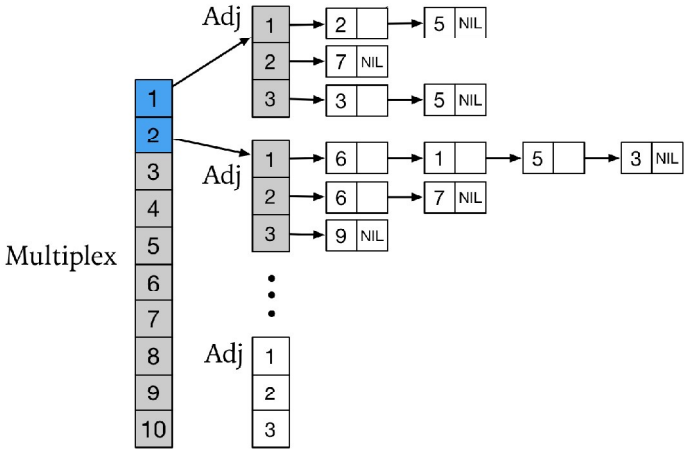


Figure 64: ADT structure used with Mx-PBFS algorithm.

of possible combinations of nodes and elementary layers, and L the set of layers, we adopt a similar concept. We use an array of pointers *Multiplex*, one for each vertex $u \in V_M$ to an array *Adj* of L linked list. For each layer $\ell \in L$, $Adj[\ell]$ stores the neighbors of node u (see Figure 64). In Python, *Adj* is a dictionary of list (or set) values, while a vertex may be represented by any hashable object. *Multiplex* is a dictionary of *Adj* dictionaries. For both directed and undirected graphs, the adjacency-list representation has the property that the amount of memory it requires is $\Theta(V + E)$.

Multiplex Parallel Breadth-First Search initialization differs from the classical one[5] to consider that each vertex in a layer have its replica in all other layers, even if it has no connections with other nodes. Therefore, the source node s is localized in all layers and corresponds to the same entity, while the neighbors of a node u are the union of two vertices sets $\Gamma_M(u) = \gamma(u) \bigcup \lambda(u)$ (Algorithm 2, line 7):

1. the set of the neighbors of u in all layers

$$\gamma(u) = \{v^\beta \in V_M \mid (u^\alpha, v^\beta) \in E_A, \alpha = \beta \in \{1, \ldots, L\}\}$$

2. the set of the neighbors counterparts in other layers

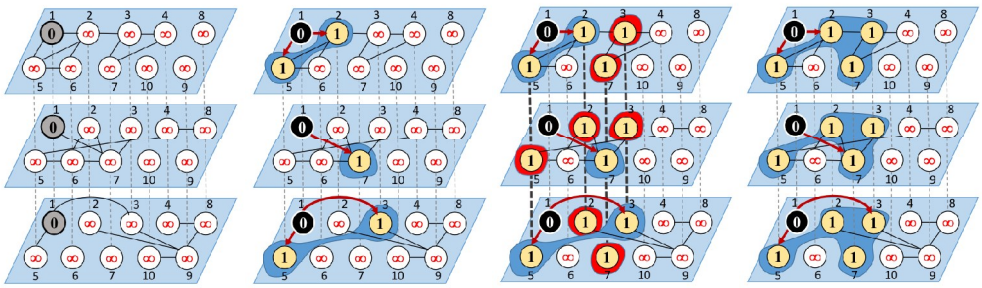
$$\lambda(u) = \{v^\beta \in V_M \mid (v^\alpha, v^\beta) \in E_C, v^\alpha \in \gamma(u), \alpha \neq \beta \in \{1, \ldots, L\}\}$$

where E_A is the set of intra-layer edges, and $E_{\tilde{C}}$ is the set of coupling edges for which the two nodes represent the same entity in different layers (see Section 2.4).

In Figure 65 is described this fundamental concept.

As well as the classical algorithm for monoplex networks, Mx-PBFS procedure explores graph level by level starting from a source node s. The idea is to look at the nodes that are reachable first in zero moves, that is the node s, then in one move: that's everything that can be reached from s in one step (adjacency of s) and so on.

[5] The serial BFS for monoplex networks



(a) **(b)** Set $\gamma(1)$ **(c)** Set $\lambda(1)$ **(d)** $\Gamma_M(1) = \gamma(1) \cup \lambda(1)$

Figure 65: The operation of Breadth-First Search in our sample multiplex network (how to determine neighbors). The value of distance (level) from source vertex appears within each vertex. (a) Start setting of multiplex network after initialization. Source vertex 1 in gray is dequeued, while all others vertices (in white) are not yet discovered. (b) Node 1 neighbors in each layer: set $\gamma(1) = \{2_{L1}, 5_{L1}, 7_{L2}, 3_{L3}, 5_{L3}\}$. They are discovered (colored in yellow within a blue convex shape), and are set at distance 1 from source node. (c) Counterpart neighbors set $\lambda(1) = \{3_{L1}, 7_{L1}, 2_{L2}, 3_{L2}, 5_{L2}, 2_{L3}, 7_{L3}\}$ of neighbors set $\gamma(1)$. They are discovered, colored in yellow within a red convex shape, and set at distance 1 from source vertex. (d) All neighbors of a source vertex 1 obtained by union of neighbors set and counterpart ones: set $\Gamma_M(1) = \gamma(1) \bigcup \lambda(1)$ (Mx-PBFS Algorithm, line 7).

Level d contains the list of vertices reachable by path of d edges. Ancestor and descendant relationships in the multiplex breadth-first tree are not defined as in BFS classical procedure. Even if not in the path, each vertex is the *predecessor* or *parent* of itself in other layers. Vertexes are discovered at most once, however they may have more then one parent (lines 20-24). In this way we can find all shortest paths from s to each other vertices of multiplex graph with no extra time computational cost. The loop over the neighbors $\Gamma_M(u)$ is done only once, in $\mathcal{O}(E)$ time (line 8).

The algorithm uses a FIFO queue as an auxiliary data structure to manage the set of candidate nodes for the next vertex to be visited. Leiserson and Schardl [240] replace the shared queue with a "bag" data structure which is more amenable for code parallelization with the Cilk++ run-time model.

The shortest paths from source node to each reachable vertex of the graph are computed considering the equivalence of entities in the different layers. For instance, in the shortest-path from node 1 in layer 1 (1_{L1}) to node 7 in layer 3 (7_{L3}) of multiplex network represented in Figure 65c $\{1_{L1}, 1_{L2}, 7_{L2}, 7_{L3}\}$, the interconnection edges between node's replicas in different layers $(1_{L1}, 1_{L2})$ and $(7_{L2}, 7_{L3})$ have no cost. Thus, each node will be at the same distance (level) in each layer: node 1_{L2} is at distance zero (level 0) from source node 1_{L1}, while both node 7_{L2} and node 7_{L3} are at distance 1 (level 1) from source vertex 1.

To extract shortest paths from parent dictionary, multiplex graph is traversed in a backtracking way, starting from the farthest nodes (or selected one) to the source (i.e. recursively take v parents until s or None).

The above parallelization will not work well for multiplex network with high-diameter due to the small number of adjacency vertices at each BFS level. Total

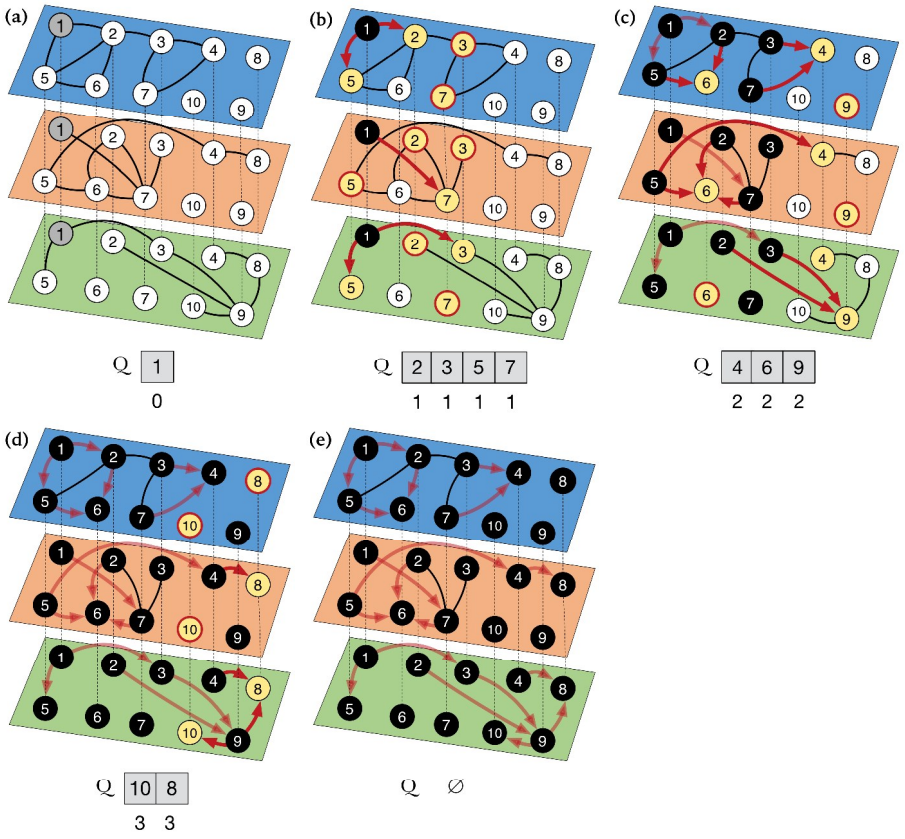


Figure 66: The execution of the parallel Breadth-First Search algorithm on the sample undirected multiplex network with $L = 3$ layers and $N = 10$ nodes per layer. Source node $s = 1$ is painted gray on each layer since we consider it to be discovered as the procedure begins. (b) For each layer, in parallel, algorithm explore neighbors of s and paints in yellow those have not yet been discovered. The neighbors counterparts (set $\lambda(s)$) are highlighted in red. The queue Q is shown at the beginning of each iteration of the while loop of lines 5-24. Vertex distances appear below vertices in the queue. A vertex is black when all its neighbors have been discovered. (c) All vertices at a level 1 in the graph are processed simultaneously as well as the adjacencies of each vertex. Edges in any shortest path from vertex s to vertex v are painted in red when they are discovered while they are shown shaded in the next iterations. (d) All vertices at a level 2 are processed in parallel. (e) The Mx-PBFS algorithms concludes in five iteration steps.

running time of sequential Multiplex BFS version is $\mathcal{O}(V + E)$. Thus, serial multiplex breadth-first search executes in linear time in the size of dictionary of dictionaries representation of M.

In Figure 66 is shown the execution of the parallel Mx-PBFS algorithm on the sample multiplex graph. The source is vertex 1 (level 0). After initialization, Mx-PBFS begins the **while** loop in line 6 which iteratively process level $d = \{1, \ldots, D\}$ where D is the diameter of the input multiplex graph (D is 3 in our example). To process V_d, the algorithm extracts each vertex u in queue Q in parallel and examines each

Step	u	$\Gamma_M(u)$	Action	Q
(a)	-	-	initialization	$\{1\}$
(b)	1	$\gamma_1 = \{2_{L1}, 5_{L1}, 7_{L2}, 3_{L3}, 5_{L3}\}$ $\bigcup \lambda_1 = \{2_{L2}, 2_{L3}, 5_{L2}, 7_{L1}, 7_{L3}, 3_{L1}, 3_{L2}\}$	set *level* $[\Gamma_M(1)] = 1$	$\{2,5,3,7\}$
(c)	2	$\gamma_2 = \{6_{L1}, 6_{L2}, 9_{L3}\} \bigcup \lambda_2 = \{6_{L3}, 9_{L1}, 9_{L2}\}$	set *level* $[\Gamma_M(2)] = 2$	$\{4,6,9\}$
	3	$\gamma_3 = \{4_{L1}, 9_{L3}\} \bigcup \lambda_3 = \{4_{L2}, 4_{L3}, 9_{L1}, 9_{L2}\}$		
	5	$\gamma_5 = \{6_{L1}, 6_{L2}, 4_{L2}\} \bigcup \lambda_5 = \{6_{L3}, 4_{L1}, 4_{L3}\}$		
	7	$\gamma_7 = \{4_{L1}, 6_{L2}\} \bigcup \lambda_7 = \{4_{L2}, 4_{L3}, 9_{L1}, 9_{L3}\}$		
(d)	4	$\gamma_4 = \{8_{L2}, 8_{L3}\} \bigcup \lambda_6 = \{8_{L1}\}$	set *level* $[\Gamma_M(3)] = 3$	$\{10,8\}$
	9	$\gamma_9 = \{10_{L3}, 8_{L3}\} \bigcup \lambda_9 = \{10_{L1}, 10_{L2}, 8_{L1}, 8_{L2}\}$		
	6	$\gamma_6 = \{\} \bigcup \lambda_6 = \{\}$		
(e)	10	$\gamma_{10} = \{\} \bigcup \lambda_{10} = \{\}$	-	$\emptyset$
	8	$\gamma_8 = \{\} \bigcup \lambda_8 = \{\}$		

Table 9: Parallel Multiplex BFS iteration steps. Letters in the first column refers to steps of algorithm illustrated in Figure 66. Other columns indicate: vertex u removed from queue Q, its adjacent vertices $\Gamma_M(u)$ in the multiplex network, action performed, and the queue Q at the beginning of each iteration.

edge (u, v^ℓ) in parallel, where $v^\ell \in \Gamma_M(u)$. If v has not yet been visited (line 10) then line 11 set $parent\left[v^\ell\right] = (u, \ell)$, line 12 set $level[v] = level[u] + 1$ and lines 13-14 inserts v into the $level$-$(d+1)$ queue Q. As illustrated in Figure 66c the update of $level[6]$ create a race, since node $\{2_{L1}\}$ and node $\{5_{L1}\}$ (as well as nodes $\{2_{L2}, 5_{L2}, 7_{L2}\}$ in layer 2) examining vertex 6_{L1} (6_{L2}) at the same time. They both check whether node 6_{L1} (6_{L2}) is not in *level* 2 in line 10, discover that it is, and both proceed to update $level[6_{L1}]$ ($level[6_{L1}]$). This race is benign and does not affect the correctness of the algorithm. Both 2_{L1} and 5_{L1} (and thus, $\{2_{L2}, 5_{L2}, 7_{L2}\}$) set the distance from source node 1 to the same value $level[6_{L1}] = 2$ ($level[6_{L2}] = 2$), and hence no inconsistency arises from both updating the location at the same time. They both go on to insert vertex 6 into queue Q in line 14, which could induce other races. Notice that inserting multiple copies of v into Q does not affect correctness, only performance for the extra work it will take when processing level $d+1$, because 6_{L1} will be encountered multiple times. We employ Cilk++ reducer functionality to avoid the race due to parallel insertion of vertices into queue Q. In Table 9 we report the iteration steps of the parallel multiplex BFS algorithm.

We illustrate a serialized Mx-BFS traversal for multiplex network in Figure **??**. The algorithm start with all white vertices except $s = 1$ that is painted gray, since we consider it to be discovered as the procedure begins. Vertex 1 is put on the queue, then the loop complete the search as follow:

1. Removes 1 from the queue and puts its adjacent vertices $\Gamma_M(1) = \{2,5,3,7\}$ on the queue, marking each and setting the *level* $[\Gamma_M(1)]$ entry for each to 1.

2. Removes 2 from the queue, checks its adjacent vertices $\{1,5,3,7\}$, which are marked, and puts its adjacent vertices 6 and 9 on the queue, marking each and setting the *level* $[\Gamma_M(2) = \{6,9\}]$ entry for each to 2.

Step	u	$\Gamma_M(u)$	Action	Q
A	-	-	initialization	$\{1\}$
B	1	$\gamma_1 = \{2_{L1}, 5_{L1}, 7_{L2}, 3_{L3}, 5_{L3}\}$ $\bigcup \lambda_1 = \{2_{L2}, 2_{L3}, 5_{L2}, 7_{L1}, 7_{L3}, 3_{L1}, 3_{L2}\}$	set *level* $[\Gamma_M(1)] = 1$	$\{2,5,3,7\}$
C	2	$\gamma_2 = \{6_{L1}, 6_{L2}, 9_{L3}\} \bigcup \lambda_2 = \{6_{L3}, 9_{L1}, 9_{L2}\}$	set *level* $[\Gamma_M(2)] = 2$	$\{5,3,7,6,9\}$
D	5	$\gamma_5 = \{4_{L2}\} \bigcup \lambda_5 = \{4_{L1}, 4_{L3}\}$	set *level* $[\Gamma_M(5)] = 2$	$\{3,7,6,9,4\}$
E	3	$\gamma_3 = \{4_{L1}, 9_{L3}\} \bigcup \lambda_3 = \{4_{L2}, 4_{L3}, 9_{L1}, 9_{L2}\}$	set *level* $[\Gamma_M(3)] = 2$	$\{7,6,9,4\}$
F	7	$\gamma_7 = \{\} \bigcup \lambda_7 = \{\}$	-	$\{6,9,4\}$
G	6	$\gamma_6 = \{\} \bigcup \lambda_6 = \{\}$	-	$\{9,4\}$
H	9	$\gamma_9 = \{10_{L3}, 8_{L3}\} \bigcup \lambda_9 = \{10_{L2}, 10_{L1}, 8_{L1}, 8_{L2}\}$	set *level* $[\Gamma_M(9)] = 3$	$\{4,10,8\}$
I	4	$\gamma_4 = \{\} \bigcup \lambda_4 = \{\}$	-	$\{10,8\}$
J	10	$\gamma_{10} = \{\} \bigcup \lambda_{10} = \{\}$	-	$\{8\}$
K	8	$\gamma_8 = \{\} \bigcup \lambda_8 = \{\}$	-	$\emptyset$

Table 10: Serial Multiplex BFS steps. Letters in the first column refers to steps of algorithm illustrated in Figure 67. Other columns indicate: vertex u removed from queue Q, its adjacent vertices $\Gamma_M(u)$ in the multiplex network, action performed, and the queue Q at the beginning of each iteration.

3. Removes 5 from the queue, checks its adjacent vertices $\{1,2,6\}$, which are marked, and puts its adjacent vertice 4 on the queue, marking it and setting the *level* $[\Gamma_M(5) = \{4\}]$ entry to 2.
4. Removes 3 from the queue, checks its adjacent vertices $\{2,7,4\}$, which are marked, and puts its adjacent vertice 9 on the queue, marking it and setting the *level* $[\Gamma_M(3) = \{9\}]$ entry to 2.
5. Removes 7 from the queue, checks its adjacent vertices $\{2,1,3,4\}$, which are marked.
6. Removes 6 from the queue, checks its adjacent vertices $\{2,5,7\}$, which are marked.
7. Removes 9 from the queue, checks its adjacent vertice $\{3\}$, which is marked, and puts its adjacent vertices 10 and 8 on the queue, marking each and setting the *level* $[\Gamma_M(9) = \{10,8\}]$ entry for each to 3.
8. Removes 4 from the queue, checks its adjacent vertices $\{3,7,8\}$, which are marked.
9. Removes 10 from the queue, checks its adjacent vertice $\{9\}$, which is marked.
10. Removes 8 from the queue, checks its adjacent vertices $\{4,9\}$, which are marked.

In Table 10 we report the sequential steps described above.

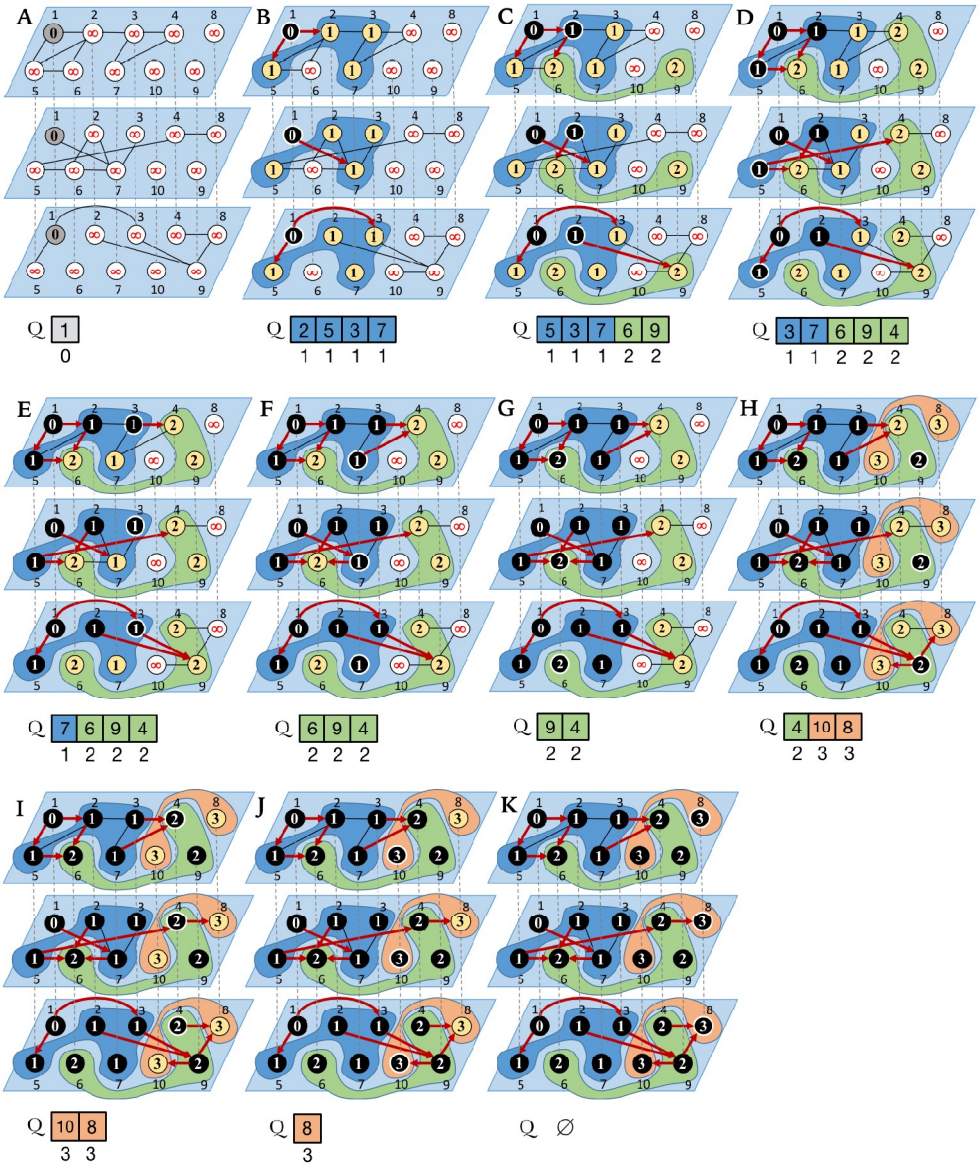


Figure 67: The operation of serialized Mx-BFS on the sample undirected multiplex graph.

10 MULTILAYER TEMPORAL EXTENSIONS

In the previous Chapters (see Section 6.2.2 and Section 6.2.2), we stressed the importance of criminal network temporal analysis by illustrating some features of our tools `LogAnalysis` and `LogViewer`. In those cases we studied criminal networks constructed from Call Detail Records (CDRs) to uncover criminals' mobility and behavioral, and evaluate their connections and positions during an offense or at a specified time window. We have dealt with monoplex networks in which nodes were connected by a single type of edges that encapsulates all relations between them.

Criminal (and terrorist) networks are *temporal* networks [197, 299]: "systems where connections between elements are active only for restricted periods of time" [225]. They are shaped both by the topological way in which participants are connected to each other and the temporal activity patterns of their dynamics (when and how they interact).

During the investigations, intelligence agencies collect data by looking at various sources of relational information to draw a picture of the network's structure [228]: credit files, bank accounts and the related transactions, telephone calling records, electronic mail, instant messaging, chat rooms, OSNs, court records and so on. Interactions among participants (or terrorists) also rely on other data-gathering methods, e.g. wiretapping, observations, informants, witnesses etc.

In these systems, each type of interaction has a given relevance and multiplex network provides a better representation of the networks.

In this Chapter we introduce a framework (`CriMuxnet`) for the temporal analysis and visualization of multilayer criminal networks. This framework is still in progress. The library supports multilayer networks with both temporal and multiplex aspect at the same time. It provides high-quality 3D visualizations of network data inside to commercial 3D computer graphics (CG) packages Autodesk Maya[1] and open source Blender [2], exploiting 3D engine features to full the goals of exploratory search and visualization strategy.

10.1 3D ANIMATION MODEL

Visual exploration is fundamental to human learning and problem solving areas (see Section 4) and it is expanding trough computer-generated imagery (CGI). Computer graphics has reached the stage where 3D models can be created and rendered, often in real time on commodity hardware, at a fidelity that is almost indistinguishable from the real thing.

In last decades, complex systems interactions (biological and chemical systems, neural networks, social interacting species, the Internet and the World Wide Web)

[1] http://www.autodesk.com/
[2] https://www.blender.org/

have been the principal focus of modern physics, mathematical, statistics, medical and bioscience research. Connections among genes, proteins, neurons, individuals and computer network structures at these scales, are often difficult or impossible to explore and visualize in depth. Multilayer networks model, also, adds extra complexity in the analysis and interpretation of these systems.

There are a number of software tools designed to help analyst understand (and visualize) networks in two dimensional space (2D), two and half space (2.5D) and 3D space, some of which have been described in Chapter 4, and in particular, for the analysis and visual representation of the multilayer networks: we mentioned the De Domenico et al. `muxViz`[3] software [120] and Kivelä `Pymnet`[4] library.

Our approach consists in employing tools built for other purposes, for representing multilayer networks in high-quality 3D visualizations, navigate, simulate interactions and to analyze every others network aspects. The challenge is to try to overcome the interactive and visualization limits of the tools mentioned above and provide a powerful mean for the analysis of complex networks in many areas.

10.1.1 Maya

Autodesk Maya, freely available for educational purposes[5], is the leading 3D application software used in the cinema, broadcast, architectural, design and gaming industries to mention a few.

Maya is a general purpose modeling, animation, and rendering application with a sophisticated engine for dynamics for simulating physical forces and collisions. Users can import or create geometry of various types (polygonal and spline-based surfaces), arrange these objects in a virtual 3D world, and change their positions and deformations over time.

Although these main aims, its programming interfaces may be accessed through a C++ application toolset (the API), via scripting in the Python language, and through Maya's embedded scripting language (MEL) [260] (see Figure 68). These features allow enormous power and flexibility in customizing Maya for scientific applications.

MEL Maya Embedded Language (MEL) was developed for use with Maya and is used extensively throughout the program. MEL scripts fundamentally define and create the Maya GUI. Maya's GUI executes MEL instructions and Maya commands. Users can also write their own MEL scripts to perform common tasks. MEL does not support object-oriented programming and can only communicate with Maya through a defined set of interfaces in the Command Engine (or by calling Python).

C++ API The Maya C++ application programming interface (API) is the most flexible way to add features to Maya and that can be executed faster than MEL. However, tools developed using the C++ API must be compiled for a specific version of Maya and also for each different target platform. Because of its compilation requirements, the C++ API cannot be used interactively with the Maya user interface.

[3] http://muxviz.net/
[4] http://people.maths.ox.ac.uk/kivela/mln_library/
[5] http://www.autodesk.com/education/free-software/maya

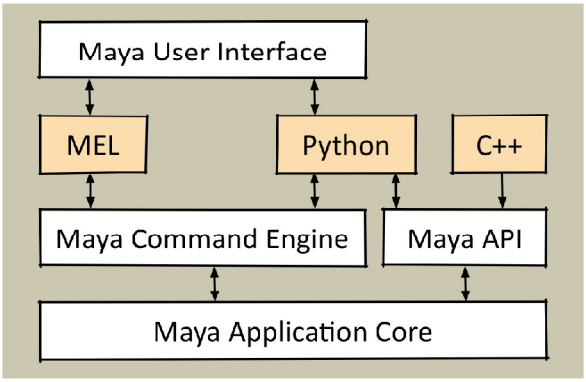


Figure 68: The architecture of Maya's programming interfaces [260].

PYTHON API Autodesk introduced Python into Maya and also created wrappers for many of the classes in the Maya C++ API. As such, developers can use much of the API functionality from Python. The total scope of classes accessible to the Python API has grown and improved with each new version of Maya. This powerful feature allows users to manipulate Maya API objects in ordinary scripts, as well as to create plug-ins and new features.

10.1.2 Blender

There are many others open source 3D computer graphics software like Maya. Blender is one of them. It supports the entirety of the 3D pipeline-modeling, rigging, animation, simulation, rendering, composing and motion tracking, even video editing and game creation. Blender has an internal fully fledged Python interpreter too. The Blender Python API is actually based on Python 3. It is integrated deeply, used for writing addons, generating user interface layouts, and import and export of many file formats. It covers all user-accessible data and functionality.

10.2 TEMPORAL MULTILAYER NETWORKS

The relationships among elements of a real networked system are dynamic. They occur at irregular intervals, even if some of them are statistically predictable.

Static networks structure analysis is considered an oversimplifying approximation. They are often time-aggregates of systems where connection happens only during an interval of time. Temporal feature is crucial for spreading of information, epidemiological application, brain signal processing, communication network and human behaviors to mention a few.

We formally define the concept of temporal multilayer network used in `CriMuxnet`, and introduce the notions of path, walk and length, following a similar scheme to that used in [217, 299].

Definition 1 (Temporal multilayer network). *A temporal multilayer network with multiple type of edges, in an interval* $[0, T]$ *is the quadruplet:*

$$M_{[0,T]} = (V^M_{\Delta t_w}, E^M_{\Delta t_w}, V, \boldsymbol{L}), \quad w = \{1, \ldots, \eta\}$$

where:

- Δt_w *is the duration of a time-window of multilayer temporal network*
- η *is the number of the graphs (time-windows) in the multilayer temporal network*
- $\boldsymbol{L} = \{L_a\}_{a=1}^{d} = \{L_1, L_2\}$ *is the set of two elementary layers (*$d = 2$ *aspects) defined as:*

  $L_1 = \{G^M_\alpha\}_{\alpha=1}^{\ell}$ *is a sequence of* ℓ *graphs corresponding to multiple type of edges of multilayer network*

  $L_2 = \{\Delta t_w\}_{w=1}^{\eta}$ *is a set of* η *successive non overlapping time-windows* $\{[t_i, t_i + \delta t]\}_{i=1}^{\eta}$ *such that* $0 < t_i < t_i + \delta t_i < \ldots < t_\eta + \delta t_\eta < T$
- V *is the set of node in the network*
- $V^M_{\Delta t_w} \subseteq V \times L_1 \times L_2$ *is the set of node-layer tuples; the set of layers in which a node* $v \in V$ *is present*
- $E^M_{\Delta t_w} \subseteq V_M \times V_M$ *is the edge set containing the set of pairs of possible combinations of nodes and elementary layers, within a time-window* Δt_w.

Each graph in the sequence can be either undirected or directed, according to kind of relationships represented by contacts.

Let $\{\boldsymbol{A}^\alpha_{\Delta t_w}\}_{\alpha=1}^{\ell} = \{A^\alpha_{\Delta t_1}, A^\beta_{\Delta t_2}, \ldots, A^\ell_{\Delta t_w}\}$ a time-dependent adjacency matrix corresponding of the multilayer network $M_{[0,T]}$ in a time-window Δt_w, then $M_{[0,T]}$ is fully described by means of a supra adjacency block matrix:

$$\mathcal{A}_{[0,T]} = \bigoplus_w \{\boldsymbol{A}^\alpha_{\Delta t_w}\}_{\alpha=1}^{\ell} + \boldsymbol{A}^{[\alpha\beta]}_{\Delta t_w}$$

where $w = \{1, \ldots, \eta\}$ are the η time-windows, $\boldsymbol{A}^{[\alpha\beta]}_{\Delta t_w}$ is the set of the interlayer block matrix.

10.2.1 Temporal walk and path

Considering the sequence of multilayer iteration in Figure 69a where the blue layer represents meetings between criminals while the next three sequence (fuchsia in transparency) ones correspond to phone calls contacts (per unit time) among them. The last graph on the right is the aggregate graph of contacts and meetings in Δt_1 time-window, generated from the union of all edges in the temporal multilayer graph.

Now, we pay attention to the path from node 2 to node 6. If we use the static aggregated graph seems to be a path from 2 to 6 via $(2, 3, 6)$, however if we take into account the time order there is no path between the nodes. This is the principal

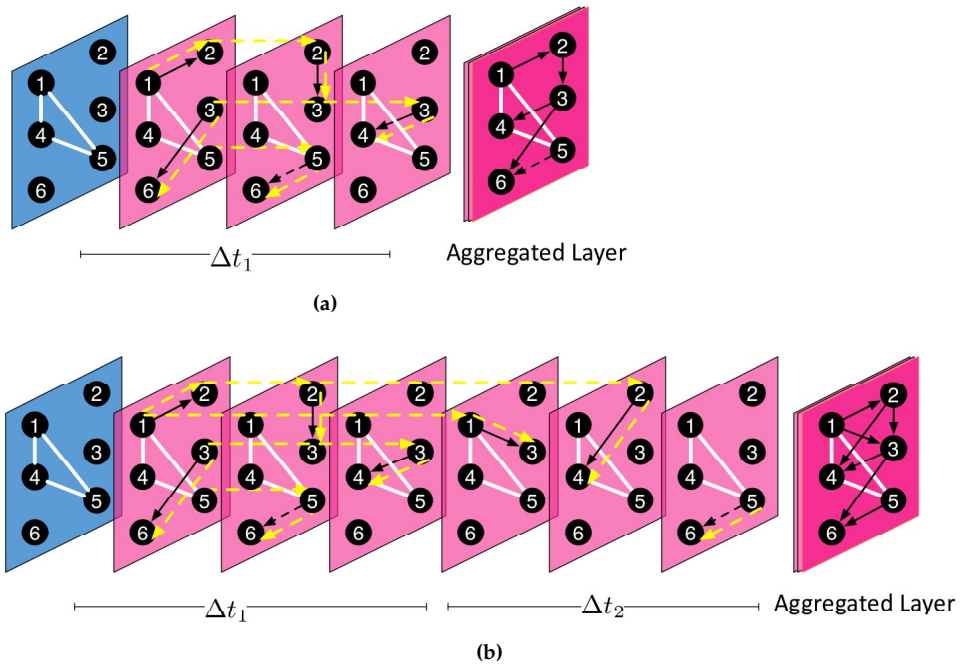


Figure 69: (a) Temporal multilayer sample graph $\mathcal{C}_{\Delta t_1}$ in the first time-window subinterval Δt_1. The blue layer represents meetings between criminals while the next three sequence (fuchsia in transparency) ones correspond to phone calls contacts (per unit time) among them. The last graph on the right is the aggregate graph of contacts and meetings in Δt_1. The dashed yellow arrows indicate the time-dependent paths. (b) Two sub sequential non-overlapped time-windows of the sample multilayer network. The addition of the Δt_2 time-window highlights more communication paths within the network.

benefit of using a temporal graph over the static counterpart. The interaction between the sub-path $(3,6)$ occur before sub-path $(2,3)$.

The path between node 2 and node 6 occur in the second time-window sequence as shown in Figure 69b. There is an indirected path via $(2,4,5,6)$ through the edge $(4,5)$ in meetings layer (blu one). This is another peculiarity of the employment of the temporal analysis in multilayer network: we can understand when contact (iteration) is made and how (through hich channel) it happened.

Walks and paths are essential concepts for many other measures in multilayer networks, including graph distance, connected components, betweenness centralities, random walks and clustering coefficients.

Walk definition on a multilayer network depends on some basic assumptions. First of all, we must evaluate if changing layers is considered to be a step, thus the same nodes in different layers are treated a set of distinct objects. In this case step and a walk are each defined as occurring between a pair of node-layer tuples [217]. Second, we have to take into account if intra-layer steps are different for layer.

Shortest path length on static (multilayer) graphs returns the number of hops from a source node to a destination node. This does not retain temporal information and hence cannot capture the true duration or speed of dissemination. Multilayer

temporal path, defined below, gives an indication of the speed of message delivery from a source to destination.

Definition 2 (Multilayer temporal path). *A multilayer temporal path, $p_u^v = n_{t_1} \rightsquigarrow n_{t_\eta}$ is a time ordered sequences of η nodes which starts form node $u = n_0$ at time window Δt_1 and finish in node $v = n_\eta$ with restriction that an edge exists between every pair of consecutive nodes $n_{\gamma-1}$ and n_γ at time window $\Delta t_{w-1} \leq \Delta t_\gamma$ and $0 \leq \Delta t_\gamma < T$.*

We define, also, the *temporal connectedness* for pairs of nodes in a time-varying multilayer graph as below: a node i of time-varying multilayer graph $\mathcal{C}_{[0,T]}$ is *temporally connected* to a node j if there exist a temporal path from i to j in $[0, T]$ [299].

10.2.2 Time-dependent multilayer sample

To describe concepts and futures of our multilayer criminal network framework (`CriMuxnet`), we will briefly introduce criminal network sample model which we will use as a case study.

First, we define a multilayer networked time-evolving system of a criminal network:

$$\mathcal{C}_{[0,T]} = (V_\mathcal{C}, E_\mathcal{C}, V, \mathbf{L})$$

where:

CRIMINAL NETWORK $\mathcal{C}_{[0,T]}$ has a total of six nodes $V = \{1, 2, 3, 4, 5, 6\}$

ASPECTS SET $\mathbf{L} = \{L_1, L_2\}$ which corresponds to an elementary layer set $L_1 = \{P, M\}$ of phone call contacts (P) and criminals meetings (M), and an elementary layer set $L_2 = \{\Delta t_w\}$ for $w = \{1, \ldots, t_\eta\}$ of a η finite non-overlapping time-windows $\{[t_i, t_i + \delta_{t_i}]\}_{i=1}^{\eta}$ where $i = \{0, \ldots, \eta\}$ is an observation interval $[0, T]$

CONTACTS AND MEETINGS between two nodes $i, j \in \mathcal{C}$ are represented by quadruplet $c = (i, j, t, \delta t)$ (contact) and quadruplet $m = (i, j, t, \delta t)$ (meeting), where $0 \leq t \leq T$ is the time at which the *contact* or the *meeting* started and δt is the duration expressed in an appropriate temporal unit.

We suppose also that we have the set of contacts and meetings among the seven nodes (criminals) within an observation period interval $[t_0, t_5]$ as illustrated in Figure 70. The observation period is divided in five successive non-overlapping time-windows Δt_w where $w = \{1, \ldots, 5\}$. Meetings (in blue) are considered *symmetric*, while contacts (in fuchsia) are not.

A multiplex adjacency matrix set $\boldsymbol{A}_{\Delta t_w}^{[P,M]}$ relating to the aspect $L_1 = \{P, M\}$ in a time-window Δt_w is described by the *time-dependent* adjacency matrix set:

$$\boldsymbol{A}_{\Delta t_w}^{[P,M]} = \{A_{\Delta t_w}^{[P]}, A_{\Delta t_w}^{[M]}\}, \tag{64}$$

where $A_{\Delta t_w}^{[P]} = \{c_{ij}^{[P]}\}$, with $c_{ij}^{[P]} = 1$ if i and j have a contact within time-window w and $c_{ij}^{[P]} = 0$ otherwise, and analogously $A_{\Delta t_w}^{[M]} = \{m_{ij}^{[M]}\}$, with $m_{ij}^{[M]} = 1$ if i and j have a meeting within time-window w, zero otherwise.

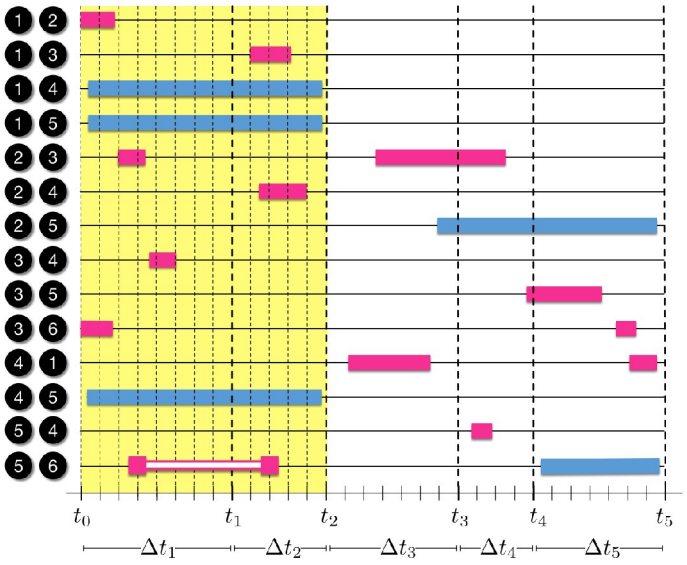


Figure 70: Timeline. Set of phone contacts and meetings among six criminals within an observation period $[t_0, t_5]$. Dashed lines correspond to five cuts of the sets in the corresponding subintervals Δt_w where $w = \{1, \ldots, 5\}$. Blue and fuchsia bars indicate the duration of each meeting and contact, respectively. The white bar within two fuchsia corresponds to an SMS that node 5 sent to node 6 in the first time-window but that is delivered in the next subinterval. Minutes are the time unit.

Next, we define the supra-adjacency matrix of the multiplex network $\boldsymbol{A}^{[P,M]}_{\Delta t_w}$ as:

$$\mathcal{A}_{\Delta t_w} = \begin{pmatrix} A^{[P]}_{\Delta t_w} & I \\ I & A^{[M]}_{\Delta t_w} \end{pmatrix} \tag{65}$$

So, the *time-varying* multilayer network is an ordered sequence of $w = 5$ time-window multiplex graphs:

$$\boldsymbol{\mathcal{A}}^{[P,M]}_{[0,T]} = \{\boldsymbol{A}^{[P,M]}_{\Delta t_1}, \ldots, \boldsymbol{A}^{[M,P]}_{\Delta t_5}\},$$

with related supra-adjacency matrix in a compact block form:

$$\boldsymbol{\mathcal{A}}^{[P,M]}_{[0,T]} = \begin{pmatrix} \boldsymbol{A}^{[P,M]}_{\Delta t_1} & \boldsymbol{A}^{[P,M]}_{\Delta t_1, t_2} & \cdots & \boldsymbol{A}^{[P,M]}_{\Delta t_1, t_5} \\ 0 & \boldsymbol{A}^{[P,M]}_{\Delta t_2} & \cdots & \boldsymbol{A}^{[P,M]}_{\Delta t_2, t_5} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{A}^{[P,M]}_{\Delta t_5} \end{pmatrix}$$

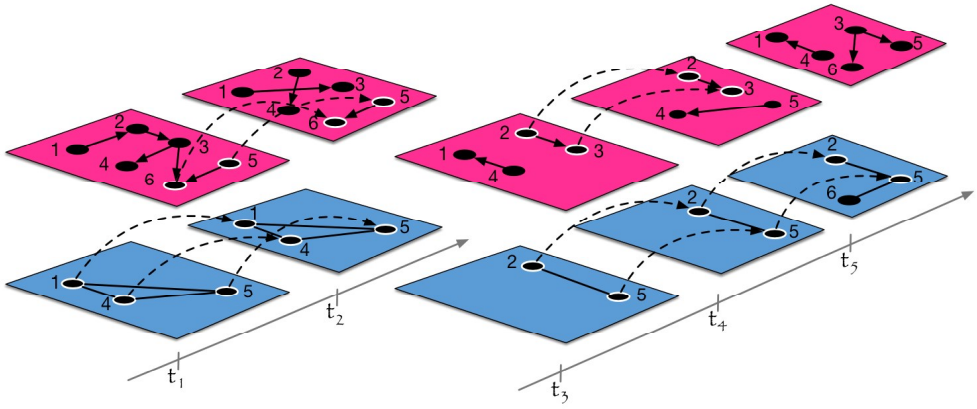


Figure 71: Schematic illustration of the mapping of a temporal network into a multilayer network. Each time-window is mapped into a different layer. Dashed lines correspond to contacts or meeting started in a time-window and finished to a subsequent one. Highlighted (white border) nodes are actives in more subsequent layers.

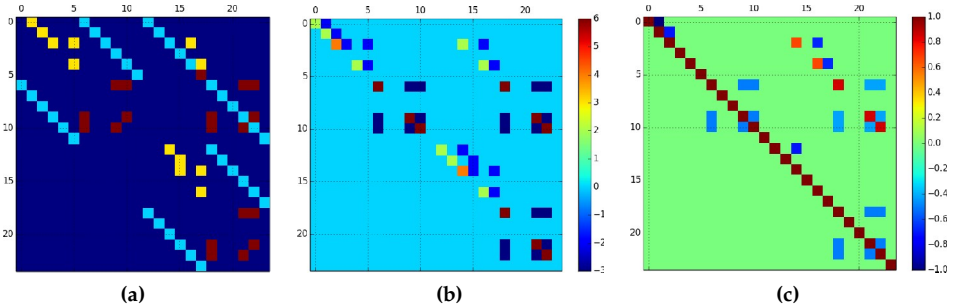
that can be expanded as follows:

$$\mathcal{A}^{[P,M]}_{[0,T]} = \left(\begin{array}{cc|cc|cc|cc} A^{[P]}_{\Delta t_1} & I & A^{[P]}_{\Delta t_1,t_2} & 0 & \cdots & \cdots & A^{[P]}_{\Delta t_1,t_5} & 0 \\ I & A^{[M]}_{\Delta t_1} & 0 & A^{[M]}_{\Delta t_1,t_2} & \cdots & \cdots & 0 & A^{[M]}_{\Delta t_1,t_5} \\ \hline 0 & 0 & A^{[P]}_{\Delta t_2} & I & \cdots & \cdots & A^{[P]}_{\Delta t_2,t_5} & 0 \\ 0 & 0 & I & A^{[M]}_{\Delta t_2} & \cdots & \cdots & 0 & A^{[M]}_{\Delta t_2,t_5} \\ \hline \vdots & \vdots & \vdots & \vdots & \ddots & & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & & \ddots & \vdots & \vdots \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & A^{[P]}_{\Delta t_5,t_4} & I \\ 0 & 0 & 0 & 0 & 0 & 0 & I & A^{[M]}_{\Delta t_5,t_5} \end{array}\right)$$

The diagonal blocks correspond to the multiplex networks within a non-overlapping ordered time-window $[t_i, t_{i+1}]$ while upper-diagonal block matrices represent the inter-layer connection. For simplicity, the lower-diagonal block is set to zero without loss of information.

In our model the only admitted inter-layer connections are between nodes and their counterpart when a contact (or a meeting) starts within a time-window and finishes to a subsequent one. In Figure 71 is shown a sketch of the multilayer temporal network $\mathcal{C}_{[0,T]}$.

In the sample network, the subintervals have been chosen to focus on meeting events, that is assuming as in a real investigation that one wants to explore network dynamics during a specific event, to answer to a specific questions: 'who has been called during an event and by whom, or who has been called next by the criminal met before?'.



(a) (b) (c)

Figure 72: Time dependent supra-adjacency matrix $\mathcal{C}_{[1,2]}$ (a) within time-windows t_1 and t_2. Yellow squares are contacts, red ones correspond to meeting between criminals. The upper right block shows interlayer links. (b) The weighted supra-Laplacian of $\mathcal{C}_{[1,2]}$ and (c) its normalized version.

10.3 CRIMUXNET VIZ

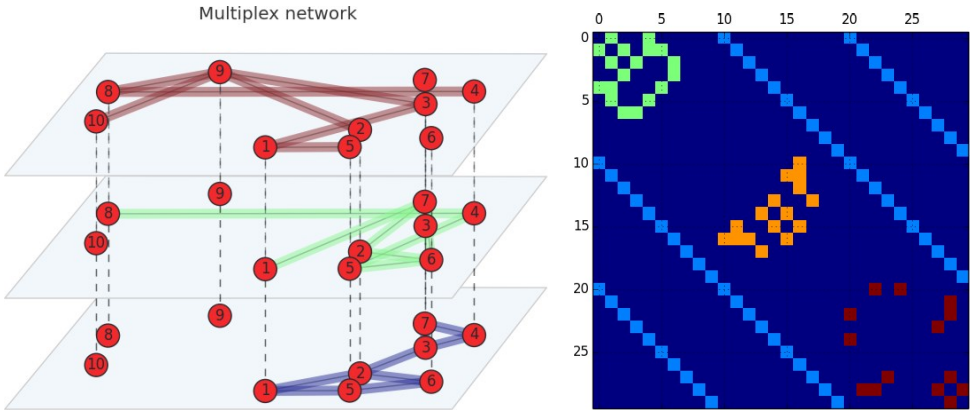
Our paradigm multilayer visualization software must satisfy several requirements. First of all the ability to import structure and interaction data. Next, it must be a robust suite of visualization tools to allow the user to complete control over how images are presented. Third, a physics engine that helps to evaluate complex network time dependent interactions, random walks, cascading failures, epidemics diffusions, percolations and other network dynamics. Until now, the framework includes only the first two steps.

`CriMuxnet` is a Python package for the manipulation and visualization of multilayer networks. It was been developed for multilayer network analysis of criminal networks, and it could be adapted for general multilayer network analysis purpose. `CriMuxnet` provide feature to: creating directed an undirected edges; weighted and unweighted multilayer graphs, analyzing spectral properties of supra-adjacency matrix and its correspondent Laplacian; visualizing networks an dynamics in both minimalistic drawing version in 2D and a high quality 3D visualization, inside to CG packages Autodesk Maya and Blender.

`CriMuxnet` is an extension of our tools `LogAnalysis` [97, 152] (see Chapter 6), and `LogViewer` [93] (see Chapter 7).

To demonstrate one of the potentialities of `CriMuxnet` in the analysis and visualization of multilayer networks, we consider the time-dependent supra-adjacency matrix of criminal networks shown in Figure 72a. It illustrates the sample multilayer network $\mathcal{C}_{[\Delta t_1, \Delta t_2]}$ in two sequential non-overlapping time-windows steps. In Figure 72b and in Figure 72c are shown both the related weighted supra-Laplacian matrix and its normalized version.

As we have highlighted in the previous Section 10.2.1, communication spreading is a fundamental topic in the study of temporal networks. It is strictly correlated to the concepts of paths and path lengths. In `CriMuxnet` we have included parallel Breadth-First Search algorithm (Mx-PBFS), described in the previous Chapter 9, which calculates all the shortest paths from single or multiple sources in the multilayer temporal network.



(a) (b)

Figure 73: (a) A multiplex network (2D) composed of three layers and ten nodes per layer. Each layer includes one elementary layer (we have $d = 1$ aspect). We represent intra-layer edges using solid curves and inter-layer edges using dotted curves. All of the inter-layer edges are coupling edges because nodes are adjacent only to themselves. (b) Supra adjacency matrix. Green squares represent the central layer link, brown squares the links of the top layer, while central orange square blocks refer to the the bottom layer edges. We use different colors in the latter case to distinguish colors of edges from that of the background of the supra-adjacency matrix.

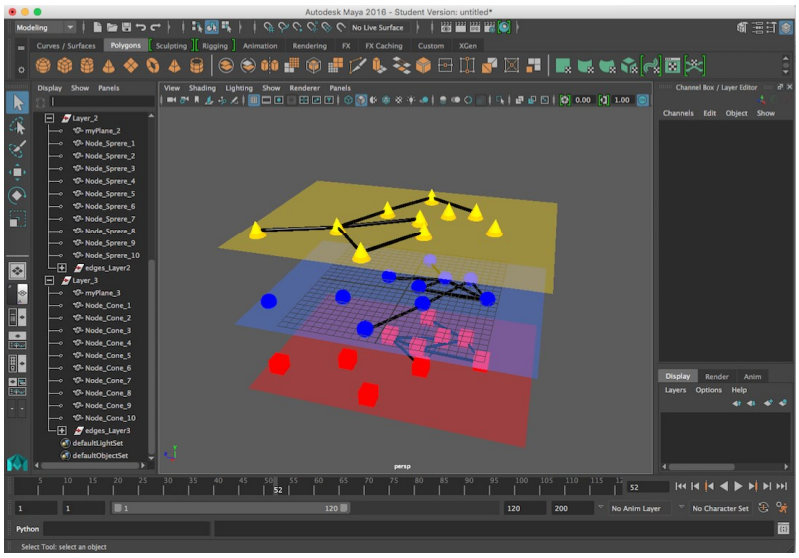
10.3.1 Python 2D visualization

In Figure 73a is shown a 2D multiplex network visualization. The network includes three layers with multiple types of edges. We show intra-layer edges as solid (and colored) lines and inter-layer edges as dashed lines. Multilayer network sample is node-aligned and couplings edges are categorical (each node is adjacent to all of its counterparts layers). The right side image correspond to supra adjacency matrix of the same multiplex network.

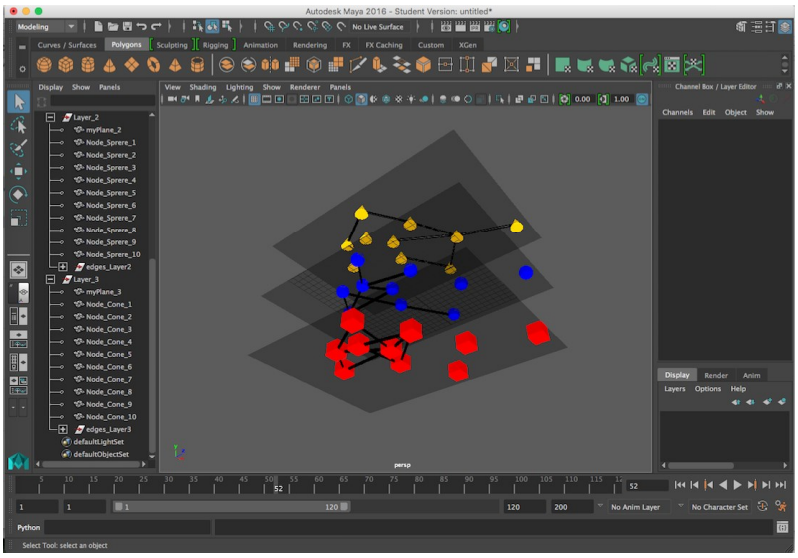
This visualization uses a force directed layout algorithm to arrange nodes and to minimize both intra- and inter-layer overlapping edges. The nodes hold the same position for aesthetic and comprehension reasons. In this way, also nodes representing the same entity (individual) are vertically aligned.

10.3.2 3D visualization

We use 3D CG software to create visual interpretation of multilayer (and complex) network phenomena. There exist many special-purpose tools for the representation and manipulation of networks structure we have cited and illustrated in several parts of this thesis. These powerful tools were designed for specific analysis tasks and users, and therefore don't fulfill a complete range of visualization needs. For instance, camera, lighting, shading, and animation options are limited even in the most advanced viewing applications. However, through an integrated workflow with more sophisticated visual software packages like Maya (or Blender), users can leverage the combined power of network modeling and advanced data visualization.



(a)



(b)

Figure 74: A multiplex network using Maya 3D software. For the sake of simplicity, the interlayer links have been omitted. Even if nodes in each layer represent the same entity, we have set them by different mesh (sphere, cube and cone) only for demonstration purpose.

Moreover, the ease with which complex, dynamic systems of interacting objects can be built, animated and visualized in 3D software makes it an effective tool for the rapid prototyping of network dynamics simulation.

If Figure 74 we show an example of 3D interactive visualization of the same multiplex network sample we used before in the previous paragraph, generated by `CriMuxnet`. We demonstrate the possibilities of our framework by using different

colors for each layer and distinct meshes for the nodes in each layer. Network exploration is very flexible thanks to the powerful graphics engine. It can be explored and analyzed from globally perspective to a very low level detail target. 3D environment significantly reduces the readability and the comprehension issues of certain complex network (multilayer) visualizations.

In Figures 75 and 76, we show multilayer networks under two perspectives: the first one is the type of relationship between the individuals (phone calls, meeting, e-mail) and the second one is the time. We assume for simplicity that all the edges are undirected, even if phone calls and e-mail messages obviously are not reciprocals. Each layer includes an elementary layer from each of the two aspects. We represent intra-layer edges using solid lines and omit inter-layer coupling edges because nodes are adjacent only to themselves. However nodes hold the same position in each layer to ensure comprehension. Nodes can display a label, embed pictures (photos) and have different meshes. In the example, colors are used to highlight the node which is an hub in different layers simultaneously and those with an certain centrality relevance. `CriMuxnet` algorithm creates the scene of network using Maya and Blender. It draws nodes, edges and layers by setting materials, meshes, camera, lights and shapes. Next, execute adapted force directed algorithm to adjust randomly distributed nodes in each layer of the system.

The `CriMuxnet` software also includes additional measures for visualizing multilayer networks and representing the results of the analysis in a meaningful way.

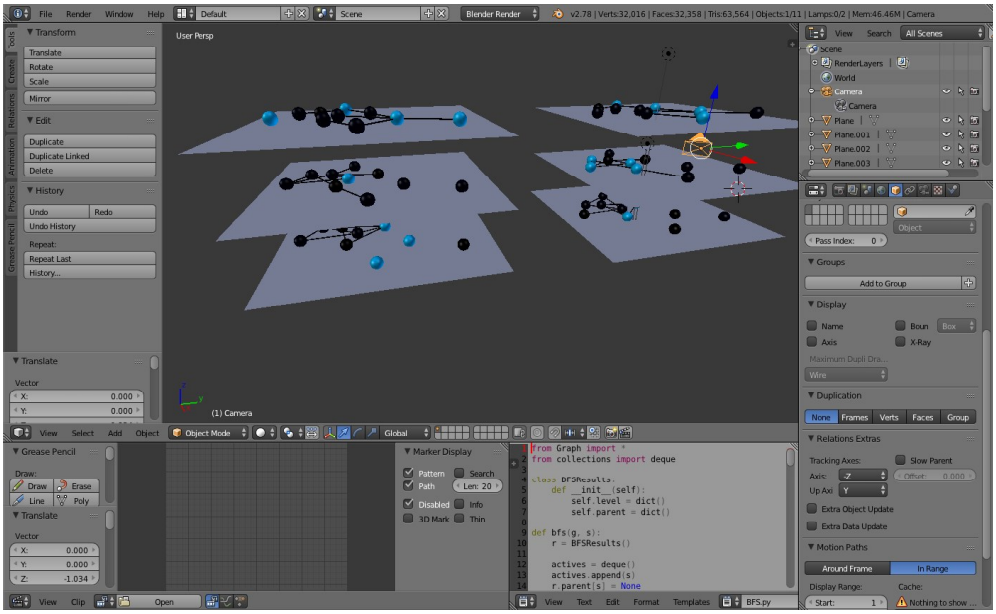
10.3.3 Features

The animation pipeline can be summarized in seven stages: modeling; characters; animation; materials and textures; lights and cameras; effects; rendering and compositing. These general stages describe the main tasks required to create an animation.

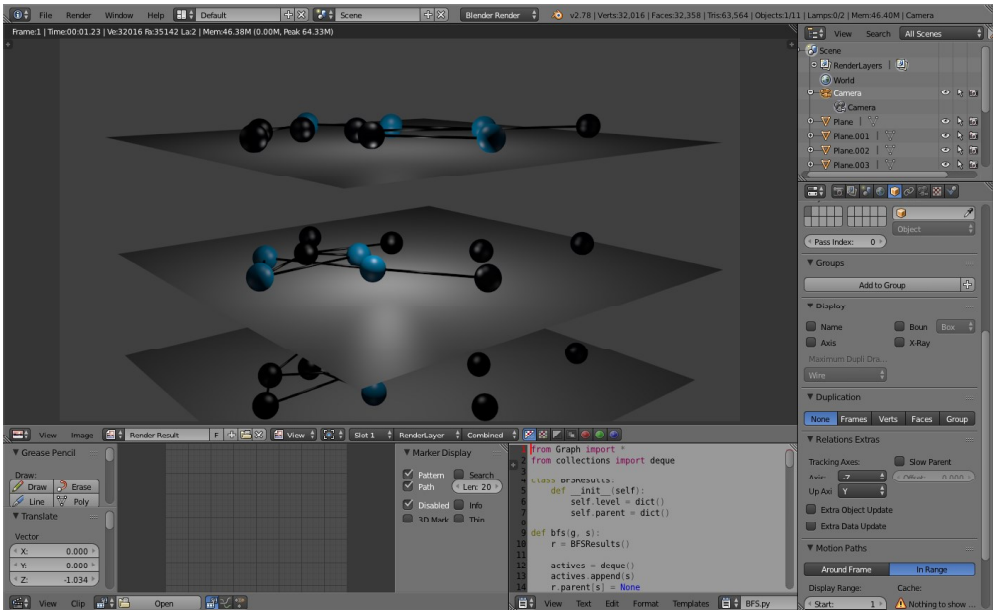
Let's see now briefly, some basic foundations and features that have been used in the 3D development process.

Scene

The scene is the 3D environment, including models and animation. It is essentially a stage for digital action. Several scenes may be developed to create a single traditional one, or one scene may contain the models, action, cameras, and lights needed to create an entire animation. At the heart of every Maya scene is the Dependency Graph. The DG represents all of the data in a scene. The DG is composed of two key components: *nodes* and *connections*. Nodes contain the actual data in the scene as well as operations being performed on the data, while connections establish relationships among the data in the nodes. A special type of DG node is the directed acyclic graph (DAG) node. These nodes are made of two specific types of connected nodes: transform and shape. The arrangement of DAG nodes consists of a hierarchy in which the shape node is a child of the transform node. Most of the objects with in the Maya viewport, such as surface geometry (cubes, spheres, planes, and so on), are DAG nodes.



(a)

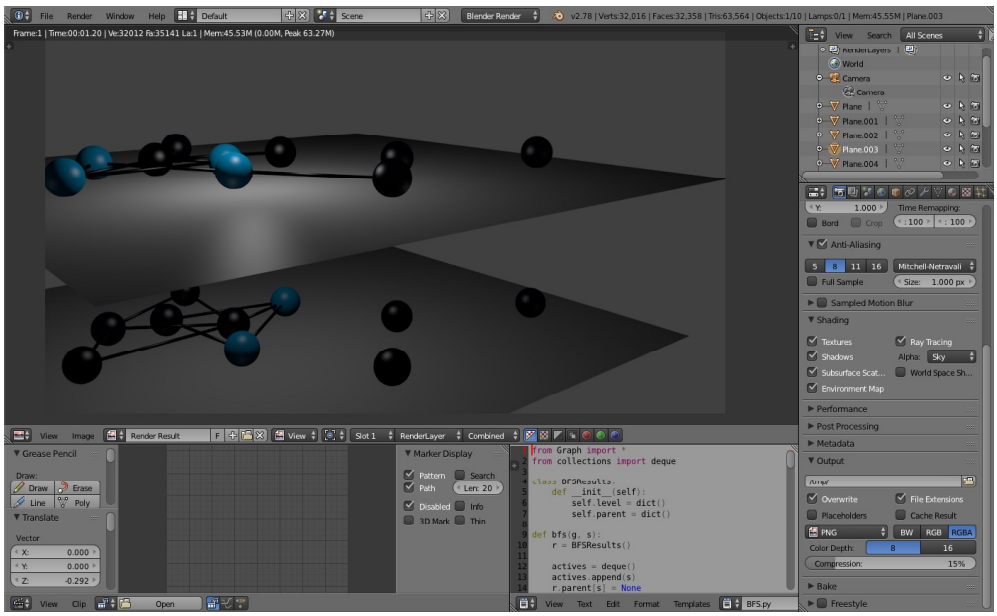


(b)

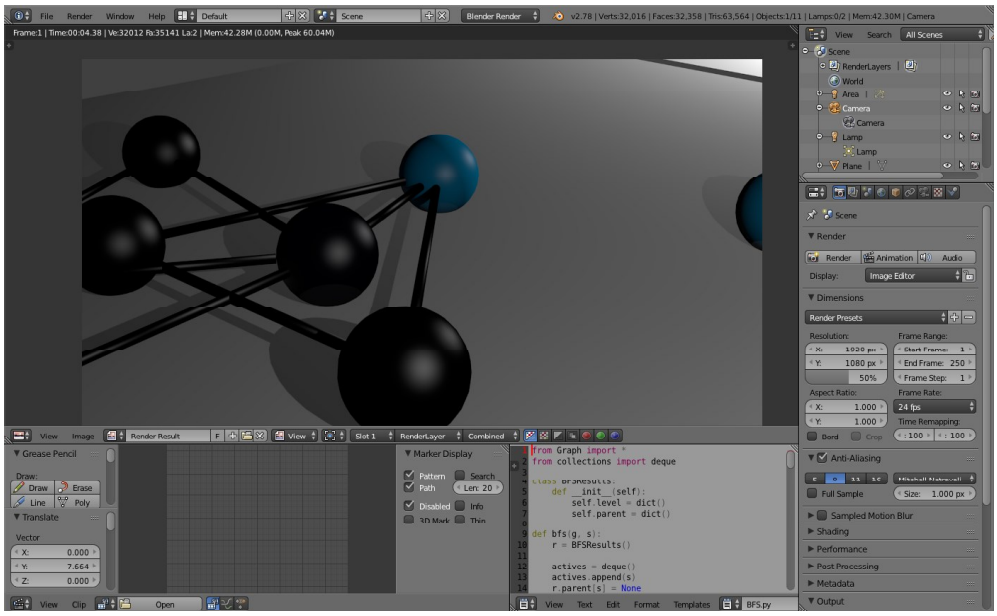
Figure 75: A three-dimensional multilayer network using Blender. Visualization generated by `CriMuxnet`. (a) It is shown the camera positioning around the scene and (b) the final rendering output.

Modeling

The creation of objects and environments in digital 3D space is called modeling. Objects in computer animation are typically modeled as shells with no solid, or



(a)



(b)

Figure 76: A three-dimensional multilayer network within Blender. Visualization generated by `CriMuxnet`.

volumetric, form to them. There are two main surface types that make up these shells. These are polygonal surfaces, comprised of many interconnected flat polygons and spline surfaces, also called NURBS, that are described by mathematical curves. NURBS modeling generally produces smoother surfaces with geometries limited by the curve properties, whereas polygonal models, comprised of many small facets,

can appear coarser but can be built in any conceivable shape, unencumbered by topological limitations [341].

Animation

Animation takes two different approaches: key-framed animation and procedural animation.The concept of animation is inseparable from that of time. The smallest unit of time in animation is the frame: many physically based calculations within a 3D application can rely on arbitrarily divided sub-frames. We can use a timeline, a linear scale divided into equal measures of seconds or frames, to locate key moments in the action. Frames containing these key moments are called key frames, for which values are assigned by the animator or by a script to the attributes being animated.

Dynamics

Many 3D applications have dynamic simulation capabilities that utilize a built-in physics engine. Natural laws of motion can be applied to a model, which has assigned physical properties, to emulate the effects of various forces acting on it. Attributes such as mass, elasticity, and friction are input by the user and elaborated by physics engine. The dynamics capabilities of 3D CG application is very useful in the scientific modeling that we're interested in exploring.

Cameras

Maya and Blender implements the same idea to plan how the events of 3D virtual visualization will be depicted. Camera provides maneuver in the space to get a desired view of the scene, you are actually translating, rotating and zooming. 3D cameras provide orthogonal and perspective views and have many of the attributes of real cameras, such as exposure settings, lens angle, and focal length. These attributes, along with the camera's translation and rotation, can be also animated.

Lights

A digital stage is dark until lights are added. Most 3D applications offer a suite of available lights that mimic those found on a movie set or in a photographer's studio. These include spotlights, area lights, point lights, and infinite lights among others, but vary from application to application. Lights can be colored and assigned a number of attributes that produce special effects such as dappling or a visible beam of light. Shadows cast by lighted objects in a scene can be a very useful device for conveying realism and for emphasizing spatial relationships within a scene (see Figure 76b). CGI shadows may have hard or soft edges and are typically set and adjusted within the controls for a given light.

Blender has several lamps available, including the point lamp, the sun, spot, hemi, and area lights. We describe someone used in `CriMuxnet`:

POINT This light emits in all directions from a single point. With appropriate falloff, it can resemble a candle or a small lightbulb. It is very useful for rim light effects, where parts of an object need to be lit in order to stand out from the background.

SUN Otherwise known as a directional light, this is light that floods a scene from a given angle. Location does not affect sun lights; it is the rotation that is important. Whichever way a sun light is rotated, the whole scene gets light from that particular angle with parallel light rays.

SPOT This is similar to a point lamp, but within a restricted V-shape direction. This light works very much like a theater spotlight. It casts a circle on a surface it is aimed at, and has settings to control the softness of the circular edges.

Shading

In 3D CGI, shading refers to the combined effects of lighting, surface color, surface texture, and geometry, determining the final rendered appearance of a models. Shading is performed during the rendering process by a program called a shader. In Maya, we can do this by creating shading networks-groups of connected render nodes which determine surface properties of models' interaction with light, including color, transparency, and relief and connecting them to geometry and other entities.

Rendering

The production of images from a 3D scene is called rendering, a complex subject which combines the effects of lights, cameras, and shading. The images are saved as individual picture files or as a group in one movie file and can then be displayed in succession using a viewing application or passed along for postproduction work. Render engines support a number of photorealism effects that may be of use in developing a look for an animation projects. Rendering involves many components to produce a final image: shading materials and textures, lighting and shadows, cameras and animation, rendering method, and visual effects. In Maya, rendering can be accomplished using software (Maya Software Renderer, mental ray for Maya Renderer, or the Maya Vector Renderer) or hardware rendering methods. Each type has its distinct advantages. Software rendering generally allows you to create more precise results but can take longer to produce each frame or image. The Maya software render type uses a process for scene illumination that directly illuminates objects in the scene or simulates illumination by reflecting neighboring objects in the scene or via texture mapping the effect on the object. Hardware rendering is generally faster but is less capable of producing detailed results. In addition, some visual effects can only be produced via one method; at times, a combination of more than one type of rendering method may be required. Rendering usually requires several iterations to achieve the final image that meets your requirements. The key is to strike the balance between producing the image that meets your requirements and producing it in the time required [6].

[6] http://docs.autodesk.com/

11 INTERCONNECTED NETWORK FAILURES

We have seen how multilayer network extend the complex network framework, under the assumption that the classical approach does not take into account the possibility that agents can be networked in different ways, and with different intensity, on multiple layers simultaneously.

Moreover, many networks interact with other networks. Consider, for example, the same individuals who participate in different online social networks while they maintain a conspicuous amount of off-line contacts. These individuals are the means by which different social networks interact, so that information can propagate from one network to the others [115].

Interdependent networks are a system in which two or more monoplex networks are connected to each other via *dependency edge*. A classical example is a network constituted by an electrical grid and a computer network where the proper functionality of a router in the computer network can depend on a power station and vice versa [80].

In the same way, *interconnected networks*, *interacting networks*, and *networks of networks* are sets of networks in which some of the nodes from the various networks are adjacent to each other, but the edges that connect different networks need not indicate dependency relations [217, 331]. *Multiplex networks* are the special cases where the same set of nodes appear across different layers while the links within the layers are different.

The existence of such multiple connections on different layers can be generalized by means of multilayer interconnected networked systems, or simply *interconnected networks*.

Interconnected systems have been examined in the engineering literature as a source of cascading failures [129, 181, 230]. Huge literature regards two very relevant, and correlated processes: resilience of the multilayer networks' structure under random failures (and/or cascades of failures), and percolation.

In this Chapter we present a tool for simulating and visualizing failures on interconnected networks. Our goal is to build a visualization system that end-users of complex networking analysts could use to facilitate discovery and increase awareness of their exploration (percolation process, cascading-failure, spreading processes, diffusions and random walks).

11.1 NODE–COLORED NETWORKS

Interconnected networks are equivalent to node-colored networks. We use a formalism developed in the review paper by Kivela et al. [217].

Node-colored networks are graphs in which each node has exactly a color: $G_c = (V_c, E_c, C, \chi)$, where V_c and E_c are the nodes and edges, C is the set of possible 'colors' (where each color is a possible categorical label for the nodes), and $\chi : V \rightarrow C$

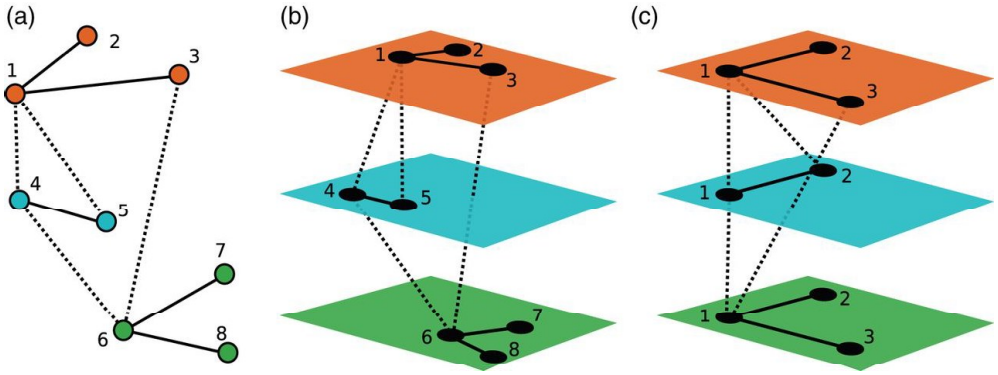


Figure 77: (a) A node-colored network example (i.e. an interconnected network). (b) Representation of the same node-colored network using our multilayer network formalism. (c) Alternative representation of the same node-colored network in Kivela et al. multilayer network formalism. Image taken from [217].

is a function that indicates the color of each node. For multitype networks and heterogeneous networks, the mapping to node-colored networks is obvious, as each type is now called a 'color'. For interdependent networks and networks of networks one needs to map the networks into a flattened graph and then assign colors to nodes according to the subnetwork to which each node belongs [217].

Node-colored graphs can be represented using multilayer-network framework with $d = 1$ by considering each layer as a color. That is, we let $V = V_c$, $L = C$, $V_M = \{(u, c) \in V \times L | \chi(u) = c\}$ and $E_M = \{((u, c_1), (v, c_2)) \in V_M \times V_M | (u, v) \in E_c\}$ (see Figure 77).

11.2 CASCADING FAILURES

Node-colored networks have concentrated mostly on spreading processes, cascading failures and network models.

A cascading-failure process in a multiplex network occurs when two nodes are considered to be in the same mutually-connected component. That is, if there is an intra-layer path between them in all of the intra-layer networks. This process can be compared to that equivalent cascade process in an interdependent network in which nodes that are adjacent to each other via interdependency edges can be merged to create a single node in a multiplex network [217]. For example, when an interdependent network has two layers and each node has exactly an undirected inter-layer dependency edge.

11.3 PERCOLATION

A connected component in an undirected monoplex network is a maximal set of nodes that are all connected to one another via some path. Definition may be extended for multilayer networks by allowing paths that include any of the possible types of

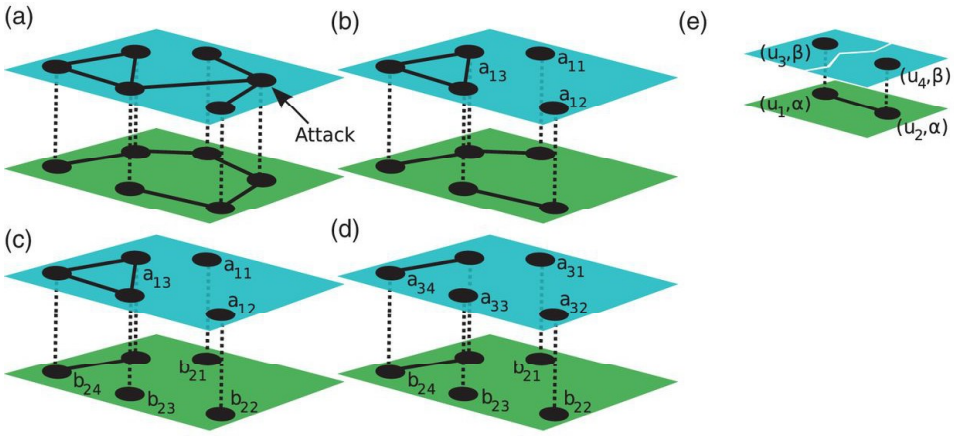


Figure 78: Percolation process on interdependent networks. (a) A node from the top layer is attacked and (b) is removed along with all of its intra-layer edges, from both layers. (c) From the bottom layer, are removed the intra-layer edges that are between nodes that are adjacent to nodes that are now in different components in the top layer and (d) vice versa. This process then continues - alternating between the two layers - and one divides the two networks into progressively smaller components until reaching a stationary state in which the nodes in connected components in each of the layers depend only on nodes that are in the same component in the other layer. (e) Schematic that illustrates the situation of an adjacent pair of nodes in one layer that are adjacent (e.g. via dependency edges) to nodes from different components of another layer. Illustration taken from [217].

edges. It is possible to use generating functions to characterize the component-size distribution for the monoplex configuration model via a mean-field approximation [291].

In site percolation (i.e. node percolation), one lets each node of a network be either occupied or unoccupied, and one constructs occupied nodes as 'operational' and unoccupied nodes as 'nonfunctional'. In bond percolation (i.e. edge percolation), it is instead the edges that are either occupied (i.e. operational) or unoccupied (i.e. nonfunctional). As with connected component, the concept of percolation can be generalized to a multilayer network framework.Usually, percolation processes are formulated in such a way that nodes or edges are removed from the network instead of labelling them as unoccupied. It is also often convenient to use a network diagnostic as a control parameter instead of the fraction of occupied nodes or edges.

In the percolation processes on node-colored networks, intra- and inter-layer edges are semantically equal, and a path that connects two nodes can include both types of edges. Buldyrev et al. [80] defined a cascade process in which intra-layer edges (*connectivity edges*) are defined in the same way as for monoplex networks, but inter-layer edges (*dependency edges*) encode dependencies between nodes. In Figure 78(aâĂŞd), is illustrated percolation process taken from [217] using a multilayer-network framework. In Figure 78(e), is shown an intra-layer edge between a pair of nodes which are adjacent (e.g. via dependency edges) to nodes from different components of another layer. Buldyrev et al. studied multilayer networks with two

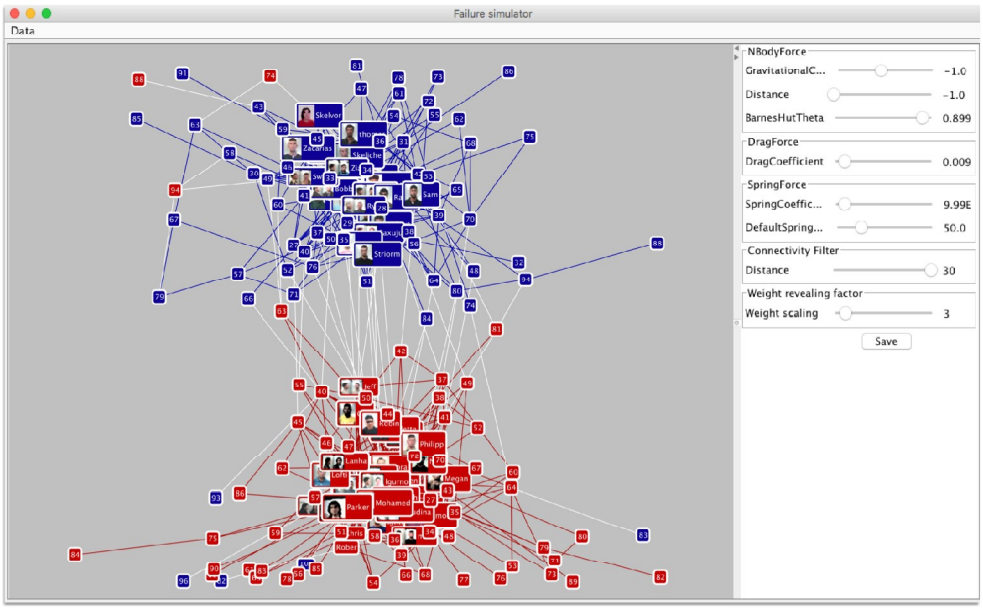


Figure 79: Cascade failure simulation. An interconnected network system with two networks: A and B. Nodes have intranetwork links within their own network but also internetwork links connecting them to the other network.

layers, arbitrary intra-layer degree distributions and inter-layer adjacencies that can exist between a node in one layer and its counterpart in the other layer.

In their cascade process, process starts by removing a fraction $1 - p$ (where $p \in [0, 1]$) of the nodes uniformly at random (panel 78a), and divides the remaining nodes into disjoint sets (panel 78b) according to their connected component in a specific layer. Process then updates the intra-layer network of the other layer by removing intra-layer edges between nodes that are adjacent to nodes from the first layer that are now in different components in that layer (panel 78c). The cascade then continues by removing intra-layer edges in the first layer that are between nodes that depend on nodes from different components in the second layer and by updating accordingly the components of the first layer (panel 78d). This process then continues - alternating between the two layers- and one divides the two networks into progressively smaller components until reaching a stationary state in which the nodes in connected components in each of the layers depend only on nodes that are in the same component in the other layer [217].

11.4 SIMULATION TOOL

Failure simulator is a tool for simulating and visualizing interconnected networks failures. User can manipulate visual data and perform animation. Interactive views are created through a highly-performant rendering system. Main representation layout is the so-called force-directed model. It is computed using the Fruchterman-Reingold layout [165] algorithm, in which nodes repel each other and edges act as

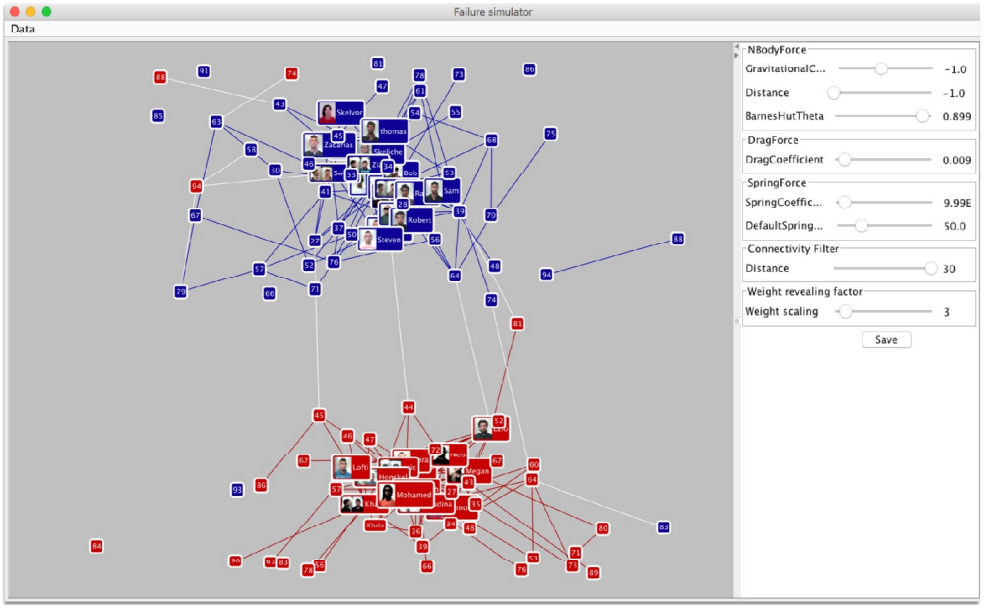


Figure 80: Illustrating interconnected networks after cascade failure simulation.

springs. The consequent displacement of nodes and links shows users clustered in groups which can be identified on the basis of their increase of connectivity. The Barnes-Hut algorithm [29] simulates a N-body repulsive system in order to continuously update position of elements. Optimization of visualization is obtained interactively by modifying parameters relative to tensions of springs that connect nodes. Nodes with minor connectivity have greater tension, thus displacing elements of a group in *orbital* position with respect to the central group to which they connect. It is possible to modify other parameters, e.g. spring constant of force, gravitation force and viscosity/drag of forces.

In our example, each node displays a criminal's name and image. Clicking a node causes the neighbours highliting. Manual panning and zooming are supported; semantic zooming is used to switch to higher resolution images of people when zoomed in.

11.5 MODEL

Two node-colored interconnected networks were used to simulate different cascading failure (see Figure 79). We assume that the two interconnected networks A and B are undirected and unweighted, and that they have the same number of nodes $N = 90$. We can use the tool with an arbitrary number of interconnected networks. Network A has 'red' nodes and friendly intra-layer network relationships, while network B has 'blu' nodes and criminal intra-layer network connections. The white edges are the interconnected connection between this two networks. It is possible to generate networks randomly, too.

Target node	Attack nr.	N_A (red)	N_B (blu)	E_A	E_B	E_{AB}	N_{Tot}	E_{Tot}
Stephen (A)	1	49,44	28,89	74,40	53,74	67,75	39,11	64,12
Stephen (A)	2	38,20	23,33	75,92	48,89	42,50	30,73	61,96
Robert (A)	1	44,44	42,22	79,83	75,27	72,25	45,81	77,32
Robert (A)	2	53,93	35,56	72,45	62,26	70,00	44,69	67,42
Robin (B)	1	41,57	24,44	75,70	49,47	47,5	32,96	61,86
Robin (B)	2	42,70	25,57	76,57	52,45	55,00	34,08	64,02
Parker (B)	1	38,02	38,44	75,05	57,14	40,00	36,31	64,95
Parker (B)	2	44,94	25,56	72,89	43,07	57,50	35,02	57,84
Rashid (A)	1	55,06	27,78	80,48	46,06	60,00	41,34	62,99
Rashid (B)	1	51,69	40,00	81,78	59,49	62,50	45,81	70,22

Table 11: Statistics (in percentage) about cascade failure under selective attack. In the table are indicated target nodes and the attack number to distinguish the repeted simulation for each node in the sample. Other columns indicate: the % of failure nodes on each network (N_A and N_B), the % of failure intra-layer edges (E_A and E_B) and of inter-layer edges (E_{AB}). Finally, they are the percentage of all nodes and all edges failed after simulation completed (N_{Tot} and E_{Tot}).

The simulation of the failure may be executed in two ways: under random failures or intentional attacks. In the former case, user can set a random function that will be automatically executes by attack algorithm. In the latter one can interactively select a node to run a singular simulation.

Visualization tool shows failure effect on the system in real-time, during attacks (see Figure 80). We can see the nodes and edges disappear as the simulation stops. Initial configuration of the network can be restored after each execution process. Simulator allows executions by enabling (animated) or disabling (static) forces.

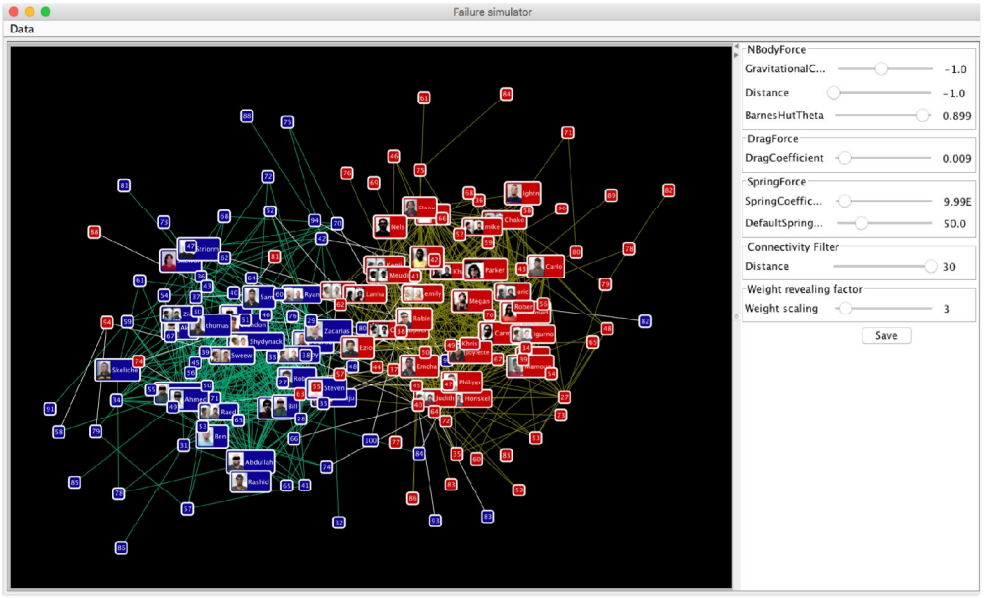
Disruption statistics are shown in the side panel: for all network nodes, for nodes of each network, distinctly, and similarly for the network edges.

In Table 11 are reported some statistics (in percentage) about cascade failure under selective attack. We tested the visualization algorithm and evaluate the effects of attacks, comparing the returned values. In particular, we compared the failures after attack of the same nodes on the same network, and damages afer the attack of the same node on the two networks.

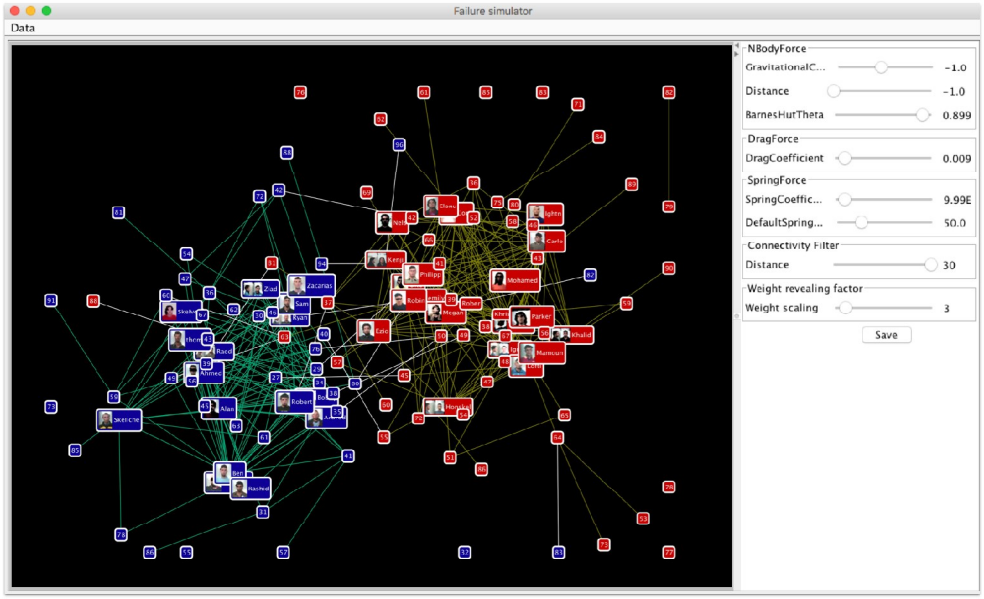
In addition to the browsing mode described above, failure simulator supports a 'blackboard' mode view (see Figure 81). In this way nodes, intra- and inter-layer edges are highlighted to improve readability of the interconnected networks.

11.6 IMPLEMENTATION

Our system is implemented in Java and integrates several open-source toolkits. Prefuse [192] is used for the network visualizations. JUNG [210] provides our underlying node-link data structures. All of the statistic features perform in real-time for the networks example used in this Chapter. The network layout has proved to be effective and stable after each action.



(a)



(b)

Figure 81: Failure simulator 'blackboard' mode view. (a) Illustrating the interconnected networks example before selective attack and (b) after a cascade failure simulation.

BIBLIOGRAPHY

[1] H. Abadinsky. In: *Organized crime*. Ed. by Thomson Wadsworth. 2010 (cit. on p. 54).

[2] E. Acar and B. Yener. "Unsupervised Multiway Data Analysis: A Literature Survey". In: *IEEE Transactions on Knowledge and Data Engineering* 21.1 (2009), pp. 6–20.

[3] E. Adar. "Guess: a language and interface for graph exploration". In: *ACM Conference on Human Factors in Computing Systems*. 2006 (cit. on p. 66).

[4] E. Adar. *GUESS: The Graph Exploration System*. http://graphexploration.cond.org/ (cit. on p. 66).

[5] V. Agarwal et al. "Scalable Graph Exploration on Multicore Processors". In: *Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*. SC '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 1–11. ISBN: 978-1-4244-7559-9. DOI: 10.1109/SC.2010.46. URL: http://dx.doi.org/10.1109/SC.2010.46 (cit. on pp. 139, 141).

[6] A. Ahmed and S. h. Hong. "Navigation techniques for 2.5D graph layout". In: *Visualization, 2007. APVIS '07. 2007 6th International Asia-Pacific Symposium on*. Feb. 2007, pp. 81–84. DOI: 10.1109/APVIS.2007.329279 (cit. on p. 48).

[7] Adel Ahmed et al. "GEOMI: GEOmetry for Maximum Insight". In: *Graph Drawing: 13th International Symposium, GD 2005, Limerick, Ireland, September 12-14, 2005. Revised Papers*. Ed. by Patrick Healy and Nikola S. Nikolov. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 468–479. ISBN: 978-3-540-31667-1. DOI: 10.1007/11618058_42. URL: http://dx.doi.org/10.1007/11618058_42 (cit. on pp. 43, 44).

[8] Y.Y. Ahn et al. "Analysis of topological characteristics of huge online social networking services". In: *Proceedings of the 16th international conference on World Wide Web*. ACM. 2007, pp. 835–844 (cit. on p. 73).

[9] T. Alahakoon et al. "K-path centrality: A new centrality measure in social networks". In: *Proc. of the 4th Workshop on Social Network Systems*. ACM. 2011, p. 1 (cit. on p. 120).

[10] Jay S. Albanese. "Organized Crime (Seventh Edition)". In: *Organized Crime (Seventh Edition)*. Seventh Edition. Anderson Publishing, Ltd., 2015, pp. 1–392. ISBN: 978-0-323-29606-9. DOI: http://dx.doi.org/10.1016/B978-0-323-29606-9.09985-5. URL: http://www.sciencedirect.com/science/article/pii/B9780323296069099855 (cit. on pp. 50, 54).

[11] R. Albert, H. Jeong, and A. Barabási. "Error and attack tolerance of complex networks". In: *Nature* 406.6794 (July 2000), pp. 378–382. ISSN: 00280836. DOI: 10.1038/35019019 (cit. on pp. 57, 64, 120–122, 132).

[12] R. Albert, H. Jeong, and A.L. Barabási. "Internet: Diameter of the world-wide web". In: *Nature* 401.6749 (1999), pp. 130–131. DOI: 10.1038/43601 (cit. on p. 64).

[13] R. Albert, H. Jeong, and A.L. Barabasi. "The diameter of the world wide web". In: *Nature* 401 (1999), pp. 130–131 (cit. on p. 4).

[14] Réka Albert and Albert-László Barabási. "Statistical mechanics of complex networks". In: *Rev. Mod. Phys.* 74.1 (1 Jan. 2002), pp. 47–97. ISSN: 1539-0756. DOI: 10.1103/RevModPhys.74.47. URL: http://link.aps.org/doi/10.1103/RevModPhys.74.47 (cit. on pp. xxv, 1, 7).

[15] J. Alstott, E. Bullmore, and D. Plenz. "powerlaw: a Python package for analysis of heavy-tailed distributions". In: *PloS one* 9.1 (2014), e85777 (cit. on p. 130).

[16] John Arquilla and David Ronfeldt. "Networks and Netwars: The Future of Terror, Crime, and Militancy". In: *Survival* 44.2 (2001), pp. 175–176 (cit. on pp. 61, 95, 97).

[17] J. Assa, D. Cohen-Or, and T. Milo. "Displaying data in multidimensional relevance space with 2D visualization maps". In: *Proc. Visualization '97*. 1997, pp. 127–134 (cit. on p. 47).

[18] David Auber. "Tulip — A Huge Graph Visualization Framework". In: *Graph Drawing Software*. Ed. by Michael Jünger and Petra Mutzel. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 105–126. ISBN: 978-3-642-18638-7. DOI: 10.1007/978-3-642-18638-7_5. URL: http://dx.doi.org/10.1007/978-3-642-18638-7_5 (cit. on p. 66).

[19] Julie Ayling. "Criminal organizations and resilience". In: *International Journal of Law, Crime and Justice* 37.4 (2009), pp. 182 –196. ISSN: 1756-0616. DOI: 10.1016/j.ijlcj.2009.10.003 (cit. on pp. 52, 62–64).

[20] Mariagiovanna Baccara and Heski Bar-Isaac. "Interrogation Methods and Terror Networks". In: *Mathematical Methods in Counterterrorism*. Ed. by Nasrullah Memon et al. Vienna: Springer Vienna, 2009, pp. 271–290. ISBN: 978-3-211-09442-6. DOI: 10.1007/978-3-211-09442-6_16. URL: http://dx.doi.org/10.1007/978-3-211-09442-6_16 (cit. on p. xxvi).

[21] D. A. Bader and K. Madduri. "Designing Multithreaded Algorithms for Breadth-First Search and st-connectivity on the Cray MTA-2". In: *2006 International Conference on Parallel Processing (ICPP'06)*. Aug. 2006, pp. 523–530. DOI: 10.1109/ICPP.2006.34 (cit. on pp. 139–141, 144).

[22] W. Baker and R Faulkner. "The social organization of conspiracy: illegal networks in the heavy electrical equipment industry". In: *Am. Social. Rev.* 58.6 (1993), pp. 837–860. DOI: 10.2307/2095954 (cit. on pp. 61, 97).

[23] Eytan Bakshy et al. "The role of social networks in information diffusion". In: *Proceedings of the 21st international conference on World Wide Web*. ACM. 2012, pp. 519–528 (cit. on p. 95).

[24] C. Ballester, A. Calvó-Armengol, and Y. Zenou. "Who's who in networks. wanted: the key player". In: *Econometrica* 74.5 (2006), pp. 1403–1417 (cit. on p. 118).

[25] Michael Balzer, Oliver Deussen, and Claus Lewerentz. "Voronoi Treemaps for the Visualization of Software Metrics". In: *Proceedings of the 2005 ACM Symposium on Software Visualization*. SoftVis '05. New York, NY, USA: ACM, 2005, pp. 165–172. ISBN: 1-59593-073-6. DOI: 10.1145/1056018.1056041. URL: http://doi.acm.org/10.1145/1056018.1056041 (cit. on pp. 38, 39).

[26] JÃÿrgen Bang-Jensen and Gregory Gutin. *Digraphs - theory, algorithms and applications*. Springer, 2002, pp. I–XXII, 1–754. ISBN: 978-1-85233-611-0.

[27] A. L. Barabasi and R. Albert. "Emergence of scaling in random networks". In: *Science* 286.5439 (1999), pp. 509–512 (cit. on pp. xxv, 1, 4, 121).

[28] Matteo Barigozzi, Giorgio Fagiolo, and Giuseppe Mangioni. "Identifying the Community Structure of the International-Trade Multi Network". In: *CoRR* (2010).

[29] J. Barnes and P. Hut. "A Hierarchical O(N log N) Force Calculation Algorithm". In: *Nature* 324 (1986), pp. 446–449 (cit. on pp. 36, 173).

[30] A. Barrat et al. "The architecture of complex weighted networks". In: *Proc. Natl. Acad. Sci.* 101 (2004), pp. 3747–3752.

[31] Louise Barrett, S. Peter Henzi, and David Lusseau. "Taking sociality seriously: the structure of multi-dimensional social networks as a source of information for individuals". In: *Philosophical Transactions of the Royal Society of London B: Biological Sciences* 367.1599 (2012), pp. 2108–2118. ISSN: 0962-8436. DOI: 10.1098/rstb.2012.0113. eprint: http://rstb.royalsocietypublishing.org/content/367/1599/2108.full.pdf. URL: http://rstb.royalsocietypublishing.org/content/367/1599/2108 (cit. on p. 27).

[32] Gereon Bartel et al. "An Experimental Evaluation of Multilevel Layout Methods". In: *Graph Drawing: 18th International Symposium, GD 2010, Konstanz, Germany, September 21-24, 2010. Revised Selected Papers*. Ed. by Ulrik Brandes and Sabine Cornelsen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 80–91. ISBN: 978-3-642-18469-7. DOI: 10.1007/978-3-642-18469-7_8. URL: http://dx.doi.org/10.1007/978-3-642-18469-7_8 (cit. on p. 37).

[33] Vladimir Batagelj. "Notes on Blockmodeling". In: *Social Networks* 19 (1997), pp. 143–155.

[34] Vladimir Batagelj, Anuska Ferligoj, and Patrick Doreian. "Generalized Blockmodeling." In: *Informatica (Slovenia)* 23.4 (1999).

[35] Vladimir Batagelj and Andrej Mrvar. "Pajek— Analysis and Visualization of Large Networks". In: *Graph Drawing: 9th International Symposium, GD 2001 Vienna, Austria, September 23–26, 2001 Revised Papers*. Ed. by Petra Mutzel, Michael Jünger, and Sebastian Leipert. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 477–478. ISBN: 978-3-540-45848-7. DOI: 10.1007/3-540-45848-4_54. URL: http://dx.doi.org/10.1007/3-540-45848-4_54 (cit. on p. 66).

[36] Giuseppe Di Battista et al. *Graph Drawing: Algorithms for the Visualization of Graphs*. 1st. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1998. ISBN: 0133016153 (cit. on pp. 31, 33, 35–37, 47, 66).

[37] Federico Battiston, Vincenzo Nicosia, and Vito Latora. "Structural measures for multiplex networks". In: *Phys. Rev. E* 89 (3 Mar. 2014), p. 032804. DOI: 10.1103/PhysRevE.89.032804. URL: http://link.aps.org/doi/10.1103/PhysRevE.89.032804 (cit. on pp. xxv, 5, 23, 27, 61, 124).

[38] Federico Battiston, Vincenzo Nicosia, and Vito Latora. "The new challenges of multiplex networks: measures and models". In: *CoRR* abs/1606.09221 (2016). URL: http://arxiv.org/abs/1606.09221 (cit. on pp. xxv, 23–27).

[39] Michael Baur et al. "Drawing the AS Graph in 2.5 Dimensions". In: *Graph Drawing: 12th International Symposium, GD 2004, New York, NY, USA, September 29-October 2, 2004, Revised Selected Papers*. Ed. by János Pach. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 43–48. ISBN: 978-3-540-31843-9. DOI: 10.1007/978-3-540-31843-9_6. URL: http://dx.doi.org/10.1007/978-3-540-31843-9_6 (cit. on p. 41).

[40] A. Bavelas. "A mathematical model for group structures". In: *Human organization* 7.3 (1948), pp. 16–30 (cit. on p. 120).

[41] Gareth J. Baxter et al. "Weak percolation on multiplex networks". In: *Phys. Rev. E* 89 (4 Apr. 2014), p. 042801. DOI: 10.1103/PhysRevE.89.042801. URL: http://link.aps.org/doi/10.1103/PhysRevE.89.042801 (cit. on p. xxv).

[42] Marya Bazzi et al. "Community Detection in Temporal Multilayer Networks, with an Application to Correlation Networks". In: *Multiscale Modeling & Simulation* 14.1 (2016), pp. 1–41. DOI: 10.1137/15M1009615. URL: http://dx.doi.org/10.1137/15M1009615.

[43] Richard Becker et al. "Human mobility characterization from cellular network data". In: *Communications of the ACM* 56.1 (2013), pp. 74–82 (cit. on pp. 52, 73).

[44] F. Benevenuto et al. "Characterizing user behavior in online social networks". In: *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*. ACM. 2009, pp. 49–62 (cit. on p. 73).

[45] Chris Bennett et al. "The Aesthetics of Graph Visualization". In: *Proceedings of the Third Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*. Computational Aesthetics'07. Alberta, Canada: Eurographics Association, 2007, pp. 57–64. ISBN: 978-3-905673-43-2. DOI: 10.2312/COMPAESTH/COMPAESTH07/057-064. URL: http://dx.doi.org/10.2312/COMPAESTH/COMPAESTH07/057-064 (cit. on p. 33).

[46] Michele Berlingerio, Michele Coscia, and Fosca Giannotti. "Finding redundant and complementary communities in multidimensional networks." In: *CIKM*. Ed. by Craig Macdonald, Iadh Ounis, and Ian Ruthven. ACM, 2011, pp. 2181–2184. ISBN: 978-1-4503-0717-8.

[47] Michele Berlingerio et al. "Foundations of Multidimensional Network Analysis." In: *ASONAM*. IEEE Computer Society, 2011, pp. 485–489.

[48] Rudolf Berrendorf and Mathias Makulla. "Level-Synchronous Parallel Breadth-First Search Algorithms For Multicore and Multiprocessor Systems". In: *Nygard, Tamir (Eds.): Future Computing 2014, the Sixth International Conference on Future Computational Technologies and Applications. Venice, Italy, May 25-29, 2014*. 2014, pp. 26–31. ISBN: 978-1-61208-339-1 (cit. on p. 141).

[49] B. Betkaoui et al. "A Reconfigurable Computing Approach for Efficient and Scalable Parallel Graph Exploration". In: *2012 IEEE 23rd International Conference on Application-Specific Systems, Architectures and Processors*. July 2012, pp. 8–15. DOI: 10.1109/ASAP.2012.30 (cit. on p. 141).

[50] Ginestra Bianconi. "Statistical mechanics of multiplex networks: Entropy and overlap". In: *Phys. Rev. E* 87 (6 June 2013), p. 062806. DOI: 10.1103/PhysRevE.87.062806. URL: http://link.aps.org/doi/10.1103/PhysRevE.87.062806 (cit. on pp. xxv, 26).

[51] Eric A. Bier et al. "Toolglass and Magic Lenses: The See-through Interface". In: *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '93. New York, NY, USA: ACM, 1993, pp. 73–80. ISBN: 0-89791-601-8. DOI: 10.1145/166117.166126. URL: http://doi.acm.org/10.1145/166117.166126 (cit. on p. 46).

[52] V.D. Blondel et al. "Fast unfolding of communities in large networks". In: *Journal of Statistical Mechanics: Theory and Experiment* 2008.10 (July 2008), P10008. ISSN: 1742-5468 (cit. on pp. 59, 101, 107).

[53] S. Boccaletti et al. "Complex Networks : Structure and Dynamics". In: *Phys. Rep.* 424.4-5 (Feb. 2006), pp. 175–308 (cit. on pp. xxv, 1, 9, 61).

[54] S. Boccaletti et al. "The structure and dynamics of multilayer networks". In: *Physics Reports* 544.1 (2014). The structure and dynamics of multilayer networks, pp. 1 –122. ISSN: 0370-1573. DOI: http://dx.doi.org/10.1016/j.physrep.2014.07.001. URL: http://www.sciencedirect.com/science/article/pii/S0370157314002105 (cit. on pp. xxv, 11, 23, 25).

[55] Stefano Boccaletti et al. "Introduction to Focus Issue: Complex Dynamics in Networks, Multilayered Structures and Systems". In: *Chaos* 26.6, 065101 (2016). DOI: http://dx.doi.org/10.1063/1.4953595.

[56] Andrey Bogomolov et al. "Once Upon a Crime: Towards Crime Prediction from Demographics and Mobile Data". In: *arXiv preprint arXiv:1409.2983* (2014) (cit. on p. 95).

[57] Bela Bollobas. *Modern Graph Theory*. New York: Springer, 1998 (cit. on p. 1).

[58] Phillip Bonacich. "Technique for analyzing overlapping memberships". In: *Sociological Methodology* 4 (1972), pp. 176–185 (cit. on p. 16).

[59] S. A. Boorman and H. C. White. "Social structure from multiple networks II. Role structures". In: *American Journal of Sociology* 81.6 (1976), pp. 1384–1446.

[60] S. Borgatti, M. G. Everett, and L. C. Freeman. *UCINET*. Analytic Technologies, 2007 (cit. on p. 66).

[61] Stephen Borgatti. "The Key Player Problem". In: *Proceedings of CASOS 2002 Conference*. National Academies Press. Pittsburgh, PA, 2002, p. 241. DOI: 10.17226/10735 (cit. on pp. xxvii, 65).

[62] StephenP. Borgatti. "Identifying sets of key players in a social network". English. In: *Computational and Mathematical Organization Theory* 12.1 (2006), pp. 21–34. ISSN: 1381-298X (cit. on p. 118).

[63] Martin Bouchard. "On the resilience of illegal drug markets". In: *Global crime* 8.4 (2007), pp. 325–344. DOI: 10.1080/17440570701739702 (cit. on pp. 63, 64).

[64] J. Bouttier, P. Di Francesco, and E. Guitter. "Geodesic distance in planar graphs". In: *Nuclear Physics B* 663.3 (2003), pp. 535–567.

[65] U. Brandes. "A faster algorithm for betweenness centrality". In: *Journal of Mathematical Sociology* 25.2 (2001), pp. 163–177 (cit. on pp. 49, 97).

[66] Ulrik Brandes. "Drawing on physical analogies". English. In: *Drawing Graphs*. Ed. by Michael Kaufmann and Dorothea Wagner. Vol. 2025. Lecture Notes in Computer Science. Springer, 2001, pp. 71–86. ISBN: 978-3-540-42062-0. DOI: 10.1007/3-540-44969-8_4 (cit. on pp. 37, 67).

[67] Ulrik Brandes. "On Variants of Shortest-Path Betweenness Centrality and their Generic Computation". In: *SOCIAL NETWORKS* 30.2 (2008) (cit. on p. 5).

[68] Ulrik Brandes, Tim Dwyer, and Falk Schreiber. "Visualizing Related Metabolic Pathways in Two and a Half Dimensions". In: *Graph Drawing: 11th International Symposium, GD 2003 Perugia, Italy, September 21-24, 2003 Revised Papers*. Ed. by Giuseppe Liotta. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 111–122. ISBN: 978-3-540-24595-7. DOI: 10.1007/978-3-540-24595-7_10. URL: http://dx.doi.org/10.1007/978-3-540-24595-7_10 (cit. on pp. 41, 66).

[69] Ulrik Brandes et al. "Explorations into the Visualization of Policy Networks". In: *Journal of Theoretical Politics* 11.1 (1999), pp. 75–106. DOI: 10.1177/0951692899011001004.

[70] David W. Brannan, Philip F. Esler, and N. T. Anders Strindberg. "Talking to Terrorists: Towards an Independent Analytical Framework for the Study of Violent Substate Activism". In: *Studies in Conflict and Terrorism* 24.1 (2001), pp. 3–24 (cit. on pp. 61, 97).

[71] Ronald L. Breiger and Philippa E. Pattison. "Cumulated social roles: The duality of persons and their algebras". In: *Social Networks* 8.3 (Sept. 1986), pp. 215–256.

[72] H. Brinton Milward and Jorg Raab. "Dark Networks as Organizational Problems: Elements of a Theory 1". In: *International Public Management Journal* 9.3 (2006), pp. 333–360. DOI: 10.1080/10967490600899747 (cit. on pp. 62, 64).

[73] South Bank Brisbane. *The network approach to evaluation: uncovering patterns, possibilities and pitfalls*. 2005 (cit. on p. 56).

[74] A. Broder et al. "Graph structure in the Web". In: *Computer Networks* 33.1 (2000), pp. 309–320 (cit. on pp. 121, 122, 132).

[75] Piotr Bródka, Katarzyna Musial, and Przemysław Kazienko. "A Method for Group Extraction in Complex Social Networks". In: *Knowledge Management, Information Systems, E-Learning, and Sustainability Research: Third World Summit on the Knowledge Society, WSKS 2010, Corfu, Greece, September 22-24, 2010. Proceedings, Part I*. Ed. by Miltiadis D. Lytras et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 238–247. ISBN: 978-3-642-16318-0. DOI: 10.1007/978-3-642-16318-0_27. URL: http://dx.doi.org/10.1007/978-3-642-16318-0_27 (cit. on p. 27).

[76] Mark Bruls, Kees Huizing, and Jarke J. van Wijk. "Squarified Treemaps". In: *Data Visualization 2000: Proceedings of the Joint EUROGRAPHICS and IEEE TCVG Symposium on Visualization in Amsterdam, The Netherlands, May 29–30, 2000*. Ed. by Willem Cornelis de Leeuw and Robert van Liere. Vienna: Springer Vienna, 2000, pp. 33–42. ISBN: 978-3-7091-6783-0. DOI: 10.1007/978-3-7091-6783-0_4. URL: http://dx.doi.org/10.1007/978-3-7091-6783-0_4 (cit. on p. 38).

[77] Charles D. Brummitt, Raissa M. D'Souza, and E. A. Leicht. "Suppressing cascades of load in interdependent networks". In: *Proceedings of the National Academy of Sciences* 109.12 (2012), E680–E689. DOI: 10.1073/pnas.1110586109. eprint: http://www.pnas.org/content/109/12/E680.full.pdf. URL: http://www.pnas.org/content/109/12/E680.abstract (cit. on p. xxvi).

[78] Charles D. Brummitt, Kyu-Min Lee, and K.-I. Goh. "Multiplexity-facilitated cascades in networks". In: *Phys. Rev. E* 85 (4 Apr. 2012), p. 045102. DOI: 10.1103/PhysRevE.85.045102. URL: http://link.aps.org/doi/10.1103/PhysRevE.85.045102 (cit. on p. xxv).

[79] Piotr BrÃşdka et al. "Analysis of Neighbourhoods in Multi-layered Dynamic Social Networks". In: *International Journal of Computational Intelligence Systems* 5.3 (2012), pp. 582–596. DOI: 10.1080/18756891.2012.696922. eprint: http://dx.doi.org/10.1080/18756891.2012.696922. URL: http://dx.doi.org/10.1080/18756891.2012.696922 (cit. on p. 27).

[80] Sergey V. Buldyrev et al. "Catastrophic cascade of failures in interdependent networks". In: *Nature* 464.7291 (Apr. 2010), pp. 1025–1028. URL: http://dx.doi.org/10.1038/nature08932 (cit. on pp. xxv, xxvi, xxxi, 169, 171).

[81] Ed Bullmore and Olaf Sporns. "Complex brain networks: graph theoretical analysis of structural and functional systems". In: *Nat Rev Neurosci* 10.3 (Mar. 2009), pp. 186–198. URL: http://dx.doi.org/10.1038/nrn2575 (cit. on p. xxv).

[82] Aydin Buluç and Kamesh Madduri. "Parallel Breadth-first Search on Distributed Memory Systems". In: *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*. SC '11. New York, NY, USA: ACM, 2011, 65:1–65:12. ISBN: 978-1-4503-0771-0. DOI: 10.1145/2063384.2063471. URL: http://doi.acm.org/10.1145/2063384.2063471 (cit. on pp. 139, 141, 144).

[83] Dennis Callahan et al. "Shaping operations to attack robust terror networks". In: *Social Informatics (SocialInformatics), 2012 International Conference on*. IEEE. 2012, pp. 13–18. DOI: 10.1109/SocialInformatics.2012.22 (cit. on p. 65).

[84] Duncan S Callaway et al. "Network Robustness and Fragility: Percolation on Random Graphs". In: *Phys. Rev. Lett.* 85.25 (Dec. 2000), pp. 5468–5471. DOI: 10.1103/PhysRevLett.85.5468 (cit. on p. 64).

[85] J. Candia et al. "Uncovering individual and collective human dynamics from mobile phone records". In: *Journal of Physics A: Mathematical and Theoretical* 41 (2008), p. 224015 (cit. on p. 73).

[86] D.V. Canter and L.J. Alison. *The Social Psychology of Crime: Groups, Teams, and Networks*. Offender Profiling Series, Vol. 111. Ashgate, 2000. ISBN: 9781840144970 (cit. on p. 62).

[87] Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman, eds. *Readings in Information Visualization: Using Vision to Think*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999. ISBN: 1-55860-533-9 (cit. on pp. 31, 34, 35).

[88] Alessio Cardillo et al. "Emergence of network features from multiplexity". In: *Scientific Report* abs/1212.2153 (2012). Ed. by Macmillan Publishers Limited. URL: http://dblp.uni-trier.de/db/journals/corr/corr1212.html#abs-1212-2153 (cit. on p. xxvi).

[89] Alessio Cardillo et al. "Modeling the multi-layer nature of the European Air Transport Network: Resilience and passengers re-scheduling under random failures". In: *The European Physical Journal Special Topics* 215.1 (2013), pp. 23–33. ISSN: 1951-6355. DOI: 10.1140/epjst/e2013-01712-8. URL: http://dx.doi.org/10.1140/epjst/e2013-01712-8.

[90] Kathleen M. Carley. "Destabilization of covert networks". In: *Computational & Mathematical Organization Theory* 12.1 (Apr. 2006), pp. 51–66. ISSN: 1572-9346. DOI: 10.1007/s10588-006-7083-y (cit. on pp. xxvii, 57, 58, 60, 61).

[91] Eoghan Casey. *Digital evidence and computer crime: forensic science, computers and the internet*. Academic press, 2011 (cit. on p. 95).

[92] Claudio Castellano, Santo Fortunato, and Vittorio Loreto. "Statistical physics of social dynamics". In: *Reviews of modern physics* 81.2 (2 May 2009), p. 591. DOI: 10.1103/RevModPhys.81.591. URL: http://link.aps.org/doi/10.1103/RevModPhys.81.591 (cit. on pp. 61, 121).

[93] Salvatore Catanese, Emilio Ferrara, and Giacomo Fiumara. "Forensic analysis of phone call networks". In: *Social Network Analysis and Mining* 3.1 (2013), pp. 15–33 (cit. on pp. 95, 96, 98, 100, 117, 128, 161).

[94] Salvatore Catanese et al. "Analyzing the Facebook Friendship Graph". In: *Proceedings of the 1st Workshop in Mining the Future Internet*. 2010, pp. 1–6.

[95] Salvatore Catanese et al. "Extraction and analysis of Facebook friendship relations". In: *Computational Social Networks*. Springer, 2012, pp. 291–324 (cit. on p. 73).

[96] Salvatore A Catanese et al. "Crawling facebook for social network analysis purposes". In: *Proceedings of the international conference on web intelligence, mining and semantics*. WIMS '11. ACM. New York, NY, USA: ACM, 2011, p. 52. ISBN: 978-1-4503-0148-0. DOI: 10.1145/1988688.1988749. URL: http://doi.acm.org/10.1145/1988688.1988749 (cit. on pp. 73, 115, 131, 139).

[97] Salvatore Amato Catanese and Giacomo Fiumara. "A Visual Tool for Forensic Analysis of Mobile Phone Traffic". In: *Proceedings of the 2Nd ACM Workshop on Multimedia in Forensics, Security and Intelligence*. MiFor '10. Firenze, Italy: ACM, 2010, pp. 71–76. ISBN: 978-1-4503-0157-2. DOI: 10.1145/1877972.1877992. URL: http://doi.acm.org/10.1145/1877972.1877992 (cit. on pp. 139, 161).

[98] B. Cayli. "Italian civil society against the Mafia: From perceptions to expectations". In: *International Journal of Law, Crime and Justice* 41.1 (2013), pp. 81–99 (cit. on p. 113).

[99] Barbara Chapman, Gabriele Jost, and Ruud van der Pas. *Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation)*. The MIT Press, 2007. ISBN: 0262533022, 9780262533027 (cit. on p. 142).

[100] H. Chen et al. "Crime data mining: a general framework and some examples". In: *Computer* 37.4 (2004), pp. 50–56 (cit. on p. 117).

[101] Hsinchun Chen et al. "COPLINK: managing law enforcement data and knowledge". In: *Communications of the ACM* 46.1 (2003), pp. 28–34 (cit. on pp. xxvii, 75, 98).

[102] Ed Huai-hsin Chi and John Riedl. "An Operator Interaction Framework for Visualization Systems". In: *Proceedings of the 1998 IEEE Symposium on Information Visualization*. INFOVIS '98. Washington, DC, USA: IEEE Computer Society, 1998, pp. 63–70. ISBN: 0-8186-9093-3. URL: http://dl.acm.org/citation.cfm?id=647341.721078 (cit. on p. 34).

[103] Fabio Ciulla et al. "Beating the news using Social Media: the case study of American Idol". In: *EPJ Data Science* 1.1 (2012), p. 8 (cit. on p. 95).

[104] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. "Finding community structure in very large networks". In: *Phys. Rev. E* 70 (6 Dec. 2004), p. 066111. DOI: 10.1103/PhysRevE.70.066111. URL: http://link.aps.org/doi/10.1103/PhysRevE.70.066111 (cit. on p. 139).

[105] Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman. "Power-law distributions in empirical data". In: (2009) (cit. on p. 1).

[106] Reuven Cohen et al. "Resilience of the Internet to Random Breakdowns". In: *Phys. Rev. Lett.* 85 (21 Nov. 2000), pp. 4626–4628. DOI: 10.1103/PhysRevLett.85.4626. URL: http://link.aps.org/doi/10.1103/PhysRevLett.85.4626 (cit. on p. 64).

[107] Christopher Collins, Sheelagh Carpendale, and Gerald Penn. "DocuBurst: Visualizing Document Content using Language Structure". In: *Computer Graphics Forum* (2009). ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2009.01439.x (cit. on p. 39).

[108] Michael D Conover et al. "Partisan Asymmetries in Online Political Activity". In: *EPJ Data Science* 1 (2012), p. 6 (cit. on p. 95).

[109] Michael D Conover et al. "The Digital Evolution of Occupy Wall Street". In: *PloS one* 8.5 (2013), e64679 (cit. on p. 73).

[110] Michael D Conover et al. "The geospatial characteristics of a social movement communication network". In: *PloS one* 8.3 (2013), e55957 (cit. on pp. 73, 95).

[111] Thomas H. Cormen et al. *Introduction to Algorithms (3. ed.)* MIT Press, 2009, pp. I–XIX, 1–1292. ISBN: 978-0-262-03384-8 (cit. on pp. 2, 3, 139, 140, 142).

[112] CarlosD. Correa and Kwan-Liu Ma. "Visualizing Social Networks". In: *Social Network Data Analytics*. Ed. by Charu C. Aggarwal. Springer US, 2011, pp. 307–326. DOI: 10.1007/978-1-4419-8462-3 (cit. on p. 66).

[113] Michele Coscia et al. "You Know Because I Know": a Multidimensional Network Approach to Human Resources Problem". In: *ASONAM 2013*. 2013.

[114] Emanuele Cozzo et al. "Contact-based Social Contagion in Multiplex Networks". In: *CoRR* abs/1307.1656 (2013).

[115] Emanuele Cozzo et al. "Multilayer Networks: Metrics and Spectral Properties". In: *Interconnected Networks*. Ed. by Antonios Garas. Cham: Springer International Publishing, 2016, pp. 17–35. ISBN: 978-3-319-23947-7. DOI: 10.1007/978-3-319-23947-7_2. URL: http://dx.doi.org/10.1007/978-3-319-23947-7_2 (cit. on p. 169).

[116] Emanuele Cozzo et al. "Structure of triadic relations in multiplex networks". In: *New Journal of Physics* 17.7 (2015), p. 073029. URL: http://stacks.iop.org/1367-2630/17/i=7/a=073029 (cit. on pp. 27, 28).

[117] R. Cross, S. P. Borgatti, and A. Parker. "Making invisible work visible: Using social network analysis to support strategic collaboration". In: *California Management Review* 44.2 (2002), pp. 25–46. DOI: 10.2307/41166121 (cit. on p. 55).

[118] Isabel F. Cruz and Joseph P. Twarog. "3D Graph Drawing with Simulated Annealing". In: *Proceedings of the Symposium on Graph Drawing*. GD '95. London, UK, UK: Springer-Verlag, 1996, pp. 162–165. ISBN: 3-540-60723-4. URL: http://dl.acm.org/citation.cfm?id=647547.728601 (cit. on p. 40).

[119] Ron Davidson and David Harel. "Drawing Graphs Nicely Using Simulated Annealing". In: *ACM Trans. Graph.* 15.4 (Oct. 1996), pp. 301–331. ISSN: 0730-0301. DOI: 10.1145/234535.234538. URL: http://doi.acm.org/10.1145/234535.234538 (cit. on p. 40).

[120] Manlio De Domenico, Mason A. Porter, and Alex Arenas. "MuxViz: a tool for multilayer analysis and visualization of networks". In: *Journal of Complex Networks* 3.2 (2015), pp. 159–176. DOI: 10.1093/comnet/cnu038. URL: http://comnet.oxfordjournals.org/content/3/2/159.abstract (cit. on pp. 44, 45, 154).

[121] Manlio De Domenico et al. "Mathematical Formulation of Multilayer Networks". In: *Phys. Rev. X* 3 (4 Dec. 2013), p. 041022. DOI: 10.1103/PhysRevX.3.041022. URL: http://link.aps.org/doi/10.1103/PhysRevX.3.041022 (cit. on pp. xxv, 11, 13, 22, 23, 29).

[122] Manlio De Domenico et al. "Navigability of interconnected networks under random failures". In: *Proceedings of the National Academy of Sciences* 111.23 (2014), pp. 8351–8356. DOI: 10.1073/pnas.1318469111. eprint: http://www.pnas.org/content/111/23/8351.full.pdf. URL: http://www.pnas.org/content/111/23/8351.abstract (cit. on p. 29).

[123] Manlio De Domenico et al. "Structural reducibility of multilayer networks". In: *Nature Communications* 6 (Apr. 2015), 6864 EP –. URL: http://dx.doi.org/10.1038/ncomms7864 (cit. on p. 69).

[124] Pasquale De Meo et al. "A novel measure of edge centrality in social networks". In: *Knowledge-based systems* 30 (2012), pp. 136–150 (cit. on p. 120).

[125] Pasquale De Meo et al. "Enhancing community detection using a network weighting strategy". In: *Information Sciences* 222 (2013), pp. 648–668 (cit. on pp. 107, 120).

[126] Pasquale De Meo et al. "Generalized Louvain method for community detection in large networks". In: *Proc. 11th International Conference on Intelligent Systems Design and Applications*. IEEE. 2011, pp. 88–93 (cit. on p. 107).

[127] Pasquale De Meo et al. "Mixing local and global information for community detection in large networks". In: *Journal of Computer and System Sciences* 80.1 (2014), pp. 72–87 (cit. on p. 107).

[128] Pasquale De Meo et al. "On Facebook, Most Ties Are Weak". In: *Commun. ACM* 57.11 (Oct. 2014), pp. 78–84. ISSN: 0001-0782. DOI: 10.1145/2629438. URL: http://doi.acm.org/10.1145/2629438 (cit. on p. 95).

[129] Mark Dickison, S. Havlin, and H. E. Stanley. "Epidemics on interconnected networks". In: *Phys. Rev. E* 85 (6 June 2012), p. 066109. DOI: 10.1103/PhysRevE.85.066109. URL: http://link.aps.org/doi/10.1103/PhysRevE.85.066109 (cit. on pp. xxvi, 169).

[130] E. W. Dijkstra. "A note on two problems in connexion with graphs". In: *Numerische Mathematik* 1.1 (1959), pp. 269–271. ISSN: 0945-3245. DOI: 10.1007/BF01386390. URL: http://dx.doi.org/10.1007/BF01386390 (cit. on pp. 2, 139).

[131] Manlio De Domenico et al. "Centrality in Interconnected Multilayer Networks". In: *CoRR* abs/1311.2906 (2013). URL: http://arxiv.org/abs/1311.2906 (cit. on pp. 15, 17, 68).

[132] Manlio De Domenico et al. "Random Walks on Multiplex Networks." In: *CoRR* (2013).

[133] J. F. Donges et al. "Investigating the topology of interacting networks". In: *The European Physical Journal B* 84.4 (2011), pp. 635–651. ISSN: 1434-6036. DOI: 10.1140/epjb/e2011-10795-8. URL: http://dx.doi.org/10.1140/epjb/e2011-10795-8 (cit. on p. xxvi).

[134] Maria R. D'Orsogna and Matjaž Perc. "Statistical physics of crime: A review". In: *Physics of life reviews* 12 (2015), pp. 1–21. ISSN: 1571-0645. DOI: http://dx.doi.org/10.1016/j.plrev.2014.11.001. URL: http://www.sciencedirect.com/science/article/pii/S1571064514001730 (cit. on pp. xxvii, 61, 118).

[135] R. Drezewski, J. Sepielak, and W. Filipkowski. "The application of social network analysis algorithms in a system supporting money laundering detection". In: *Information Sciences* 295 (2015), pp. 18–32 (cit. on p. 117).

[136] "Mixing Patterns and Community Structure in Networks". In: *The Globalization of Crime: A Transnational Organized Crime Threat Assessment*. Ed. by UN Office on Drugs and Crime (UNODC). UN Office on Drugs and Crime (UNODC), 2010. ISBN: 978-92-1-130295-0. URL: http://www.refworld.org/docid/4cad7f892.html (cit. on pp. 50, 51).

[137] Jordi Duch and Alex Arenas. "Community detection in complex networks using extremal optimization". In: *Phys. Rev. E* 72 (2 Aug. 2005), p. 027104. DOI: 10.1103/PhysRevE.72.027104. URL: http://link.aps.org/doi/10.1103/PhysRevE.72.027104 (cit. on p. 139).

[138] P.A.C. Duijn, V. Kashirin, and P.M.A. Sloot. "The Relative Ineffectiveness of Criminal Network Disruption". In: *Sci. Rep.* 4 (2014), p. 4238. DOI: 10.1038/srep04238. URL: http://www.nature.com/srep/2014/140228/srep04238/abs/srep04238.html (cit. on pp. 62, 63, 65).

[139] Daniel M. Dunlavy et al. "Multilinear algebra for analyzing data with multiple linkages". In: *Graph Algorithms in the Language of Linear Algebra*. Ed. by Jeremy Kepner and John Gilbert. Fundamentals of Algorithms. Philadelphia: SIAM, 2011, pp. 85–114.

[140] Jennifer A. Dunne, Richard J. Williams, and Neo D. Martinez. "Food-web structure and network theory: The role of connectance and size". In: *Proceedings of the National Academy of Sciences* 99.20 (2002), pp. 12917–12922. DOI: 10.1073/pnas.192407699 (cit. on p. 64).

[141] Jennifer A. Dunne, Richard J. Williams, and Neo D. Martinez. "Network structure and biodiversity loss in food webs: robustness increases with connectance". In: *Ecology Letters* 5.4 (July 2002), pp. 558–567. DOI: 10.1046/j.1461-0248.2002.00354.x (cit. on p. 64).

[142] Peter Eades. "A Heuristic for Graph Drawing". In: *Congressus Numerantium* 42 (1984). Ed. by D. S. Meek and van G. H. J. Rees, pp. 149–160 (cit. on pp. xxvii, 36).

[143] Peter Eades. "Drawing Free Trees". In: 1992, pp. 10–36 (cit. on p. 37).

[144] Peter Eades and Roberto Tamassia. *Algorithms for Drawing Graphs: An Annotated Bibliography*. Tech. rep. Providence, RI, USA, 1988.

[145] N. Eagle, A. Pentland, and D. Lazer. "Mobile phone data for inferring social network structure". In: *Social Computing, Behavioral Modeling, and Prediction* (2008), pp. 79–88 (cit. on p. 73).

[146] N. Eagle, A.S. Pentland, and D. Lazer. "Inferring friendship network structure by using mobile phone data". In: *Proc. Natl. Acad. Sci.* 106.36 (2009), p. 15274 (cit. on p. 73).

[147] P. Erdős and A. Rényi. "On random graphs". In: *Publicationes Mathematicae* 6.26 (1959), pp. 290–297 (cit. on p. 7).

[148] P. Erdős and A. Rényi. "On the Evolution of Random Graphs". In: *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*. 1960 (cit. on p. 7).

[149] M. Faloutsos, P. Faloutsos, and C. Faloutso. "On power-law relationships of the Internet topology". In: *Proc. of the ACM SIGCOMM computer communication review*. Vol. 29. 4. ACM Press. Cambridge, Massachusstes, USA: ACM, Aug. 1999, pp. 251–262. DOI: 10.1145/316194.316229 (cit. on pp. 4, 64, 121).

[150] Thomas J. Fararo and Patrick Doreian. "Tripartite structural analysis: Generalizing the Breiger-Wilson formalism". In: *Social Networks* 6.2 (1984), pp. 141–175. ISSN: 03788733. DOI: 10.1016/0378-8733(84)90015-7.

[151] Paolo Federico et al. "A Visual Analytics Approach to Dynamic Social Networks". In: *Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies*. i-KNOW '11. Graz, Austria: ACM, 2011, 47:1–47:8. ISBN: 978-1-4503-0732-1. DOI: 10.1145/2024288.2024344. URL: http://doi.acm.org/10.1145/2024288.2024344 (cit. on pp. 42, 43).

[152] E. Ferrara et al. "Detecting criminal organizations in mobile phone networks". In: *Expert Systems with Applications* 41.13 (2014), pp. 5733–5750 (cit. on pp. 95, 98, 114, 117, 161).

[153] Emilio Ferrara. "A large-scale community structure analysis in Facebook". In: *EPJ Data Science* 1.9 (2012), pp. 1–30 (cit. on p. 95).

[154] Emilio Ferrara and Giacomo Fiumara. "Topological Features of Online Social Networks". In: *Communications on Applied and Industrial Mathematics* 2.2 (2011), pp. 15–33. ISSN: 2038-0909.

[155] Emilio Ferrara et al. "Traveling trends: social butterflies or frequent fliers?" In: *Proceedings of the first ACM conference on Online social networks*. ACM. 2013, pp. 213–222 (cit. on p. 95).

[156] Santo Fortunato. "Community detection in graphs". In: *Physics Reports* 486 (2010), pp. 75–174 (cit. on pp. xxv, 1, 59).

[157] Santo Fortunato and Marc Barthelemy. "Resolution limit in community detection". In: *Proceedings of the National Academy of Sciences* 104.1 (2007), pp. 36–41 (cit. on pp. 59, 81).

[158] Terrill Frantz and Kathleen M. Carley. *A Formal Characterization of Cellular Networks*. Tech. rep. CMU-ISRI-05-109. Carnegie Mellon University School of Computer Science Institute for Software Research International, 2005 (cit. on p. 54).

[159] L.C. Freeman. "A set of measures of centrality based on betweenness". In: *Sociometry* 40 (1977), pp. 35–41 (cit. on pp. 49, 97, 119).

[160] Linton C Freeman. "Centrality in social networks: conceptual clarification". In: *Social networks* 1.3 (1979), pp. 215–239. ISSN: 0378-8733. DOI: 10.1016/0378-8733(78)90021-7. URL: http://www.sciencedirect.com/science/article/pii/0378873378900217 (cit. on pp. 65, 139).

[161] Linton C. Freeman. "Visualizing Social Networks." In: *Journal of Social Structure* 1 (2000) (cit. on pp. xxvii, 49, 65, 98).

[162] Arne Frick, Andreas Ludwig, and Heiko Mehldau. "A fast adaptive layout algorithm for undirected graphs (extended abstract and system demonstration)". In: *Graph Drawing: DIMACS International Workshop, GD '94 Princeton, New Jersey, USA, October 10–12, 1994 Proceedings*. Ed. by Roberto Tamassia and Ioannis G. Tollis. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 388–403. ISBN: 978-3-540-49155-2. DOI: 10.1007/3-540-58950-3_393. URL: http://dx.doi.org/10.1007/3-540-58950-3_393 (cit. on p. 37).

[163] Carsten Friedrich and Peter Eades. "Graph Drawing in Motion". In: *J. Graph Algorithms Appl.* 6.3 (2002), pp. 353–370 (cit. on p. 47).

[164] Matteo Frigo, Charles E. Leiserson, and Keith H. Randall. "The Implementation of the Cilk-5 Multithreaded Language". In: *SIGPLAN Not.* 33.5 (May 1998), pp. 212–223. ISSN: 0362-1340. DOI: 10.1145/277652.277725. URL: http://doi.acm.org/10.1145/277652.277725 (cit. on p. 142).

[165] T.M.J. Fruchterman and E.M. Reingold. "Graph drawing by force-directed placement". In: *Software: Practice and Experience* 21.11 (1991), pp. 1129–1164. ISSN: 1097-024X (cit. on pp. xxvii, 36, 79, 98, 104, 172).

[166] G. W. Furnas. "Generalized Fisheye Views". In: *SIGCHI Bull.* 17.4 (Apr. 1986), pp. 16–23. ISSN: 0736-6906. DOI: 10.1145/22339.22342. URL: http://doi.acm.org/10.1145/22339.22342 (cit. on pp. 47, 105).

[167] V. Furtado et al. "Collective intelligence in law enforcement–The WikiCrimes system". In: *Information Sciences* 180.1 (2010), pp. 4–17 (cit. on p. 118).

[168] Riccardo Gallotti and Marc Barthelemy. "Anatomy and efficiency of urban multimodal mobility". In: *Scientific Reports* 4.6911 (2014).

[169] M. Ghoniem, J. D. Fekete, and P. Castagliola. "A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations". In: *Symposium on Information Visualization*. IEEE. 2004 (cit. on p. 66).

[170] Helen Gibson, Joe Faith, and Paul Vickers. "A survey of two-dimensional graph layout techniques for information visualisation". In: *Information Visualization* 12.3-4 (2013), pp. 324–357. DOI: 10.1177/1473871612455749. eprint: http://ivi.sagepub.com/content/12/3-4/324.full.pdf+html. URL: http://ivi.sagepub.com/content/12/3-4/324.abstract (cit. on p. 37).

[171] John R. Gilbert, Steve Reinhardt, and Viral B. Shah. "A Unified Framework for Numerical and Combinatorial Computing". In: *Computing in Science and Engg.* 10.2 (Mar. 2008), pp. 20–25. ISSN: 1521-9615. DOI: 10.1109/MCSE.2008.45. URL: http://dx.doi.org/10.1109/MCSE.2008.45 (cit. on p. 141).

[172] M. Girvan and M.E.J. Newman. "Community structure in social and biological networks". In: *Proc. Natl. Acad. Sci.* 99.12 (2002), p. 7821 (cit. on pp. 59, 79, 88).

[173] M. Gjoka et al. "Walking in facebook: a case study of unbiased sampling of OSNs". In: *Proceedings of the 29th conference on Information communications*. IEEE Press. 2010, pp. 2498–2506 (cit. on p. 139).

[174] E. Glaeser, B. Sacerdote, and J. Scheinkman. "Crime and Social Interactions". In: *The Quarterly journal of economics* 111.2 (1996), pp. 507–548 (cit. on p. 118).

[175] Anna Goldenberg et al. "A Survey of Statistical Network Models". In: *Found. Trends Mach. Learn.* 2.2 (Feb. 2010), pp. 129–233.

[176] S. Gómez et al. "Diffusion Dynamics on Multiplex Networks". In: *Phys. Rev. Lett.* 110 (2 Jan. 2013), p. 028701. DOI: 10.1103/PhysRevLett.110.028701. URL: http://link.aps.org/doi/10.1103/PhysRevLett.110.028701 (cit. on p. xxv).

[177] Jesús Gómez-Gardeñes et al. "Evolution of cooperation in multiplex networks." In: *Scientific reports* 2 (Aug. 2012).

[178] Bruno Gonçalves, Nicola Perra, and Alessandro Vespignani. "Modeling Users' Activity on Twitter Networks: Validation of Dunbar's Number". In: *PloS one* 6.8 (2011), e22656 (cit. on p. 95).

[179] Sandra González-Bailón et al. "The dynamics of protest recruitment through an online network". In: *Scientific reports* 1 (2011) (cit. on p. 95).

[180] Mark Granovetter. "The Strength of Weak Ties: A Network Theory Revisited". In: *Sociological Theory* 1.1983 (1983), pp. 201–233. DOI: 10.2307/202051 (cit. on p. 63).

[181] Chang-Gui Gu et al. "Onset of cooperation between layered networks". In: *Phys. Rev. E* 84 (2 Aug. 2011), p. 026101. DOI: 10.1103/PhysRevE.84.026101. URL: http://link.aps.org/doi/10.1103/PhysRevE.84.026101 (cit. on pp. xxv, 169).

[182] Stefan Hachul and Michael Jünger. "Drawing Large Graphs with a Potential-Field-Based Multilevel Algorithm". In: *Graph Drawing: 12th International Symposium, GD 2004, New York, NY, USA, September 29-October 2, 2004, Revised Selected Papers*. Ed. by János Pach. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 285–295. ISBN: 978-3-540-31843-9. DOI: 10.1007/978-3-540-31843-9_29. URL: http://dx.doi.org/10.1007/978-3-540-31843-9_29 (cit. on pp. xxvii, 37).

[183] Arda Halu et al. *Multiplex PageRank*. June 2013. eprint: 1306.3576. URL: http://arxiv.org/abs/1306.3576 (cit. on p. xxvi).

[184] F. van Ham and B. Rogowitz. "Perceptual Organization in User-Generated Graph Layouts". In: *IEEE Transactions on Visualization and Computer Graphics* 14.6 (Nov. 2008), pp. 1333–1339. ISSN: 1077-2626. DOI: 10.1109/TVCG.2008.155 (cit. on p. 33).

[185] Imen Hamed and Malika Charrad. "Recognizing Information Spreaders in Terrorist Networks: 26/11 Attack Case Study". In: *Information Systems for Crisis Response and Management in Mediterranean Countries: Second International Conference, ISCRAM-med 2015, Tunis, Tunisia, October 28-30, 2015, Proceedings*. Ed. by Narjès Bellamine Ben Saoud, Carole Adam, and Chihab Hanachi. Springer International Publishing, 2015, pp. 27–38. ISBN: 978-3-319-24399-3. DOI: 10.1007/978-3-319-24399-3_3. URL: http://dx.doi.org/10.1007/978-3-319-24399-3_3 (cit. on p. 50).

[186] David Harel and Yehuda Koren. "A Fast Multi-Scale Method for Drawing Large Graphs". In: *Journal of Graph Algorithms and Applications* 6.3 (2002), pp. 179–202. DOI: 10.7155/jgaa.00051 (cit. on p. xxvii).

[187] David Harel and Yehuda Koren. "Graph Drawing by High-Dimensional Embedding". In: *Graph Drawing: 10th International Symposium, GD 2002 Irvine, CA, USA, August 26–28, 2002 Revised Papers*. Ed. by Michael T. Goodrich and Stephen G. Kobourov. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 207–219. ISBN: 978-3-540-36151-0. DOI: 10.1007/3-540-36151-0_20. URL: http://dx.doi.org/10.1007/3-540-36151-0_20 (cit. on p. xxvii).

[188] Andreas Harrer and Alona Schmidt. "An Approach for the Blockmodeling in Multi-Relational Networks." In: *ASONAM*. IEEE Computer Society, 2012, pp. 591–598. ISBN: 978-0-7695-4799-2.

[189] Christopher Healey and James Enns. "Attention and Visual Memory in Visualization and Computer Graphics". In: *IEEE Transactions on Visualization and Computer Graphics* 18.7 (July 2012), pp. 1170–1188. ISSN: 1077-2626. DOI: 10.1109/TVCG.2011.127. URL: http://dx.doi.org/10.1109/TVCG.2011.127 (cit. on p. 32).

[190] Christopher G. Healey, Kellogg S. Booth, and James T. Enns. "High-speed Visual Estimation Using Preattentive Processing". In: *ACM Trans. Comput.-Hum. Interact.* 3.2 (June 1996), pp. 107–135. ISSN: 1073-0516. DOI: 10.1145/230562.230563. URL: http://doi.acm.org/10.1145/230562.230563 (cit. on pp. 31, 32).

[191] J. Heer and D. Boyd. "Vizster: Visualizing Online Social Networks". In: *Proc. IEEE Symposium on Information Visualization*. 2005, p. 5 (cit. on p. 80).

[192] Jeffrey Heer, Stuart K. Card, and James A. Landay. "prefuse: A Toolkit for Interactive Information Visualization". In: *Conf. on Human Factors in Computing Systems*. ACM. Portland, 2005 (cit. on pp. xxvii, 66, 174).

[193] Nathalie Henry and Jean-Daniel Fekete. "MatrixExplorer: A Dual-Representation System to Explore Social Networks". In: *IEEE Transactions on Visualization and Computer Graphics* 12.5 (Sept. 2006), pp. 677–684. ISSN: 1077-2626. DOI: 10.1109/TVCG.2006.160. URL: http://dx.doi.org/10.1109/TVCG.2006.160 (cit. on pp. 40, 98).

[194] Ivan Herman, Guy Melançon, and M. Scott Marshall. "Graph Visualization and Navigation in Information Visualization: A Survey". In: *IEEE Transactions on Visualization and Computer Graphics* 6.1 (Jan. 2000), pp. 24–43. ISSN: 1077-2626. DOI: 10.1109/2945.841119. URL: http://dx.doi.org/10.1109/2945.841119 (cit. on pp. 35, 37).

[195] Colin Hirsch and R. Fleischer. "Graph Drawings and Its Applications". In: *Drawing Graphs – Methods and Models*. Ed. by M. Kaufmann and D. Wagner. LNCS. Springer, 2001. Chap. 1, pp. 1–22 (cit. on p. 37).

[196] Joshua Ho and Seok-Hee Hong. "Drawing Clustered Graphs in Three Dimensions". In: *Graph Drawing: 13th International Symposium, GD 2005, Limerick, Ireland, September 12-14, 2005. Revised Papers*. Ed. by Patrick Healy and Nikola S. Nikolov. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 492–502. ISBN: 978-3-540-31667-1. DOI: 10.1007/11618058_44. URL: http://dx.doi.org/10.1007/11618058_44 (cit. on p. 41).

[197] Petter Holme and Jari Saramäki. "Temporal networks". In: *Physics Reports* 519.3 (2012), pp. 97–125 (cit. on p. 153).

[198] Petter Holme and Jari Saramäki. "Temporal Networks as a Modeling Framework". In: *Temporal Networks*. Ed. by Petter Holme and Jari Saramäki. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 1–14. ISBN: 978-3-642-36461-7. DOI: 10.1007/978-3-642-36461-7_1. URL: http://dx.doi.org/10.1007/978-3-642-36461-7_1.

[199] Petter Holme et al. "Attack vulnerability of complex networks". In: *Phys. Rev. E* 65 (5 May 2002), p. 056109. DOI: 10.1103/PhysRevE.65.056109 (cit. on pp. 5, 64).

[200] Seok-Hee Hong. "MultiPlane: A New Framework for Drawing Graphs in Three Dimensions". In: *Graph Drawing: 13th International Symposium, GD 2005, Limerick, Ireland, September 12-14, 2005. Revised Papers*. Ed. by Patrick Healy and Nikola S. Nikolov. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 514–515. ISBN: 978-3-540-31667-1. DOI: 10.1007/11618058_49. URL: http://dx.doi.org/10.1007/11618058_49 (cit. on p. 42).

[201] Seok-Hee Hong and Tom Murtagh. "Visualisation of Large and Complex Networks Using PolyPlane". In: *Graph Drawing: 12th International Symposium, GD 2004, New York, NY, USA, September 29-October 2, 2004, Revised Selected Papers*. Ed. by János Pach. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 471–481. ISBN: 978-3-540-31843-9. DOI: 10.1007/978-3-540-31843-9_49. URL: http://dx.doi.org/10.1007/978-3-540-31843-9_49 (cit. on p. 41).

[202] Seok-Hee Hong and Nikola S. Nikolov. "Layered Drawings of Directed Graphs in Three Dimensions". In: *Proceedings of the 2005 Asia-Pacific Symposium on Information Visualisation - Volume 45*. APVis '05. Sydney, Australia: Australian Computer Society, Inc., 2005, pp. 69–74. ISBN: 1-920-68227-9. URL: http://dl.acm.org/citation.cfm?id=1082315.1082326 (cit. on pp. 41, 42).

[203] Seok-Hee Hong, Nikola S. Nikolov, and Alexandre Tarassov. "A 2.5D Hierarchical Drawing of Directed Graphs." In: *J. Graph Algorithms Appl.* 11.2 (2007), pp. 371–396. URL: http://dblp.uni-trier.de/db/journals/jgaa/jgaa11.html#HongNT07.

[204] Mikko Hypponen. "Malware goes mobile". In: *Scientific American* 295.5 (2006), pp. 70–77 (cit. on p. 95).

[205] *Intel Cilk++ SDK Programmer's Guide*. Intel Corp. 2009 (cit. on p. 142).

[206] H. Jeong et al. "Lethality and centrality in protein networks". In: *Nature* 411.6833 (May 2001), pp. 41–42. DOI: 10.1038/35075138 (cit. on p. 64).

[207] H. Jeong et al. "The large-scale organization of metabolic networks". In: *Nature* 407.6804 (2000), pp. 651–654. ISSN: 0028-0836 (cit. on p. 4).

[208] Yvonne Jewkes and Majid Yar. *Handbook of Internet crime*. Routledge, 2013 (cit. on p. 95).

[209] Brian Johnson and Ben Shneiderman. "Tree-Maps: A Space-filling Approach to the Visualization of Hierarchical Information Structures". In: *Proceedings of the 2Nd Conference on Visualization '91*. VIS '91. San Diego, California: IEEE Computer Society Press, 1991, pp. 284–291. ISBN: 0-8186-2245-8. URL: http://dl.acm.org/citation.cfm?id=949607.949654 (cit. on p. 38).

[210] *JUNG: Java Universal Network/Graph Framework*. http://jung.sf.net/ (cit. on pp. 66, 174).

[211] T. Kamada and S. Kawai. "An Algorithm for Drawing General Undirected Graphs". In: *Inf. Process. Lett.* 31.1 (Apr. 1989), pp. 7–15. ISSN: 0020-0190. DOI: 10.1016/0020-0190(89)90102-6. URL: http://dx.doi.org/10.1016/0020-0190(89)90102-6 (cit. on p. 36).

[212] B. Kapferer. *Norms and the Manipulation of Relationships in a Work Context*. Ed. by J. Clyde Mitchell. Manchester University Press, 1969.

[213] Leo Katz. "A new status index derived from sociometric analysis". In: *Psychometrika* 18.1 (1953), pp. 39–43. DOI: 10.1007/BF02289026. URL: http://dx.doi.org/10.1007/BF02289026 (cit. on pp. 6, 16).

[214] M. Kenney. *From Pablo to Osama: Trafficking and Terrorist Networks, Government Bureaucracies, and Competitive Adaptation*. Pennsylvania State University Press, 2007. ISBN: 9780271045313. DOI: 10.1017/S1537592709990156 (cit. on p. 65).

[215] J.V. Kepner and J.R. Gilbert. *Graph Algorithms in the Language of Linear Algebra*. Ed. by Jeremy Kepner and John Gilbert. Software, environments, tools. Society for Industrial and Applied Mathematics, 2011. ISBN: 9780898719918. DOI: 10.1137/1.9780898719918. eprint: http://epubs.siam.org/doi/pdf/10.1137/1.9780898719918. URL: http://epubs.siam.org/doi/abs/10.1137/1.9780898719918 (cit. on p. 141).

[216] Jung Yeol Kim and K.-I. Goh. "Coevolution and Correlated Multiplexity in Multiplex Networks". In: *Phys. Rev. Lett.* 111 (5 July 2013), p. 058702. DOI: 10.1103/PhysRevLett.111.058702. URL: http://link.aps.org/doi/10.1103/PhysRevLett.111.058702 (cit. on p. xxv).

[217] Mikko Kivelä et al. "Multilayer networks". In: *Journal of complex networks* 2.3 (2014), pp. 203–271 (cit. on pp. xxv, xxxi, 11, 23, 29, 44, 142, 155, 157, 169–172).

[218] J. Kleinberg. "The small-world phenomenon: an algorithm perspective". In: *Proc. of the 32nd annual symposium on Theory of computing*. ACM. 2000, pp. 163–170. ISBN: 1581131844 (cit. on p. 8).

[219] J. M. Kleinberg. "Authoritative Sources in a Hyperlinked Environment". In: *Journal of the ACM* 46.5 (1999), pp. 604–632 (cit. on p. 8).

[220] Peter Klerks and Eysink Smeets. "The Network Paradigm Applied to Criminal Organizations: Theoretical nitpicking or a relevant doctrine for investigators? Recent developments in the Netherlands". In: *Connections* 24 (2001), pp. 53–65 (cit. on pp. xxvi, 61, 97).

[221] Alden S Klovdahl. "A note on images of networks". In: *Social Networks* 3.3 (1981), pp. 197 –214. ISSN: 0378-8733. DOI: 10.1016/0378-8733(81)90016-2.

[222] S. G. Kobourov. "Force-directed drawing algorithms". In: *Handbook of Graph Drawing and Visualization*. Ed. by Roberto Tamassia. CRC Press, 2013, pp. 383–408 (cit. on p. 37).

[223] Tamara G. Kolda and Brett W. Bader. "Tensor Decompositions and Applications". In: *SIAM REVIEW* 51.3 (2009), pp. 455–500 (cit. on p. 18).

[224] Stuart Koschade. "A Social Network Analysis of Jemaah Islamiyah: The Applications to Counterterrorism and Intelligence". In: *Studies in Conflict & Terrorism* 29.6 (2006), pp. 559–575. DOI: 10.1080/10576100600798418. URL: http://dx.doi.org/10.1080/10576100600798418 (cit. on p. 61).

[225] L. Kovanen et al. "Temporal motifs in time-dependent networks". In: *Journal of Statistical Mechanics: Theory and Experiment* 2011.11 (2011), P11005. URL: http://stacks.iop.org/1742-5468/2011/i=11/a=P11005 (cit. on p. 153).

[226] D. Krackhardt. "Cognitive social structures". In: *Social Networks* 9.2 (1987), pp. 109–134.

[227] David Krackhardt, Jim Blythe, and Cathleen McGrath. "KRACKPLOT 3.0: An Improved Network Drawing Program". In: *Connections* 17.2 (2000), pp. 53–55.

[228] Valdis Krebs. "Mapping Networks of Terrorist Cells". In: *Connections* 24.3 (2002), pp. 43–52 (cit. on pp. xxvi, xxx, 54, 55, 61, 79, 97, 153).

[229] Ravi Kumar, Jasmine Novak, and Andrew Tomkins. "Structure and evolution of online social networks". In: *Link mining: models, algorithms, and applications*. Springer, 2010, pp. 337–357 (cit. on p. 95).

[230] Maciej Kurant and Patrick Thiran. "Layered Complex Networks". In: *Phys. Rev. Lett.* 96 (13 Apr. 2006), p. 138701. DOI: 10.1103/PhysRevLett.96.138701. URL: http://link.aps.org/doi/10.1103/PhysRevLett.96.138701 (cit. on pp. xxv, 169).

[231] Klaus von Lampe. "Human capital and social capital in criminal networks: introduction to the special issue on the 7th Blankensee Colloquium". In: *Trends in Organized Crime* 12.2 (2009), pp. 93–100. ISSN: 1936-4830. DOI: 10.1007/s12117-009-9067-z. URL: http://dx.doi.org/10.1007/s12117-009-9067-z (cit. on p. 60).

[232] Andrea Lancichinetti et al. "Characterizing the Community Structure of Complex Networks". In: *PLoS ONE* 5.8 (Aug. 2010), e11976. DOI: 10.1371/journal.pone.0011976 (cit. on p. 1).

[233] Vito Latora and Massimo Marchiori. "How the science of complex networks can help developing strategies against terrorism". In: *Chaos, Solitons & Fractals* 20.1 (Apr. 2004), pp. 69–75. DOI: http://dx.doi.org/10.1016/S0960-0779(03)00429-6. URL: http://www.sciencedirect.com/science/article/pii/S0960077903004296 (cit. on pp. xxvii, 62).

[234] Neal Leavitt. "Mobile phones: the next frontier for hackers?" In: *Computer* 38.4 (2005), pp. 20–23 (cit. on p. 95).

[235] C. Y. Lee. "An Algorithm for Path Connections and Its Applications". In: *IRE Transactions on Electronic Computers* EC-10.3 (Sept. 1961), pp. 346–365. ISSN: 0367-9950. DOI: 10.1109/TEC.1961.5219222 (cit. on p. 139).

[236] Kyu-Min Lee, Byungjoon Min, and Kwang-Il Goh. "Towards real-world complexity: an introduction to multiplex networks". In: *The European Physical Journal B* 88.2 (2015), p. 48. ISSN: 1434-6036. DOI: 10.1140/epjb/e2015-50742-1. URL: http://dx.doi.org/10.1140/epjb/e2015-50742-1 (cit. on pp. 26, 27).

[237] Kyu-Min Lee et al. "Correlated multiplexity and connectivity of multiplex random networks". In: *New Journal of Physics* 14.3 (2012), p. 033027. URL: http://stacks.iop.org/1367-2630/14/i=3/a=033027 (cit. on pp. xxv, 26).

[238] Janette Lehmann et al. "Dynamical classes of collective attention in Twitter". In: *Proceedings of the 21st international conference on World Wide Web*. ACM. 2012, pp. 251–260 (cit. on p. 95).

[239] E. A. Leicht and R. M. D'Souza. "Percolation on interacting networks". In: *ArXiv e-prints* (July 2009). eprint: 0907.0894 (cit. on p. xxvi).

[240] Charles E. Leiserson. "The Cilk++ Concurrency Platform". In: *J. Supercomput.* 51.3 (Mar. 2010), pp. 244–257. ISSN: 0920-8542. DOI: 10.1007/s11227-010-0405-3. URL: http://dx.doi.org/10.1007/s11227-010-0405-3 (cit. on pp. 139, 147).

[241] Charles E. Leiserson and Tao B. Schardl. "A Work-efficient Parallel Breadth-first Search Algorithm (or How to Cope with the Nondeterminism of Reducers)". In: *Proceedings of the Twenty-second Annual ACM Symposium on Parallelism in Algorithms and Architectures*. SPAA '10. New York, NY, USA: ACM, 2010, pp. 303–314. ISBN: 978-1-4503-0079-7. DOI: 10.1145/1810479.1810534. URL: http://doi.acm.org/10.1145/1810479.1810534.

[242] Jure Leskovec and Eric Horvitz. "Planetary-scale Views on a Large Instant-messaging Network". In: *Proceedings of the 17th International Conference on World Wide Web*. WWW '08. Beijing, China: ACM, 2008, pp. 915–924. ISBN: 978-1-60558-085-2. DOI: 10.1145/1367497.1367620. URL: http://doi.acm.org/10.1145/1367497.1367620.

[243] Y. K. Leung and M. D. Apperley. "A Review and Taxonomy of Distortion-oriented Presentation Techniques". In: *ACM Trans. Comput.-Hum. Interact.* 1.2 (June 1994), pp. 126–160. ISSN: 1073-0516 (cit. on p. 47).

[244] Robert Levinson. "Towards Domain-Independent Machine Intelligence". In: *Proceedings on Conceptual Graphs for Knowledge Representation*. ICCS '93. London, UK, UK: Springer-Verlag, 1993, pp. 254–273. ISBN: 3-540-56979-0. URL: http://dl.acm.org/citation.cfm?id=645486.757708 (cit. on p. 139).

[245] Kevin Lewis et al. "Tastes, Ties, and Time: A New (Cultural, Multiplex, and Longitudinal) Social Network Dataset Using Facebook.com". In: *Social Networks* (2008).

[246] Roy Lindelauf, Peter Borm, and Herbert Hamers. "The influence of secrecy on the communication structure of covert networks". In: *Social Networks* 31.2 (2009), pp. 126 –137. ISSN: 0378-8733. DOI: 10.2139/ssrn.1096057. URL: http://www.sciencedirect.com/science/article/pii/S0378873309000021 (cit. on pp. xxvi, 63).

[247] X. Liu et al. "Criminal networks: Who is the key player?" In: 2012.39 (2012) (cit. on p. 119).

[248] Francois Lorrain and Harrison C. White. "Structural equivalence of individuals in social networks". In: *The Journal of Mathematical Sociology* 1.1 (1971), pp. 49–80. DOI: 10.1080/0022250X.1971.9989788 (cit. on pp. 49, 97).

[249] V. H. P. Louzada et al. "Breathing synchronization in interconnected networks". In: *Scientific Reports* 3 (Nov. 2013), 3289 EP –. URL: http://dx.doi.org/10.1038/srep03289 (cit. on p. xxvi).

[250] M.D. Lyman and G.W. Potter. *Organized Crime*. Pearson/Prentice Hall, 2007. ISBN: 9780131730366. URL: https://books.google.it/books?id=GAqoSwAACAAJ (cit. on p. 52).

[251] P. Maas. *The Valachi papers*. Panther, 1968 (cit. on p. 131).

[252] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. New York, NY, USA: Henry Holt and Co., Inc., 1982. ISBN: 0716715678 (cit. on p. 41).

[253] Charles Z. Marshak et al. "Growth and containment of a hierarchical criminal network". In: *Phys. Rev. E* 93 (2 Feb. 2016), p. 022308. DOI: 10.1103/PhysRevE.93.022308. URL: http://link.aps.org/doi/10.1103/PhysRevE.93.022308 (cit. on p. 61).

[254] Carla D. Martin and Mason A. Porter. "The Extraordinary SVD". In: *American Mathematical Monthly* 119 (10 2011), pp. 838–851.

[255] J. Martin-Hernandez et al. *On Synchronization of Interdependent Networks*. Mar. 2013 (cit. on p. xxvi).

[256] G. Mastrobuoni and E. Patacchini. "Organized crime networks: An application of network analysis techniques to the American Mafia". In: *Review of Network Economics* 11.3 (2012) (cit. on pp. 113, 114, 117).

[257] D. McAndrew. "The structural analysis of criminal networks". In: *D. Canter and L. Alison Eds. The Social Psychology of Crime: Groups, Teams, and Networks, Offender Profiling Series* 3 (1999), pp. 53–94 (cit. on pp. xxvi, 49, 62).

[258] Kathleen M.Carley, Jeffrey Reminga, and Natasha Kamneva. "Destabilizing Terrorist Networks". In: *Institute for Software Research* 45.45 (1998) (cit. on pp. 58, 97).

[259] J.M. McGloin. "Policy and intervention considerations of a network analysis of street gangs". In: *Criminology & Public Policy* 4(3) (2005), pp. 607–635 (cit. on pp. 113, 117).

[260] "Front Matter". In: *Maya Python for Games and Film*. Ed. by Adam Mechtley and Ryan Trowbridge. Boston: Morgan Kaufmann, 2012, pp. 1 –381. ISBN: 978-0-12-378578-7. DOI: http://dx.doi.org/10.1016/B978-0-12-378578-7.00016-8 (cit. on pp. 154, 155).

[261] Nasrullah Memon et al. "Harvesting Covert Networks: a Case Study of the iMiner Database". In: *Int. J. Netw. Virtual Organ.* 8.1/2 (Nov. 2011), pp. 52–74. ISSN: 1470-9503. DOI: 10.1504/IJNVO.2011.037161. URL: http://dx.doi.org/10.1504/IJNVO.2011.037161 (cit. on p. xxvi).

[262] Nasrullah Memon et al. "Retracted: Small World Terrorist Networks: A Preliminary Investigation". In: *Applications and Innovations in Intelligent Systems XV: Proceedings of AI-2007, the Twenty-seventh SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*. Ed. by Richard Ellis, Tony Allen, and Miltos Petridis. London: Springer London, 2008, pp. 339–344. ISBN: 978-1-84800-086-5. DOI: 10.1007/978-1-84800-086-5_28. URL: http://dx.doi.org/10.1007/978-1-84800-086-5_28 (cit. on p. 61).

[263] J. Mena. *Investigative Data Mining for Security and Criminal Detection*. Butterworth-Heinemann, 2003. ISBN: 9780750676137. URL: http://books.google.it/books?id=WbSNpYHoNWMC.

[264] Tom Michoel and Bruno Nachtergaele. "Alignment and integration of complex networks by hypergraph-based spectral clustering". In: *Physical Review E* 86 (2012), pp. 056111+. DOI: 10.1103/physreve.86.056111. URL: http://dx.doi.org/10.1103/physreve.86.056111.

[265] S. Milgram. "The small world problem". In: *Psychology today* 2.1 (1967), pp. 60–67.

[266] R. Milo et al. "Network Motifs: Simple Building Blocks of Complex Networks". In: *Science* 298.5594 (2002), pp. 824–827. ISSN: 0036-8075. DOI: 10.1126/science.298.5594.824. eprint: http://science.sciencemag.org/content/298/5594/824.full.pdf. URL: http://science.sciencemag.org/content/298/5594/824 (cit. on p. 9).

[267] Byungjoon Min and K.-I. Goh. "Layer-crossing overhead and information spreading in multiplex social networks." In: *CoRR* (2013).

[268] Byungjoon Min and K.-I. Goh. "Multiple resource demands and viability in multiplex networks". In: *Phys. Rev. E* 89 (4 Apr. 2014), p. 040802. DOI: 10.1103/PhysRevE.89.040802. URL: http://link.aps.org/doi/10.1103/PhysRevE.89.040802 (cit. on p. xxv).

[269] Alan Mislove et al. "Measurement and analysis of online social networks". In: *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM. 2007, pp. 29–42 (cit. on pp. 95, 139).

[270] J. Clyde Mitchell, ed. *Social networks in urban situations: Analysis of personal relationships in central African towns*. Manchester: Manchester University Press, 1969.

[271] Delia Mocanu et al. "The Twitter of Babel: Mapping World Languages through Microblogging Platforms". In: *PloS one* 8.4 (2013), e61981 (cit. on p. 95).

[272] Edward F. Moore. "The shortest path through a maze". In: *Proceedings of the International Symposium on the Theory of Switching, and Annals of the Computation Laboratory of Harvard University*. Harvard University Press, 1959, pp. 285–292 (cit. on p. 139).

[273] J.L. Moreno. *Who shall survive? A new approach to the problem of human interrelations.* Washington: Nervous and Mental Disease Publishing Company, 1934 (cit. on p. 36).

[274] Yamir Moreno, Romualdo Pastor-Satorras, and Alessandro Vespignani. "Epidemic outbreaks in complex heterogeneous networks". In: *The European Physical Journal B-Condensed Matter and Complex Systems* 26.4 (2002), pp. 521–529 (cit. on p. 121).

[275] C. Morselli. "Career opportunities and network-based privileges in the Cosa Nostra". In: *Crime, Law and Social Change* 39.4 (2003), pp. 383–418 (cit. on pp. 114, 117).

[276] C. Morselli. *Inside Criminal Networks*. Studies of Organized Crime. Springer, 2008. ISBN: 9780387095264. DOI: 10.1007/978-0-387-09526-4 (cit. on pp. 52, 58, 63).

[277] Carlo Morselli. "Assessing vulnerable and strategic positions in a criminal network". In: *Journal of Contemporary Criminal Justice* 26.4 (2010), pp. 382–392 (cit. on pp. 73, 99, 103).

[278] Carlo Morselli. *Contacts, opportunities, and criminal enterprise*. University of Toronto Press, 2005 (cit. on p. 95).

[279] Carlo Morselli. "The Efficiency–Security Trade-Off". In: *Inside Criminal Networks*. New York, NY: Springer New York, 2009, pp. 1–9. ISBN: 978-0-387-09526-4. DOI: 10.1007/978-0-387-09526-4_4 (cit. on pp. xxvi, xxvii, 51, 95).

[280] Carlo Morselli, Cynthia Giguere, and Katia Petit. "The efficiency/security trade-off in criminal networks". In: *Social Networks* 29.1 (2007), pp. 143 –153. ISSN: 0378-8733. DOI: 10.1016/j.socnet.2006.05.001 (cit. on p. 52).

[281] Peter J. Mucha and Mason A. Porter. "Communities in multislice voting networks". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 20.4 (2010), pp. 041108+.

[282] Peter J Mucha et al. "Community structure in time-dependent, multiscale, and multiplex networks". In: *Science* 328.5980 (2010), pp. 876–878 (cit. on p. xxv).

[283] Sam Mullins and Adam Dolnik. "An exploratory, dynamic application of Social Network Analysis for modelling the development of Islamist terrorâĂŘcells in the West". In: *Behavioral Sciences of Terrorism and Political Aggression* 2.1 (2010), pp. 3–29. DOI: 10.1080/19434470903319441. URL: http://dx.doi.org/10.1080/19434470903319441 (cit. on p. 61).

[284] T. Munzner. "H3: Laying out Large Directed Graphs in 3D Hyperbolic Space". In: *Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97)*. INFOVIS '97. Washington, DC, USA: IEEE Computer Society, 1997, pp. 2–. ISBN: 0-8186-8189-6. URL: http://dl.acm.org/citation.cfm?id=857188.857627 (cit. on p. 40).

[285] Seth A Myers, Chenguang Zhu, and Jure Leskovec. "Information diffusion and external influence in networks". In: *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2012, pp. 33–41 (cit. on p. 95).

[286] M. Natarajan. "Understanding the structure of a large heroin distribution network: A quantitative analysis of qualitative data". In: *Journal of Quantitative Criminology* 22.2 (2006), pp. 171–192 (cit. on p. 117).

[287] M Newman. "A measure of betweenness centrality based on random walks". In: *Social Networks* 27.1 (2005), pp. 39–54. DOI: 10.1016/j.socnet.2004.11.009 (cit. on pp. 57, 120).

[288] M. E. J. Newman. "Analysis of weighted networks". In: *Physical Review* E 70 (2004).

[289] M. E. J. Newman. "Detecting community structure in networks". In: *The European Physical Journal B* 38.2 (2004), pp. 321–330. ISSN: 1434-6036. DOI: 10.1140/epjb/e2004-00124-y. URL: http://dx.doi.org/10.1140/epjb/e2004-00124-y (cit. on p. 139).

[290] M. E. J. Newman. "The structure and function of complex networks". In: *SIAM REVIEW* 45 (2003), pp. 167–256 (cit. on pp. xxv, 1, 5–7).

[291] Mark Newman. *Networks: An Introduction*. New York, NY, USA: Oxford University Press, Inc., 2010 (cit. on pp. xxv, 1, 6, 114, 119–121, 171).

[292] Mark Newman, Albert-Laszlo Barabasi, and Duncan J. Watts. *The Structure and Dynamics of Networks: (Princeton Studies in Complexity)*. Princeton, NJ, USA: Princeton University Press, 2006. ISBN: 0691113572 (cit. on p. xxvi).

[293] M.E.J. Newman. "Fast algorithm for detecting community structure in networks". In: *Phys. Rev. E* 69.6 (6 June 2004), p. 066133. DOI: 10.1103/PhysRevE.69.066133. URL: http://link.aps.org/doi/10.1103/PhysRevE.69.066133 (cit. on pp. 79, 81, 89, 139).

[294] M.E.J. Newman. "Modularity and community structure in networks". In: *PNAS* 103.23 (2006), p. 8577 (cit. on p. 59).

[295] M.E.J. Newman and M. Girvan. "Finding and evaluating community structure in networks". In: *Phy. Rev. E* 69.2 (Feb. 2004), p. 26113. ISSN: 1550-2376 (cit. on pp. 58, 59, 139).

[296] M.E.J. Newman and M. Girvan. "Mixing Patterns and Community Structure in Networks". In: *Statistical Mechanics of Complex Networks*. Ed. by Romualdo Pastor-Satorras, Miguel Rubi, and Albert Diaz-Guilera. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 66–87. ISBN: 978-3-540-44943-0. DOI: 10.1007/978-3-540-44943-0_5. URL: http://dx.doi.org/10.1007/978-3-540-44943-0_5 (cit. on p. 58).

[297] V. Nicosia et al. "Growing Multiplex Networks". In: *Phys. Rev. Lett.* 111 (5 July 2013), p. 058701. DOI: 10.1103/PhysRevLett.111.058701. URL: http://link.aps.org/doi/10.1103/PhysRevLett.111.058701.

[298] Vincenzo Nicosia and Vito Latora. "Measuring and modeling correlations in multiplex networks". In: *Phys. Rev. E* 92 (3 Sept. 2015), p. 032805. DOI: 10.1103/PhysRevE.92.032805. URL: http://link.aps.org/doi/10.1103/PhysRevE.92.032805 (cit. on pp. 26, 27).

[299] Vincenzo Nicosia et al. "Graph Metrics for Temporal Networks". In: *Temporal Networks*. Ed. by Petter Holme and Jari Saramäki. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 15–40. ISBN: 978-3-642-36461-7. DOI: 10.1007/978-3-642-36461-7_2. URL: http://dx.doi.org/10.1007/978-3-642-36461-7_2 (cit. on pp. 153, 155, 158).

[300] Vincenzo Nicosia et al. "Remote Synchronization Reveals Network Symmetries and Functional Modules". In: *Phys. Rev. Lett.* 110 (17 Apr. 2013), p. 174102.

[301] Wouter de Nooy, Andrej Mrvar, and Vladimir Batagelj. *Exploratory Social Network Analysis with Pajek*. Cambridge: Cambridge University Press, 2005.

[302] FranH. Norris et al. "Community Resilience as a Metaphor, Theory, Set of Capacities, and Strategy for Disaster Readiness". In: *Americhe Journal of Community Psychology* 41.1-2 (2008), pp. 127–150. ISSN: 0091-0562. DOI: 10.1007/s10464-007-9156-6 (cit. on p. 63).

[303] J. O'Madadhain et al. "Analysis and Visualization of Network Data using JUNG". In: *Journal of Statistical Software* VV (2005) (cit. on p. 66).

[304] J.P. Onnela et al. "Structure and tie strengths in mobile communication networks". In: *Proc. Natl. Acad. Sci.* 104.18 (2007), p. 7332 (cit. on p. 73).

[305] Jukka-Pekka Onnela et al. "Analysis of a large-scale weighted network of one-to-one human communication". In: *New Journal of Physics* 9.6 (2007), p. 179 (cit. on p. 73).

[306] Fatih Ozgul. "Classification of Terrorist Networks and Their Key Players". In: *Multidisciplinary Social Networks Research: International Conference, MISNC 2014, Kaohsiung, Taiwan, September 13-14, 2014. Proceedings*. Ed. by Leon Shyue-Liang Wang et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 145–157. ISBN: 978-3-662-45071-0. DOI: 10.1007/978-3-662-45071-0_12. URL: http://dx.doi.org/10.1007/978-3-662-45071-0_12 (cit. on p. 54).

[307] Padgett and C.K. J.F. Ansell. "Robust action and the rise of the Medici, 1400-1434". In: *American Journal of Sociology* 98.6 (1993), pp. 1259–1319.

[308] G. Palla et al. "Uncovering the overlapping community structure of complex networks in nature and society". In: *Nature* 435.7043 (2005), pp. 814–818. ISSN: 0028-0836 (cit. on pp. 59, 88).

[309] Philippa Pattison. "Social Networks, Algebraic Models for." In: *Encyclopedia of Complexity and Systems Science*. Ed. by Robert A. Meyers. Springer, 2009, pp. 8291–8306.

[310] Georgios A. Pavlopoulos et al. "Arena3D: visualization of biological networks in 3D". In: *BMC Systems Biology* 2.1 (2008), p. 104. ISSN: 1752-0509. DOI: 10.1186/1752-0509-2-104. URL: http://dx.doi.org/10.1186/1752-0509-2-104 (cit. on p. 43).

[311] A. Perer and B. Shneiderman. "Balancing systematic and flexible exploration of social networks". In: *IEEE Transactions on Visualization and Computer Graphics* (2006), pp. 693–700. ISSN: 1077-2626 (cit. on pp. xxvii, 98).

[312] Adam Perer and Ben Shneiderman. "Integrating statistics and visualization: case studies of gaining clarity during exploratory data analysis". In: *Conference on Human Factors in Computing Systems*. ACM. New York, 2008, pp. 53–55 (cit. on p. 66).

[313] M.B. Peterson. *Applications in Criminal Analysis: A Sourcebook*. Praeger, 1998. ISBN: 9780275964689. DOI: 10.1016/0047-2352(95)91181-I. URL: http://books.google.it/books?id=TPkrJAAACAAJ (cit. on p. 58).

[314] Nicholas J. Pioch and John O. Everett. "POLESTAR: collaborative knowledge management and sensemaking tools for intelligence analysts." In: *CIKM*. Ed. by Philip S. Yu et al. ACM, 2006, pp. 513–521. ISBN: 1-59593-433-2 (cit. on pp. xxviii, 75, 99).

[315] Mason Porter. "Small-world network." In: *Scholarpedia* 7.2 (2012), p. 1739 (cit. on p. 1).

[316] Mason A Porter, Jukka-Pekka Onnela, and Peter J Mucha. "Communities in networks". In: *Notices of the AMS* 56.9 (2009), pp. 1082–1097 (cit. on pp. 1, 58).

[317] R. C. Prim. "Shortest connection networks and some generalizations". In: *The Bell System Technical Journal* 36.6 (Nov. 1957), pp. 1389–1401. ISSN: 0005-8580. DOI: 10.1002/j.1538-7305.1957.tb01515.x (cit. on p. 139).

[318] JÃűrg Raab and H. Brinton Milward. "Dark Networks as Problems". In: *Journal of Public Administration Research and Theory* 13.4 (2003), pp. 413–439. DOI: 10.1093/jpart/mug029. eprint: http://jpart.oxfordjournals.org/content/13/4/413.full.pdf+html. URL: http://jpart.oxfordjournals.org/content/13/4/413.abstract (cit. on p. 50).

[319] Filippo Radicchi and Alex Arenas. "Abrupt transition in the structural formation of interconnected networks". In: *Nat Phys* 9.11 (Nov. 2013), pp. 717–720. URL: http://dx.doi.org/10.1038/nphys2761 (cit. on p. xxvi).

[320] A. Rasheed and U. K. Wiil. "A Tool for Analysis and Visualization of Criminal Networks". In: *2015 17th UKSim-AMSS International Conference on Modelling and Simulation (UKSim)*. Mar. 2015, pp. 97–102. DOI: 10.1109/UKSim.2015.64 (cit. on p. xxvi).

[321] James Reinders. *Intel threading building blocks - outfitting C++ for multi-core processor parallelism*. O'Reilly, 2007, pp. I–XXV, 1–303. ISBN: 978-0-596-51480-8 (cit. on p. 142).

[322] E. M. Reingold and J. S. Tilford. "Tidier Drawings of Trees". In: *IEEE Trans. Softw. Eng.* 7.2 (Mar. 1981), pp. 223–228. ISSN: 0098-5589. DOI: 10.1109/TSE.1981.234519. URL: http://dx.doi.org/10.1109/TSE.1981.234519 (cit. on p. 37).

[323] Jun Rekimoto and Mark Green. "he Information Cube: Using Transparency in 3D Information Visualization". In: *Proceedings of the Third Annual Workshop on Information Technologies & Systems*. WITS '93. 1993 (cit. on p. 40).

[324] R. S. Renfro and R. F. Deckro. "A Social Network Analysis of the Iranian Government". In: *69th Military Operational Research Symposium* (2001) (cit. on p. 61).

[325] M. M. G. Ricci and T. Levi-Civita. "Méthodes de calcul différentiel absolu et leurs applications". In: *Mathematische Annalen* 54.1 (1900), pp. 125–201. ISSN: 1432-1807. DOI: 10.1007/BF01454201. URL: http://dx.doi.org/10.1007/BF01454201 (cit. on p. 13).

[326] George G. Robertson, Jock D. Mackinlay, and Stuart K. Card. "Cone Trees: Animated 3D Visualizations of Hierarchical Information". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '91. New York, NY, USA: ACM, 1991, pp. 189–194. ISBN: 0-89791-383-3. DOI: 10.1145/108844.108883. URL: http://doi.acm.org/10.1145/108844.108883 (cit. on p. 40).

[327] Matthew Rocklin and Ali Pinar. "Latent Clustering on Graphs with Multiple Edge Types". In: *Algorithms and Models for the Web Graph: 8th International Workshop, WAW 2011, Atlanta, GA, USA, May 27-29, 2011. Proceedings*. Ed. by Alan Frieze, Paul Horn, and Paweł Prałat. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 38–49. ISBN: 978-3-642-21286-4. DOI: 10.1007/978-3-642-21286-4_4. URL: http://dx.doi.org/10.1007/978-3-642-21286-4_4.

[328] Daniel M Romero, Brendan Meeder, and Jon Kleinberg. "Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter". In: *Proceedings of the 20th international conference on World wide web*. ACM. 2011, pp. 695–704 (cit. on p. 95).

[329] M. Sageman. *Understanding Terror Networks*. University of Pennsylvania Press, 2004 (cit. on p. 79).

[330] R.K. Sah. "Social Osmosis and Patterns of Crime". In: *Journal of Political Economy* 99.6 (1991), pp. 1272–1295 (cit. on p. 118).

[331] F. D. Sahneh, C. Scoglio, and F. N. Chowdhury. "Effect of coupling on the epidemic threshold in interconnected complex networks: A spectral analysis". In: *2013 American Control Conference*. June 2013, pp. 2307–2312. DOI: 10.1109/ACC.2013.6580178 (cit. on pp. xxxi, 169).

[332] Jari Saramäki and Esteban Moro. "From seconds to months: an overview of multi-scale dynamics of mobile telephone calls". In: *The European Physical Journal B* 88.6 (2015), p. 164. ISSN: 1434-6036. DOI: 10.1140/epjb/e2015-60106-6. URL: http://dx.doi.org/10.1140/epjb/e2015-60106-6 (cit. on p. 73).

[333] Manojit Sarkar and Marc H. Brown. "Graphical Fisheye Views." In: *Comm. ACM* 37.12 (1994), pp. 73–84. URL: http://dblp.uni-trier.de/db/journals/cacm/cacm37.html#SarkarB94 (cit. on pp. 47, 105).

[334] J. Sarnecki. *Delinquent networks: Youth co-offending in Stockholm*. Cambridge University Press, 2001 (cit. on p. 117).

[335] Anna Saumell-Mendiola, M. Ángeles Serrano, and Marián Boguñá. "Epidemic spreading on interconnected networks". In: *Phys. Rev. E* 86 (2 Aug. 2012), p. 026106 (cit. on p. xxvi).

[336] F. Schneider et al. "Understanding online social network usage from a network perspective". In: *Proc. 9th SIGCOMM conference on Internet measurement conference*. ACM. 2009, pp. 35–48 (cit. on pp. 98, 106).

[337] J. Scott. *Social Network Analysis*. Sage, Newbury Park CA, 1992.

[338] J. Scott. *Social Network Analysis: A Handbook*. Second. Sage Publications, 2000. ISBN: 0761963391. DOI: 10.1037/033461 (cit. on p. 55).

[339] Maria Secrier et al. "Arena3D: visualizing time-driven phenotypic differences in biological systems". In: *BMC Bioinformatics* 13.1 (2012), p. 45. ISSN: 1471-2105. DOI: 10.1186/1471-2105-13-45. URL: http://dx.doi.org/10.1186/1471-2105-13-45 (cit. on p. 43).

[340] Raimund Seidel. "On the all-pairs-shortest-path problem". In: *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*. STOC '92. New York, NY, USA: ACM, 1992, pp. 745–749. ISBN: 0-89791-511-9 (cit. on p. 3).

[341] Jason Sharpe, Charles John Lumsden, and Nicholas Woolridge. *In Silico: 3D Animation and Simulation of Cell Biology with Maya and MEL*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007. ISBN: 9780080879253 (cit. on p. 167).

[342] X. Shen et al. "Visualization and analysis of email networks". In: *Asia-Pacific Symposium on Visualisation 2007* 00.undefined (2007), pp. 1–8. DOI: doi.ieeecomputersociety.org/10.1109/APVIS.2007.329302 (cit. on p. 44).

[343] B. Shneiderman and A. Aris. "Network Visualization by Semantic Substrates". In: *IEEE Trans. Visual. and Computer Graphics* 12.5 (Sept. 2006), pp. 733–740. ISSN: 1077-2626. DOI: 10.1109/TVCG.2006.166 (cit. on p. 96).

[344] Ben Shneiderman. "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations". In: *Proceedings of the 1996 IEEE Symposium on Visual Languages*. VL '96. Washington, DC, USA: IEEE Computer Society, 1996, pp. 336–. ISBN: 0-8186-7508-X. URL: http://dl.acm.org/citation.cfm?id=832277.834354 (cit. on p. 33).

[345] Ben Shneiderman. "Tree Visualization with Tree-maps: 2-d Space-filling Approach". In: *ACM Trans. Graph.* 11.1 (Jan. 1992), pp. 92–99. ISSN: 0730-0301. DOI: 10.1145/102377.115768. URL: http://doi.acm.org/10.1145/102377.115768 (cit. on p. 38).

[346] M. B. Short et al. "A statistical model of criminal behavior". In: *Mathematical Models and Methods in Applied Sciences* 18.suppo1 (2008), pp. 1249–1267. DOI: 10.1142/S0218202508003029. URL: http://www.worldscientific.com/doi/abs/10.1142/S0218202508003029 (cit. on p. 62).

[347] A. Slike. "The Devil You Know: Continuing Problems with Research on Terrorism". In: *Terrorism and Political Violence* 13 (2001), pp. 1–14 (cit. on pp. 61, 97).

[348] Peter M.A. Sloot, George Kampis, and Laszlo Gulyas. "Advances in dynamic temporal networks: Understanding the temporal dynamics of complex adaptive networks". In: *The European Physical Journal Special Topics* 222.6 (2013), pp. 1287–1293. ISSN: 1951-6355. DOI: 10.1140/epjst/e2013-01926-8 (cit. on p. 52).

[349] Marc A. Smith and al. "Analyzing Social Media Networks with NodeXL". In: *Fourth International Conference on Communities and Technologies*. ACM. Pennsylvania, 2009 (cit. on p. 66).

[350] Albert Solé-Ribalta et al. "Centrality Rankings in Multiplex Networks". In: *Proceedings of the 2014 ACM Conference on Web Science*. WebSci '14. New York, NY, USA: ACM, 2014, pp. 149–155. ISBN: 978-1-4503-2622-3. DOI: 10.1145/2615569.2615687. URL: http://doi.acm.org/10.1145/2615569.2615687 (cit. on p. 29).

[351] Albert Solé-Ribalta et al. "Spectral properties of the Laplacian of multiplex networks". In: *CoRR* abs/1307.2090 (3 Sept. 2013), p. 032807. URL: http://arxiv.org/abs/1307.2090 (cit. on pp. 20, 21).

[352] Luis SolÃą et al. "Eigenvector centrality of nodes in multiplex networks". In: *Chaos* 23.3, 033131 (2013) (cit. on pp. 22, 24).

[353] Seung-Woo Son, Peter Grassberger, and Maya Paczuski. "Percolation Transitions Are Not Always Sharpened by Making Networks Interdependent". In: *Phys. Rev. Lett.* 107 (19 Nov. 2011), p. 195702. DOI: 10.1103/PhysRevLett.107.195702. URL: http://link.aps.org/doi/10.1103/PhysRevLett.107.195702 (cit. on p. xxvi).

[354] S. Sonnino and L. Franchetti. *La Sicilia nel 1876*. G. Barbèra, 1877 (cit. on pp. 113, 116).

[355] M. K. Sparrow. "The application of network analysis to criminal intelligence: An assessment of the prospects". In: *Social Networks* 13.3 (1991), pp. 251 – 274. ISSN: 0378-8733. DOI: http://dx.doi.org/10.1016/0378-8733(91)90008-H. URL: http://www.sciencedirect.com/science/article/pii/037887339190008H (cit. on pp. xxvi, xxvii, 52, 55, 60, 62, 96).

[356] Francesca Spezzano, VS Subrahmanian, and Aaron Mannes. "Reshaping terrorist networks". In: *Communications of the ACM* 57.8 (2014), pp. 60–69. DOI: 10.1145/2632661.2632664 (cit. on p. 65).

[357] Steven H. Strogatz. "Exploring complex networks". In: *Nature* 410 (Mar. 2001), pp. 268–276 (cit. on pp. xxvi, 1, 7).

[358] K. Sugiyama, S. Tagawa, and M. Toda. "Methods for Visual Understanding of Hierarchical System Structures". In: *IEEE Transactions on Systems, Man, and Cybernetics* 11.2 (Feb. 1981), pp. 109–125. ISSN: 0018-9472. DOI: 10.1109/TSMC.1981.4308636 (cit. on p. 42).

[359] P.G. Sun, L. Gao, and S. Shan Han. "Identification of overlapping and non-overlapping community structure by fuzzy clustering in complex networks". In: *Information Sciences* 181.6 (2011), pp. 1060–1071 (cit. on p. 88).

[360] Michael Szell, Renaud Lambiotte, and Stefan Thurner. "Multirelational organization of large-scale social networks in an online world". In: *Proceedings of the National Academy of Sciences* 107.31 (July 2010), pp. 13636–13641. DOI: 10.1073/pnas.1004008107. eprint: http://www.pnas.org/content/107/31/13636.full.pdf. URL: http://www.pnas.org/content/107/31/13636.abstract.

[361] Robert Endre Tarjan. "Depth-First Search and Linear Graph Algorithms". In: *SIAM J. Comput.* 1.2 (1972), pp. 146–160 (cit. on p. 2).

[362] Alexandru Telea and Jarke J. van Wijk. "Visualization of Generalized Voronoi Diagrams". In: *Data Visualization 2001: Proceedings of the Joint Eurographics — IEEE TCVG Symposium on Visualization in Ascona, Switzerland, May 28–30, 2001*. Ed. by David S. Ebert, Jean M. Favre, and Ronald Peikert. Vienna: Springer Vienna, 2001, pp. 165–174. ISBN: 978-3-7091-6215-6. DOI: 10.1007/978-3-7091-6215-6_18. URL: http://dx.doi.org/10.1007/978-3-7091-6215-6_18 (cit. on p. 38).

[363] T.P. Thornberry et al. "The role of juvenile gangs in facilitating delinquent behavior". In: *Journal of research in Crime and Delinquency* 30.1 (1993), pp. 55–87 (cit. on p. 114).

[364] M. Todd and A. Nomani. *The Truth Left Behind: Inside the Kidnapping and Murder of Daniel Pearl*. New York (2011) - http://www.publicintegrity.org/2011/01/20/2190/, 2011 (cit. on p. 79).

[365] Noemi Toth et al. "The importance of centralities in dark network value chains". In: *The European Physical Journal Special Topics* 222.6 (2013), pp. 1413–1439. ISSN: 1951-6355. DOI: 10.1140/epjst/e2013-01935-7 (cit. on p. 52).

[366] Vincent A Traag et al. "Social event detection in massive mobile phone data using probabilistic location inference". In: *2011 IEEE 3rd international conference on social computing (socialcom)*. IEEE. 2011, pp. 625–628 (cit. on p. 110).

[367] J. Travers and S. Milgram. "An experimental study of the small world problem". In: *Sociometry* 32.4 (1969), pp. 425–443. ISSN: 0038-0431.

[368] Anne Treisman. "Preattentive Processing in Vision". In: *Comput. Vision Graph. Image Process.* 31.2 (Aug. 1985), pp. 156–177. ISSN: 0734-189X. DOI: 10.1016/S0734-189X(85)80004-9. URL: http://dx.doi.org/10.1016/S0734-189X(85)80004-9 (cit. on p. 31).

[369] J. Ugander et al. "The anatomy of the Facebook social graph". In: *arXiv preprint arXiv:1111.4503* (2011) (cit. on pp. 115, 121, 131).

[370] Y. Umuroglu, D. Morrison, and M. Jahre. "Hybrid breadth-first search on a single-chip FPGA-CPU heterogeneous platform". In: *2015 25th International Conference on Field Programmable Logic and Applications (FPL)*. Sept. 2015, pp. 1–8 (cit. on p. 141).

[371] Lois M. Verbrugge. "Multiplexity in Adult Friendships". In: *Social Forces* 57.4 (June 1979), pp. 1286–1309.

[372] Alessandro Vespignani. "Predicting the behavior of techno-social systems". In: *Science* 325.5939 (2009), p. 425 (cit. on p. 95).

[373] Le Vy. "Organised Crime Typologies: Structure, Activities and Conditions". In: *International Journal of Criminology and Sociology* 1 (2012), pp. 121–131 (cit. on p. 52).

[374] Xiaofeng Wang, Matthew S Gerber, and Donald E Brown. "Automatic crime prediction using events extracted from twitter posts". In: *Social Computing, Behavioral-Cultural Modeling and Prediction*. Springer, 2012, pp. 231–238 (cit. on p. 95).

[375] Colin Ware. "Designing with a 2 and half dimension attitude". In: *Information Design Journal* 10.3 (2001), pp. 171–182 (cit. on p. 41).

[376] Colin Ware. *Information Visualization: Perception for Design*. 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2012. ISBN: 9780123814647, 9780123814654 (cit. on pp. 31–33).

[377] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Ed. by Mark Granovetter. Structural Analysis in the Social Sciences. Cambridge University Press, 1994. ISBN: 9780521387071. DOI: 10.1017/CBO9780511815478 (cit. on pp. xxvi, 1, 5, 49, 56–58, 65, 98, 104).

[378] D.J. Watts and S.H. Strogatz. "Collective dynamics of 'small-world' networks". In: *Nature* 393.393 (1998), pp. 440–442. ISSN: 0028-0836. DOI: 10.1038/30918 (cit. on pp. xxv, xxvii, 1, 5, 7, 8, 57).

[379] Duncan J. Watts. "The "New" Science of Networks". In: *Annual Review of Sociology* 30.1 (2004), pp. 243–270 (cit. on p. xxvii).

[380] Lilian Weng et al. "Competition among memes in a world with limited attention". In: *Scientific Reports* 2 (2012) (cit. on p. 95).

[381] Uffe Kock Wiil, Jolanta Gniadek, and Nasrullah Memon. "Measuring Link Importance in Terrorist Networks." In: *ASONAM*. Ed. by Nasrullah Memon and Reda Alhajj. IEEE Computer Society, 2010, pp. 225–232. ISBN: 978-0-7695-4138-9. URL: http://dblp.uni-trier.de/db/conf/asunam/asonam2010.html#WiilGM10 (cit. on p. 79).

[382] P. Williams. "Transnational Criminal Networks". In: *Networks and Netwars: The Future of Terror, Crime, and Militancy*. RAND Corporation, 2001. Chap. 3, pp. 61 –98. DOI: 10.1080/00396338.2002.9688556 (cit. on p. 63).

[383] Graham J. Wills. "NicheWorks — Interactive visualization of very large graphs". In: *Graph Drawing: 5th International Symposium, GD '97 Rome, Italy, September 18–20, 1997 Proceedings*. Ed. by Giuseppe DiBattista. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 403–414. ISBN: 978-3-540-69674-2. DOI: 10.1007/3-540-63938-1_85. URL: http://dx.doi.org/10.1007/3-540-63938-1_85 (cit. on p. 37).

[384] C. Wilson et al. "User interactions in social networks and their implications". In: *Proceedings of the 4th ACM European conference on Computer systems*. ACM. 2009, pp. 205–218 (cit. on p. 139).

[385] Christopher Winship and Michael Mandel. "Roles and Positions: A Critique and Extension of the Blockmodeling Approach". In: *Sociological Methodology*. Ed. by Samuel Leinhardt. Jossey-Bass, 1984.

[386] O. Woolley-Meza et al. "Complexity in human transportation networks: a comparative analysis of worldwide air transportation and global cargo-ship movements". In: *The European Physical Journal B* 84.4 (2011), pp. 589–600. ISSN: 1434-6036. DOI: 10.1140/epjb/e2011-20208-9 (cit. on p. xxvi).

[387] William Wright et al. "The Sandbox for Analysis: Concepts and Methods". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '06. New York, NY, USA: ACM, 2006, pp. 801–810. ISBN: 1-59593-372-7. DOI: 10.1145/1124772.1124890. URL: http://doi.acm.org/10.1145/1124772.1124890 (cit. on pp. xxviii, 75, 99).

[388] Jierui Xie, Stephen Kelley, and Boleslaw K Szymanski. "Overlapping community detection in networks: the state of the art and comparative study". In: *ACM Computing Surveys* 45.4 (2013) (cit. on p. 59).

[389] J. Xu and H. Chen. "CrimeNet explorer: a framework for criminal network knowledge discovery". In: *ACM Transactions on Information Systems (TOIS)* 23.2 (2005), pp. 201–226 (cit. on pp. xxvii, 75, 95, 98, 117).

[390] Jennifer Xu and Hsinchun Chen. "Criminal network analysis and visualization". In: *Communications of the ACM* 48.6 (2005), pp. 100–107 (cit. on pp. 73, 74, 95, 99).

[391] Jennifer Xu et al. "Analyzing and visualizing criminal network dynamics: A case study". In: *Intelligence and Security Informatics*. Springer, 2004, pp. 359–377 (cit. on p. 95).

[392] Christopher Yang, Hsinchun Chen, and Kay Hong. "Visualization of Large Category Map for Internet Browsing". In: *Decis. Support Syst.* 35.1 (Apr. 2003), pp. 89–102. ISSN: 0167-9236 (cit. on pp. 47, 105).

[393] Christopher Yang, Nan Liu, and Marc Sageman. "Analyzing the terrorist social networks with visualization tools". In: *Intelligence & security informatics*. 2006 (cit. on p. 98).

[394] Jianmei Yang, Wenjie Wang, and Guanrong Chen. "A two-level complex network model and its application". In: *Physica A: Statistical Mechanics and its Applications* 388.12 (2009), pp. 2435 –2449. ISSN: 0378-4371. DOI: http://dx.doi.org/10.1016/j.physa.2009.02.046. URL: http://www.sciencedirect.com/science/article/pii/S0378437109001563 (cit. on p. xxv).

[395] K.P. Yee et al. "Animated Exploration of Dynamic Graphs with Radial Layout". In: *Proc. IEEE Symposium on Information Visualization*. INFOVIS '01. Washington, DC, USA: IEEE Computer Society, 2001, p. 43. ISBN: 0-7695-1342-5. URL: http://dl.acm.org/citation.cfm?id=580582.857705 (cit. on pp. 37, 76).

[396] Ji Soo Yi et al. "Toward a Deeper Understanding of the Role of Interaction in Information Visualization". In: *IEEE Transactions on Visualization and Computer Graphics* 13.6 (Nov. 2007), pp. 1224–1231. ISSN: 1077-2626. DOI: 10.1109/TVCG.2007.70515. URL: http://dx.doi.org/10.1109/TVCG.2007.70515 (cit. on p. 45).

[397] Wayne Zachary. "An information flow model for conflict and fission in small groups". In: *Journal of Anthropological Research* 33.4 (1977), pp. 452–473. ISSN: 0091-7710.

[398] Hui Zang, Francois Baccelli, and Jean Bolot. "Bayesian inference for localization in cellular networks". In: *2010 Proceedings IEEE INFOCOM*. IEEE. 2010, pp. 1–9 (cit. on p. 110).

[399] Yang Zhang and Eric A. Hansen. "Parallel breadth-first heuristic search on a shared-memory architecture". In: *In AAAI Workshop on Heuristic Search, Memory-Based Heuristics and Their Applications*. 2006 (cit. on p. 139).

[400] Ales Ziberna. "Generalized blockmodeling of valued networks." In: *Social Networks* 29.1 (2007), pp. 105–126.

DECLARATION

I herewith declare that I have produced this paper without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This Thesis has not previously been presented in identical or similar form to any other Italian or foreign examination board. The thesis work was conducted from January 2014 to November 2016 under the supervision of Prof. Giacomo Fiumara at the University of Messina.

Catania, November 2016

Salvatore A. Catanese