SAPIENZA
UNIVERSITÀ DI ROMA

Bounds On The Maximal Number Of Corrupted Nodes Via Boolean Network Tomography

Faculty of Engineering, Computer Science and Statistics

Dottorato di Ricerca in Computer Science – XXXIII Ciclo

Candidate

Fariba Ranjbar
ID number 1792535

Thesis Advisor

Prof. Nicola Galesi

Academic Year 2020/2021

Thesis defended on July 8, 2021;
in front of a Board of Examiners composed by:

Prof. Emanuele Rodolà (chairman)

Prof. Maurizio Bonuccelli

Prof. Dario Catalano

Prof. Andrea Marin

Bounds On The Maximal Number Of Corrupted Nodes Via Boolean Network Tomography

Ph.D. thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: fariba.ranjbar@uniroma1.it, ranjbar@di.uniroma1.it

Sapienza University of Rome

Faculty of Engineering, Computer Science and Statistics

Bounds On The Maximal Number Of Corrupted Nodes Via Boolean Network Tomography

by

Fariba Ranjbar

Ph.D. Thesis
Rome, July 2021

Thesis Committee
Prof. Nicola Galesi
Prof. Michele Zito
Prof. Annalisa Massini

Reviewers
Prof. Liang Ma
Prof. Francesco Lo Presti

to my lovely family,

Abstract

In this thesis we are concentrating on identifying defective items in larger sets which is a main problem with many applications in real life situations, e.g., fault diagnosis, medical screening and DNA screening. We consider the problem of localizing defective nodes in networks through an approach based on Boolean Network Tomography (BNT), which is grounded on inferring informations from the Boolean outcomes of end-to-end measurement paths. In particular, we focus on the following three:

- Studying *Maximal Identifiability*, which was recently introduced in BNT to measure the maximal number of corrupted nodes which can be uniquely localized in sets of end-to-end measurement paths on networks;
- Central role of *Vertex-Connectivity* in maximal identifiability;
- Investigating identifiability conditions on the set of paths which guarantee discovering or counting unambiguously the defective nodes and contributing this problem both from a theoretical and applied perspectives.

We prove tight upper and lower bounds on the maximal identifiability for sets of end-to-end paths in network topologies obtained from trees and d-(dimensional) grids over n^d nodes. For trees (both directed and undirected) we show that the maximal identifiability is 1. For undirected d-grids we prove that, using only $2d$ monitors, maximal identifiability is at least $d-1$ and at most d. In the directed case proving that the maximal identifiability is d and can be reached at the cost of placing $2d(n-1)+2$ monitors on the d-grid. This monitor placement is optimal and adding more monitors will not increase the identifiability. We also study maximal identifiability for directed topologies under embeddings establishing new relations with embeddability, graph dimension and proving that under the operation of transitive closure maximal identifiability grows linearly.

Our results suggest the design of networks over n nodes reaching maximal identifiability $\Omega(\log n)$ using $O(\log n)$ monitors and a heuristic to boost maximal identifiability increasing the minimal degree of the network which we test experimentally.

Moreover we prove tight bounds on the maximal identifiability first in a particular class of graphs, the Line of Sight networks and then slightly weaker bounds for arbitrary networks. Furthermore we initiate the study

of maximal identifiability in random networks. We investigate two models: the classical Erdős-Rényi model, and that of Random Regular graphs. The proposed framework allows a probabilistic analysis of the identifiability in random networks giving a tradeoff between the number of monitors to place and the maximal identifiability.

Further in this thesis, we work on the precise tradeoff between number of nodes and number of paths such that at most k nodes can be identified unambiguously. The answer to this problem is known only for $k = 1$ and we answer it for any k, setting a problem implicitly left open in previous works. We focus on upper and lower bounds on the number of unambiguously identifiable nodes, introducing new identifiability measures (Separability and Distinguishability) which strictly imply and are strictly implied by the notion of identifiability introduced in [39]. We utilize these new measures to design algorithmic heuristics to count failure nodes in a fine-grained way and further to prove the first complexity hardness results on the problem of identifying failure nodes in networks via BNT.

At last but not least, we introduce a random model so as to achieve lower bounds on the number of unambiguously identifiable defective nodes. We use this model to approximate that number on real networks by a maximum likelihood estimate approach.

Acknowledgments

Throughout the writing of this dissertation I have received a great deal of support and assistance. I would first like to express my sincere gratitude to my advisor Professor Nicola Galesi whose expertise was invaluable in formulating the research questions and methodology. He made me interested into Boolean Network Tomography and Parameterized Complexity. His insightful feedback pushed me to sharpen my thinking and brought my work to a higher level. This thesis without him simply wouldn't have existed.

I would like to acknowledge my co-author of one of my papers that led to this thesis, Professor Michele Zito for his collaboration. Thank you for expressing your thoughts so eloquently. The meetings and conversations were vital in inspiring me to think outside the box, from multiple perspectives to form a comprehensive and objective critique.

More in general, I am grateful to the many people from the theory group of the Computer Science department and the Network Tomography community that I had the privilege to meet in the last years. I wish to thank Navid Talebanfard, Liang Ma and Annalisa Massini for insightful discussions around Boolean Network Tomography.

In addition, I would like to thank my family, in particular my brother Professor Saeed Ranjbar for his wise counsel. I could not have completed this dissertation without their unconditional support in this very intense academic year.

Last, but just in order of time, I want to thank the reviewers of this thesis for their careful reading.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 What Is Network Tomography?

A central issue in communication networks is to ensure that the structure works reliably. To this end detecting as quickly as possible those components that develop some sort of failure is of the prime priority. In other words, monitoring a network to localize corrupted components is essential to guarantee a correct behaviour and the reliability of a network. In general, identifying a subset of defective items out of a much larger set of items is a problem that has numerous applications in a variety of situations such as medical screening, network reliability, DNA screening, streaming algorithms. In many real networks direct access and direct monitoring of the individual components are not possible (for instance because of limited access to the network) or unfeasible in terms of available resources (protocols, communications, response-time etc.).

Network Tomography as a way to check network reliability

A well-studied approach to localization of failing components is *Network Tomography*. Network Tomography is a general inference technique based on end-to-end *measurements* aimed to extract internal network characteristics such as link delays and link loss rates but also defective items. Network Tomography focuses on detecting the state of single components in the network by running a *measurement* process along the network. The process starts by sending packets (containing suitable data to capture interesting failures) from specific *input monitor* nodes and terminates receiving another data packet on other specific *output monitor* nodes. Otherwise stated Network Tomography is a family of distributed failure detection algorithms

based on the spreading of end-to-end measurements [12, 49] rather than directly measuring individual network components.

Typically a network $G = (V, E)$ is given as a graph along with a collection of paths $\mathbb{P}$ in it and the goal is to take measurements along such paths to infer properties of the given network. Quoting from [16] "A key advantage of tomographic methods is that they require no participation from network elements other than the usual forwarding of packets. This distinguishes them from well-known tools such as `trace-route` and `ping`, that require ICMP responses to function. In some networks, ICMP response has been restricted by administrators, presumably to prevent probing from external sources. Another feature of tomography is that probing and the recovery of probe data may be embedded within transport protocols, thus co-opting suitably enabled hosts to form impromptu measurement infrastructures".

Connection to group testing

The approach is strongly related to group testing [15] where, in general, one is interested in making statements about individuals in a population by taking group measurements. In other words, the problem of group testing concerns the design and evaluation of tests on pools of items (a pool is a selected subset of all items) to identify the defective items. The main concern is to do so with the minimum number of tests. In our setting, for instance the connectivity structure of the network constrains the set of feasible tests. Graph-constrained group testing has been studied before, starting with [11]. One of our main interests is using structural graph-theoretic properties to make statements about the quality of the testing process.

Research in Network Tomography is vast. The seminal works of Vardi [49], and Coates *et al.* [12], or more recent surveys like [10] each have more that 500 citations, according to Google Scholar. Methods and algorithms vary dramatically depending on the network property of interest, or the measurements one has to rely on.

Boolean Network Tomography

In this thesis we focus on the problem of detecting *node states* (failing/working), using a *Boolean Network Tomography* approach [16, 17]. Duffield, who as first introduced Boolean Network Tomography [17] to identify network failure components, proposed an inference algorithm based on Boolean Network Tomography to identify the set of failure links. The Boolean Network Tomography approach was later studied also to identify node failures in networks [4, 24, 26, 37–39]. Boolean Network Tomography (BNT) aims to identify corrupted components in a network using Boolean

measurements (i.e. assuming that elementary network components can be in one of two states: "working" or "failing"). A (Boolean) *measurement* is done along a set of *end-to-end* paths, each one starting and ending with a monitor node and moreover the final measurement in the path[1] at each monitor is one bit (failure (1) /working (0)), capturing the presence or the absence of a failure along a path. Introduced in [16], the paradigm has recently attracted a lot of interest [29, 38] because of its importance.

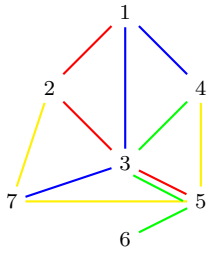
We are interested in conditions under which identifying (uniquely) failure nodes. Assume to have a set $\mathbb{P}$ of measurement paths over a node set V. We would like to know the state x_v (with $x_v = 0$ corresponding to "v in working order" and $x_v = 1$ corresponding to "v in a faulty state") of each node $v \in V$. In simple words, Receiving a 0 (working state) at an output monitor of a path means that each node in the path is working properly. Then the localization of failing nodes in a set of paths $\mathbb{P}$ is captured by the solutions to the following Boolean system: *Boolean system*

$$\bigwedge_{p \in \mathbb{P}} \left(\bigvee_{v \in p} x_v \equiv b_p \right) \tag{1.1}$$

where $\vec{b}_p$ is a vector of Boolean values (corresponding to final measurement in the paths) and x_v's are Boolean variables, one for each node v. Any solution to this system is a possible location of node-failures satisfying the measurements. Hence in the case of identifying failure nodes, the BNT approach deals with extracting as much information as possible on the number and the positions of the corrupted nodes from the solutions $\vec{x}$ of a Boolean system $\mathbb{P}\vec{x} = \vec{b}$, where $\mathbb{P}$ is the incidence matrix of the m measurement paths over the n nodes and $\vec{b}$ is the m-vector of the Boolean outcomes of the measurement paths (see Figure 1.1).

The solutions to Eq. 1.1 are often multiple. Namely, systems of this form may have several solutions and therefore, in general, the availability of a collection of end-to-end measurements does not necessarily lead to the unique identification of the failing nodes. This leads to pose the following question: Given the set of paths $\mathbb{P}$, what is the maximal set of nodes we can hope to identify unambiguously? Identifiability conditions on the matrix $\mathbb{P}$ under which failure nodes can be localized unambiguously (or also counted accurately) from the solution of the system $\mathbb{P}\vec{x} = \vec{b}$ are of course of the utmost interest. We will investigate properties of the underlying network

[1] As a generic model, we assume failure if no result is obtained at the terminal monitor after the given time.



$\mathbb{P}$	1	2	3	4	5	6	7	$\vec{b}$
1	1	1	1	0	1	0	0	1
2	1	0	1	1	0	0	1	0
3	0	0	1	1	1	1	0	1
4	0	1	0	1	1	0	1	1

$$\begin{cases} x_1 \vee x_2 \vee x_3 \vee x_5 = 1 \\ x_1 \vee x_3 \vee x_4 \vee x_7 = 0 \\ x_3 \vee x_4 \vee x_5 \vee x_6 = 1 \\ x_2 \vee x_4 \vee x_5 \vee x_7 = 1 \end{cases}$$

Figure 1.1. (1) A set $\mathbb{P}$ of 4 paths over 7 nodes. (2) The incidence matrix of $\mathbb{P}$ and a measurement vector $\vec{b}$. (3) The associated Boolean system. Notice that the outcome 1 in the measurement of a path indicates the presence of at least a node failure.

that facilitate the solution of this problem. In particular, we follow the approach initiated by Ma et. al. [39] and later studied in [37,38] based on the notion of *maximal identifiability* (see Section 2.4 Definition 2.4.1 for a precise definition). The parameter aims to count the maximal number of simultaneously failure nodes which are uniquely identifiable along a set of measurement paths through a Boolean measurement. It turns out that the network maximal identifiability is an interesting combinatorial measure.

Maximal identifiability

The condition introduced as *k-identifiability* (for $\mathbb{P}$) states that any two distinct node sets of size at most k can be separated by at least a path in $\mathbb{P}$. k-identifiability initially introduced for link failure detection [36,41], was later studied with success also for node failure detection [4,24,26,37–39]. If this condition is true for a set of measurement paths $\mathbb{P}$, it ensures that if there are at most k failure nodes in $\mathbb{P}$ then these nodes can be identified unambiguously. Hence the optimization problem of computing the maximal $k \leq n$ such that a set $\mathbb{P}$ is k-identifiable (k-ID) admits that the k-identifiability property is very relevant to the problem of node failure localization. We refer to this maximal value as $\mu(\mathbb{P})$ (it was called $\Omega(\mathbb{P})$ in [37]).

Maximal identifiability was recently studied under several aspects, in-

cluding network topologies, routing protocols and probing mechanisms. Ma et. al. in [37, 38] investigated under what conditions one can uniquely localize failed nodes from path measurements available in the entire network. Such questions depend on the network topology, on the placement of monitors, and the implemented routing mechanism. Several studies [3, 24, 39, 41] have investigated variants of this measure in connection with various types of path systems. However, it seems difficult to come up with simple graph-theoretic properties that affect the given network identifiability. We contend that the maximal identifiability using measures over the collection of all simple paths between two sets of vertices S and T enables us to make good progress on this issue. The works [4, 37, 38] focused on how much to increase the number of monitors on generic topologies and what nodes to link them, with the aim of optimizing maximal identifiability. However structural limitations due to the network topology might affect the feasibility of such approaches. For instance, as we notice in this thesis, the minimal degree of the graph modeling the network is a structural limit on the maximal identifiability one can hope for, independently of the number of monitors and their placement. Despite of the importance of detecting failing nodes in real networks, there is still a lack of a complete understanding of what maximizing identifiability of failure nodes requires in terms of network properties as the topology and the monitor placement. Our work contributes to this line of research.

1.2 Main Results

Network Tomography provides one of the main approach to detect failure components in networks. The identification of failure nodes via Boolean Network Tomography is a computational problem of Boolean nature. The purpose of this dissertation is to contribute both from a theoretical and applied perspective to the understanding of node failure identification via Boolean Network Tomography. In our analysis we use basic tolls from discrete math in Graph Theory (Menger's Theorem), Extremal Combinatorics (Union-Free Families), Complexity Theory (NP-Completeness) and Probability (Maximal Likelihood Estimate).

Network Tomography is an approach based on distributed algorithms aimed to detect fault components in networks through end-to-end path measurements. In this thesis we focus on the problem of detecting failing

nodes in networks through binary (working/failing) measurements. We consider the problem of detecting the maximal number of failing nodes that can be simultaneously detected in a network as a measure of network's reliability, an approach devised in [9, 16–18, 32, 37–39]. The study of this measure turns out to be an interesting combinatorial problem with several real-world applications.

Node failure identification is carried on set of (end-to-end) measurement paths that we usually denote with $\mathbb{P}$. When we study failure identification in networks whose topology is defined by a graph $G = (V, E)$ (either directed or undirected), we specify a monitor placement on G by $\chi = (S, T)$ where $S, T \subseteq V$. Expressly, we consider $\mathbb{P} = \mathbb{P}_\chi(G) =$ the set of all paths in G from a node in S to a node in T. Now in this thesis we are focusing on:

1. studying how structural properties of G limit the maximal identifiability of nodes in $\mathbb{P}$, cf. Chapter 3;

2. studying upper and lower bounds for maximal identifiability when G is a tree, a d-dimensional grid, an augmented hypergrid and random networks, possibly independently of the monitor placement, cf. Chapter 4;

3. understanding how embeddability between directed graphs interferes with the maximal identifiability, cf. Chapter 3;

4. exploring experimentally a heuristic (suggested by the results in items 2 and 3) to increase maximal identifiability by increasing the minimal degree of the network topology, cf. Chapter 6;

5. studying the maximal identifiability of any network G using vertex-connectivity $\kappa(G)$ and other tools in Graph Theory such as Menger's Theorem, cf. Chapter 3;

6. combinatorial upper bounds on the maximal number of failure nodes using union-free families, cf. Chapter 5;

7. studying lower bounds on maximal identifiability using random models, cf. Chapter 5 and 6;

8. counting and localizing failure nodes in real networks, cf. Chapter 5 and 6;

9. complexity of k-identifiability and the minimum hitting set, cf. Chapter 6.

Structural results

The first results we show are structural. Namely we identify how maximal identifiability depends on the minimal degree (both in the directed and undirected case), cf. Theorems 3.2.1 and 3.2.3, on the number of monitors (Theorem 3.1.1) and the number of edges and nodes, cf. Corollary 3.2.2. It is noteworthy to mention that we provide heuristics on how to design topologies and related monitoring schemes to achieve the maximum identifiability under various network settings.

Trees

Later, we prove that when the topology G is a tree (both directed or not), the maximal identifiability is very low, namely 1, cf. Theorems 4.1.1 and 4.1.4. This result means that the maximal number of failing nodes we can uniquely identify on such topologies is 1.

Grids

In searching for topologies where maximal identifiability is greater than 1, we considered the case of grids (directed and not). We prove that in 2-dimensional grids we can reach a maximal identifiability strictly greater than 1, namely 2. Our analysis easily extends to the case of d-dimensional grids where we prove that the maximal identifiability is the dimension d. We prove these results for both directed and undirected grids, cf. Theorems 4.2.1 and 4.2.9. In the former case we prove a tight bound of d for the maximal identifiability under a suitable monitor placement of $O(nd)$ monitors and we prove this is optimal. For undirected d-grids we greatly improve the directed case. First we show a $d-1$ lower bound and a d upper bound using only d input and d output monitors (so independently of n). Second, we prove the bounds *for any* monitor placement of d input and d output monitors. To prove our lower bounds we use a combinatorial analysis, which hence leads to algorithms to design network topologies with a guarantee of reaching a tight maximal identifiability. Our results are the first known on precise bounds for maximal identifiability for specific network topologies.

Embeddings and transitive closure

d-grids are related to the *dimension* of directed acyclic graphs (DAG) through the operation of *embedding*. Namely the dimension of a DAG G , is the smallest integer d such that G is embeddable in the d-grid [5]. Motivated by the results on d-grids. we start the study of maximal identifiability under isomorphic embeddings of DAGs. While the most general definition of embedding can drastically decrease maximal identifiability, yet we explore two directions: (1) restricting the class of topologies we want to embed (Theorem 3.4.2) and (2) restricting the mapping that defines the embedding, cf. Theorem 3.4.4 and Corollary 3.4.5. In both cases we show new results on

how maximal identifiability can be preserved under embeddings. Among our results we prove that the k-transitive closure on directed graphs increases the maximal identifiability linearly in k, cf. Theorem 3.4.12.

Hypergrids

d-grids are examples of concrete topologies which reach an optimal value of the maximal identifiability. The results on embeddability suggest that for increasing the maximal identifiability of real networks (which often are very low since many real topologies are trees, quasi-trees or grids) one can try to add edges to the network in such a way to get closer to a graph which is embeddable into a d-hypergrid.

We propose an algorithm `Agrid` that given a network G and a parameter d outputs a new network $G^{\mathtt{A}}$ having minimal degree d by adding random edges. We discuss the feasibility of `Agrid` on real networks, and we test it on examples of real networks, as well as on random graphs and also for random placement of monitors, obtaining positive results.

All the above mentioned results in this thesis rely on the following works:

[24] N. GALESI, AND F. RANJBAR, Tight bounds for maximal identifiability of failure nodes in boolean network tomography. In *38th IEEE International Conference on Distributed Computing Systems, ICDCS 2018, Vienna, Austria, July 2-6, 2018* (2018), IEEE Computer Society, pp. 212–222.

[25] N. GALESI, AND F. RANJBAR, Tight bounds to localize failure nodes on trees, grids and through embeddings under boolean network tomography. Submitted, 2020.

The paper [25] is an extended version of the ICDCS paper [24].

Maximal identifiability of LoS networks

More specifically, we show that the proposed approach provides an almost tight characterization of the maximal identifiability in *augmented hypergrids* and more general Line-of-Sight (LoS) networks (see definition in Section 2.2). LoS networks were introduced by Frieze *et al.* in [23] and have been widely studied (see for instance [13, 14, 42, 43]) as models for communication patterns in a geometric environment containing obstacles. Like grids, LoS networks can be embedded in a finite cube of $\mathbb{Z}^d$, for some positive integer d. But LoS networks generalize grids that edges are allowed between nodes that are not necessarily next to each other in the network embedding. We employed *Menger's Theorem* establishing a precise relation of $\mu(\mathbb{P})$ with the vertex connectivity of the graph (i.e. the size of the

Vertex-connectivity and Menger's Theorem

minimal set of nodes disconnecting the graph) underlying $\mathbb{P}$.

Using the network vertex-connectivity, $\kappa(G)$, we are able to prove the following:

Theorem 1.2.1. *Let $\mathcal{H}$ be an undirected augmented hypergrid. For every pair of disjoint $S, T \subseteq V(\mathcal{H})$, the maximal identifiability of $\mathcal{H}$, $\mu(\mathcal{H})$ using measures over simple paths between S and T satisfies: $\mu(\mathcal{H}) \leq \kappa(\mathcal{H})$. Furthermore, for all pair of disjoint S and T we have $\mu(\mathcal{H}) \geq \kappa(\mathcal{H}) - 1$.*

The result on hypergrids immediately suggests the related question about general graphs. In this thesis we prove upper and lower bounds on the maximal identifiability of any network G. The following statement summarizes our findings (here $\kappa_{ST}(G)$ is the size of smallest set of vertices separating S and T):

Theorem 1.2.2. *Let $G = (V, E)$ be an undirected graph. For every pair of disjoint $S, T \subseteq V(G)$, the maximal identifiability of G, $\mu(G)$ using measures over simple paths between S and T satisfies: $\mu(G) \leq \min(\delta(G), \kappa_{ST}(G))$. Furthermore, there is a way to choose S and T that guarantees $\mu(G) \geq \lfloor \kappa(G)/2 \rfloor - 1$.*

In both results, the upper bound is proved by showing that there are sets of $\kappa(G) + 1$ vertices that cannot be identified. The lower bounds which require the construction of paths separating large sets of nodes in the graph, are based on a well-known relationship between $\kappa(G)$ and the existence of collections of vertex-disjoint paths between certain sets of nodes in G. In fact a much higher lower bound can be proved for graphs with low connectivity. The following result applies to arbitrary LoS networks, and to many topologies studied in relation to communication problems including various types of grids, butterflies, hypercubes, and sparsely connected sensor networks.

Theorem 1.2.3. *Let $G = (V, E)$ be an undirected network with $\kappa(G) \leq |V|/3$. Let $\mu(G)$ denote the maximal identifiability of G using measures over simple paths between two disjoint sets of vertices S and T.*

1. *For all pairs of disjoint $S, T \subseteq V$, $\mu(G) \leq \kappa(G)$.*
2. *There is a pair of disjoint $S, T \subseteq V(G)$ such that $\mu(G) \geq \kappa(G) - 2$.*

Furthermore, we look at random networks (Erdős-Rényi and Random Regular Graphs), cf. Theorems 4.4.4 and 4.4.5. In these structures we are able to show a trade-off between the success probability of the relevant path

construction processes and the size of the sets S and T defining the path set $\mathbb{P}$. Random graphs also give us alternative constructions of networks with large identifiability.

All these results on LoS networks, arbitrary graphs and random graphs are based on:

[26] N. GALESI, F. RANJBAR AND M. ZITO. Vertex-connectivity for node failure identification in boolean network tomography. In *Algorithms for Sensor Systems - 15th International Symposium on Algorithms and Experiments for Wireless Sensor Networks, ALGOSENSORS 2019, Munich, Germany, September 12-13, 2019, Revised Selected Papers* (2019), F. Dressler and C. Scheideler, Eds., vol. 11931 of Lecture Notes in Computer Science, Springer, pp. 79–95.

Finally, we try to understand the combinatorics and the complexity of the theoretical problem of unambiguously identifying failure node sets under the BNT approach, and moreover to devise new algorithms and heuristics to count or localize failure nodes in networks as more precisely as possible. we introduce new identifiability measures and deepen the study of k-identifiability obtaining several new results and new heuristics to test networks against the number and position of failing nodes.

Tradeoffs between paths and nodes

In this matter, our first contribution is regarding the minimal number of measurement paths m on n nodes such that below m, $\mathbb{P}$ is not k-identifiable. We set this question about limits of upper bounds on identifiability of failures via Boolean Network Tomography which is implicitly left open in some previous works [4, 36]. What are the precise tradeoffs between number of nodes n and number of paths m of $\mathbb{P}$ such that $\mathbb{P}$ is no longer k-identifiable, that is $\mu(\mathbb{P}) < k$? The answer is known only for $k = 1$ where the tradeoff $n \geq 2^m - 1$ implies $\mu(\mathbb{P}) < 1$, and it is obtained by a straightforward counting argument (see Lemma 5.2.1 and [4]). Using the notion of *regular union-free* families, we answer to the problem for any $2 \leq k \leq n$, showing that $n \geq 2^{\frac{k}{k-1}(m+k-1)^{(1+\epsilon)}}$ implies $\mu(\mathbb{P}) < k$, for any $\epsilon > 0$. (Theorem 5.2.7 and Corollary 5.2.8).

The result mentioned above can be used as an estimate of upper bounds on the number of k-identifiable nodes in $\mathbb{P}$. As [4] uses the result for $k = 1$ to prove that $|\mathsf{ID}_1(\mathbb{P})| \leq \min(n, 2^m - 1)$ (see Theorem 5.2.2), our bound proves the general statement that for all $2 \leq k \leq n$, $|\mathsf{ID}_k(\mathbb{P})| \leq \min\{n, 2^{\frac{k}{k-1}(m+k-1)^{(1+\epsilon)}}\}$ (Theorem 5.2.9). Our bound can also

be used as a black-box in algorithms and heuristics aimed at approximating the number of identifiable nodes ([37–39]) which use the bound for $k = 1$. For instance the ICE heuristic of [4], that creates a set of paths $\mathbb{P}$ reaching a certain value of $\mu(\mathbb{P})$, is generating paths according to the result for $k = 1$.

Refining Identifiability

Secondly, we introduce two new identifiability notions, namely, *k-separability* (k-SEP) and *k-distinguishability* (k-DIS)(see Section 5.3). Analogously to identifiability we define these notions on nodes and we consider the corresponding node sets $\mathsf{SEP}_k(\mathbb{P})$ and $\mathsf{DIS}_k(\mathbb{P})$. These conditions provide significant upper and lower bounds to identifiability: namely we prove that for all $k \leq n$, k-SEP *implies* k-ID and k-ID *implies* k-DIS, both strictly. Hence $\mathsf{SEP}_k(\mathbb{P}) \subseteq \mathsf{ID}_k(\mathbb{P}) \subseteq \mathsf{DIS}_k(\mathbb{P})$. We use these measures to get upper and lower bounds for $|\mathsf{ID}_k(\mathbb{P})|$ and $\mu(\mathbb{P})$, to study the computational complexity of identifiability conditions and to estimate the number of k-identifiable nodes through a random model. Namely:

1. *Complexity results and MHS problem*

   We prove that the problem of deciding the non k-identifiability of a given node in $\mathbb{P}$ is polynomial time reducible to the *minimum hitting set* problem (MHS), cf. Theorem 6.4.1. Furthermore we prove that the optimization problem of *finding the minimal k such that a given node is not k-separable* in $\mathbb{P}$ is NP- complete (Theorem 6.4.3). To our knowledge these are the first known hardness results of identifiability problems arising from Boolean Network Tomography. The fact that the MHS problem is reducible to the non-separability problem suggests the idea of using the *minimal hypergraph transversal* (instead of a minimum hitting set) to lower bound the number of separable nodes (hence identifiable nodes) in $\mathbb{P}$. Given an order of the variables a minimum hypergraph transversal in a set-system can be efficiently computed. We propose two algorithms based on the hypergraph transversal (`Simple`-SEP and `Decr`-SEP). In particular in the second algorithm we use a new idea which partitions the set of nodes of $\mathbb{P}$ in family of subsets of nodes called 0-*decreasing* which allows to apply in a more efficient way the hypergraph transversal heuristic (`Decr`-SEP).

2. *A random model to estimate the number of identifiable nodes*

   We introduce and study a random model for $\mathbb{P}$ based on the binomial distribution and we estimate lower bounds on the number of k-identifiable nodes $|\mathsf{ID}_k(\mathbb{P})|$ in this model by analyzing the number of k-separable nodes in $\mathbb{P}$ (Theorem 5.4.2). We employ the random model to approximately count the number of k-identifiable nodes on concrete networks using an approach based on the *maximum likelihood*

estimate for binomial distributions. Our experimental results indicate that a lower bound for the number of k-identifiable nodes of a real network can be computed very accurately using a relatively simple random model based on the binomial distributions and computing the probability that a node is k-separable in this model. We then consider a real set of measurement paths $\hat{\mathbb{P}}$ as it is a random experiment, we plug in the MLE estimates on $\hat{\mathbb{P}}$ in the probability formula of the random model to estimate the cardinality of the set $\mathsf{SEP}_k(\hat{\mathbb{P}})$.

Fine-grained search of failure nodes

3. We use node distinguishability to study upper bounds on the number of k-identifiable nodes parameterizing the search of such nodes in terms of specific subset of nodes and specific subset of paths in $\mathbb{P}$. We introduce the relation (Definition 5.5.1) *u k-equal W modulo $\mathcal{P}$*, where u is node, W a set of nodes and $\mathcal{P}$ a family of paths in $\mathbb{P}$ that characterizes non-distinguishability of u restricted to the set W with respect to $\mathcal{P}$. A recursive construction (Definition 5.5.3 of τ_k) built on the previous relation allows to upper bound efficiently the number of k-identifiable nodes in a fine-grained way (Theorem 5.5.4). In other words, we use the definition of τ_k to upper bound the number of k-identifiable nodes in $\mathbb{P}$ according to specific families of subset of nodes and subset of paths. As we show in Section 5.5, this can be used to compute approximations of the value of $\mu(\mathbb{P})$ and $|\mathsf{ID}_k(\mathbb{P})|$ which are efficiently computable (Algorithm $\texttt{lb-DIS}_k$).

All the mentioned results on counting and localizing failure nodes by Boolean Network Tomography rely on our following paper:

[27] N. GALESI, AND F. RANJBAR, Counting and localizing defective nodes by boolean network tomography. Submitted, 2020. arXiv:2101.04403

1.3 Organization of The Thesis

In the thesis we have firstly studied k-identifiability from the topological point of view of the graph underlying $\mathbb{P}$. Then we generalized our results and established a precise relation of identifiability with the structural properties of our networks especially with vertex connectivity and started the study of identifiability conditions on random graphs and random regular graphs. Furthermore we worked on counting and localizing failure nodes in networks and finally we end this thesis with some algorithms, data and experiments.

Each chapter starts with some more detailed overview of the results/ techniques introduced.

Chapter 2 In this chapter we focus on the preliminary and basic definitions and notations used in the thesis. It is including five sections as the following: *Section 2.1* which is regarding Sets, Graphs, Paths and Connectivity; *Section 2.2* introduces grids and LoS Networks; *Section 2.3* offers a brief introduction on embeddings; and the last two *Sections 2.4* and *2.5* contain the main definitions related to the Boolean Network Tomography and identifiability.

Chapter 3 This chapter consists of several sections, each focusing on bounds on maximal identifiability under some particular structural properties of the networks. In *Section 3.1* we study how the number of input and output nodes effects the maximal identifiability. *Sections 3.2* and *3.5* contain some results on the relation of the minimal degree and vertex connectivity of our networks with the k-identifiability. Moreover in *Section 3.4* we discuss some results on maximal identifiability under isomorphic embeddings of DAGs. Finally this chapter ends with a list of related open problems, cf. *Section 3.6*.

Chapter 4 In this chapter we construct the framework to prove lower and upper bounds on maximal identifiability for some specific networks such as trees (*Section 4.1*), grids (*Section 4.2*), augmented hypergrids (*Section 4.3*) and last but not least random networks, cf. *Section 4.4*. This chapter is largely dependent on Chapter 3.

Chapter 5 This chapter switch the focus from topological and structural point of view to the combinatorics and the complexity point of view. We first introduce some preliminary definitions, cf. *Section 5.1*. Then we study the tradeoffs between number of nodes and number of paths in *Section 5.2*. In the *Section 5.3* we give the definitions of k-separability and k-distinguishability and prove the relation with identifiability. Then in *Section 5.4* we introduce the random model and show how to count k-separable nodes (hence lower bounds on k-identifiable nodes) on real networks through a *maximum likelihood estimate* method. Finally in *Section 5.5* we prove some properties and introduce a corresponding method based on distinguishability to compute upper bounds on identifiable nodes in a

fine-grained way, when the set of paths is obtained by taking all the paths in a graph from a set of source nodes to a set of target nodes.

Chapter 6 This last chapter is consisting of algorithms, data and experiments or in other words all applications of our results in previous chapters. In *Sections 6.1* and *6.2* our results in Chapter 4 culminated with a heuristic `Agrid` to design networks with a high degree of identifiability or to modify a network to boost identifiability and we study the feasibility of `Agrid` and provide some experimental data. In *Section 6.3* we present some experiments on the number of k-separable nodes on real networks using maximum likelihood estimate method on the random model. Lastly, *Sections 6.4* and *6.5* contain algorithms and experimental data based on the results in Chapter 5. We present an algorithm regarding counting k-identifiable nodes in a network on real set of measurement paths. Furthermore we present the results on the computational complexity of k-identifiability and we introduce two algorithms based on hypergraph transversal to count identifiable nodes.

Appendix For the convenience of the reader we collect in an appendix some additional proofs which we have omitted due to their similarity to the other proofs given in the thesis.

Chapter 2

Preliminaries

This chapter gives a concise introduction to most of the terminology used later in the thesis. Fortunately, much of standard graph theoretic terminology is so intuitive that it is easy to remember; However some few terms better understood in their proper setting will be introduced later, when their time has come. *Sections 2.1*, *2.2* and *2.3* offer a brief but precise summary of the most basic definitions in graph theory, those centered round the notion of a graph. Most readers will have met these definitions before, or will have them explained to them as they begin to read this thesis. For this reason, *Sections 2.1*, *2.2* and *2.3* do not dwell on these definitions more than clarity requires: its main purpose is to collect the most basic terms in one place, for easy reference later.

From *Sections 2.4* and *2.5* onwards, primary concepts and definitions related to Network Tomography are being introduced. We become familiar with identifying (uniquely) failure nodes. We study the maximal size of sets of failure nodes one can guarantee identifiability for which is a measure of the ability to identify failure sets uniquely using paths in a Given Network. Furthermore in *Section 2.5* we formally prove that how and why maximal identifiability implies unique localization of sets of failure nodes (Theorem 2.5.2).

2.1 Sets, Graphs, Paths and Connectivity

For a real number x we denote by $\lfloor x \rfloor$ the greatest integer $\leq x$, and by $\lceil x \rceil$ the least integer $\geq x$. For sets U, V, $U \triangle V = (U \setminus V) \cup (V \setminus U)$ is the symmetric difference between U and V.

Symmetric difference

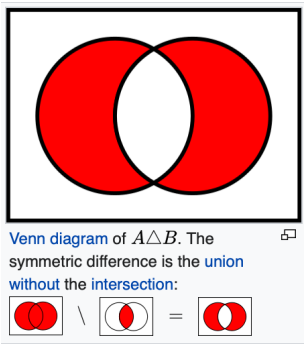


Figure 2.1. Venn diagram of symmetric difference.

Graph

A *graph* is a pair $G = (V, E)$ of sets such that $E \subseteq V \times V$; thus, the elements of E are 2-element subsets of V. The elements of V are the *vertices (or nodes, or points)* of the graph G, the elements of E are its *edges (or lines)* (we will use the terms network and graph interchangeably). The set of *neighbours* of a vertex u in G is denoted by $N_G(u)$, or briefly by $N(u)$ and it is $N(u) = \{v \in V \mid (u, v) \in E\}$. The *degree* of u, $\deg(u)$, is the cardinality of $N(u)$. The number $\delta(G) = \min_{u \in V} \deg(u)$ is the minimum degree of G, the number $\Delta(G) = \max_{u \in V} \deg(u)$ its maximum degree. If all the vertices of G have the same degree r, then G is *r-regular*, or simply regular.

Neighbours of a node

Degree

r-regular

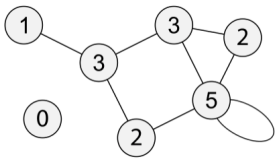


Figure 2.2. A graph with a loop having vertices labeled by degree.

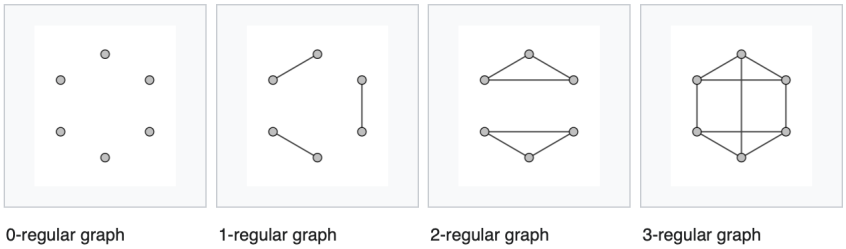


Figure 2.3. Some examples of r-regular graphs.

Path

A *path* p in G from a node u to a node v is a sequence of edges $p = (u_1, u_2) \cdot (u_2, u_3) \cdot \ldots \cdot (u_{k-1}, u_k)$ such that $u_1 = u$ and $u_k = v$ and

$(u_i, u_{i+1}) \in E$ for all $i \in [k-1]$. $\cdot$ is the operation of path concatenation. We often refer to a path by the natural sequence of its vertices, writing, say, $p = u_1 \cdot \ldots \cdot u_k$. The number of edges of a path is its *length*, and the path of length k is denoted by $L(p) = k$. The path p is *simple* if no two u_i and u_j in p are the same. Any sub-sequence $u_x, \ldots, u_{x+y}$ ($x \in \{1, \ldots, k+1\}$, $y \in \{0, \ldots, k+1-x\}$) is said to be *contained in* p, and dually we say that p *contains* the sequence or *passes through* it. We say that path p and q *intersect* if they contain a common sub-sequence. The intersection of a path p and an arbitrary set of nodes W is the set of elements of W that are contained in p. When p intersects W sometimes we say that p *touches* W. A *cycle* is a path where $u_1 = u_k$. An acyclic graph is a graph not containing any cycles. The distance $d_G(x, y)$ in G of two vertices x, y is the length of a shortest $x - y$ path in G; if no such path exists, we set $d(x, y) := \infty$.

Cycle

Distance of two vertices

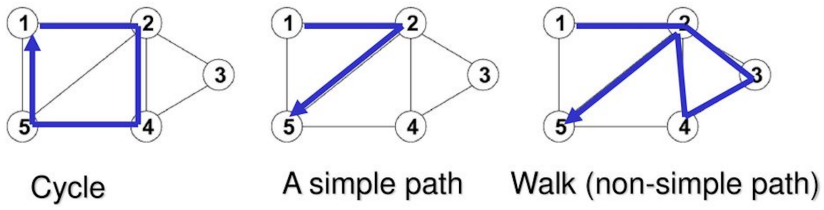


Figure 2.4. Some examples of paths.

G is *directed* if pairs in E are ordered. Otherwise G is *undirected*. G is DAG if it is *directed* and with no cycles. If G is directed then we also distinguish $N_{\mathtt{i}}(u)$, the set of neighbours v of u s.t. $(v, u) \in E$, from $N_{\mathtt{o}}(u)$, the neighbours v of u s.t. $(u, v) \in E$. For all degree measures on G we distinguish the in-degree $\Delta_{\mathtt{i}}(G)$ and $\delta_{\mathtt{i}}(G)$ and the out-degree $\Delta_{\mathtt{o}}(G)$, and $\delta_{\mathtt{o}}(G)$.

Directed graph

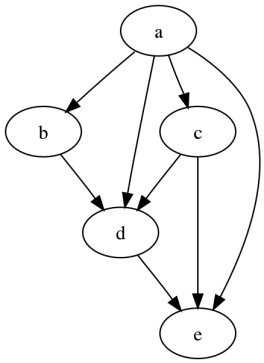


Figure 2.5. Example of a directed acyclic graph.

Connected graph

A graph G is called connected if it is non-empty and any two of its vertices are linked by a path in G. Moreover a maximal connected subgraph of G is a component of G. In Connectivity, the vertex-connectivity of the given graph $G = (V, E)$, namely $\kappa(G)$ is the size of the minimal subset K of V, such that removing K nodes from G disconnects G. In particular it is well-known (see for example [30], Theorem 5.1, pag 43) that

Vertex-connectivity

$$\kappa(G) \leq \delta(G). \tag{2.1}$$

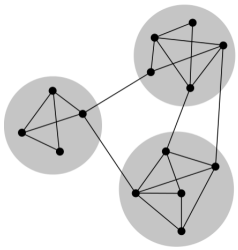


Figure 2.6. This graph becomes disconnected when the right-most node in the gray area on the left is removed.

It will also be convenient to work with sets of vertices disconnecting particular parts of G. If $S, T \subseteq V$, then $\kappa_{ST}(G)$ is the size of the smallest vertex separator of S and T in G, i.e. the smallest set of vertices whose removal disconnects S and T (set $\kappa_{ST}(G) = \infty$ if $S \cap T \neq \emptyset$ or there are $s \in S$ and $t \in T$ such that $\{s, t\} \in E$). Notice that $\kappa_{ST}(G) \geq \kappa(G)$.

Trees

An undirected tree $\mathcal{T}_n$ over n nodes is an acyclic graph where any two nodes are connected by exactly one path. In the directed case we say that $\mathcal{T}_n$ is a *downward directed tree* (respectively *upward directed tree*) if all vertices v except the root have $\Delta_{\mathtt{i}}(v) = 1$ (respectively $\Delta_{\mathtt{o}}(v) = 1$). See Figure 4.1. The vertices of degree 1 in a tree are its leaves.

Hypergraph

A hypergraph is a pair (V, E) of disjoint sets, where the elements of E are non-empty subsets (of any cardinality) of V . Thus, graphs are special hypergraphs.

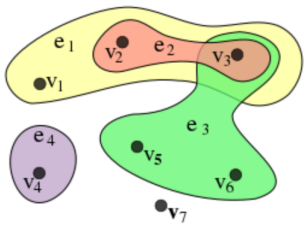


Figure 2.7. An example of an undirected hypergraph with $E = \{e_1, e_2, e_3, e_4\} = \{\{v_1, v_2, v_3\}, \{v_2, v_3\}, \{v_3, v_5, v_6\}, \{v_4\}\}$.

2.2 Grids and LoS Networks

Hypergrid

We consider the following graphs. Let $d \in \mathbb{N}^+$ and $n \in \mathbb{N}$, $n \geq 3$. The *(directed) hypergrid* of dimension d (d-grid) over support $[n]$, $\mathcal{H}_{n,d}$, is the graph with vertex set $[n]^d$ and where there is a directed edge from a node $x = (x_1, x_2, ..., x_d)$ to a node $y = (y_1, y_2, ..., y_d)$ if for some $i \in [d]$ we have $y_i - x_i = 1$ and $x_j = y_j$ for all $j \neq i$. In the case of *undirected* d-grid in $\mathcal{H}_{n,d}$ there is an edge between a node x and a node y if for some $i \in [d]$ we have $|x_i - y_i| = 1$ and $x_j = y_j$ for all $j \neq i$. In the case of grids over n nodes, i.e. $d = 2$, we use the notation $\mathcal{H}_n$. ∂_i is the set of nodes $x = (x_1, x_2, ..., x_d)$ such that $x_i = 1$. A *border node* is a node of $\mathcal{H}_{n,d}$ which is also in some ∂_i.

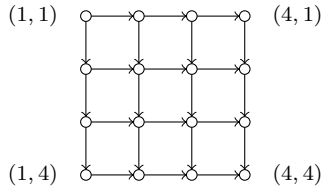
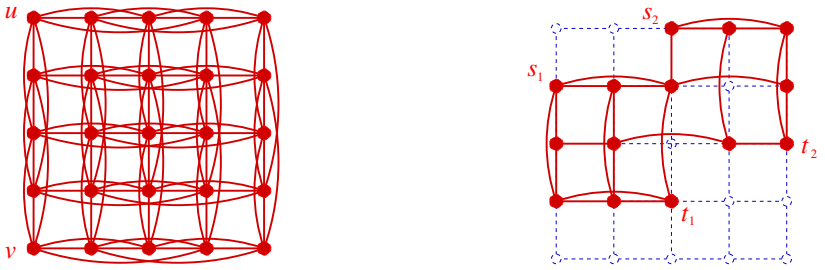


Figure 2.8. *Directed hypergrid* $\mathcal{H}_4 = \mathcal{H}_{4,2}$.

For positive integers d, and $n \geq 2$, let $\mathbb{Z}_n^d$ be the d-dimensional cube $\{1, \ldots, n\}^d$. We say that distinct points P_1 and P_2 in a cube *share a line of sight* if their coordinates differ in a single place. A graph $G = (V, E)$ is said to be a *Line of Sight (LoS) network of size n, dimension d, and range parameter ω* if there exists an embedding $f_G : V \to \mathbb{Z}_n^d$ such that $\{u, v\} \in E$ if and only if $f_G(u)$ and $f_G(v)$ share a line of sight and the (Euclidean) distance between $f_G(u)$ and $f_G(v)$ is less than ω. In the rest of the paper a LoS network G is always given along with some embedding f_G in $\mathbb{Z}_n^d$ for some d and n, and with slight *abuse of language* we will often refer to the vertices of G, $u, v, \ldots \in V$ in terms of their corresponding points $f_G(u), f_G(v), \ldots$ in $\mathbb{Z}_n^d$, and in fact the embedding f_G will not be mentioned explicitly.

Line of Sight (LoS) network

Note that d-dimensional hypergrids, $\mathcal{H}_{n,d}$ are particular LoS networks with $\omega = 2$ and all possible n^d vertices. In the forthcoming sections we will study *augmented* hypergrids $\mathcal{H}_{n,d,\omega}$ (or simply $\mathcal{H}_{n,\omega}$ in the 2-dimensional case), namely d-dimensional LoS networks with range parameter $\omega > 2$ containing all possible n^d nodes.



(a) (b)

Figure 2.9. On the left, the network $\mathcal{H}_{n,\omega}$ for $n = 5$ and $\omega = 4$ (note that vertices u and v are not adjacent); on the right a more general example of LoS network, having $\omega = 3$, embedded in $\mathbb{Z}_5^2$ (represented as a dashed grid).

2.3 Embeddings

Each DAG $G = (V, E)$ is equivalent to a *poset* with elements V and partial order $\preceq_G$, where $u \preceq_G v$ if v is reachable from u in G. Elements u and v are *comparable* if $u \preceq_G v$ or $v \preceq_G u$, and *incomparable* otherwise. We write $u \prec_G v$ if $u \preceq_G v$ and $u \neq v$.

Now that we have met all the standard basics about graphs, we can also define what it means to embed one graph in another. Basically, an embedding of G in G' is an injective map $f : G(V, E) \to G'(V', E')$ that preserves the kind of structure we are interested in. In other words, a mapping f from a poset $G = (V, E)$ to a poset $G' = (V', E')$ is called an *embedding* if f is injective and it respects the partial order, that is, all $u, v \in V$ are mapped to $u', v' \in V'$ such that $u \preceq_G v$ iff $u' \preceq_{G'} v'$. If G is embeddable into G' we write $G \hookrightarrow G'$. *Embedding*

Figure 2.10. *Example of embedding* $G_1 \hookrightarrow_f G_2$*:* $f(u_i) = w_i$.

2.4 Monitors, Measurement Paths and Maximal Identifiability μ

In Boolean Network Tomography one takes measurements along paths, and the quality of the monitoring scheme depends on the choice of such paths. Let $\mathbb{P}$ be a set of paths over some node set V. For a node $v \in V$, let $\mathbb{P}(v)$ be the set of paths $p \in \mathbb{P}$ passing through v, i.e. such that v is a node in p. For a set of nodes U, $\mathbb{P}(U) = \bigcup_{u \in U} \mathbb{P}(u)$. Hence if $U \subseteq V$, $\mathbb{P}(U) \subseteq \mathbb{P}(V)$. Crucially, we identify two sets of vertices S and T, and assume that $\mathbb{P}$ is the set of all S-T paths in G, i.e. simple paths with one end-point in S and the other one in T.

Traditionally in Network Tomography all measurements originate and end at special monitoring stations that are connected to the structure under observation. In other words, in end-to-end measurement paths, messages are routed and received through monitor nodes. For any tomographic process to have any chance of succeeding, one has to assume that such monitors are infallible. It is therefore customary to assume that *physical monitors are external to the given network* but connected to it through a designated set of nodes ($S \cup T$ is such set in our case). This assumption is justified by two reasons: (1) Monitors by default must be reliable, hence there is no failure to identify for them; (2) Since we study maximal identifiability in set of paths associated to given topologies $G = (V, E)$, the assumption allows to consider all the nodes in G as equally potentially identifiable for a failure.

Monitor placement

Let I, O be sets of physical monitors. A *monitor placement* for $G = (V, E)$ is a pair of injective mappings $\chi = (\chi_{\mathtt{i}}, \chi_{\mathtt{o}})$ such that $\chi_{\mathtt{i}} : I \to V$ and $\chi_{\mathtt{o}} : O \to V$. We always denote by (S, T) the pair $(\chi_{\mathtt{i}}(I), \chi_{\mathtt{o}}(O))$, where clearly $S = \bigcup_{i \in I} \chi_{\mathtt{i}}(i)$ and $T = \bigcup_{i \in O} \chi_{\mathtt{o}}(i)$. Notice that there is no relation between cardinalities of S and I or cardinalities of T and O. The interpretation is that S is the set of the nodes in G (*input nodes*) linked to input monitors and T (*output nodes*) the nodes in G linked to output monitors. We use to denote measurement paths in G under χ as $S \cdot (v_1, v_2) \cdot \ldots \cdot (v_{k-1}, v_k) \cdot T$ where $v_1 \in S$ is an input node, $v_k \in T$ is an output node and $(v_1, v_2) \cdot \ldots \cdot (v_{k-1}, v_k)$ a path in G.

Given a graph $G = (V, E)$ and a monitor placement $\chi = (S, T)$ we denote by $\mathbb{P}(G|\chi)$ the set of all *distinct* paths from a node in S to a node in T. Let $\mathbb{P}$ be a set of paths over a set of nodes V. Following [38] we define:

k-identifiability

Definition 2.4.1 (k-identifiability)**.** *A set of vertices V is k-identifiable*

with respect to $\mathbb{P}$ *if and only if for all* $U, W \subseteq V$*, with* $U \triangle W \neq \emptyset$ *and* $|U|, |W| \leq k$*, it holds that* $\mathbb{P}(U) \triangle \mathbb{P}(W) \neq \emptyset$*.*

That is, for any two distinct node sets U and W in V of cardinality at most k there exists always a path in $\mathbb{P}$ traversing exactly one node-set between U and W.

Definition 2.4.2 (Maximal identifiability)**.** *The maximal identifiability of* V *with respect to* $\mathbb{P}$*,* $\mu(V)$ *is the* $\max k \geq 0$ *such that* V *is* k*-identifiable with respect to* $\mathbb{P}$*.*

Maximal identifiability

Monotonicity of identifiability (a property noticed also in [38, 39]), i.e. that k-identifiability of V wrt $\mathbb{P}$ implies k'-identifiability of V wrt $\mathbb{P}$ for $k' < k$, is trivial from our definition.

Monotonicity of identifiability

In the Boolean system as in Equation 1.1 we can have equations made by only one variable, $x_v = b$ for some $b \in \{0, 1\}$. This situation occurs when in $\mathbb{P}$ a node v is linked to both input and output monitors. But one node is no path. We then consider such equation to correspond to a loop path of one node $S \cdot (v, v) \cdot T$. We call it a DLP-path (from *degenerate loop path*). Notice that since $x_v = b$ has a trivial solution which can be immediately propagated in the whole system of Equation 1.1, DLP does not have a key role in localizing node failures. Given a graph G and the monitor placement $\chi = (S, T)$, a routing mechanism determines the set of measurement paths. All our results hold under *any routing which does not create loops* and therefore also without DLP.

DLP-*path*

For a graph $G = (V, E)$, and a monitor placement χ for G, we write $\mu(G|\chi)$ and call it the maximal identifiability of $G|\chi$, to indicate the maximal identifiability of V with respect to $\mathbb{P}(G|\chi)$. We might omit the χ, when it is either clear from the context, or when the result holds for all possible monitor placements.

$\mu(G|\chi)$

Symbol	Meaning
$\mathbb{P}$	set of paths
$\mathbb{P}(u)$	paths in $\mathbb{P}$ passing through u
$\mathbb{P}(U)$	$\bigcup_{u \in U} \mathbb{P}(u)$
(I, O)	physical input and output monitors
χ	monitor placement: $\chi = (\chi_{\mathtt{i}}(I), \chi_{\mathtt{o}}(O))$
(S, T)	nodes in V linked to I and O by χ
$\mathbb{P}(G\|\chi)$	set of all paths in G from S to T
$\mu(G\|\chi)$	maximal identifiability of V wrt $\mathbb{P}(G\|\chi)$

Table 2.1. *Notations for paths, monitors, identifiability.*

2.5 Bounds on μ and Unique Localization of Failure Nodes

To prove that $\mu(G|\chi) \leq k-1$ we have to show that $G|\chi$ is not k-identifiable. By Definition 2.4.1 it is sufficient to show the existence of two distinct node sets U and W of cardinality at most k such that $\mathbb{P}(U) \triangle \mathbb{P}(W) = \emptyset$. By the monotonicity property of identifiability, this implies that $\mu(G|\chi) \leq k-1$.

Conversely, to prove that $\mu(G|\chi) \geq k$ for some k, then by Definition 2.4.1 it is enough to argue that for all distinct node sets U and W of cardinality $|U|, |W| \leq k$, $\mathbb{P}(U) \triangle \mathbb{P}(W) \neq \emptyset$. To prove this we have to show that for any two distinct node sets U and W of cardinality at most k there exists a path in $\mathbb{P}$ intersecting exactly one between U and W.

To make clear the importance of the measure μ, we formally addresses how maximal identifiability implies unique localization of sets of failure nodes, which is known but did not appear before anywhere.

Let us motivate our definition in the context of the approach of Boolean Network Tomography to detect failure nodes in networks. Assume to have a set $\mathbb{P}$ of m end-to-end paths over n nodes. A *binary measurement* $\mathbb{M}$ along a path $p \in \mathbb{P}$ is obtained by sending a message through p and recording the outcome $\mathbb{M}(p)$, a bit, which identifies (in the case $\mathbb{M}(p) = 1$) that some node in p is failing, or (in the case $\mathbb{M}(p) = 0$) that no node is failing along the path p.

Binary measurement

We claim that if $\mathbb{P}$ is k-identifiable, then under any binary measurement $\mathbb{M}$, we can *uniquely localize* in $\mathbb{P}$ up to k failing nodes.

Let $\mathbb{P}$ be a set of paths over nodes V and let $\mathbb{M}$ be a Boolean measurement running along the paths in $\mathbb{P}$. Let $\mathsf{fail}_{\mathbb{M}}(\mathbb{P}) = \{p \in \mathbb{P} | \mathbb{M}(p) = 1\}$, i.e. the set of paths where a failure is detected under $\mathbb{M}$.

Definition 2.5.1 (Unique failure)**.** *Let $\mathbb{P}$ be a set of m paths over V and $\mathbb{M}$ a binary measurement on $\mathbb{P}$. A set of nodes $W \subseteq V$ is* failing *in $\mathbb{P}$ under $\mathbb{M}$ if $\mathbb{P}(W) \subseteq \mathsf{fail}_{\mathbb{M}}(\mathbb{P})$. Furthermore W is* uniquely failing *if $\overline{\mathbb{P}(W)} \subseteq \overline{\mathsf{fail}_{\mathbb{M}}(\mathbb{P})}$ (where $\overline{\mathbb{P}(W)} = \mathbb{P} \setminus \mathbb{P}(W)$ and $\overline{\mathsf{fail}_{\mathbb{M}}(\mathbb{P})} = \mathbb{P} \setminus \mathsf{fail}_{\mathbb{M}}(\mathbb{P})$), i.e. on any path not touching W the measurement $\mathbb{M}$ is not failing.*

Unique failure

Theorem 2.5.2. *Let $\mathbb{P}$ be a set of paths over V. If $\mu(\mathbb{P}) \geq k$, then under any Boolean measurement $\mathbb{M}$ there is exactly one set W of nodes in V of size at most k that is uniquely failing in $\mathbb{P}$.*

Proof. Let $\mathbb{M}$ be a measurement over $\mathbb{P}$. Assume that $\mu(\mathbb{P}) \geq k$ and that for the sake of contradiction there are two distinct sets $W, W' \subseteq V$ of size at most k both uniquely failing in $\mathbb{P}$ under $\mathbb{M}$. Since $W \neq W'$, and $\mathbb{P}$ is k-identifiabale, hence by Definition 2.4.1 we know that either there is path $p \in \mathbb{P}(W) \setminus \mathbb{P}(W')$ or $p \in \mathbb{P}(W') \setminus \mathbb{P}(W)$. Say wlog the first case. Since $p \in \mathbb{P}(W)$ and W is failing, then by Definition 2.5.1 $p \in \mathsf{fail}_{\mathbb{M}}(\mathbb{P})$. On the other hand since $p \notin W'$, by unique failure of W' $p \in \overline{\mathsf{fail}_{\mathbb{M}}(\mathbb{P})}$. Contradiction. □

Chapter 3

Tight Bounds on Maximal Identifiability by Structural Results

One of the modern methods in Pure and Applied Mathematics is to switch the problem into the language of Graph Theory and then by using the theorems, rules and structures in Graph Theory we can find a better depiction to solve the problem. Using different structural graphs may reveal new insights on the problem. Our main purposes in this chapter are developing new techniques in terms of structural property of the topology to detect fault components in networks through main tools in Graph Theory.

In the first two *Sections 3.1* and *3.2* we present some upper bounds for maximal identifiability in terms of structural properties of the topology. We analyse: the number of nodes linked to monitors (Theorem 3.1.1), the minimal degree (Lemmas 3.2.1 and 3.2.3) and the number of edges and nodes (Corollary 3.2.2).

Section 3.4 contains all the results about maximal identifiability and isomorphic embeddings. We establish relations between maximal identifiability and embeddability when networks are modeled by DAGs. In Theorem 3.4.2 we prove that upper bounds for maximal identifiability (for DAGs) are preserved under isomorphic embeddings. For isomorphic embeddings we prove that lower bounds on maximal identifiability are preserved (Theorem 3.4.4). This result in turn is used to prove that, for DAGs closed under transitivity, maximal identifiability is lower bounded by the dimension of the graph (Theorem 3.4.7). The end of this section is devoted to the study of maximal identifiability under the operation of k-transitive closure of

DAGs. We show that μ grows linearly in k (Theorem 3.4.12).

Last but not least, *Section 3.5* focuses on arbitrary graphs. We study the upper and lower bounds on the maximal identifiability of any network G in terms of the vertex connectivity using tools from Graph Theory (such as Menger's Theorem). First we look at the proof of Theorem 1.2.2. Then we describe a different construction that leads to the proof of Theorem 1.2.3.

3.1 Number of Input and Output Nodes

Having monitors external to the network, we look at the maximal identifiability we can hope for in a graph, knowing how many internal nodes are linked to monitors. Let $G = (V, E)$ be a graph. Let $\chi = (\chi_{\mathtt{i}}, \chi_{\mathtt{o}})$ be a monitor placement for G of physical monitors I and O. Let $S = \bigcup_{i \in I} \chi_{\mathtt{i}}(i)$ and $\hat{S} = |S|$. Let $T = \bigcup_{i \in O} \chi_{\mathtt{o}}(i)$ and $\hat{T} = |T|$.

Theorem 3.1.1. *Let $G = (V, E)$ be an arbitrary connected graph without loop paths and therefore without* DLP*s and χ an arbitrary monitor placement for G. Then $\mu(G|\chi) < \max(\hat{S}, \hat{T})$.*

Proof. Define $U = S$ and $W = T$. Hence $|U|, |W| \leq \max(\hat{S}, \hat{T})$. If $U \neq W$, then since G is connected, there is no way of separating U from W with a path going from an input-node to an output-node. We will always touch both. Then $\mathbb{P}(U) \triangle \mathbb{P}(W) = \emptyset$ and hence $\mu(G|\chi) < \max(\hat{S}, \hat{T})$. If $U = W$ then it must be that $|U| = |W| \geq 2$, since otherwise $U = W = \{u\}$ and we would have a DLP-path which is not allowed. Then define $U' = U - \{u\}$ where u is one of the nodes that is both the termination of a path and the source of another (u necessarily exists since $U = W$). It is obvious that $\mathbb{P}(U') \subseteq \mathbb{P}(W)$. Now, if $p \in \mathbb{P}(W)$ is not touching u, then $p \in \mathbb{P}(U')$ since $U' = W - \{u\}$. If $p \in \mathbb{P}(W)$ is touching u, then the source of this path is in U' and p touches U' as well, unless the source of p is u. If the source of p is u, then the termination of this path is in $W - \{u\} = U'$ (since loop paths are not allowed) and touches U' as well. Therefore $\mathbb{P}(U') \triangle \mathbb{P}(W) = \emptyset$ and hence $\mu(G|\chi) < \max(\hat{S}, \hat{T})$. □

3.2 Degree

Next results hold for any monitor placement and we omit χ. We start with the undirected case.

Lemma 3.2.1. *Let $G = (V, E)$ be undirected. Then $\mu(G) \leq \delta(G)$.*

Proof. Let $u \in V$ be such that $\deg(u) = \delta(G)$. Fix $U = N(u)$ and $W = \{u\} \cup N(u)$. Each path touching u is passing through at least a node in $N(u)$. Hence $\mathbb{P}(\{u\}) \subseteq \mathbb{P}(N(u))$. Hence $\mathbb{P}(W) = \mathbb{P}(\{u\}) \cup \mathbb{P}(N(u)) = \mathbb{P}(N(u)) = \mathbb{P}(U)$ and then $\mathbb{P}(U) \triangle \mathbb{P}(W) = \emptyset$. We have found two sets U, W of cardinality at most $\delta(G) + 1$. Hence $\mu(G) \leq \delta(G)$. □

Notice that if a node v in V is disconnected in G, then $\mu(G) = \delta(G) = 0$. Hence in the rest of the paper, we assume the graphs always to be connected.

Corollary 3.2.2. *Let $G = (V, E)$ be defined over n nodes and m edges. Then $\mu(G) \leq \min\{n, \lceil \frac{2m}{n} \rceil\}$.*

Proof. Assume a graph G has n nodes and minimal degree d. Then there are at least $nd/2$ edges in G. So $m \geq nd/2$. Hence $d \leq 2m/n$. By Lemma 3.2.1 $\mu(G) \leq d = 2m/n$. □

Let us now consider the directed case. Let $G = (V, E)$ be a directed graph and $\chi = (S, T)$ be a monitor placement. A node $v \in V$ is called a *complex source* if $v \in S$ and $\deg_{\mathtt{i}}(v) > 0$ and a *simple source* if $v \in S$ and $\deg_{\mathtt{i}}(v) = 0$. Let K (resp. L) be the set of complex (resp. simple) source nodes and $R = V \setminus (K \cup L)$. We let $\hat{\delta}(G) = \min\{ \min_{v \in R} \deg_{\mathtt{i}}(v), \min_{v \in K}(\deg_{\mathtt{i}}(v) + \deg_{\mathtt{o}}(v)) \}$.

Complex and simple sources

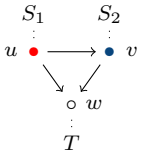


Figure 3.1. *Example of simple (●) and complex (●) source nodes.*

Lemma 3.2.3. *Let $G = (V, E)$ be directed. Then $\mu(G) \leq \hat{\delta}(G)$.*

Proof. Let w be a node in G which minimizes $\hat{\delta}(G)$. If $w \in R$, then $\hat{\delta}(G) = \delta_{\mathtt{i}}(w)$. Define $W = N_{\mathtt{i}}(w)$ and $U = N_{\mathtt{i}}(w) \cup \{w\}$. Since $w \in R$, then each path passing through w is necessarily proceeding from a node in $N_{\mathtt{i}}(w)$, hence $\mathbb{P}(\{w\}) \subseteq \mathbb{P}(N_{\mathtt{i}}(w))$. Therefore $\mathbb{P}(U) = \mathbb{P}(W)$, which proves the claim since $|U| = \delta_{\mathtt{i}}(G) + 1 = \hat{\delta}(G) + 1$. If $w \in K$, then define $W = N_{\mathtt{i}}(w) \cup N_{\mathtt{o}}(w)$ and $U = W \cup \{w\}$. If a path is passing from w and raising from an input monitor linked to w, it is necessarily continuing to a node in $N_{\mathtt{o}}(w)$. □

3.3 Lines

We call a path p in an undirected graph $G = (V, E)$ a *line* if $p := (u_0, u_1) \dots (u_k, u_{k+1})$ and $N(u_i) = \{u_{i-1}, u_{i+1}\}$ for any $i \in [k]$ (see also [9, 18]). Reasoning exactly as in Lemma 3.2.1 is easy to observe that if $\mathbb{P}(G|\chi)$ includes a path which is a line, the maximal identifiability of G is less than 1. Hence meaningful topologies should not include a line. We define an undirected topology G to be *Line-Free* (LF) if each node u is linked to at least three other nodes in G.

Line

3.4 Embeddings

Let $G = (V, E)$ and $H = (V', E')$ be two DAGs and consider f to be an embedding $G \hookrightarrow_f H$. Let χ be a monitor placement in G, and χ^f be the monitor placement for H defined by $(f \circ \chi_{\mathtt{i}}, f \circ \chi_{\mathtt{o}})$. We want to explore what can be said on $\mu(H|\chi^f)$ in terms of $\mu(G|\chi)$.

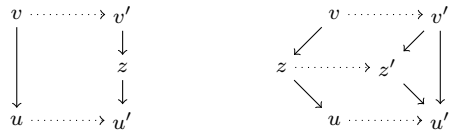


Figure 3.2. *Injective and bijective embeddings.*

As it can be seen from the left example in Figure 3.2, a $1-1$ mapping can map an edge into a line. Hence it can map a line-free graph G into a graph H which contains a line and whose $\mu(H)$ is 0 (see Section 3.3) independently of $\mu(G)$. We then consider embeddings defined by $1-1$ and onto mappings which are known as *order-isomorphisms*, see [45, 47]. Though weaker than general embeddings, order-isomorphisms are still interesting to study with respect to maximal identifiability. The order-isomorphism on the right in Figure 3.2 indicates that new paths can appear under an order-isomorphism and hence there is hope that μ can potentially grow (since it will be potentially easier to find a path to separate bigger sets of nodes). In fact we will show that property in Theorem 3.4.12.

Order-isomorphisms

In the rest of the section we study what can be said on $\mu(H|\chi^f)$ from $\mu(G|\chi)$ when $G \hookrightarrow_f H$ and f is an order-isomorphism, i.e. a bijective

embedding [45, 47]. To simplify readability we always omit the χ's, writing simply $\mu(G)$ and $\mu(H)$. Consider the following definition given in [4, 21].

Routing consistent

Definition 3.4.1. *A set of paths $\mathbb{P}$ is* routing consistent *if any two distinct paths p and p' in $\mathbb{P}$ and any distinct nodes u and w traversed by both paths (if any) p and p' follow the same subpath between u and w.*

In the directed case we can prove the following result.

Theorem 3.4.2. *Assume that $G = (V, E)$ is a routing consistent directed graph and that $G \hookrightarrow_f G'$. Then $\mu(G) \leq \mu(G')$.*

Proof. Let $\chi = (S, T)$ be a monitor placement on G. Assume $\mu(G') = k$. We prove that $\mu(G) \leq k$. Since $\mu(G') \leq k$, there are two sets $U', W' \subseteq V'$ such that $U' \triangle W' \neq \emptyset$, and at least one of them, wlog say U', has cardinality $k+1$, and $\mathbb{P}_{G'}(U') \triangle \mathbb{P}_{G'}(W') = \emptyset$. Fix $U = f^{-1}(U')$ and $W = f^{-1}(W')$. By bijectivity of f, U has cardinality $k+1$ and $U \triangle W \neq \emptyset$ (since otherwise $U' \triangle W' = \emptyset$). Assume by contradiction that $\mathbb{P}_G(U) \triangle \mathbb{P}_G(W) \neq \emptyset$. That is, there exists a path p in G from S to T touching nodes of only one between U and W, say U. Let $p = (u_1, u_2) \cdot \ldots \cdot (u_r, u_{r+1})$, $r \geq 1$. Hence $u_i \leq u_{i+1}$ for all $i \in [r]$. Let $u'_i = f(u_i)$. Clearly if $u_i \in U$, then $u'_i \in U'$. Since f is an embedding (i.e. $x \leq y$ iff $f(x) \leq f(y)$), then $u'_i \leq u'_{i+1}$, $u'_1 \in S'$ and $u'_{r+1} \in T'$. Hence there are paths p'_i in G' from u'_i to u'_{i+1} and the path $p' = p'_1 \cdot \ldots \cdot p'_r$ is a path from S' to T' in G'. If all nodes in p' are in $V' \setminus W$, this is a contradiction with the fact $\mathbb{P}_{G'}(U') \triangle \mathbb{P}_{G'}(W') = \emptyset$. Then there is an $i \in [r]$ such that p'_i is touching a node $w' \in W'$. Hence we have that in G' , $u'_i \leq w' \leq u'_{i+1}$. Since f is an embedding and since $u_i = f^{-1}(u'_i)$, this means that in G, $u_i \leq f^{-1}(w) \leq u_{i+1}$. Then in G there is a path from u_i to u_{i+1} passing through $f^{-1}(w)$. This contradicts the routing consistency of G since between u_i and u_{i+1} there is another path, the edge that is in p. □

Distance-increasing, distance decreasing and distance preserving

The previous result shows that restricting the class of graphs one can still hope to bound identifiability using embeddability. We now restrict the class of embeddings, obtaining similar relationships but for broader classes of topologies. Assume that f is an embedding between two DAGs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Let us say that f is *distance-increasing (d.i)* (respectively *distance decreasing (d.d.)* if for all $x, y \in V_1$, $d_{G_1}(x, y) \leq d_{G_2}(f(x), f(y))$ ($d_{G_1}(x, y) \geq d_{G_2}(f(x), f(y))$). Here $d_G(x, y)$ is the length of the shortest path between x and y in G. We call f *distance preserving* (d.p.) if $d_{G_1}(x, y) = d_{G_2}(f(x), f(y))$. Distance-increasing of f immediately implies that the inverse image under f of edges of G_2 are edges of G_1.

Lemma 3.4.3. *Let $G = (V, E)$ and $H = (W, F)$. If $G \hookrightarrow_f H$, where f is d.i.-embedding and $(w_1, w_2) \in F$, then $(f^{-1}(w_1), f^{-1}(w_2)) \in E$.*

Theorem 3.4.4. *Let G and G' be two DAGs such that $G \hookrightarrow_f G'$, where f is a (d.i.)-embedding (respectively a d.d.- embedding). Then $\mu(G) \geq \mu(G')$ (respectively $\mu(G') \geq \mu(G)$).*

Proof. Let $\chi = (S, T)$ be a monitor placement on G. Let us prove the statement for a d.i.-embedding. Assume $\mu(G) = k$, we prove that $\mu(G') \leq k$. Since $\mu(G) \leq k$, there are two sets $U, W \subseteq V$ such that $U \triangle W \neq \emptyset$, at least one of them, say U, has cardinality $k + 1$, and $\mathbb{P}_G(U) \triangle \mathbb{P}_G(W) = \emptyset$. Fix $U' = f(U)$ and $W' = f(W)$ and let $S' = f(S)$ and $T' = f(T)$. By injectivity of f, U' has cardinality $k + 1$ and clearly $U' \triangle W' \neq \emptyset$ (since otherwise $U \triangle W = \emptyset$). Assume by contradiction that $\mathbb{P}_{G'}(U') \triangle \mathbb{P}_{G'}(W') \neq \emptyset$. That means that there exists a path p' from S' to T' touching nodes of only one between U' and W', say U'. Let $p' = (u'_1, u'_2) \cdot \ldots \cdot (u'_{r-1}, u'_r)$ and $u_i = f^{-1}(u'_i)$. By Lemma 3.4.3 for all $i \in [r - 1]$, (u_i, u_{i+1}) is an edge in G and since f is an embedding, then $u_i \in U$, $u_1 \in S$ and $u_r \in T$. But then $p = (u_1, u_2) \cdot \ldots \cdot (u_r, u_{r+1})$ is a path from S to T touching only nodes in U. This is a contradiction with the fact $\mathbb{P}_G(U) \triangle \mathbb{P}_G(W) = \emptyset$.

In the case of d.d.-embedding the claim follows from the first part since if $G \hookrightarrow_f G'$ and f is d.d.-embedding, then by bijectivity of f $G' \hookrightarrow_{f^{-1}} G$ is a d.i.-embedding from G' to G. □

Hence if f is distance-preserving, then equality holds.

Corollary 3.4.5. *Let G and G' be two DAGs such that $G \hookrightarrow_f G'$, where f is a (d.p.)-embedding. Then $\mu(G) = \mu(G')$.*

Dimention of a graph G

The *dimension* of G, $\dim(G)$ is the smallest integer d such that $G \hookrightarrow \mathcal{H}_{n,d}$. Dushnik and Miller [19] proved that for any $n > 1$, the hypergrid $\mathcal{H}_{n,d}$ has dimension exactly d. We explore how to bound $\mu(G)$ in terms of $\dim(G)$. Let $G = (V, E)$ be a directed graph. The transitive closure $G^* = (V, E^*)$ of the graph G is formed by the reachability factor, i.e an edge $(v, w) \in E^*$ if and only if the vertex w is reachable in the graph G from the vertex v (means there exists at least one path from v to w).

Transitive closure of a graph

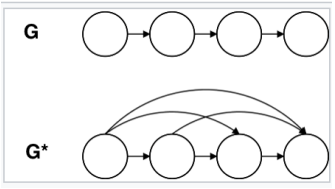


Figure 3.3. Transitive closure constructs the graph G^* from the graph G.

Now let G^* be the *transitive closure* of a DAG G.

Lemma 3.4.6. *Let G and H be DAGs. If G is closed under transitivity and $G \hookrightarrow_f H$, then $\mu(G) \geq \mu(H)$. In particular $\mu(G^*) \geq \mu(G)$.*

Proof. Since G is closed under transitivity then the embedding f is necessarily a distance-increasing one. Hence the first claim follows by Theorem 3.4.4. The second claim follows since the identity is a bijective embedding from G^* to G. □

Theorem 3.4.7. *Let G be a DAG closed under transitivity. Then $\mu(G) \geq \dim(G)$.*

Proof. Let f be the function witnessing the embedding $G \hookrightarrow \mathcal{H}_{n,\dim(G)}$. Since G is closed under transitivity and by Theorem 4.2.10 $\mu(\mathcal{H}_{n,\dim(G)}) = \dim(G)$, the claim follows by previous lemma. □

Corollary 3.4.8. *For all DAGs G, for all $k \in \mathbb{N}$, $\mu(G^k) \geq \mu(G)$.*

Transitive closure The result in Corollary 3.4.8 is new but unsatisfactory since it is not giving a precise estimate of how much it can increase maximal identifiability building the k-transitive closure of a graph. Here we obtain a precise lower bound.

Definition 3.4.9 (Segment)**.** *Let $G = (V, E)$ be a directed graph. Let $\{u_1, ..., u_r\} \subseteq V$ and p be a path in G. We say that $\{u_1, ..., u_r\}$ is a* segment of length r *in p if $u_1 \cdot u_2 \cdot \ldots \cdot u_r$ is a subpath of p.*

Segment

We say that the segment I is *internal* to a path if neither endpoints of I coincide with the endpoints of p. Let us denote with $|p|$ the length of p, i.e. the number of edges in p.

Internal segment

Length of a path p

Lemma 3.4.10. *Let $G = (V, E)$ be a directed graph and $G^k = (V, E^k)$ its k-transitive closure and let p be a path in G^k from a node s to a node t. Any internal segment I of length at most $|I| \leq |p| - 1 \leq k - 1$ in p can be removed and replaced by one edge obtaining another path in G^k from s to t.*

Proof. Since the segment I is internal, it is surrounded by at least two nodes, say u and w. Since G^k is closed under k-transitivity and $|I| \leq k - 1$, then in G^k there is an edge (u, w). We can then replace in p, the segment I by the edge (u, w), obtaining a new path from s to t. □

Given $U \subseteq V$ and a path p, the set of *U-segments in p* is the set of all $I \subseteq U$ forming an internal segment in p. The set of U-segments is closed

under inclusion. We consider $U^p_{\max}$ to be the maximal subset I of U such that I is an internal segment in p.

Lemma 3.4.11. *Let $G = (V, E)$ be a DAG and $G^k = (V, E^k)$ the k-transitive closure of G. Let $U, W \subseteq V$, with $U \neq W$ and such that $\mathbb{P}_{G^k}(U) = \mathbb{P}_{G^k}(W)$. Then for all $p \in \mathbb{P}_{G^k}(U)$, $|U^p_{\max}|, |W^p_{\max}| > k - 1$. Hence $|U|, |W| > k - 1$.*

Proof. Assume for the sake of contradiction that there exits a $p \in \mathbb{P}_{G^k}(U)$ such that $|U^p_{\max}| \leq k - 1$. Hence for any U-segment I in p, it holds that $|I| \leq k - 1$. Since $\mathbb{P}_{G^k}(U) = \mathbb{P}_{G^k}(W)$, then p must necessarily touch also nodes in W. Hence any two consecutive U-segments I' and I'' in p are either interleaved by elements of W as in $p = \ldots w \cdot I' \cdot W' \cdot I'' \cdot w'' \ldots$, where $W' \subseteq W$, or separated by elements of W and surrounded by input and output monitors, that is $p = S \cdot I' \cdot W' \cdot I'' \cdot T$. Consider the first case and assume that $W' = \{w_1, \ldots w_r\}$, $r \geq 1$. Since $|I'|, |I''| \leq k - 1$, and G^k is closed under k-transitivity, then in G^k there are the edges (w, w_1) and (w_r, w''). Hence according to previous lemma the two segment I' and I'' can be replaced by the consecutive edges $(w, w') \cdot (w', w'')$. Obtaining a new path from S to T. We can repeat this replacement for all consecutive pairs of U-segments in p, eventually to obtain a path connecting S with T but not touching U at all. This means that there is a path in $\mathbb{P}_{G^k}(W)$ but not touching U and hence $\mathbb{P}_{G^k}(U) \neq \mathbb{P}_{G^k}(W)$, contradicting the hypothesis of the lemma. The same argument proves that $|W^p_{\max}| > k - 1$. □

Let $G = (V, E)$ be a DAG and $G^k = (V, E^k)$ the k-transitive closure of G. Notice that the identity function id is a (d.d.)-isomorphic-embedding $G \hookrightarrow_{\mathsf{id}} G^k$ (respectively a (d.i.)-isomorphic embedding $G^k \hookrightarrow_{\mathsf{id}} G$).

Theorem 3.4.12. *Let $G = (V, E)$ be a DAG. Then $\mu(G^k) \geq \mu(G) + (k-1)$.*

Proof. Assume that $\mu(G) = r$, we want to prove that $\mu(G^k) \geq r + (k - 1)$. Assume for the sake of contradiction that $\mu(G^k) < r + (k - 1)$. Then there exist $\hat{U}, \hat{W} \subseteq V$, and at least one of them, say wlog $\hat{U}$, of size strictly less than $r + (k - 1)$ such that $\mathbb{P}_{G^k}(\hat{U}) = \mathbb{P}_{G^k}(\hat{W})$. By Lemma 3.4.11 we have that $|\hat{U}|, |\hat{W}| > k - 1$. Let $I \subseteq \hat{U}$ such that $|I| = k - 1$ and let $\tilde{U} = \hat{U} \setminus I$. Let $J \subseteq \hat{W}$ such that $|J| = k - 1$ and let $\tilde{W} = \hat{W} \setminus J$. Notice that $|\tilde{U}| < r$.

We first claim that $\mathbb{P}_{G^k}(\tilde{U}) = \mathbb{P}_{G^k}(\tilde{W})$. Let $p \in \mathbb{P}_{G^k}(\tilde{U})$. Since $\tilde{U} \subset \hat{U}$, then $p \in \mathbb{P}_{G^k}(\hat{U}) = \mathbb{P}_{G^k}(\hat{W})$. Hence p is also touching $\hat{W}$. Assume that p is not touching $\tilde{W} = \hat{W} \setminus J$. Then p must necessarily touch a node in J, since otherwise $p \notin \mathbb{P}_{G^k}(\hat{W})$. Consider the set of J-segments for p. All

such subsets of J are of cardinality at most $|J| \leq k-1$. Hence according to Lemma 3.4.10 all those segments can be replaced by edges with both endpoints in $\hat{U}$ obtaining a path from S to T, touching only $\hat{U}$ but not $\hat{W}$. This is not possible since $\mathbb{P}_{G^k}(\hat{U}) = \mathbb{P}_{G^k}(\hat{W})$. This proves that $\mathbb{P}_{G^k}(\tilde{U}) \subseteq \mathbb{P}_{G^k}(\tilde{W})$ and a symmetric argument proves also that $\mathbb{P}_{G^k}(\tilde{W}) \subseteq \mathbb{P}_{G^k}(\tilde{U})$, giving the claim.

Observe that $G \hookrightarrow_{\mathsf{id}} G^k$ and we now claim that equality is kept on G passing to the counter-image, i.e. $\mathbb{P}_G(\mathsf{id}^{-1}(\tilde{U})) = \mathbb{P}_G(\mathsf{id}^{-1}(\tilde{W}))$. This is the same as saying that $\mathbb{P}_G(\tilde{U}) = \mathbb{P}_G(\tilde{W})$ and would immediately give a contradiction since we know that $\mu(G) \geq r$ and hence for any sets U and W of size at most r, $\mathbb{P}_G(U) \neq \mathbb{P}_G(W)$ but $|\hat{U}| < r$. To see the last claim notice that since $G \hookrightarrow_{\mathsf{id}} G^k$, then the same argument used in Theorem 3.4.4 proves that $\mathbb{P}_G(\tilde{U}) = \mathbb{P}_G(\tilde{W})$. □

3.5 Vertex Connectivity and Menger's Theorem

We now look at the maximal identifiability in arbitrary networks. In this section in order to get our desired results we assume that the monitor placement (S, T) are disjoint i.e., $S \cap T = \emptyset$. Theorem 1.2.2 stated in Section 1.2 will be a consequence of two independent results. In [24] it was proved that $\mu(G) \leq \delta(G)$, for any monitor placement (S, T). Here we show that $\mu(G)$ can be upper bounded in terms of κ_{ST}, the size of the minimal node set separating S from T (refer to Section 2.1 for a more precise definition) [26].

Theorem 3.5.1. *Let $G = (V, E)$ be an undirected graph and (S, T) be a monitor placement with $S \cap T = \emptyset$. Then $\mu(G) \leq \kappa_{ST}(G)$.*

Proof. If there is no vertex set in G separating S and T, $\kappa_{ST}(G) = \infty$ and the result is trivial. Let K be the set witnessing the minimal separability of S from T in G. Hence $|K| = \kappa_{ST}(G)$. Let $N(K)$ be the set of neighbours of nodes in K and notice this cannot be empty since K is disconnecting G. Pick one $w \in N(K)$ and define $U := K$ and $W := U \cup \{w\}$. Clearly $\mathbb{P}(U) \subseteq \mathbb{P}(W)$. To see the opposite inclusion assume that there exists a path from S to T passing from w but not touching $U = K$. Then K is not separating S from T in G. Contradiction. □

Note that, while in general $\kappa_{ST}(G)$ may be larger than $\delta(G)$, if S and T are separated by a set of $\kappa(G)$ vertices then, by inequality (2.1), the bound

in Theorem 3.5.1 is at least as good as the minimum degree bound proved earlier by the two authors [24]. This implies the upper bound in Theorem 1.2.2.

Moving to lower bounds, in this section we prove the following:

Theorem 3.5.2. *Let $G = (V, E)$ be an undirected connected graph and (S, T) be a monitor placement for G with $S \cap T = \emptyset$. Then $\mu(G) \geq \min(\kappa(G) - 1, |S|, |T|) - 1$.*

The lower bound in Theorem 1.2.2 can be derived easily from Theorem 3.5.2. Let K be a vertex separator in G of size $\kappa(G)$, set S^K to be the first $\lfloor \kappa(G)/2 \rfloor$ elements of K and $T^K = K \setminus S^K$. By Theorem 3.5.2 the maximal identifiability of G is at least $|S^K| - 1 = \lfloor \kappa(G)/2 \rfloor - 1$.

The proof of Theorem 3.5.2 uses Menger's Theorem, a well-known result in graph theory (see [30, Theorem 5.10, p. 48] for its proof).

Theorem 3.5.3 (Menger's Theorem)**.** *Let $G = (V, E)$ be a connected graph. Then $\kappa(G) \geq k$ if and only if each pair of nodes in V is connected by at least k node-disjoint paths in G.*

Menger's Theorem

Menger's Theorem is central to the following lemma which is used in the proof of Theorem 3.5.2.

Lemma 3.5.4. *Let $G = (V, E)$ be a connected graph. Let $W \subseteq V$ such that $|W| \leq \kappa(G) - 2$. Then any pair of vertices in $V \setminus W$ is connected by at least two vertex-disjoint simple paths not touching W.*

Proof. By Menger's Theorem, for any pair of nodes u and v in $V \setminus W$ there are at least $\kappa(G)$ vertex-disjoint paths from u to v in G. Call $\mathbb{P}$ the set of such paths. Since $|W| \leq \kappa(G) - 2$, then the nodes of W can be in at most $\kappa(G) - 2$ of paths in $\mathbb{P}$. Hence there are at least two paths in $\mathbb{P}$ not touching W. □

Let $G = (V, E)$ be a graph, $W \subseteq V$ and p a path in G. We say that p is *W-free* if no node of p is in W.

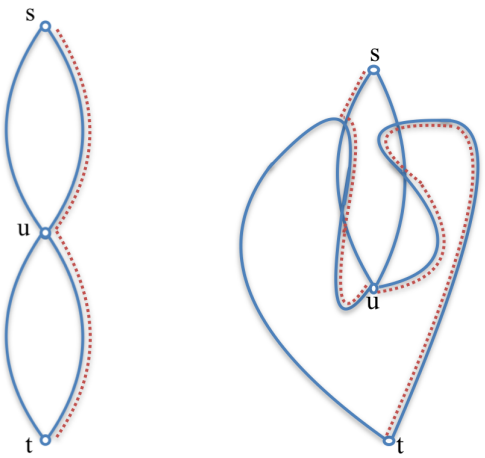
W-free

Lemma 3.5.5. *Let $G = (V, E)$ be an undirected connected graph. Let s, t, u be three distinct nodes in V. If in G there are two vertex-disjoint paths from s to u and two vertex-disjoint paths from u to t, then there is a simple path in G from s to t passing through u.*

Proof. Let π_1^s, π_2^s be the two vertex-disjoint paths from s to u. Let π_1^t, π_2^t be the two vertex-disjoint paths from u to t. If at least one between π_1^s,

and π_2^s only intersects one of π_1^t, and π_2^t at u then the concatenation of such paths is a (longer) simple path from s to t passing through u (Figure 3.4 (a)). Otherwise the concatenation of one between π_1^s, and π_2^s with one between π_1^t, and π_2^t is a non-simple path. In what follows we show that the subgraph of G induced by the four paths does contain a simple path from s to t passing through u (as an example see Figure 3.4 (b)). In the construction below we exploit the fact that π_1^s, and π_2^s (resp. π_1^t, and π_2^t) are simple and vertex disjoint. Let p be a path from s to u. Define an order on the nodes of p as follows: $v \prec_p w$ if going from v to u we pass through w. From now on we will use $\prec$ instead of $\prec_p$ when the path under consideration will be clear from the context. Let Z_1^j be the nodes in $\pi_1^s \cap \pi_j^t$. Z_1^1 and Z_1^2 are disjoint but there will be a node in those sets, say z, which is minimal according to $\prec$. Without loss of generality let us say that $z \in Z_1^1$. The subpath $\pi_1^s[s \dots z]$ of π_1^s going from s to z, is intersecting neither π_1^t nor π_2^t. Hence the concatenation of the following three disjoint paths defines a simple path from s to t passing through u, hence a path with the required properties:

1. $\pi_1^s[s \dots z]$, going from s to z;
2. $\pi_1^t[z \dots u]$ a sub-path of π_1^t going from u to z and traversed in the other direction;
3. π_2^t, connecting u to t.



(a) (b)

Figure 3.4. Examples of simple paths from s to t passing through u .

□

Proof of Theorem 3.5.2. Let $G = (V, E)$ be an undirected connected graph and (S, T) be a monitor placement in G with $S \cap T = \emptyset$. Note that without loss of generality $\min(\kappa(G) - 1, |S|, |T|) > 1$ (for otherwise there is nothing to prove).

Assume first that $|S| \geq \kappa(G) - 1$ and $|T| \geq \kappa(G) - 1$. We claim that

$$\mu(G) \geq \kappa(G) - 2.$$

We show that for any distinct non-empty subsets U and W of V of size at most $\kappa(G) - 2$, there is a path in $\mathbb{P}$ touching exactly one between U and W. Given such U and W, fix a node $u \in U \triangle W$ and assume w.l.o.g. that $u \in U$. Since $|W| \leq \kappa(G) - 2$ and $|S| \geq \kappa(G) - 1$ there is at least a node in $s \in S \setminus W$. By Lemma 3.5.4 applied to nodes s and u and to the set W, there are two vertex-disjoint simple paths π_1^s, π_2^s from s to u not touching W. The same reasoning applied to T, guarantees the existence of a node $t \in T \setminus W$ and two vertex-disjoint paths π_1^t, π_2^t from u to t not touching W. Let $\pi_1^s, \pi_2^s, \pi_1^t, \pi_2^t$ be the paths in the graph $G - W$ (the subgraph of G obtained by removing nodes in W). Hence Lemma 3.5.5 gives a simple path from s to t passing through u in $G - W$, hence a simple path from s to t passing through u in G, avoiding W.

Now assume that at least one between $|S|$ and $|T|$ is less than $\kappa(G) - 1$. Let $r = \min(|S|, |T|) - 1$. As before we prove that for all distinct non-empty U and W subsets of V of size at most r, there is an $S - T$ path in G, hence in $\mathbb{P}$, touching exactly one between U and W. Let $u \in U \triangle W$ and without loss of generality assume $u \in U$. Notice that $r + 1 = \min(|S|, |T|)$, then both $|S| \geq r + 1$ and $|T| \geq r + 1$. Since $|W| \leq r$, as before there are $s \in S \setminus W$ and $t \in T \setminus W$. Furthermore, since $\kappa(G) \geq \min(|S|, |T|)$, then by previous observation on $|S|$ and $|T|$, $\kappa(G) \geq r + 1$ and, since $|W| \leq r$, then $\kappa(G) - |W| \geq 2$, that is $|W| \leq \kappa(G) - 2$. As in the previous case we can apply the claim above once to s, u and W getting the vertex-disjoint paths π_1^s and π_2^s from s to u, and once to t, u and W getting the vertex-disjoint paths π_1^t and π_2^t from t to u. The proof then follows by the same steps as in the previous case. We then have proved that if $|S|$ or $|T|$ are smaller than $\kappa(G) - 1$, then $\mu(G) \geq \min(|S|, |T|) - 1$ and the proof of Theorem 3.5.2 is complete. □

Proof of Theorem 1.2.3. We complete this section investigating a different way to relate the graph vertex connectivity to $\mu(G)$. It is easy to see that, in general, the bounds in Theorem 1.2.2 are not very tight,

particularly when $\kappa(G)$ is large. However, if $\kappa(G)$ is small, we can do better.

In what follows let K be a minimal vertex separator in G. Let $G_i^K = (V_i^K, E_i^K)$, $i \in \{1, \dots, r_K\}$ be the $r_K \geq 2$ connected components remaining in G after removing K. Since $\kappa(G) \leq \frac{n}{3}$, then $2\kappa(G) \leq n - \kappa(G)$ and one can define disjoint sets S, and T with $\kappa(G)$ vertices each in such a way that the smallest among the V_i^K's contains only elements of S. This can be done as follows: if the smallest of V_i^K's has less than $\kappa(G) - \ell$ nodes, then assign all its nodes to S. Then use the other components G_j^K's to assign ℓ nodes to S and $\kappa(G)$ other nodes to T. If the smallest V_i^K has more than $\kappa(G)$ nodes, choose $\kappa(G)$ nodes among them and put them in S. Choose $\kappa(G)$ nodes in other components and assign them to T.

We now prove that the set of simple paths between S and T defined as above allows a very high identifiability. The lower bound on $\mu(G)$ follows from Theorem 3.5.2 noticing that $|S| = |T| > \kappa(G) - 1$. We now prove that $\mu(G) \leq \kappa(G)$. Let G_i^K be the component where all the S-nodes are assigned. Let w be a node in $V_i^K \cap N(K)$. This node has to exists since G was connected and the removal of K is disconnecting G_i^K from K. Fix $U = K$ and $W = K \cup \{w\}$. We will show that $\mathbb{P}(U) = \mathbb{P}(W)$. It suffices to prove that $\mathbb{P}(\{w\}) \subseteq \mathbb{P}(K)$, since clearly $\mathbb{P}(U) \subseteq \mathbb{P}(W)$. Observe that no $S - T$ path p in G can live entirely inside G_i^K, i.e. have all of its nodes in V_i^K. This is because at least one end-point (that in T) is necessarily missing in any path entirely living only in G_i^K. Hence a path touching w is either entering or leaving G_i^K. But outside of G_i^K, w is connected only to K, since otherwise K would not be a minimal vertex separator. Hence it must be $\mathbb{P}(\{w\}) \subseteq \mathbb{P}(K)$. We have found U, W of size $\leq \kappa(G) + 1$ such that $\mathbb{P}(U) = \mathbb{P}(W)$. The upper bound follows. □

Arbitrary LoS networks have minimum degree, and hence also vertex connectivity at most $2d(\omega - 1)$. The next corollary follows directly from Theorem 1.2.3.

Corollary 3.5.6. *Let G be an undirected LoS network over n nodes and with fixed range parameter ω, independent of n, such that $n \geq \omega$. Then $\mu(G) \geq \kappa(G) - 2$.*

3.6 Open Problems

We shortly address some directions related to our approach which might be further explored in the analysis of identifiability of failure nodes. In 1982, [51] showed that for $k \geq 3$ to test if a partial order has dimension $\leq k$ is NP-complete. Nevertheless there are some algorithms to compute the dimension of poset [48,50] which are practically used. It would be interesting to further explore connections between Boolean Network Tomography and poset dimension theory to get better estimates on the maximal identifiability for DAG network topologies. It is a well-known result [44] that planar graphs over n nodes can be embedded through a straight line embedding into a $(n-2) \times (n-2)$ 2-dimensional grid. It seems not difficult to see that our results on embeddability can be generalized to obtain a lower bound of 2 for the maximal identifiability when a network is a planar graph.

k-Transitive-Closure-Spanner of a graph G

A k-Transitive-Closure-Spanner of a graph G is a graph H with a small diameter - k- that preserves the connectivity of the original graph. These graphs and their relations with dimension of poset were recently studied in [5]. From our results it is clear that adding edges to a graph G can strengthen the potential of failure identifiability. Are k-TC-Spanners and in particular Steiner-k-TC-Spanners (see [5]) useful to maximize failure identifiability of a network?

Chapter 4

Bounds on Identifiability for Some Network Topologies

In this chapter we want to explore maximal identifiability for specific class of topologies such as trees, grids and augmented hypergrids, which are among the topologies most used and implemented in real networks and random networks.

First we prove upper and lower bounds on the maximal identifiability for trees and d-(dimensional) grids, in both directed and undirected cases. For trees we obtain that the maximal node identifiability is 1. This result has to be interpreted as saying that if our network topology is a tree then maximal number of failed nodes we can hope to uniquely identify is 1. Then searching for topologies which are better than trees with respect to maximal node-failure identifiability, we proved that d-grids, under a suitable optimal monitor placement, can reach a better identifiability strictly greater than 1. We prove that directed d-grids with support n have maximal identifiability d using $2d(n-1)+2$ monitors on the d-grid; and in the undirected case we show that $2d$ monitors suffice to get identifiability of at least $d-1$ and at most d. When one consider the minimal number of monitors to reach the maximal identifiability on d-grids, our results mark an important difference between the directed and undirected cases. In the latter we can show how to get tight lower and upper bound results using only $2d$ monitors (so independently of n). In the directed case instead the number of monitors to reach a maximal identifiability depends linearly on the number of nodes and cannot be improved.

Moreover we are handling class of graphs which are very regular combinatorial objects. In order to prove lower bounds on maximal identifiability,

instead of checking experimentally the optimality of the upper bounds as in previous works, we use an algorithmic/combinatorial analysis, so obtaining tight results. This approach directly leads to algorithms to design network topologies with a guarantee of reaching a precise maximal identifiability of failure nodes.

Furthermore in an attempt to generalize to more general graphs than hypergrids, we start by studying the identifiability of Line-of-Sight (LoS) networks. We consider Line-of-Sight networks and we characterize the maximal identifiability of such networks highlighting that vertex-connectivity plays a central role.

Finally we initiate the study of maximal identifiability for random networks. We focus on two models: Erdős-Rényi model and Random Regular graphs. The framework proposed in the thesis allows a probabilistic analysis of the identifiability in random networks giving tradeoffs between the number of monitors to place and the maximal identifiability. The last section of this chapter is divided into three parts: First we analyze the case of Erdős-Rényi graphs and we show a simple analysis to prove sub-linear maximal identifiability. A more refined analysis is sketched in the second part for Erdős-Rényi graphs reaching an optimal linear separability. Finally we analyze maximal identifiability for the case of random regular graphs. Random graphs give us constructions of networks with large identifiability.

To sum up, the organization of this chapter is as the following:

Section 4.1 includes the bounds on the maximal identifiability of trees, both in the directed (Theorem 4.1.1) and the undirected case (Theorem 4.1.4).

In *Section 4.2* we prove tight bounds for maximal identifiability for d-grids undirected (Theorem 4.2.1) and directed (Theorem 4.2.9).

In *Section 4.3* we show tight bounds on the maximal identifiability in a particular class of graphs, augmented hypergrids, cf. Theorem 4.3.1.

At last *Section 4.4* is dedicated to the analysis of the maximal identifiability on random graphs. First we look at Erdős-Rényi graphs (Theorem 4.4.4), then random regular graphs, cf. Theorem 4.4.5. All our results gave immediate tradeoffs between maximal identifiability and number of monitors.

4.1 Trees

Let $\mathcal{T}_n$ be a directed tree. For downward trees we consider the monitor placement $\chi_{\mathtt{t}}$ which includes in S the root of $\mathcal{T}_n$ and in T all the leaves. Vice versa in an upward tree $\chi_{\mathtt{t}}$ assigns the root of $\mathcal{T}_n$ in T and the leaves in S (see Figure 4.1).

Theorem 4.1.1. *Let $\mathcal{T}_n$ be a directed tree and $\chi_{\mathtt{t}}$ the monitor placement which assigns the roots and the leaves of $\mathcal{T}_n$ in S and T according to be a downward tree or an upward tree with $S \cap T = \emptyset$. Then $\mu(\mathcal{T}_n|\chi_{\mathtt{t}}) = 1$.*

Proof. We assume the tree to be line-free (LF) so that the bound depends only on the topology and not on the fact that contains a line. Consider a node u in $\mathcal{T}_n$. Since $\mathcal{T}_n$ is LF u has either in-degree ≥ 2 or out-degree ≥ 2. According to whether the tree is downward or upward, one of the two cases in Figure 4.1 can happen:

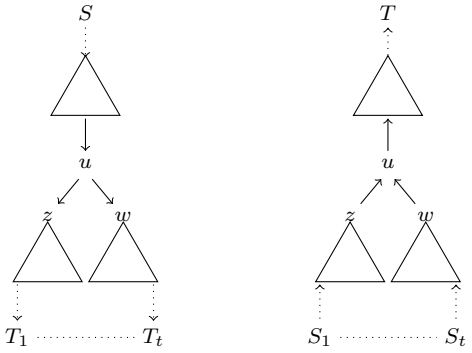


Figure 4.1. *Directed trees with monitor placement $\chi_{\mathtt{t}}$.*

For the upper bound: fix $W = \{u, w\}$ and $U = \{u\}$. $\mathbb{P}(U) \subseteq \mathbb{P}(W)$. Moreover in both cases each path passing through w is also touching u. Hence $\mathbb{P}(\{w\}) \subseteq \mathbb{P}(\{u\})$. Therefore $\mathbb{P}(W) \subseteq \mathbb{P}(U)$ and then $\mathbb{P}(U) = \mathbb{P}(W)$ and $\mathbb{P}(U) \triangle \mathbb{P}(W) = \emptyset$. For the lower bound, let u and w be two distinct nodes in $\mathcal{T}_n$. Let $U = \{u\}$ and $W = \{w\}$. Each node in $\mathcal{T}_n$ is on some path from the root to a leaf. If u and w lie on different paths, then clearly there are paths in $\mathbb{P}(U)$ but not in $\mathbb{P}(W)$. Hence $\mathbb{P}(U) \triangle \mathbb{P}(W) \neq \emptyset$. If u and w lie on the same path p and say that p meets w before u. Let p_w be the subpath of p truncated at node w. Let $w_1 \in N_{\mathtt{o}}(w)$ be the neighbour of w lying on p. Since $\mathcal{T}_n$ is LF there is necessarily another node $w_2 \neq w_1, w_2 \in N_{\mathtt{o}}(w)$

and in $\mathcal{T}_n$ there is a path q from w_2 to a leaf. Hence the concatenation of p_w with q is a path from the root to a leaf touching w but not u. Hence in $\mathbb{P}(W)$ but not in $\mathbb{P}(U)$. Hence $\mathbb{P}(U) \triangle \mathbb{P}(W) \neq \emptyset$. □

Optimality of $\chi_{\mathtt{t}}$**.** Notice that the monitor placement $\chi_{\mathtt{t}}$ in both cases is optimal. Consider the downward case: if we modify $\chi_{\mathtt{t}}$ by removing one output monitor from a leaf, say u, then $\mu(\mathcal{T}_n) = 0$: let v be the node parent of u and let w be its other son. From $\{w\}$ and $\{v\}$ pass exactly one path. Hence $\mu(\mathcal{T}_n) < 1$.

We now consider undirected trees. We first start by observing that a large class of monitor placement on undirected trees make maximal identifiability 0. We later see that under any other monitor placement undirected trees have μ exactly 1 as well. Let $\mathcal{T}_n$ be an undirected tree and $\chi = (S, T)$ be a monitor placement for $\mathcal{T}_n$. We say that $\mathcal{T}_n$ is an *input tree* (respectively *output tree*) with respect to χ if there is a node of $\mathcal{T}_n$ in S (respectively in T). Notice that a tree can be both an input and an output tree. Given a tree $\mathcal{T}_n$ and one of its edges $e = (u, v)$, let $T^e(u)$ (respectively $T^e(v)$) be the subtrees of $\mathcal{T}_n$ obtained from cutting in $\mathcal{T}_n$ the edge (u, v) and taking the tree rooted at u (respectively v). For a node $u \in \mathcal{T}_n$, we call the *u-subtrees of* $\mathcal{T}_n$ the family of trees $\{T^{(w,u)}(w)\}_{w \in N(u)}$.

Definition 4.1.2 (Monitor-balanced tree)**.** *An undirected tree $\mathcal{T}_n$ is* monitor-balanced *under χ if for each non-leaf node u in $\mathcal{T}_n$ the family $\{T^{(w,u)}(w)\}_{w \in N(u)}$ of the u-subtrees of $\mathcal{T}_n$ contains at least two input trees and at least two output trees.* *Monitor-balanced tree*

Lemma 4.1.3. *If an undirected tree $\mathcal{T}_n$ is not monitor-balanced under χ, then $\mu(\mathcal{T}_n|\chi) = 0$.*

Proof. If $\mathcal{T}_n$ is not monitor-balanced, then there is a non-leaf node u in $\mathcal{T}_n$ such that the family $\{T^{(w,u)}(w)\}_{w \in N(u)}$ contains either only one input tree or only one output tree. There are hence only three possible cases at such a node u that can happen and which are visualized in Figure 4.2.

In all the cases we set $U = \{u\}$ and $W = \{w\}$. Since any path must necessarily go from an input node to an output node, then $\mathbb{P}(U) = \mathbb{P}(W)$. This proves that $\mu(\mathcal{T}_n|\chi) < 1$. □

On the other hand when χ is balanced, a similar proof as in Theorem 4.1.1 (that we omit) proves that,

Theorem 4.1.4. *Let $\mathcal{T}_n$ be an undirected tree and χ a monitor-balanced monitor placement for $\mathcal{T}_n$. Then $\mu(\mathcal{T}_n|\chi) = 1$.*

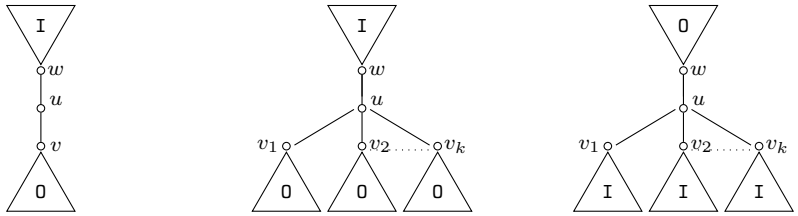


Figure 4.2. *The three possible cases of Lemma 4.1.3 for u when $\mathcal{T}_n$ is not monitor-balanced.*

We remark that precise relations between maximal identifiability and vertex connectivity were found in [26].

4.2 Grids

Can we find topologies whose maximal identifiability is strictly greater than 1? We consider the case of d-dimensional grids and we start with the undirected case.

We show that d input and d output monitors suffice to get maximal identifiability at least $d-1$ and at most d in the case of undirected d-dimensional grids *for any* monitor placement.

Theorem 4.2.1. *Let $n \geq 3$ and $\mathcal{H}_{n,d}$ be an undirected d-dimensional grid. Then $d-1 \leq \mu(\mathcal{H}_{n,d}|\chi) \leq d$ for any monitor placement $\chi = (S,T)$ such that $|S|, |T| = d$.*

The rest of the section contains the proof for the case $d = 2$. The proof for dimension $d > 2$ is essentially the same and we omit the proof.

Proof. (of Theorem 4.2.1) The upper bound follows from Lemma 3.2.1. For the lower bound we consider the following claim.

Claim 4.2.2. *Let $z_1 = (i_1, j_1), z_2 = (i_2, j_2), z_3 = (i_3, j_3)$ be three nodes in $\mathcal{H}_n$ such that z_1 and z_3 are distinct nodes. There exists a simple path from z_1 to z_3 touching z_2.*

Proof. First we consider a rectangle/square of four paths in $\mathcal{H}_n$ such that all these three nodes are lying on the edges of this rectangle/square (see Figure 4.3 for an example). Then we start from the node that we want to be the origin of our path and move along the edge towards our second node that we want to be touched by our path. After reaching the second

node we continue moving along the edge which will lead to the third node that our path terminates at. We then build a path from z_1 to z_3 touching z_2. □

Now we have to prove that independently of what nodes form S and T, for any $U, W \subseteq V$ with $U \triangle W \neq \emptyset$ such that $|U|, |W| \leq 1$, then $\mathbb{P}(U) \triangle \mathbb{P}(W) \neq \emptyset$. Since $U \neq W$, then there is at least one $S \notin W$, at least one $T \notin W$ and wlog a $u \in U \setminus W$. By Claim 4.2.2 we get a simple path from S to T passing through u. If this path touches w, then we can avoid it. If w is an internal node (not on the borders), in order to avoid w we remove this node and all the edges linked to it. Then we have a hole in our grid. By previous observation, after removing w, at least one node in S and one node in T are in the remaining network and they must be different since we we do not have loops or DLP. By previous claim applied to S, T and u we have an $S - T$ path in $\mathcal{H}_n$ touching U but not W. Notice that if a part of the rectangle/square that we are considering in Claim 4.2.2 intersects with our hole then we can move along the borders of our hole (see Figure 4.4 for an example). If w is on the border but u is an internal node, then by the same argument as above we can touch w and avoid u. If both w and u are on the same border and one of them say u is isolated by w, S and T (see Figure 4.5 for an example), then we remove u and the edges linked to it and again by the same argument as above we have a path $S - T$ touching W but not U. □

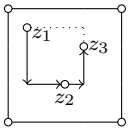


Figure 4.3. *Building a path in $\mathcal{H}_n$ touching three points.*

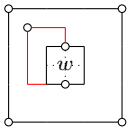


Figure 4.4. *Avoiding a hole in $\mathcal{H}_n$.*

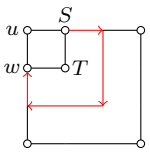


Figure 4.5. *Avoiding a hole in a corner in $\mathcal{H}_n$.*

We now turn on considering directed grids. Paths are more constrained in a directed grid, hence we expect that to reach a maximal identifiability of d we have to place more monitors. Our next result proves that in fact we can have a tight maximal identifiability of d on directed grids but the number of monitors is now linear in n, the support of the grid. Furthermore

the result does not hold for any monitor placement but for a specific one, which however we prove to be optimal.

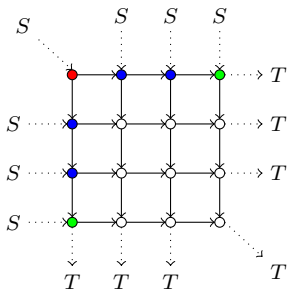


Figure 4.6. *A directed grid $\mathcal{H}_4$ with the monitor placement $\chi_{\mathtt{g}}$.*

Let us consider the monitor placement $\chi_{\mathtt{g}}$ for $\mathcal{H}_n$ as in Figure 4.6. Precisely $\chi_{\mathtt{g}}$ is defined by $S = \{(1,1), \dots (1,n), (2,1), \dots, (n,1)\}$ and $T = \{(n,1), (n,2), \dots (n,n), (1,n), (2,n), \dots, (n-1,n)\}$. $(1,1)$ is the only simple source node and Lemma 3.2.3 can be applied to this case.

Lemma 4.2.3. *Let $n \geq 3$ and $\mathcal{H}_n$ be a directed grid. Then $\mu(\mathcal{H}_n|\chi_{\mathtt{g}}) \leq 2$.*

To prove a matching lower bound on $\mu(\mathcal{H}_n|\chi_{\mathtt{g}})$ we prove that any two distinct node sets U and W of size at most 2 can be separated by a path in $\mathbb{P}(\mathcal{H}_n|\chi_{\mathtt{g}})$. Since we do not have DLP or loop paths, we have to be careful that no path separating U from W could be a DLP path. Since the only nodes which are both input and output are $(1,n)$ and $(n,1)$, then to separate U from W by a path we have to avoid the two DLP paths $S{\cdot}(1,n){\cdot}T$ or $S{\cdot}(n,1){\cdot}T$ and any other loop path starting and ending in the same node. Hence we assign a special role to the complex sources $(1,n)$ and $(n,1)$ (green nodes in Figure 4.6) and we consider the following assumption (which we prove to fulfill) about paths that separate sets of nodes.

Assumption 4.2.4. *Nodes $(1,n)$ and $(n,1)$ can be endpoint but never starting point of a path starting in S and ending in T.*

Let us denote with V the nodes in the directed grid $\mathcal{H}_n$ and with V^- the nodes of $\mathcal{H}_n$ except for $(1,n)$ and $(n,1)$. By the definition of $\chi_{\mathtt{g}}$, S and T are both formed by the border nodes. Hence to fulfill our assumption we define $S' = S \setminus \{(1,n),(n,1)\}$, and $T' = T$. Given a node u in V, let $S(u) = \{v \in V^- \mid \exists \text{ a path from } v \text{ to } u \text{ in } \mathcal{H}_n\}$ and $T(u) = \{v \in V \mid \exists \text{ a path from } u \text{ to } v \text{ in } \mathcal{H}_n\}$.

The following lemmas give a way to build paths avoiding specific nodes. We always assume $n \geq 3$ since otherwise, independently of d, the directed d-dimensional grid $\mathcal{H}_{n,d}$ would have no node with degree $2d$.

Lemma 4.2.5. *Let $n \geq 3$. Let u be a node in V^- and $w \in S(u)$ with $w \neq u$. There is a path p_u^w from a node in S' to u not touching w.*

Proof. By induction on $S(u)$. If $S(u) = \{u\}$ for some $u \in S'$, then u is linked to an input monitor (notice in S' we do not have the two mentioned complex sources) and since $u \neq w$, then p_u^w is the path made by the only node u. In the inductive hypothesis $|N_{\mathtt{i}}(u)| = 2$, Hence there is $w_1 \in N_{\mathtt{i}}(u)$ such that $w_1 \neq w$. Since $|N_{\mathtt{i}}(u)| = 2$, then $S(w_1) \subset S(u)$. By induction there is a path $p_{w_1}^w$ from S' to w_1 avoiding w. Then define as p_u^w, the path concatenating $p_{w_1}^w$ with u. □

A similar proof holds also for the nodes in T reachable from u without worrying about the two nodes $\{(1,n),(n,1)\}$.

Lemma 4.2.6. *Let $n \geq 3$. Let u be a node in the directed grid $\mathcal{H}_n$ and $w \in T(u)$ with $w \neq u$. There is path q_u^w from u to a node in T not touching w.*

Next claim handles the case when one among U and W contains at least a complex source. In this case to fulfill our assumption, we have to show an $S - T$ path touching exactly one between U and W which is not starting neither with $(1,n)$ nor with $(n,1)$. This immediately implies that this path can be neither $S \cdot (1,n) \cdot T$ nor $S \cdot (n,1) \cdot T$.

Claim 4.2.7. *Let U, W be non-empty sets of nodes of the directed grid $\mathcal{H}_n$, $n \geq 3$ such that $|U|, |W| \leq 2$ and at least one of the complex sources $(1,n)$ or $(n,1)$ belongs to one of them. Then there is a path from a node in S to a node in T passing though exactly one between U and W fulfilling Assumption 4.2.4.*

Proof. Assume without loss of generality that $(1,n) \in U$ (the case where $(n,1) \in U$ is symmetric). Let $N((1,n))$ be the neighbours of $(1,n)$ in $\mathcal{H}_n$, i.e. $N((1,n)) = \{(1,n-1),(2,n)\}$. We distinguish the following 4 cases:

1. $W \cap N((1,n)) = \emptyset$;
2. $W = N((1,n))$
3. $W \cap N((1,n)) = \{(1,n-1)\}$

4. $W \cap N((1,n)) = \{(2,n)\}$

In each of these cases we find an $S-T$ path touching only one between U and W fulfilling Assumption 4.2.4.

In case (1) and in case (4) the path $(1,n-1)\cdot(1,n)$, is a path touching U but not W fulfilling our assumption.

In case (2) $(1,n)$ is completely surrounded by W. So we will build a path touching W but not U. If the node $(2,n-1)$ is not in U then the path $(1,n-1)\cdot(2,n-1)\cdot(2,n)$ proves the claim. If instead $(2,n-1)$ is in U we have to avoid it. We use here that $n \geq 3$ to build the path starting in $(1,n-1)$ going up to $(1,n-2)$, then going right until the node $(3,n-2)$ and finally going down to the T node $(3,n)$.

In case (3) we distinguish the following two cases according to whether $U \cap \{(2,n-1),(2,n)\} = \emptyset$ or not. In the first case the path $(1,n-1)\cdot(2,n-1)\cdot(2,n)$ touches only W and fulfill the assumption. In the second case, we follow case (2) and avoid both nodes in $\{(2,n-1),(2,n)\}$ using the fact that $n \geq 3$. We start in $(1,n-1)$, go up to $(1,n-2)$, then right up to $(3,n-2)$ and finally down to $(3,n)$. This path touches W but not U and fulfill the assumption. □

Lemma 4.2.8. *(Main Lemma) Let $n \geq 3$ and $\mathcal{H}_n$ be a directed grid. $\mu(\mathcal{H}_n|\chi_{\mathsf{g}}) \geq 2$.*

Proof. Let V be the set of nodes of $\mathcal{H}_n$. We have to prove that for any $U, W \subseteq V$ with $U \neq W$ and such that $|U|, |W| \leq 2$, $\mathbb{P}(U) \triangle \mathbb{P}(W) \neq \emptyset$. It is sufficient to find a path $p \in \mathcal{H}_n$ from S to T touching exactly one between U and W. By Claim 4.2.7 we can assume that neither of U and W contain $(1,n)$ and $(n,1)$. So in the rest of the proof we work only with S and no node will be ever $(1,n)$ and $(n,1)$. We split in the following cases:

1. at least one between U and W has cardinality 1;
2. both U and W have cardinality 2.

Case 1. Assume wlog that $W = \{w\}$. Since $U \triangle W \neq \emptyset$, then there is a node $u \in U \setminus W$, such that $w \neq u$. w can be either in (1) $S(u)$, or (2) in $T(u)$; or (3) in $V \setminus (S(u) \cup T(u))$.

In case (3) any path p from S to T passing through u is not touching w and proves the claim. In case (1) we use Lemma 4.2.5 to have a path p_u^w from S to u avoiding W. Moreover, any path p from u to T is avoiding w. Then the path $p_u^w \cdot p$, concatenation of p_u^w with p proves the claim. In case

(2) any path p from S to u avoids w, and Lemma 4.2.6 guarantees a path q_u^w from u to T avoiding w. Hence the path $p \cdot q_u^w$, concatenation of p and q_u^w proves the claim.

Case 2. Observe that though $U \triangle W \neq \emptyset$, they might share a node. So there might be two cases: (A) $|U \cap W| = 1$ and (B) $|U \cap W| = 0$. In case (A) we fix u to be the node of U not in W. In case (B) say $U = \{u_0, u_1\}$ we fix u to be the node in U not reachable in $\mathcal{H}_n$ by the other node in U, i.e. the u_i such that $u_i \notin S(u_{1-i})$. Notice that this node always exists since the nodes in U cannot reach each other in $\mathcal{H}_n$. As in case (1) we divide in three cases according to the position of W wrt u.

i. $W \subseteq S(u)$;

ii. $W \subseteq T(u)$;

iii. $|S(u) \cap W| \leq 1$ and $|T(u) \cap W| \leq 1$;

In case (iii) a similar argument as above works. Since $|S(u) \cap W| \leq 1$, then either (if $|S(u) \cap W| = 0$) any path from S to u avoids W, or (if $|S(u) \cap W| = 1$) we can apply Lemma 4.2.5 to find a path p_u from S to u avoiding W. Using $|T(u) \cap W| \leq 1$, a similar argument works for finding a path q_u from u to T avoiding W. Hence the path $p_u \cdot q_u$, concatenation of p_u and q_u is a path from S to T passing from u but avoiding W. In case (i) we further distinguish two cases and fix the w as follows:

(A) $|U \cap W| = 1$. w is the only node in $U \cap W$.

(B) w is any node in W. Denote by v be the other node in W.

In case (A) $U = \{u, w\}$ and $W = \{w, v\}$, hence since $u \neq w$, then by Lemma 4.2.5 there is a path p_u^w from S to u avoiding w. Moreover, since $W \subseteq S(u)$ any path q_u from u to T avoids W. Hence the path $p_u := p_u^w \cdot q_u$ which is the concatenation of p_u^w with q_u, touches U and avoids W. This path proves the claim unless $v \in p_u$, and precisely $v \in p_u^w$, since $W \subseteq S(u)$ and q_u lives only in $T(u)$.

If $v \in p_u$ then we modify p_u into a new path p_v touching v, hence the set W, but avoiding U and this will prove the claim. To do this we first identify a node z lying on p_u^w before u but after v and we consider the subpath p_z^w of p_u^w stopping at z, hence touching v. The node z is defined as follows: assume u to be the node $u = (x_1, x_2)$ with $x_1, x_2 \in [n]$. Since p_u^w is ending at u and $\mathcal{H}_n$ is directed, there is a first node z_1 in p_u^w such that

starting from z_1 all the nodes $z_1, \dots z_r, u$ of the subpath of p_u^w starting at z_1 lie either on the same row (x_1) or on the same column (x_2) of u. z is defined to be either z_1 or z_i if $v = z_i$ for some $i \in [r]$. The main properties of z are that: $u \in T(z)$ and that $w \notin p_z^w$. The first is straightforward. For the latter first notice that before z no node on p_z^w can be w because p_z^w is a subpath of p_u^w. Furthermore $w \notin T(z)$ since z is by definition on the same border of u and hence $S(u) \cap T(z)$ is the set of nodes $\{z_1, \dots z_r\}$ and none of them can be w.

Since $u \in T(z)$ and $z \neq u$[1], then we can use Lemma 4.2.6 on z and u to find a path q_z^u from z to T avoiding u. Define $p_v := p_z^w \cdot q_z^u$ the path concatenating p_z^w with q_z^u. p_v touches v but avoids both u and w, hence touches W but avoids U. Case (A) is proved.

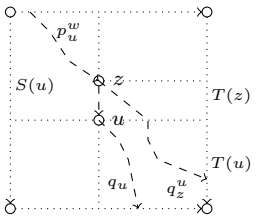


Figure 4.7. *Case (i).A*

In Case (B) $w \notin U$ and let u_1 be the other node of U. So $U = \{u, u_1\}$ and $W = \{w, v\}$. The same proof of Case (A) works here too. If $v \in p_u$ however we have to be slightly more careful. Assume without loss of generality that u_1 appears before u on p_u (the other case is exactly the same swapping u and u_1). We want to build a path p_v avoiding both u_1 and u. Since $v \in S(u)$, we can have two cases: (1) v is before both u_1 and u; and (2) v is in between u_1 and u. Let t_v be the subpath of p_u ending in v. In case (1) we use a first time Lemma 4.2.6 on v and u_1 to find a path t from v to T avoiding u_1. If t still passes through u, then we notice that u cannot be on the border of the grid, since otherwise u_1 would also be on the same border and hence t would not avoid u_1. Hence u is an internal node in the grid. Let $\{i_1, i_2\}$ be the incoming nodes in u. Only one can be u_1, say i_1. Hence t is entering in u through i_2. Let t_2 be the subpath of t ending at i_2. Again by Lemma 4.2.6 on i_2 and u we can find a path q from i_2 to T avoiding u. The path $p_v := t_2 \cdot q$ concatenating t_2 with q proves the claim. Case (2) is easier. First we apply Lemma 4.2.5 on v and

[1] If v is a source node and $v \in N_{\text{i}}(u)$, then z is v itself.

u_1 to have path t from S to v avoiding u_1. Then we apply Lemma 4.2.6 on v and u to have a path q from v to T avoiding u. Then the concatenation of t with q proves the claim. In case (ii) a similar argument of case (i) but on $T(u)$ works. We left the details to the reader. □

Together previous Lemma and Lemma 4.2.3, imply the following.

Theorem 4.2.9. *Let $n \in \mathbb{N}$, $n \geq 3$ and $\mathcal{H}_n$ be a directed grid. Then $\mu(\mathcal{H}_n|\chi_{\mathrm{g}}) = 2$.*

Previous result can be easily proved for grids of dimension $d > 2$ generalizing the definitions and the proofs to the case of a generic d. We omit the details.

Theorem 4.2.10. *Let $d, n \in \mathbb{N}$, $d > 2$, $n \geq 3$ ans $\mathcal{H}_{n,d}$ be a d-dimensional directed grid. Then $\mu(\mathcal{H}_{n,d}|\chi_{\mathrm{g}}) = d$.*

Optimality of χ_{g}. In the case $d = 2$ we were using $4n - 2$ monitors. We wonder whether the number of monitors can be reduced. The answer is essentially no. Namely, it is easy to see that if in the monitor placement used in χ_{g} for Theorem 4.2.9 we remove the input links to nodes $(1, 2)$ and $(2, 1)$ (so we have $4n - 4$ monitors), the sets $U = \{(1, 2), (2, 1)\}$ and $W = \{(1, 1)\}$ cannot be separated by any path in $\mathcal{H}_n$.

4.3 Augmented Hypergrids

Let $\omega > 2$ be an integer. In this section we analyze the maximal identifiability of undirected augmented hypergrids. To maximize clarity, we provide full details for $\mathcal{H}_{n,d,\omega}$, the d-dimensional augmented hypergrid. In [24], we showed that $\mu(G) \leq \delta(G)$ for any $(\mathbb{P}, S, T)$. In $\mathcal{H}_{n,d,\omega}$ each node u has $\omega - 1$ edges for each one of the possible directions (we have d directions). Hence the minimal degree in $\mathcal{H}_{n,d,\omega}$ is reached at the corner nodes and it is $d(\omega - 1)$. Thus $\mu(\mathcal{H}_{n,d,\omega}) \leq d(\omega - 1)$ for any $(\mathbb{P}, S, T)$. In the remainder of this section we pair this up with a tight lower bound for any monitor placement. Note that these results readily imply the upper and lower bounds in Theorem 1.2.1 as in the augmented hypegrids the vertex connectivity is actually equal to the network's minimum degree. The rest of this section focuses on the second inequality in that theorem. First notice that we shorten $\mathcal{H}_{n,d,\omega}$ with $\mathcal{H}$.

Theorem 4.3.1. *Let $n, \omega, d \in \mathbb{N}$, $\omega > 2$, $n > 3(\omega-1)$ and $d \geq 2$. Let (S, T) be any monitor placement such that $|S| = |T| = d(\omega - 1)$ and $S \cap T = \emptyset$. Then $d(\omega - 1) - 1 \leq \mu(\mathcal{H}) \leq d(\omega - 1)$.*

Notice that this strengthen the result on hypergrids obtained in [24].

We start by proving preliminary technical lemmas. For U a set of nodes in $\mathcal{H}$, we let $\mathcal{H} - U$ to be the subgraph of $\mathcal{H}$ obtained by removing the nodes in U.

Lemma 4.3.2. *Let n, d, ω be integers, $d \geq 2$ and $n \geq \omega \geq 3$. Let A be a set of vertices in $\mathcal{H}$. Then:*

1. *$\kappa(\mathcal{H}) = \delta(\mathcal{H})$;*
2. *If $|A| < \delta(\mathcal{H})$, then $\mathcal{H} - A$ is connected;*
3. *if $|A| = \delta(\mathcal{H})$, then $\mathcal{H} - A$ is disconnected only if $A = N(c)$ for some corner node c in $\mathcal{H}$;*
4. *if $A = N(c)$ for some corner node c, then $\mathcal{H} - A$ is made by two connected components: $\{c\}$ and $\mathcal{H} - (A \cup \{c\})$.*

Proof. First notice that (2) is an obvious consequence of (1).

We start with (1). Let $P_{n,\omega}$ be the *generalized path* on n vertices, obtained from a simple path on the given vertices by connecting any pair of vertices at distance less than ω.

Claim 4.3.3. $\kappa(P_{n,\omega}) = \omega - 1$.

To see this, observe that a potential minimal vertex separator in $P_{n,\omega}$ is a set of at most $\omega - 1$ vertices of $P_{n,\omega}$, not including the two end-points of the path, which we call u and v. Clearly $N(u)$ and $N(v)$ are two separators of size $\omega - 1$. We claim that a set of vertices in $P_{n,\omega}$ is a minimal separator if and only if it is formed by $\omega - 1$ consecutive vertices of $P_{n,\omega}$, excluding u and v. It is not difficult to see that the removal of $\omega - 1$ consecutive vertices excluding u and v disconnects $P_{n,\omega}$. A smaller set, or a set of $\omega - 1$ non-consecutive vertices, will not cover sufficiently long stretches of $P_{n,\omega}$ to disconnect the path.

Next observe that obviously,

$$\mathcal{H} = \underbrace{P_{n,\omega} \times \ldots \times P_{n,\omega}}_{d \text{ times}} \tag{4.1}$$

(where $\times$ here denotes the cartesian graph product). Equality (1) now fol-

-lows from Claim 4.3.3 and the following formula [35]:

$$\kappa(G \times H) = \min\{\kappa(G) \cdot |H|, \kappa(H) \cdot |G|, \delta(G \times H)\}$$

expressing the vertex connectivity of the cartesian product of two graphs G and H in terms of the properties of its contributing graphs.

Coming to (3), we will prove that statement in its contrapositive form: if there is no corner vertex c such that $A = N(c)$, then $\mathcal{H} - A$ is connected. Denote by $B\mathcal{H}$ the set of vertices on the border of $\mathcal{H}$. We first observe some simple facts from definitions. Let $\mathcal{H}'$ be the smallest $[n']^d$ subgrid of $\mathcal{H}$ such that A is contained among vertices of $\mathcal{H}'$. Hence $A \cap B\mathcal{H}' \neq \emptyset$, but $A \cap (\mathcal{H} - \mathcal{H}')$ is empty. Let $\mathcal{H}'' = \mathcal{H}' - B\mathcal{H}'$ be the $[n' - 2]^d$ subgrid of $\mathcal{H}'$ obtained by removing the border nodes from $\mathcal{H}'$.

We first prove the claim in the simpler case when $n' = n$ (hence $\mathcal{H}' = \mathcal{H}$). Then we extend the argument to any $n' < n$, when $\mathcal{H}'$ is a strict subgraph of $\mathcal{H}$. Observe that $\mathcal{H}'' - A$ is connected: the reason is that, since at least a node in A is in $B\mathcal{H}$, then $|A \cap \mathcal{H}''| < \delta(\mathcal{H}'') = \delta(\mathcal{H})$. Hence by case (2), $\mathcal{H}'' - A$ is connected. It remains to prove that adding nodes in $B\mathcal{H} - A$ to $\mathcal{H}'' - A$ the connectivity is preserved. Let x be any node in $B\mathcal{H} - A$ not in the corner. x has at least $\delta(\mathcal{H}) + 1 = d(w - 1) + 1 = |A| + 1$ neighbours. Hence x is connected to either $B\mathcal{H} - A$ or $\mathcal{H}'' - A$, which proves our claim. If x is a corner node in $\mathcal{H}$, then it is necessarily connected to one of its neighbours on $B\mathcal{H}$, since $A \not\subseteq N(c)$. In conclusion the graph $(\mathcal{H}'' - A) \cup (B\mathcal{H} - A)$ is connected.

When $n' < n$, $\mathcal{H}'$ is a strict subgraph of $\mathcal{H}$. As for the case $n' = n$, we have that $\mathcal{H}'' - A$ is connected. But we also know that $\mathcal{H} - \mathcal{H}'$ is A-free and connected. We want to prove that the graph $(\mathcal{H} - \mathcal{H}') \cup (B\mathcal{H}' - A) \cup (\mathcal{H}'' - A)$ is connected. Let x be any node in $(B\mathcal{H}' - A) \cup (\mathcal{H}'' - A)$. x is not a corner node in $\mathcal{H}$, hence its degree is $\geq \delta(\mathcal{H}) + 1 = |A| + 1$. Then there is at least a node y in $(\mathcal{H} - \mathcal{H}') \cup (B\mathcal{H}' - A) \cup (\mathcal{H}'' - A)$ such that x is connected to and the claim is proved.

Finally to prove (4), clearly removing $N(c)$ disconnects the graph. Obviously c becomes isolated. If $\mathcal{H} - (A \cup \{c\})$ had more than one component, each of those would have to be connected only to some of the neighbours of c but this would imply the existence of a smaller separator. □

Lemma 4.3.4. *Let n, d, ω be integers, $d \geq 2$, $n \geq 4$ and $\omega \geq 2$. Let A be a set of nodes in $\mathcal{H}_{n,d,\omega}$ such that $|A| < \delta(\mathcal{H})$ and for each corner node c in $\mathcal{H}$, it holds that $A \not\subseteq N(c)$. Then $\kappa(\mathcal{H} - A) \geq 2$.*

Proof. Notice that $\kappa(\mathcal{H} - A) > 0$, since $|A| < \delta(\mathcal{H})$ and by Lemma 4.3.2 case (2), $\mathcal{H} - A$ is connected. Assume by contradiction that $\kappa(\mathcal{H} - A) < 2$, then it must be that $\kappa(\mathcal{H} - A) = 1$. That means that there is a node z in $\mathcal{H}$ such that $\mathcal{H} - (A \cup \{z\})$ is disconnected. This is not possible because $|A \cup \{z\}| = \delta(\mathcal{H})$ but A is not included in any corner node of $\mathcal{H}$, and according to Lemma 4.3.2 case (3), that set does not disconnect $\mathcal{H}$. □

Proof of Theorem 4.3.1. As already mentioned the upper bound follows by the result in [24] that maximal identifiability is upper bounded by the minimal degree. $\delta(\mathcal{H}) = d(\omega - 1)$ and the result is proved.

Let S and T be disjoint set of $d(\omega - 1)$ nodes in $\mathcal{H}$ linked to input and output monitors. Let $\mathbb{P}$ be the set of all paths in $\mathcal{H}$ from a node in S to a node in T. We have to prove that for all U,W with $U \neq W$ and $|U|, |W| \leq d(\omega - 1) - 1$, $\mathbb{P}(U) \neq \mathbb{P}(W)$.

Assume wlog that $u \in U \setminus W$ and let $s \in S \setminus W$ and $t \in T \setminus W$ (since $|W| < |S| = |T|$, such s and t exist). If for each corner node c, $W \not\subseteq N(c)$, then by Lemma 4.3.4 $\kappa(\mathcal{H} - W) \geq 2$. Then by Lemma 3.5.4, there exist two vertex-disjoint paths between s and u (and respectively between u and t) in $\mathcal{H} - W$. These are then W-free paths in $\mathcal{H}$. Hence by Lemma 3.5.5 there is a simple path in $\mathcal{H}$ from s to t touching u and avoiding W, and the claim is proved.

Assume now that there exists a corner node c in $\mathcal{H}$ such that $W \subseteq N(c)$. Notice that if $|W| = d(\omega - 1) - 1$, then $\delta(\mathcal{H} - W) = 1$ because c has degree 1 in $\mathcal{H} - W$. We claim that:

Claim 4.3.5. $\kappa(\mathcal{H} - (W \cup \{c\})) \geq 2$.

Proof. (of the Claim) Notice that $\kappa(\mathcal{H} - (W \cup \{c\})) > 0$ because $|W \cup \{c\}| \leq \delta(\mathcal{H})$ and by Lemma 4.3.2 case (3) to disconnect $\mathcal{H}$, $W \cup \{c\}$ should be equal to the set of neighbours of a corner node, which is not the case since c is a corner itself. Assume by contradiction that $\kappa(\mathcal{H} - (W \cup \{c\})) = 1$. Then there should be a node z in $\mathcal{H} - (W \cup \{c\})$ such that $\mathcal{H} - (W \cup \{c\} \cup \{z\})$ is disconnected. If $z = N(c) \setminus W$, then we have the two following cases: (I) If $|W| = d(\omega - 1) - 1$, then z is the last remaining neighbour of c outside W. Thus by Lemma 4.3.2 case (4) $\mathcal{H} - (W \cup \{c\} \cup \{z\})$ is still connected because $\mathcal{H} - (W \cup \{z\})$ is disconnected into two connected components: $\{c\}$ and $\mathcal{H} - (W \cup \{c\} \cup \{z\})$; (II) If $|W| < d(\omega - 1) - 1$, then there is still another node in $\mathcal{H}$ that is connected to the corner node c. Therefore by Lemma 4.3.2 case (3) to disconnect $\mathcal{H}$, $W \cup \{c\} \cup \{z\}$ should be equal to the set of neighbours of a corner node, which is not the case. If $z \notin N(c)$,

then by Lemma 4.3.2 case (3), $\mathcal{H} - (W \cup \{z\})$ is connected and since the degree of c in $\mathcal{H} - (W \cup \{z\})$ is ≥ 1, then $H - (W \cup \{c\} \cup \{z\})$ cannot be disconnected. □

Now we distinguish the following cases:

1. $c \neq s, c \neq t, c \neq u$. Then s, t, u are nodes in $\mathcal{H} - (W \cup \{c\})$ and the theorem follows by the same argument as above since by the previous claim $\kappa(\mathcal{H} - (W \cup \{c\}) \geq 2$.

2. $c = s$. Then there is at least a node $z_1 \in N(c) \setminus W$. If $z_1 \neq u$ and $z_1 \neq t$, then by the same argument as above we can find a simple path p in $\mathcal{H} - (W \cup \{c\})$ from z_1 to t passing from u. And finally the theorem follows considering the path $(s, z_1)p$. If $z_1 = u$, we know (by the previous claim) that $\mathcal{H} - (W \cup \{c\})$ is connected and hence there is a simple path p from u to t in $\mathcal{H} - (W \cup \{c\})$. Then $(s, u)p$ proves the theorem. If $z_1 = t$ and $|N(c) \setminus W| > 1$, then there exits another node $z_2 \in N(c) \setminus W$, such that $z_2 \neq z_1$. We can reason as the last case with z_2 in the place of z_1. If $z_1 = t$ and $|N(c) \setminus W| = 1$, then that means $|W| = \delta(\mathcal{H}) - 1$. Therefore there is a node $w' \in W \setminus U$. Moreover $\mathcal{H} - U$ is connected by Lemma 4.3.2 case (2) since $|U| < \delta(\mathcal{H})$. Hence there is a simple path q in $\mathcal{H} - U$ from w' to t. Therefore the path $(s, w')q$ in $\mathcal{H}$ is connecting s and t, touching W and avoiding U, thus it proves the theorem.

3. $c = t$. It is the same as the case $c = s$.

4. $c = u$. If $|W| \leq \delta(\mathcal{H}) - 2$, then we claim that $\kappa(\mathcal{H} - W) \geq 2$. From this fact and by Lemma 3.5.4 and Lemma 3.5.5, we have a simple path in $\mathcal{H} - W$ from s to t touching u and this is giving the theorem. The claim $\kappa(\mathcal{H} - W) \geq 2$ follows since: (1) $\kappa(\mathcal{H} - W) > 0$ since $|W| < \delta(\mathcal{H})$ and by Lemma 4.3.2 case (2) we have that $\mathcal{H} - W$ is connected; (2) $\kappa(\mathcal{H} - W) \neq 1$ since if it is 1, then there should exist a node z in $\mathcal{H} - W$ such that $\mathcal{H} - (W \cup \{z\})$ is disconnected. This is not possible since $|W \cup \{z\}| < \delta(\mathcal{H})$ and by Lemma 4.3.2 case (2) $\mathcal{H} - (W \cup \{z\})$ is connected.

   If $|W| = \delta(\mathcal{H}) - 1$, then we can use the same argument as in the case (2), when $|W| = \delta(\mathcal{H}) - 1$ finding a path from s to t passing through W and avoiding U.

□

4.4 Random Networks

The main aim of this work is to characterize the identifiability in terms of the vertex connectivity. In this section we prove that tight results are possible in random graphs. Also we show an interesting trade-off between the success probability of the various random processes and the size of the sets S and T.

Random graph is the general term to refer to probability distributions over graphs. Random graphs may be described simply by a probability distribution, or by a random process which generates them. Random graphs give us constructions of networks with large identifiability. Its practical applications are found in all areas in which complex networks need to be modeled. Many random graph models are thus known, mirroring the diverse types of complex networks encountered in different areas. A random graph is obtained by starting with a set of n isolated vertices and adding successive edges between them at random. Here let us become familiar with the random graphs which we work on in this section.

Erdős-Rényi graphs $G(n,p)$

Definition 4.4.1 (Erdős-Rényi graphs $G(n,p)$)**.** *In the Erdős-Rényi $G(n,p)$ model, a graph is constructed by connecting labeled nodes randomly. Each edge is included in the graph with probability p, independently from every other edge.*

Figure 4.8. A graph generated by the binomial model of Erdős-Rényi ($p = 0.01$).

Random r-regular graphs

Definition 4.4.2 (Random r-regular graphs)**.** *A random r-regular graph is a graph selected from $\mathcal{G}_{n,r}$, which denotes the probability space of all r-regular graphs on n vertices, where $3 \leq r < n$ and nr is even.*

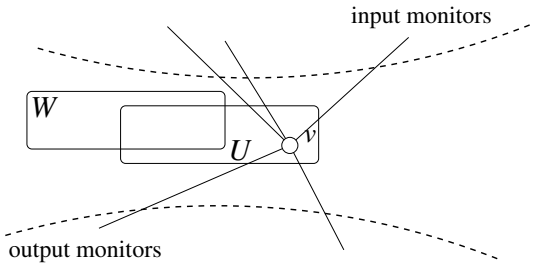


Figure 4.9. A node $v \in U \Delta W$ and a possible way to connect it to S and T.

Sub-Linear Separability in Erdős-Rényi Graphs We start our investigation of the identifiability of node failures in random graphs by looking at the binomial model $G(n,p)$, for fixed $p \leq 1/2$ (in this section we only follow the traditional random graph jargon and use p to denote the graph edge probability rather than a generic path). The following equalities, which hold with probability approaching one as n tends to infinity (that is *with high probability* (w.h.p.)), are folklore:

$$\kappa(G(n,p)) = \delta(G(n,p)) = np - o(n), \tag{4.2}$$

(see [8]). Here we describe a simple method which can be used to separate sets of vertices of sub-linear size.

We assume, for now, that S and T are each formed by $\gamma = \gamma(n)$ nodes with $\kappa(G(n,p)) \leq \gamma < n/2$. Let $M = S \cup T$.

Let $\mathbb{P}$ be the set of all distinct paths in $G(n,p)$. Let U and W be two arbitrary subsets of $V \setminus M$ of size k. We want to estimate the probability that U and W are separated by a path in G, i.e.

$$\Pr[\exists p \in \mathbb{P}(U) \triangle \mathbb{P}(W)]. \tag{4.3}$$

We focus on specific simple paths separating U from W, that is paths made by only two edges $(s,v)(v,t)$ where $v \in U \Delta W$. The probability in 4.3 is then at least the probability that there is such path, i.e. there is an element v of $U \Delta W$ (w.l.o.g. assume $v \in U \setminus W$) which is directly connected to a node in S and to a node in T.

The probability that a node u is an endpoint of an edge in a set of size γ is $(1-(1-p)^{\gamma})$. Hence the probability that u is directly linked to both S and T is $(1-(1-p)^{\gamma})^2$. Hence:

$$\Pr[\exists p \in \mathbb{P}(U) \triangle \mathbb{P}(W)] \geq (1-(1-p)^{\gamma})^2. \tag{4.4}$$

On the other hand the probability that U and W cannot be separated by any path, i.e. $\mathbb{P}(U) = \mathbb{P}(W)$ is hence:

$$\Pr[\mathbb{P}(U) = \mathbb{P}(W)] \leq 1 - (1 - (1-p)^{\gamma})^2 = 2(1-p)^{\gamma} - (1-p)^{2\gamma}$$

and therefore the probability that some pairs of sets U and W of size k (not intersecting M) fail is at most $2\binom{n-2\gamma}{k}\binom{2k}{k}(1-p)^{\gamma}$.

Theorem 4.4.3. *For fixed p with $p \leq 1/2$, under the assumptions above about the way monitors are placed in $G(n,p)$, the probability that $G(n,p)$ is not k-vertex separable is at most $2k\binom{n}{k}^2 \mathrm{e}^{(2k-\gamma)p}$.*

Proof. The argument above works if both U and W contain no vertex in M. The presence of vertices of M in U or W may affect the analysis in two ways. First v could be in M (say $v \in S$). In this case U and W are separable if v is directly connected to a vertex in T. This happens with probability $(1 - (1-p)^{\gamma}) > (1 - (1-p)^{\gamma})^2$. Second, M might contain some elements of U and W different from v. In the worst case when v is trying to connect to M, it must avoid at most $2k$ elements of such set. There are at most $\sum_{h \leq k} \binom{n}{h}^2 \leq k\binom{n}{k}^2$ pairs of U and W of size at most k. Thus the probability that $G(n,p)$ fails to be k-vertex separable is at most $2k\binom{n}{k}^2(1-p)^{\gamma-2k}$. and the result follows as $1 - p \leq \mathrm{e}^{-p}$. □

Note that the bound in Theorem 4.4.3 can only be small if $k = o(n)$ for otherwise the factor $\mathrm{e}^{(2k-\gamma)p}$ is large. In fact it has to be $k = O(n^{\epsilon})$ for sufficiently small positive ϵ otherwise the large factor $\binom{n}{k}^2$ is not "killed off" by the magnitude of the small exponential.

Linear Separability in Erdős-Rényi Graphs The argument above cannot be pushed all the way up to $\kappa(G(n,p))$. When trying to separate vertex sets containing $\Omega(n)$ vertices, the problem is that these sets can form a large part of M and the existence of direct links from v to $S \setminus W$ and $T \setminus W$ is not guaranteed with sufficiently high probability.

However a different argument allows us to prove the following:

Theorem 4.4.4. *For any fixed $p \leq 1/4$, there is a $\beta > 0$ such that for any monitor placement (S,T) with $S \cap T = \emptyset$, $\mu(G(n,p)) \geq \beta n$ w.h.p.*

Proof. We claim that the chance that two sets of size at most βn are not vertex separable is small. To believe this, pick two sets U and W, assuming without loss of generality that $U \setminus W \neq \emptyset$, and remove W from the graph. $G(n,p) - W$ is still a random graph on at least $n - np = \Omega(n)$ vertices and constant edge probability. In the remainder of the proof, we prove that the main result in [40] is still valid for $G(n,p) - W$. Hence, in particular $G(n,p) - W$ has a Hamiltonian cycle with sufficiently high probability and therefore there is a good chance that $G(n,p) - W$ is Hamiltonian for any choice of W (recall that a Hamiltonian path/cycle is a path/cycle that visits each vertex of the graph exactly once and a graph that contains a Hamiltonian cycle is called a Hamiltonian graph). Any Hamiltonian cycle in $G(n,p) - W$, by definition, contains a path from S to T passing through $v \notin W$, for every possible choice of v. This proves, w.h.p., the separability of sets of size up to βn.

Hamiltonian path/cycle

Hamiltonian graph

Assume $p \leq 1/4$. In what follows we claim that there is an $\epsilon > 0$ such that

$$\Pr[G(n,p) \text{ not Hamiltonian}] \leq en^5(1-\epsilon p)^{n/5} + n(1-\epsilon p)^{n/4} + \left(\frac{1-p}{1-\epsilon p}\right)^{n/4}. \tag{4.5}$$

Also, using (4.5), we then prove that, in fact, for each $p < 1/4$ there exists a value β such that if $|W| \leq \beta n$ the probability that $G(n,p) - W$ is not Hamiltonian is bounded as above by function of the form $f(\epsilon, \beta, p)^{n+o(n)}$ with $f(\epsilon, \beta, p) < 1$. The proof of the theorem follows.

Proof of (4.5): We obtain the bound above by just working through the proof of the main result in [40] and quantifying the error probability assuming p constant. Pósa uses a two-stage exposure defining $G(n,p)$ as the union of $G(n,p_1)$ and $G(n,p_2)$ using $p_1 = \epsilon p$ and $p_2 = (1-\epsilon)p/(1-\epsilon p)$, solution of the equation: $p = p_1 + p_2 - p_1 p_2$. The two stage exposure allows him to first work with enough edges to claim the existence of a Hamiltonian path, and then showing that few more edges are likely to close at least one of those paths into a fully fledged Hamiltonian cycle. following Pósa's calculations, we argue that:

$$\Pr[G(n,p_1) \text{ has NO Hamiltonian path}] \leq en^5(1-\epsilon p)^{n/5} + n(1-\epsilon p)^{n/4}$$

and furthermore the probability that assuming $G(n,p_1)$ does have a Hamiltonian path, the edges of $G(n,p_2)$ do not close any of those, is at most

$$\left(\frac{1-p}{1-\epsilon p}\right)^{n/4}.$$

These two together give (4.5).

Now we move to the second part of the argument which focuses on $G(n,p) - W$. Assume ϵ is large enough (say $\epsilon > 1/2$). Note that:

- $1 - p < 1 - \epsilon p$, so all exponentials in (4.5) tend to zero.
- The sum of the first two terms is at most

$$(\mathrm{e}n^5 + n)(1-\epsilon p)^{n/5}.$$

- The argument of the right-most exponential is smaller than $(1-\epsilon p)^{1/5}$ if

$$(1-p)^{1/4} < (1-\epsilon p)^{9/20},$$

  which can be verified resorting to the Taylor expansion of the two sides.

So, given our constraints on p and ϵ, the upper bound on the probability that for any W of size at most βn, $G(n,p) - W$ is NOT Hamiltonian is at most

$$(\mathrm{e}n^5 + n + 1)\sum_{k=1}^{\beta n}\binom{n}{k}(1-\epsilon p)^{(n-k)/5}.$$

This is at most

$$(\mathrm{e}n^5 + n + 1)\left(\frac{\mathrm{e}}{\beta}\right)^{\beta n}(1-\epsilon p)^{n/5}\sum_{k=1}^{\beta n}\left[\frac{1}{(1-\epsilon p)^{1/5}}\right]^k.$$

The geometric sum satisfies:

$$\begin{aligned}
\frac{\left(\frac{1}{(1-\epsilon p)^{1/5}}\right)^{\beta n+1} - 1}{\frac{1}{(1-\epsilon p)^{1/5}} - 1} - 1 &= \frac{\left(\frac{1}{(1-\epsilon p)^{1/5}}\right)^{\beta n+1} - \frac{1}{(1-\epsilon p)^{1/5}}}{\frac{1}{(1-\epsilon p)^{1/5}} - 1} \\
&= \frac{(1-\epsilon p)^{1/5}}{1-(1-\epsilon p)^{1/5}}\left[\left(\frac{1}{(1-\epsilon p)^{1/5}}\right)^{\beta n+1} - \frac{1}{(1-\epsilon p)^{1/5}}\right] \\
&= \frac{1}{1-(1-\epsilon p)^{1/5}}\left[\left(\frac{1}{(1-\epsilon p)^{1/5}}\right)^{\beta n} - 1\right]
\end{aligned}$$

$$\leq \frac{1}{1-(1-\epsilon p)^{1/5}} \left(\frac{1}{(1-\epsilon p)^{1/5}} \right)^{\beta n}.$$

Hence, the overall error probability is at most

$$\frac{\mathrm{e}n^5 + n + 1}{1-(1-\epsilon p)^{1/5}} \left[\left(\frac{\mathrm{e}}{\beta} \right)^{\beta} (1-\epsilon p)^{(1-\beta)/5} \right]^n$$

which is exponentially small for an appropriate choice of β provided $p < 1/4$. □

Random Regular Graphs A standard way to model random graphs with fixed vertex degrees is Bollobàs' configuration model [6]. There are n buckets, each with r free points. A random pairing of these free points has a constant probability of not containing any pair containing two points from the same bucket or two pairs containing points from just two buckets. These configurations are in one-to-one correspondence with r-regular n-vertex simple graphs. Denote by $\mathcal{C}_{n,r}$ the set of all configurations $C(n,r)$ on n buckets each containing r points, and let $G(r\text{-reg})$ be a random r-regular graph.

As before assume $|S| = |T| = \gamma$ with $S \cap T = \emptyset$. The main result of this section is the following:

Theorem 4.4.5. *Let $r \geq 3$ be a fixed integer. $r-2-o(1) \leq \mu(G(r\text{-reg})) \leq r$ w.h.p.*

The result resembles Theorem 1.2.3 but its proof uses different techniques. The upper bound is true for any r-regular graph G as $\mu(G) \leq \delta(G) = r$. The lower bound is a consequence of the following:

Lemma 4.4.6. *Let $r \geq 3$ be a fixed integer. Two sets U and W with $U, W \subseteq V(G(r\text{-reg}))$ and $\max(|U|, |W|) \leq k$ are separable w.h.p. if $k = r - 2 - o(1)$.*

Proof. In what follows we often use graph-theoretic terms, but we actually work with a random configuration $C(n,r)$. Let U and W be two sets of k buckets. For simplicity assume that (the vertices corresponding to the elements of) both U and W are subsets of $V \setminus M$. The probability that U and W can be separated is at least the probability that a (say) random

element v of $U \triangle W$ (w.l.o.g. $v \in U \setminus W$) is connected to S by a path of length at most ℓ_s and to T by a path of length at most ℓ_t, neither of which "touches" W. Figure 4.10 provides a simple example of the event under consideration. The desired paths can be found using algorithm PathFinder below that builds the paths and $C(n, r)$ at the same time.

PathFinder(v, ℓ_s, ℓ_t, W)

SimplePaths(v, ℓ_s, ℓ_t, W)**.** Starting from v, build a simple path p^s of length ℓ_s that avoids W. Similarly, starting from v, build a simple path p^t of length ℓ_t that avoids W.

RandomShooting(p^s, p^t)**.** Pair up all un-matched points in p^s and p^t.

Complete the configuration $C(n, r)$ by pairing up all remaining points.

Sub-algorithm SimplePaths can complete its constructions by pairing points starting from elements of the bucket v, then choosing a random un-matched point in a bucket u, then picking any other point u and then again a random un-matched point and so on, essentially simulating two random walks RW_s and RW_t on the set of buckets. Note that the process may fail if at any point we re-visit a previously visited bucket or if we hit W or even M. However the following can be proved easily.

Claim 4.4.7. RW_s *and* RW_t *succeed w.h.p. provided* $\ell_s, \ell_t \in o(n)$.

As to RandomShooting, the process succeeds if we manage to hit an element of S from p^s and an element of T from p^t.

Claim 4.4.8. RandomShooting(q_s, q_t, S, T) *succeeds w.h.p. if* $\ell_s, \ell_t \in \omega(1)$.

Any un-matched point in p^s or p^t after SimplePaths is completed is called *useful*. Path p^s (resp. p^t) contains $q_s = (r-2)\ell_s + 1$ (resp $q_t = (r-2)\ell_t + 1$) useful points. During the execution of RandomShooting a single useful point "hits" its target set, say S, with probability proportional to the cardinality of S. Hence the probability that none of the q_s useful points hits S is $(1-\frac{\gamma}{n})^{q_s}$ and the overall success probability is $(1-(1-\frac{\gamma}{n})^{q_s})(1-(1-\frac{\gamma}{n})^{q_t})$.

Back to the proof of Lemma 4.4.6, set $\ell_s = \ell_t = \ell$ and q the common value of q_s and q_t. The argument above implies that the success probability

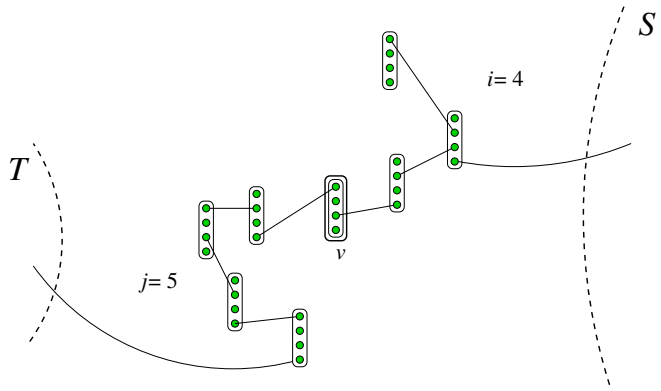


Figure 4.10. Assume $r = 4$. The picture represents a bucket (i.e. vertex) $v \in U \triangle W$ and two possible "paths" (sequences of independent edges such that consecutive elements involve points from the same bucket) of length 3 and 5, respectively connecting it to S and T.

for U and W is asymptotically approximately $(1 - (1 - \frac{\gamma}{n})^q)^2$ and the rest of the argument (and its conclusion) is very similar to the $G(n, p)$ case (the final bound is slightly weaker, though). The chance that a random r-regular graph is not k-vertex separable is at most

$$O(n^{2k}) \times (1 - (1 - (1 - \frac{\gamma}{n})^q)^2) \leq O(n^{2k}) \times 2(1 - \frac{\gamma}{n})^q \leq O(n^{2k}) \times 2\mathrm{e}^{-\frac{\gamma}{n}q},$$

which goes to zero as n^{-C} provided ℓ is at least logarithmic in n. The constraints on ℓ from the claims above imply that the parameter can be traded-off against γ to achieve high identifiability. □

Chapter 5

Counting and Localizing Failure Nodes in Networks

As observed in [37, 38] k-identifiability can be *scaled* to each single node yet preserving the property for the whole set of paths. A node u is k-ID if any two sets of size at most k differing on u are separated by at least a path in $\mathbb{P}$. Hence understanding the combinatorics of the set $\mathsf{ID}_k(\mathbb{P})$ of the k-identifiable nodes in $\mathbb{P}$ and study upper and lower bounds for $|\mathsf{ID}_k(\mathbb{P})|$ is of great importance to develop algorithms to maximize the identification of failure nodes in real networks.

In [38] they started this study quantifying the capability of failure localization through (1) the maximum number of failures such that failures within a given node set can be localized unambiguously, and (2) the largest node set, failures can be uniquely localized under a given bound on the total number of failures. These measures were used to evaluate the impact of maximum identifiability on various parameters of the network (underlying the set of paths) like the topology, the number of monitor and the probing mechanisms. They presented a set of sufficient and necessary conditions (testable in polynomial time) for identifying a bounded number of failures within an arbitrary node set.

Moreover, the works [4, 37] considered the problem of optimizing the capability of identifying network failures through different monitoring schemes and giving upper bounds on the maximum number of identifiable nodes, given the number of monitoring paths, the routing scheme and the maximum path length. In particular in [4] they studied upper bounds on the set of $|\mathsf{ID}_1(\mathbb{P})|$ of the 1-identifiable nodes in $\mathbb{P}$.

In this chapter we study upper and lower bounds on the number of

unambiguously identifiable nodes for any k, introducing new identifiability conditions which strictly imply and are strictly implied by unambiguous identifiability. We use these new conditions on one side to design new algorithms and heuristics to count or localize as more precisely as possible failure nodes in networks, on the other side to prove the first complexity hardness results on the problem of identifying defective nodes in networks via BNT.

More precisely, we first give some preliminary definitions on Boolean Network Tomography and identifiability, showing the connection with unambiguous identification of failure nodes in *Section 5.1*.

In *Section 5.2*, we study upper bounds on the number of paths such that $\mathbb{P}$ is no longer k-identifiable (Corollary 5.2.8) using union-free families notion. Moreover this result will give us an estimate of upper bounds on the number of k-identifiable nodes in $\mathbb{P}$ (Theorem 5.2.9).

Section 5.3 includes the definitions of k-separability, k-distinguishability and the relation of these new definitions with identifiability (Lemmas 5.3.2 and 5.3.4).

In *Section 5.4* we introduce a random model to study lower bounds on the number of unambiguously identifiable defective nodes (Theorem 5.4.2) and then use this model to estimate this number of k-identifiable nodes on real networks by a maximum likelihood estimate approach.

Finally in *Section 5.5* we present a method based on distinguishability to compute upper bounds on identifiable nodes in a fine-grained way (Theorem 5.5.4).

5.1 Preliminary Definitions

Let $n, k \in \mathbb{N}$ and $k \leq n$. $\binom{[n]}{k}$ is the set of subsets of $[n]$ of size k. $\binom{[n]}{\leq k}$ is the set of subsets of $[n]$ of size at most k. 2^A is the set of subsets of the set A. $A \oplus B$ is the symmetric difference between A and B. $\overline{A}$ denotes the complement of A.

Let n and m be positive integers and $V = [n]$ be the set of nodes. We encode a *set of m paths over nodes in V* as a collection $\mathbb{P}$ of n *distinct* m-bit vectors such that $\mathbb{0} \notin \mathbb{P}$, i.e. the m-bit zero vector is not in $\mathbb{P}$ (this condition means that each node in V is used in at least a path).

We can view $\mathbb{P}$ in three different ways: as a Boolean $m \times n$-matrix, as a collection of n m-bit vectors and as a collection of m n-bit vectors. For a node $u \in V$, $\mathbb{c}_u$ is then the m-bit vector whose p-th coordinate indicates whether the node u is in the p-th path or not. $\mathbb{c}_u$

We use also $\mathbb{P}$ in a graph notation as follows: if $u \in V$ is a node, then $\mathbb{P}(u)$ identifies the set of all paths touching u, in other words the set $\{p \in \mathbb{P} = [m] : \mathbb{P}[p, u] = 1\}$. If $U \subseteq V$ is a set of nodes, $\mathbb{P}(U)$ denotes the set of paths in $\mathbb{P} = [m]$ touching at least a node in U, i.e. $\mathbb{P}(U) = \bigcup_{u \in U} \mathbb{P}(u)$. $\mathbb{P}(u)$ $\mathbb{P}(U)$

Now let $\mathbb{P}$ be a set of m paths over n nodes. Notice that in terms of the column-vector notation, the definition of k-identifiability says that for all distinct sets $U, W \subseteq V$ of size at most k,

$$\bigvee_{u \in U} \mathbb{c}_u \oplus \bigvee_{w \in W} \mathbb{c}_w \neq \mathbb{0}.$$

The definition of k-identifiability can be equivalently given for nodes $u \in V$ as follows (see also [37]).

Definition 5.1.1 (k-identifiable nodes)**.** *A node $u \in V$ is k-identifiable with respect to $\mathbb{P}$, if for all $U, W \subseteq V$ of size at most k and such that $U \cap \{u\} \neq W \cap \{u\}$, it holds that $\mathbb{P}(U) \neq \mathbb{P}(W)$.* *$k$-identifiable nodes*

$\mathsf{ID}_k(\mathbb{P})$ denotes the set of k-identifiable nodes in $\mathbb{P}$. As we already know k-identifiability implies k'-identifiability for $k' < k$. Hence, $\mathsf{ID}_k(\mathbb{P})$

Lemma 5.1.2. *Let $\mathbb{P}$ be a set of m paths over n nodes. Then $\mathsf{ID}_k(\mathbb{P}) \subseteq \mathsf{ID}_{k'}(\mathbb{P})$ for $k' \leq k \leq n$.*

Furthermore scaling to identifiability of nodes does not affect the main property of k-identifiability (which we see below). Next theorem is proved in [37] (Theorem 4).

Theorem 5.1.3. *([37]) Let $\mathbb{P}$ be a set of m paths over n nodes. $\mathbb{P}$ is k-identifiable if and only if every node in V is k-identifiable with respect to $\mathbb{P}$.*

5.2 Union-Free Families and Upper Bounds on $\mu(\mathbb{P})$

In this section we show that under what bounds on the number of paths m in $\mathbb{P}$, we have that $\mu(\mathbb{P}) < k$. We start by showing under what conditions

on m, $\mu(\mathbb{P}) < 1$.

Notice that to prove that $\mu(\mathbb{P}) < 1$, by Definition 2.4.1 it is sufficient to find two distinct nodes $u, w \in V$ such that $\mathbb{c}_u \oplus \mathbb{c}_w = \mathbb{0}$, that is for all $p \in \mathbb{P} = [m] : \mathbb{c}_u[p] = \mathbb{c}_w[p]$. $\mu(\mathbb{P}) < 1$ will follow from an easy information, theoretic bound on sets of m-vectors.

Lemma 5.2.1. *Let $\mathbb{P}$ be a set of m paths built on n nodes. If $m < \log_2(n+1)$, then $\mu(\mathbb{P}) < 1$.*

Proof. $\mathbb{P}$ is a collection of n m-bit strings. There are at most $2^m - 1$ different such strings ($\mathbb{0} \notin \mathbb{P}$). Hence whenever $n > 2^m - 1$ there are two elements $u \neq w \in V$ such that $\mathbb{c}_u = \mathbb{c}_w$, which means $\mathbb{c}_u \oplus \mathbb{c}_w = \mathbb{0}$. □

Corollary IV.1 in [4] can be obtained by previous observation immediately.

Theorem 5.2.2. *([4]) Let $\mathbb{P}$ be a set of m paths over n nodes. Then $|\mathsf{ID}_1(\mathbb{P})| \leq \min\{n, 2^m - 1\}$.*

Proof. $|\mathsf{ID}_1(\mathbb{P})| \leq n$ since it is a set of nodes. Assume that $n > 2^m - 1$, hence by previous Lemma 5.2.1 $\mu(\mathbb{P}) = 0$, hence there are at least two nodes $u_1 \neq u_2$ not 1-identifiable. Hence $|\mathsf{ID}_1(\mathbb{P})| \leq 2^m - 1$. □

We will prove similar results for $\mu(\mathbb{P}) < k$ for a generic $k \leq n$. In order to obtain the desired results we first need to become familiar with union-free families.

Hypergraph and r-regular

A hypergraph $\mathcal{F}$ on the set $[m]$ is a family of distinct subsets of $[m]$, called edges of $\mathcal{F}$. If each edge is of fixed size $r \leq m$, then $\mathcal{F}$ is said to be *r-regular*, i.e., $\mathcal{F} \subset \binom{[m]}{r}$.

Union-free families

Definition 5.2.3 (Union-free families)**.** *For a positive integer k, $\mathcal{F}$ is called* k-union-free *if for any two distinct subsets of edges $\mathcal{A}, \mathcal{B} \subseteq \mathcal{F}$, with $1 \leq |\mathcal{A}|, |\mathcal{B}| \leq k$, it holds that $\cup_{A \in \mathcal{A}} A \neq \cup_{B \in \mathcal{B}} B$.*

Union-free regular hypergraphs are investigated in extremal combinatorics [22]. It is immediate to see that a set $\mathbb{P}$ of m paths over n nodes defines a hypergraph $\mathcal{F}_{\mathbb{P}}$ on the set $[m]$ in the following way: for $i \in V$ let $A_i = \{j \in [m] | \mathbb{c}_i[j] = 1\}$ and define $\mathcal{F}_{\mathbb{P}} = \{A_1, \ldots, A_n\}$. Given a $U \subseteq V$, consider the subset of $\mathcal{F}_{\mathbb{P}}$, $\mathcal{U} = \{A_i \in \mathcal{F}_{\mathbb{P}} | i \in U\}$. Observe that then $\mathbb{P}(U) = \bigcup_{A \in \mathcal{U}} A$. Hence immediately by definition of k-identifiability and that of k-union-freeness it follows that:

Lemma 5.2.4. *If $\mathbb{P}$ is a set of m paths over n nodes and $\mu(\mathbb{P}) \geq k$, then $\mathcal{F}_{\mathbb{P}}$ is k-union free.*

$\mathcal{F}_{\mathbb{P}}$ is not necessary a regular hypergraph. For $r \in [m]$ let $\mathcal{F}_{\mathbb{P}}(r) = \{A \in \mathcal{F}_{\mathbb{P}} || A| = r\}$. Notice that each $\mathcal{F}_{\mathbb{P}}(r)$ is now an r-regular hypergraph on $[m]$. Moreover the family of the $\mathcal{F}_{\mathbb{P}}(r)$'s partitions $\mathcal{F}_{\mathbb{P}}$ and hence $|\mathcal{F}_{\mathbb{P}}| = \sum_{r\in[m]} |\mathcal{F}_{\mathbb{P}}(r)|$. Since $|\mathcal{F}_{\mathbb{P}}| = n$, it follows that:

Lemma 5.2.5. $\sum_{r\in[m]} |\mathcal{F}_{\mathbb{P}}(r)| = n$.

Furthermore notice that if $\mathcal{F}_{\mathbb{P}}$ is k-union free then so will be $\mathcal{F}_{\mathbb{P}}(r)$ for each $r \in [m]$.

Let $m > r$, $k \in [m]$ with $k \geq 2$, and let $f(k, r, m)$ denote the maximum cardinality of a k-union-free r-regular hypergraph over $[m]$.

Theorem 5.2.6 ([22, 46]). $\Omega(m^{\frac{r}{k-1}}) \leq f(k, r, m) \leq O(m^{\lceil \frac{r}{k-1} \rceil})$.

Let $m_0 \in \mathbb{N}$ and C be the constant such that for all $m \geq m_0$, $f(k, r, m) \leq Cm^{\lceil \frac{r}{k-1} \rceil}$.

Theorem 5.2.7. *Let m be an integer such that $m \geq m_0$. Let $\mathbb{P}$ be a set of m paths over n nodes. If $n > \sum_{r\in[m]} Cm^{\lceil \frac{r}{k-1} \rceil}$, then $\mu(\mathbb{P}) < k$.*

Proof. Assume by contradiction that $n > \sum_{r\in[m]} Cm^{\lceil \frac{r}{k-1} \rceil}$ and $\mu(\mathbb{P}) \geq k$. By Lemma 5.2.4 $\mathcal{F}_{\mathbb{P}}$ is k-union free. Hence (see observation after Lemma 5.2.5) for each $r \in [m]$, $\mathcal{F}_{\mathbb{P}}(r)$ is an r-regular k-union free hypergraph and hence by previous theorem $|\mathcal{F}_{\mathbb{P}}(r)| \leq Cm^{\lceil \frac{r}{k-1} \rceil}$. The $\mathcal{F}_{\mathbb{P}}(r)$ partition $\mathcal{F}_{\mathbb{P}}$ and by Lemma 5.2.5 we have

$$n = \sum_{r\in[m]} |\mathcal{F}_{\mathbb{P}}(r)| \leq \sum_{r\in[m]} Cm^{\lceil \frac{r}{k-1} \rceil}$$

. □

Corollary 5.2.8. *Let $\mathbb{P}$ be a set of m paths over n nodes and $2 \leq k \leq m$. If $m < \sqrt[1+\epsilon]{\frac{(k-1)}{k}(\log_2 n - D)} - (k-1)$, for some $\epsilon > 0$ and where $D = \log C$, then $\mu(\mathbb{P}) < k$.*

Proof. Assume for the moment that m divides $k-1$. We prove that if $m < \sqrt[1+\epsilon]{\frac{(k-1)}{k}(\log_2 n - D)}$, then

$$n > Cmm^{\frac{m}{k-1}}. \tag{5.1}$$

This immediately implies

$$n > \sum_{r\in[m]} Cm^{\lceil \frac{r}{k-1} \rceil} \tag{5.2}$$

since $\sum_{r\in[m]} Cm^{\lceil \frac{r}{k-1} \rceil} \leq Cmm^{\frac{m}{k-1}}$. Equation 5.1 follows by the following implications:

$$m < \sqrt[1+\epsilon]{\frac{k-1}{k}(\log_2 n - D)} \tag{5.3}$$

$$m^{1+\epsilon} < \frac{k-1}{k}(\log_2 n - D) \tag{5.4}$$

$$\frac{k}{k-1}m^{1+\epsilon} < \log_2 n - D \tag{5.5}$$

$$\log C + \frac{k}{k-1} m \log m < \log n \tag{5.6}$$

$$\log C + \log m + \frac{m}{k-1} \log m < \log n \tag{5.7}$$

$$Cmm^{\frac{m}{k-1}} < n \tag{5.8}$$

Equation 5.7 follows from Equation 5.6 since $K = \frac{k}{k-1} > 1$ and $Km \log m > \log m + \frac{m \log m}{k-1}$ for all m.

If m does not divide $(k-1)$, let $a < (k-1)$ be the smallest non-negative integer such that $m + a$ divides $k - 1$. Hence $m + a < m + (k-1)$. Let $\hat{m} = m + a$. Since $m < \sqrt[1+\epsilon]{\frac{(k-1)}{k}(\log_2 n - D)} - (k-1)$, then $\hat{m} < \sqrt[1+\epsilon]{\frac{(k-1)}{k}(\log_2 n - D)}$. Hence the previous argument proves that $n > C\hat{m}\hat{m}^{\frac{\hat{m}}{k-1}}$. Since $m < \hat{m}$, then $n > Cmm^{\frac{m}{k-1}}$. Now by Theorem 5.2.7, this implies that $\mu(\mathbb{P}) < k$. □

Theorem 5.2.9. *Let $\mathbb{P}$ be a set of m paths over n nodes. Then for all $k \leq n$, $|\,\mathsf{ID}_k(\mathbb{P})| \leq \min\{n, 2^{\frac{k(m+k-1))^{(1+\epsilon)}}{k-1}}\}$.*

Proof. Notice that if $n \geq 2^{\frac{k(m+k-1)^{(1+\epsilon)}}{k-1}}$, then $m < \sqrt[1+\epsilon]{\frac{(k-1)}{k}(\log_2 n - D)} - (k-1)$. Hence Corollary 5.2.8 and the same proof of Theorem 5.2.2 imply the claim. □

5.3 Refining Identifiability: Separability and Distinguishability

We introduce two new definitions approximating identifiability from above and from below. Moreover we are going to use them to prove upper and lower bounds on the number of k-identifiable nodes.

k-separable nodes

Definition 5.3.1 (k-separable nodes)**.** *A node $u \in V$ is* k-separable *in $\mathbb{P}$, if for all $U \subseteq V$ of size at most k and such that $u \notin U$, it holds that there is a path $p \in \mathbb{P}(u) \setminus \mathbb{P}(U)$, i.e. there is at least a path passing though u but not touching any node of U.*

We say that $\mathbb{P}$ *is k-separable* if each node $u \in V$ is k-separable. k-separability is a stronger notion than k-identifiability as captured by the following lemma.

Lemma 5.3.2. *If u is k-separable in $\mathbb{P}$, then u is k-identifiable in $\mathbb{P}$.*

Proof. Let u be k-separable and U and W be distinct subset of V of size at most k such that $U \cap \{u\} \neq W \cap \{u\}$. Then either $u \in U \setminus W$ or $u \in W \setminus U$. Assume wlog the former. Then $u \notin W$. u is k-separable in $\mathbb{P}$, there is a path $p \in \mathbb{P}(u) \setminus \mathbb{P}(W)$. Since $u \in U$, then $p \in \mathbb{P}(U) \setminus \mathbb{P}(W)$ and then $\mathbb{P}(U) \neq \mathbb{P}(W)$. □

Notice that opposite direction is not true as we argue: assume that $\mathbb{P}$ is k-identifiable and that $u \notin W$ for W a set of at most k nodes. The k-identifiability of $\mathbb{P}$ implies that $\mathbb{P}(u) \neq \mathbb{P}(W)$, yet this condition alone does not guarantee that the path separating $\{u\}$ from W, pass through u and not touching W.

k-distinguishable nodes

Definition 5.3.3 (k-distinguishable nodes)**.** *A node $u \in V$ is* k- distinguishable *in $\mathbb{P}$, if for all $U \subseteq V$ of size at most k and such that $u \notin U$, it holds $\mathbb{P}(u) \neq \mathbb{P}(U)$.*

We say that $\mathbb{P}$ *is k-distinguishable* if each node $u \in V$ is k-distinguishable.

Lemma 5.3.4. *If u is a k-identifiable node in $\mathbb{P}$, then u is k-distinguishable in $\mathbb{P}$.*

Proof. Assume that $u \in V$ is k-ID in $\mathbb{P}$. Let $W \subseteq V$ be of size at most k such that $u \notin W$. We want to prove that $\mathbb{P}(u) \neq \mathbb{P}(W)$. By k-ID of u we know that for all U' and W' in V of size at most k such that

$U' \cap \{u\} \neq W' \cap \{u\}$, it holds that $\mathbb{P}(U') \neq \mathbb{P}(W')$. Fix $U' = \{u\}$ and $W' = W$. Since $u \notin W$, then $U' \cap \{u\} \neq W' \cap \{u\}$, hence $\mathbb{P}(u) \neq \mathbb{P}(W)$, as required. □

Notice that the opposite direction is not necessary true: indeed if $u \in U \setminus W$, knowing that $\mathbb{P}(u) \neq \mathbb{P}(W)$ it is not sufficient to conclude $\mathbb{P}(U) \neq \mathbb{P}(W)$, exactly in those case when $\mathbb{P}(u) \neq \mathbb{P}(W)$ is witnessed by a path in $\mathbb{P}(W) \setminus \mathbb{P}(u)$, which can touch other nodes in U but not u.

We denote by $\mathsf{ID}_k(\mathbb{P}), \mathsf{SEP}_k(\mathbb{P}), \mathsf{DIS}_k(\mathbb{P})$ the sets of nodes which are respectively k-identifiable, k-separable and k-distinguishable in $\mathbb{P}$ and we use to say respectively that u is k-ID, k-SEP and k-DIS in $\mathbb{P}$. $\mathsf{ID}_k(\mathbb{P}), \mathsf{SEP}_k(\mathbb{P})$ *and* $\mathsf{DIS}_k(\mathbb{P})$

By previous lemmas it holds that,

Lemma 5.3.5. *For all $k \in [n]$, $|\mathsf{SEP}_k(\mathbb{P})| \leq |\mathsf{ID}_k(\mathbb{P})| \leq |\mathsf{DIS}_k(\mathbb{P})|$.*

Furthermore since the three properties are clearly anti-monotone, it holds that:

Lemma 5.3.6. *For all $k \in [n]$, $\mathsf{ID}_k(\mathbb{P}) \subseteq \mathsf{ID}_{k-1}(\mathbb{P})$, $\mathsf{SEP}_k(\mathbb{P}) \subseteq \mathsf{SEP}_{k-1}(\mathbb{P})$, $\mathsf{DIS}_k(\mathbb{P}) \subseteq \mathsf{DIS}_{k-1}(\mathbb{P})$.*

We denote by $\sigma(\mathbb{P})$ (respectively $\delta(\mathbb{P})$) the maximal $k \leq n$ such that $\mathbb{P}$ is k-separable (respectively k-distinguishable). Hence we have $\sigma(\mathbb{P}) \leq \mu(\mathbb{P}) \leq \delta(\mathbb{P})$. $\sigma(\mathbb{P})$ *and* $\delta(\mathbb{P})$

5.4 Lower Bounds on $\mu(\mathbb{P})$ by A Random Model

To study lower bounds on $\mathsf{ID}_k(\mathbb{P})$ (or on $\mu(\mathbb{P})$) for real set of paths we introduce a simple random model. We are given m and n natural numbers and n real numbers $\lambda_i \in [0, 1]$. The random set of $\mathbb{P}$ of m paths over n nodes is obtained by taking independently n binary strings of length m such that the i-th string is distributed according to the binomial distribution $\mathsf{Bin}(m, \lambda_i)$. That means that node $i \in V$ will be present on each path with probability λ_i and absent with probability $(1 - \lambda_i)$.

Our approach to estimate $|\mathsf{ID}_k(\mathbb{P})|$ is the following:

1. by Lemma 5.3.5, $|\mathsf{SEP}_k(\mathbb{P})| \leq |\mathsf{ID}_k(\mathbb{P})|$;
2. For $u \in V$ we obtain $\nu_{n,m,\lambda}(u) = \Pr[u \in \mathsf{SEP}_k(\mathbb{P})]$;

3. Given a real set $\hat{\mathbb{P}}$ of M paths on N nodes, we assume $\hat{\mathbb{P}}$ to be a random experiment and from $\hat{\mathbb{P}}$ we compute a *maximum likelihood estimate* (MLE) $\hat{\lambda}_i$ for each of the λ_i;
4. We estimate $|\,\mathsf{SEP}_k(\hat{\mathbb{P}})| = \sum_{u \in V=[N]} \nu_{N,M,\hat{\lambda}}(u)$.

Let $u \in V$ and $W \in \binom{V\setminus\{u\}}{\leq k}$. Let us say that (u, W) is GOOD if there is a path $p \in \mathbb{P} = [m]$ such that $p \in \mathbb{P}(u) \setminus \mathbb{P}(W)$. (u, W) is BAD if it is not GOOD.

Lemma 5.4.1. *Let $u \in V$ and $W \subseteq V \setminus \{u\}$ of size at most k.*

$$\Pr[(u, W)\,\mathsf{BAD}] = \left(1 - \lambda_u \prod_{w \in W} (1 - \lambda_w)\right)^m.$$

Proof. (u, W) is BAD if and only if for all $p \in \mathbb{P} = [m] : (p(u) \to p(W))$. This means that if p touches u which happens with probability λ_u ($\Pr[p(u)] = \lambda_u$), then (for (u, W) to be BAD) p must touch at least one node of W which happens with probability $1 - \prod_{w \in W}(1 - \lambda_w)$ since the probability of a path not touching any nodes of W is $\prod_{w \in W}(1 - \lambda_w)$ ($\Pr[\neg p(w)] = (1 - \lambda_w)$). Notice that if p does not touch u, which happens with probability $(1 - \lambda_u)$, then u is separated from any W. Then $\Pr[(u, W)\,\mathsf{BAD}] = (\Pr[(p(u) \to p(W))])^m$. We thus have $\Pr[(u, W)\,\mathsf{BAD}] = (1 - \lambda_u + \lambda_u(1 - \prod_{w \in W}(1 - \lambda_w)))^m = (1 - \lambda_u \prod_{w \in W}(1 - \lambda_w))^m$.

The condition $p(u) \to p(W)$ is also the same as $\neg p(u) \vee \bigvee_{w \in W} p(w)$ which is the same as $\neg\,(p(u) \wedge \bigwedge_{w \in W} \neg p(w))$. Hence the claim. □

Let $k \leq n.$ and let $S(k) = \binom{[n-1]}{\leq k}$.

Theorem 5.4.2. *Let $n, m, k \in \mathbb{N}$, $u \in V$, and $k \leq n$.*

$$\Pr[u \in \mathsf{SEP}_k(\mathbb{P})] = \prod_{W \in S(k)} \left(1 - (1 - \lambda_u \prod_{w \in W} (1 - \lambda_w))^m\right).$$

Proof. Observe that $\Pr[u \in \mathsf{SEP}_k(\mathbb{P})] = \Pr[\ u$ is k-SEP in $\mathbb{P}] = \Pr[\forall W, u \notin W, |W| \leq k : (u, W)\,\mathsf{GOOD}]$. By previous lemma $\Pr[(u, W)\,\mathsf{GOOD}] = 1 - (1 - \lambda_u \prod_{w \in W}(1 - \lambda_w))^m$. Hence the theorem follows. □

Assume we have a set $\hat{\mathbb{P}}$ of m paths over n nodes. We consider $\hat{\mathbb{P}}$ as a random experiment. The standard approach to compute an MLE estimate $\hat{\lambda}_i$ of the λ_i in the case of binomial distribution is to compute $\hat{\lambda}_i$ as the zero of the polynomial obtained by the prime derivative of the function expressing the probability that the node i touches N_i paths in $\hat{\mathbb{P}}$.

Let $p_i = \Pr[$ node i touches N_i paths in $\hat{\mathbb{P}}]$. Since in $\hat{\mathbb{P}}$ the column i is distributed accordingly to the $\mathsf{Bin}(m, \lambda_i)$, then $p_i = \binom{m}{N_i}\lambda_i^{N_i}(1-\lambda_i)^{m-N_i}$. We study $\frac{d}{d\lambda_i}p_i$ and compute $\hat{\lambda}_i$ by setting $\frac{d}{d\lambda_i}p_i = 0$. It is easy to see that this happen for $\hat{\lambda}_i = \frac{N_i}{m}$.

5.5 Localizing Failure Nodes in Real Networks

In this section we study some heuristics to compute as more precisely as possible the number of k-identifiable nodes in a set of measurement paths defined on concrete networks, that is the set of all paths between monitor nodes. According to Section 5.3, we study upper bounds on the number of k-distinguishable nodes.

To upper bound $|\,\mathsf{DIS}_k\,|$ we lower bound the number of nodes which are *not* distinguishable in $\mathbb{P}$. In fact we will localize specific set of nodes which we can guarantee to be not k-distinguishable.

Let $\mathbb{P}$ be given and let $u \in V$. We let $\mathbb{W}_k(u)$ be a subset of $\binom{V\setminus\{u\}}{\leq k}$. This should be meant as (a method to generate) a collection of subsets of at most k nodes in $V \setminus \{u\}$ based on a function of the node u. An example can be: the subsets of V made by at most k nodes which are at distance at most d from u. For any $v \in V$, let $\mathcal{P}(u, v) \subseteq \mathbb{P}(u) \cap \mathbb{P}(v)$. This should be meant as (a method to generate) a subset of all paths touching both nodes u and v.

Definition 5.5.1 (k-equal modulo $\mathcal{P}$)**.** *Let $\mathbb{W}$ and $\mathcal{P}$ be given for $\mathbb{P}$. We say that $u \in V$ and $W \in \mathbb{W}_k(u)$ are* k-equal modulo $\mathcal{P}$ *in $\mathbb{P}$ if* *k-equal modulo $\mathcal{P}$*

1. *$\exists w, w' \in W$ such that $\mathbb{P}(u) \setminus \mathcal{P}(u, w) \subseteq \mathbb{P}(w')$, and*
2. *$\forall w \in W$, $\mathbb{P}(w) \setminus \mathcal{P}(u, w) \subseteq \mathbb{P}(u)$.*

Let

$$E_{V,k}[\mathbb{W}, \mathcal{P}] := \{u \in V : \text{there is a } W \in \mathbb{W}_k(u) \text{ s.t. } u \text{ and } W \text{ are } k\text{-equal modulo } \mathcal{P}\}.$$

Lemma 5.5.2. *For all $V \subseteq [n]$, $E_{V,k}[\mathbb{W}, \mathcal{P}] \subseteq \overline{\mathsf{DIS}_k(\mathbb{P})}$.*

Proof. Let $u \in E_{V,k}[\mathbb{W}, \mathcal{P}]$. We have to find a $W \in \binom{V}{\leq k}$ with $u \notin W$ such that $\mathbb{P}(u) = \mathbb{P}(W)$. Fix W as the one in $\mathbb{W}_k(u)$ given by the the definition of $E_{V,k}[\mathbb{W}, \mathcal{P}]$. We first argue that $\mathbb{P}(u) \subseteq \mathbb{P}(W)$. By Definition 5.5.1 case (1) we know that there exist $w, w' \in W$ such that $\mathbb{P}(u) - \mathcal{P}(u, w) \subseteq \mathbb{P}(w')$. Consider $p \in \mathbb{P}(u)$. If $p \in \mathcal{P}(u, w)$, then $p \in \mathbb{P}(w)$ and hence $p \in \mathbb{P}(W)$. If $p \notin \mathcal{P}(u, w)$, then $p \in \mathbb{P}(u) \setminus \mathcal{P}(u, w)$ and then by Definition 5.5.1 case (1) p is in $\mathbb{P}(w')$ and hence in $\mathbb{P}(W)$.

Let $q \in \mathbb{P}(W)$, then $q \in \mathbb{P}(w)$ for some $w \in W$. If $q \in \mathcal{P}(u, w)$, then $q \in \mathbb{P}(u)$. If $q \notin \mathcal{P}(u, w)$, then $q \in \mathbb{P}(w) \setminus \mathcal{P}(u, w)$ and then, by Definition 5.5.1 case (2), $q \in \mathbb{P}(u)$. □

By Lemma 5.5.2, nodes in $E_{V,k}[\mathbb{W}, \mathcal{P}]$ are not k-distinguishable and, by the anti-monotonicity, are not $(k+1)$-,$(k+2)$-,..., n-distinguishable.

We now study how to upper bound the number of k-distinguishable nodes in $\mathbb{P}$ given the specific definition of $\mathbb{W}$ and $\mathcal{P}$. Consider the following family of vertices in V:

$$\begin{cases} V_1 = V \\ V_k = V - \bigcup_{j<k} E_{V_j,j}[\mathbb{W}, \mathcal{P}] \qquad k > 1 \end{cases}$$

Definition 5.5.3. *Let $k \leq n$. $\tau_k := |E_{V_k,k}[\mathbb{W}, \mathcal{P}]|$.* *Definition of τ_k*

Theorem 5.5.4. $|\mathsf{DIS}_k(\mathbb{P})| \leq n - \sum_{j=1}^{k} \tau_j$.

Proof. We abbreviate $E_{V_j,j}[\mathbb{W}, \mathcal{P}]$ with $E_{V_j,j}$. First we claim that $\left|\bigcup_{j\leq k} E_{V_j,j}\right| \leq \sum_{j=1}^{k} \tau_j$. This is because for all $k \leq n$, if $u \in E_{V_k,k}$, then $u \notin \bigcup_{j\leq k-1} E_{V_j,j}$, by definition of $E_{V_k,k}$.

Further we claim that

$$E_{V_k,k} \subseteq \overline{\mathsf{DIS}_k(\mathbb{P})} \setminus \bigcup_{j\leq k-1} E_{V_j,j}.$$

Indeed by Lemma 5.5.2 $E_{V_k,k} \subseteq \overline{\mathsf{DIS}_k(\mathbb{P})}$ and again by definition of $E_{V_k,k}$, if $u \in E_{V_k,k}$, then $u \notin \bigcup_{j\leq k-1} E_{V_j,j}$. Therefore:

$$|\overline{\mathsf{DIS}_k(\mathbb{P})}| \geq |E_{V_k,k}| + \left|\bigcup_{j\leq k-1} E_{V_j,j}\right|.$$

By definition of τ_k it follows that $|\overline{\mathsf{DIS}_k(\mathbb{P})}| \geq \tau_k + \sum_{j=1}^{k-1} \tau_j$, and hence that $|\,\mathsf{DIS}_k(\mathbb{P})| \leq n - \sum_{j=1}^{k} \tau_j$. □

Notice that the proof of the theorem is constructive and is counting well-defined nodes in the network, so that nodes can also be precisely localized.

Chapter 6

Algorithms, Data and Experiments

Algorithms are one of the most important aspects of Computing and are a very important topic in Computer Science because they help software developers create efficient and error free programs. With the best algorithm, a computer programme will be able to produce very accurate results. Algorithms give us the most ideal option of accomplishing a task.

On the other hand, experiments play fundamental roles in science. One of its important roles is to test theories and to provide the basis for scientific knowledge. It can also acquire new knowledge. Experiment can provide hints toward the structure or mathematical form of a theory and it can provide evidence for the existence of the entities involved in our theories. The role of experimentation in Informatics in order to understand the nature of information processes is highly essential. Experiments must be sound and verifiable.

Here in this chapter we provide algorithms and experiments for the results we have proven in previous chapters. We get some results as applications of the main results of the previous chapters.

In *Sections 6.1 and 6.2* we first present a heuristic, `Agrid` (Algorithm 1) to boost maximal identifiability increasing the minimal degree of the network. Then we run some experiments and using several Tables we comment on the performance of `Agrid` on concrete examples of networks. We end by discussing feasibility of `Agrid` on real networks.

Section 6.3 includes experiments on how to count k-separable nodes on real networks through a maximum likelihood estimate method on the random model.

In *Sections 6.4 and 6.5*, respectively first we study the combinatorics and the complexity of the theoretical problem of k-identifiability, cf. Theorems 6.4.1 and 6.4.3 and then we present some algorithms and heuristics to count failure nodes in networks as more accurately as possible.

6.1 Agrid (Adding Edges to Increase μ)

In this section we discuss how Theorem 4.2.1 suggests the design of a network on N nodes potentially reaching a maximal identifiability of $O(\log N)$. We describe a heuristic, `Agrid`, to boost maximal identifiability in a network by adding random edges to increase the original minimal degree and simulate a d-hypergrid.

Assume we have to design a network over $N \geq 4$ nodes and we aim to have maximal identifiability of failure nodes. Theorem 4.2.1 suggests how to set edges between the nodes in the network and how to place monitors in such a way to reach an identifiability of at most $\log_3 N$. Let $n \geq 3$ and set a dimension d in such a way $N = n^d$. Since $n \geq 3$, then $N \geq 3^d$. Hence as long as $d \leq \log_3 N$, Theorem 4.2.1 applies. Assume that all values are integers. Assign an address to each node as a d-dimensional vector in $[n]$ and place edges between nodes following $\mathcal{H}_{n,d}$.

Now assume to have a network with very low maximal identifiability. We explore the idea to add edges to the network to get better maximal identifiability. The following algorithm modifies a graph G to approximate a d-grid (d will be a parameter), adding random edges to the topology, to increase the minimal degree to d (and choosing d input and d output monitors).

Algorithm 1 `AGrid`

Data: $G = (V, E)$, d
Result: $G^{\mathtt{A}} = (V, E^{\mathtt{A}})$, $I_S \subseteq V$, $I_T \subseteq V$

/* Boost minimal degree as close to d as possible */
1: **for all** $v \in V : \deg(v) < d$ **do**
2: $W =$ choose at random $d - |N(v)|$ nodes in $V \setminus N(v)$
3: **for all** $w \in W$ **do**
4: $E = E \cup (v, w)$

/* Select input and output nodes*/
5: **for** $i = 1, \ldots, d$ **do**
6: Select $x, y \in V$ according to heuristic MDMP
7: $S = S \cup \{x\}; T = T \cup \{y\}$;
8: $V = V - \{x, y\}$

Given a network G, Agrid adds a number of random edges so that the minimal degree of the network increases to some suitable $d = d(N)$, a slow-growing function of the number of nodes N in the network. Agrid assumes to work with a network where monitors are not placed. To place monitors we follow the heuristic of placing monitors on the nodes of minimal degree. We call this heuristic MDMP. Agrid gets in input the graph G and the value $d = d(N)$ and release in output a graph $G^{\mathtt{A}}$ whose minimal degree is d. The addition of edges is performed between Lines 1 and 4. For each node v with degree smaller than d, we choose at random a number of neighbours w, namely $d - |N(v)|$, and we add an edge in the network between v and w, keeping updated the set of edges (Line 4). In Line 5 we choose $2d$ nodes according to MDMP to be linked to input and output monitors in $G^{\mathtt{A}}$. We use the same monitor placement on G.

Algorithm Complexity. Algorithmic complexity is a measure of how long an algorithm would take to complete given an input of size n. If an algorithm has to scale, it should compute the result within a finite and practical time bound even for large values of n. While complexity is usually in terms of time, sometimes complexity is also analyzed in terms of space, which translates to the algorithm's memory requirements. Analysis of an algorithm's complexity is helpful when comparing algorithms or seeking improvements. The Agrid algorithm has the complexity of $O(N^2)$ since the size of the input graph G (the number of nodes) is N and also we have the dimension $d = d(N)$. So we have $O(Nd)$ as the complexity of the Agrid.

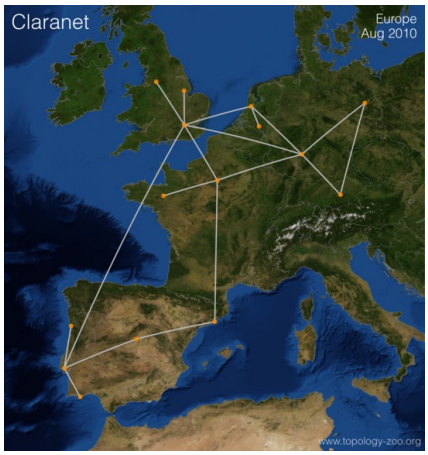
6.2 Experimental Data and Feasibility of Agrid

For any topology G on N nodes and for the parameter d that we set d as either $\log N$ or $\sqrt{\log N}$, Agrid generates $G^{\mathtt{A}}$, the super graph of G that simulates a d-hypergrid. After computing the monitor placements χ and $\chi^{\mathtt{A}}$, we proceed to compute $\mu(G^{\mathtt{A}}|\chi^{\mathtt{A}})$ and $\mu(G|\chi)$. We generate all possible paths, and hence the number of paths tends to highly grow. That is the reason why our examples are with less than 30 nodes.

Real networks

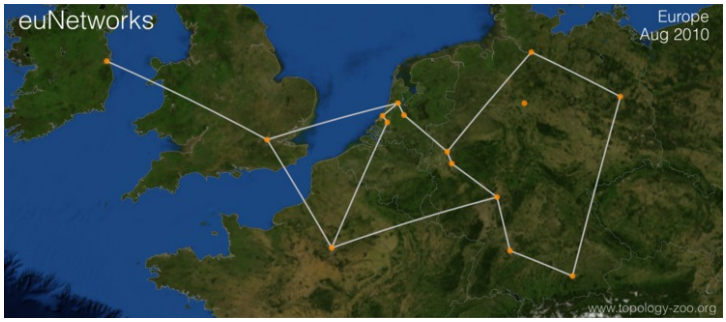
We test examples of real internet networks whose topologies are on the data set Internet Topology Zoo ([33]). We compute maximal identifiability,

number of paths, minimal degree and number of edges for eight networks for the both cases $d = \log N$ and $d = \sqrt{\log N}$ (Tables 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7 and 6.8). d is the dimension of simulated hypergrid. In some examples (such as Table 6.3), when the number of nodes is so small that $d \leq \delta(G)$ (so $G^{\mathtt{A}}$ would not change with respect to G) we decide to add one dimension to d. The examples show an increment of the maximal identifiability more evident in the case when $d = \log N$. For example in Tables 6.2 on a network of 16 edges, adding 6 monitors and 9 links, we pass from not having identifiability at all (using the same number of monitors) to detect uniquely in $G^{\mathtt{A}}$ any two node-failures.



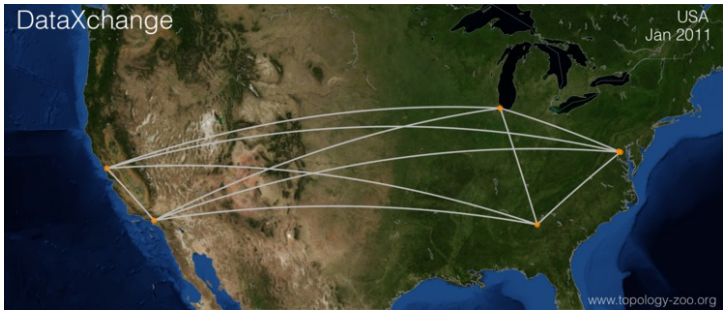
	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathtt{A}}$	G	$G^{\mathtt{A}}$
μ	**0**	**1**	**1**	**2**
$\lvert\mathbb{P}\rvert$	18	247	39	16528
$\lvert E \rvert$	18	22	18	29
δ	1	2	1	3
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 3$	

Table 6.1. *Claranet,* $|V| = 15$.



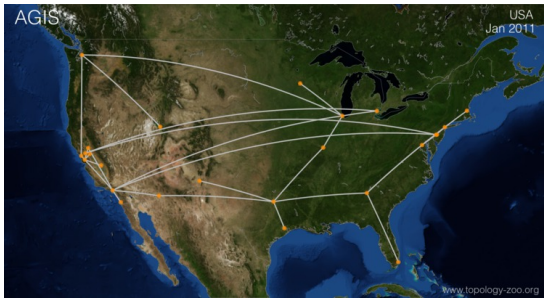
	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathtt{A}}$	G	$G^{\mathtt{A}}$
μ	**0**	**1**	**0**	**2**
$\lvert\mathbb{P}\rvert$	20	40	46	4917
$\lvert E \rvert$	16	17	16	25
δ	1	2	1	3
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 3$	

Table 6.2. *EuNetworks,* $|V| = 14$.



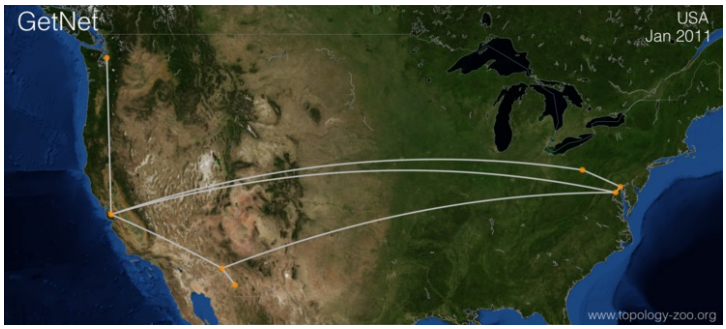
	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathbb{A}}$	G	$G^{\mathbb{A}}$
μ	**1**	**1**	**1**	**2**
$\lvert\mathbb{P}\rvert$	64	108	129	291
$\lvert E \rvert$	11	12	11	13
δ	1	2	1	3
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 3$	

Table 6.3. *DataXchange*, $|V| = 6$.



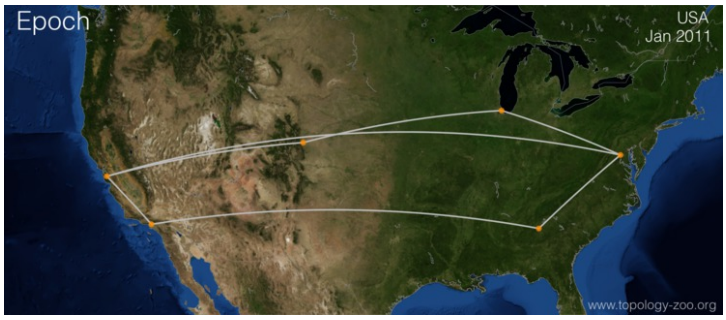
	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathbb{A}}$	G	$G^{\mathbb{A}}$
μ	**1**	**2**	**1**	**2**
$\lvert\mathbb{P}\rvert$	73	3050	242	10110
$\lvert E \rvert$	30	39	30	60
δ	1	2	1	4
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 4$	

Table 6.4. *AGIS*, $|V| = 25$.



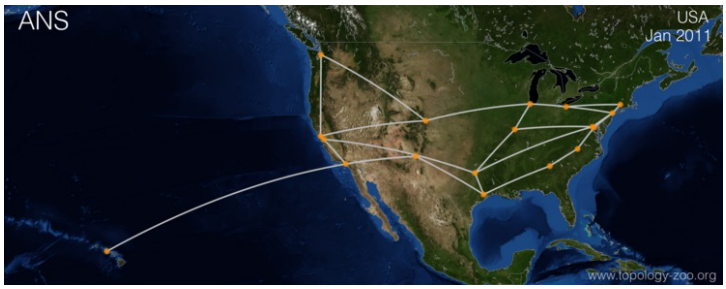
	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathbb{A}}$	G	$G^{\mathbb{A}}$
μ	**1**	**1**	**1**	**2**
$\lvert\mathbb{P}\rvert$	12	52	24	181
$\lvert E \rvert$	8	11	8	12
δ	1	2	1	3
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 3$	

Table 6.5. *GetNet*, $|V| = 7$.



	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathbb{A}}$	G	$G^{\mathbb{A}}$
μ	**1**	**1**	**1**	**2**
$\lvert\mathbb{P}\rvert$	14	110	41	112
$\lvert E \rvert$	7	10	7	10
δ	2	2	2	3
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 3$	

Table 6.6. *Epoch*, $|V| = 6$.



	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathbb{A}}$	G	$G^{\mathbb{A}}$
μ	**1**	**2**	**1**	**2**
$\lvert\mathbb{P}\rvert$	292	541	1222	2264
$\lvert E \rvert$	25	26	25	41
δ	1	2	1	3
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 3$	

Table 6.7. *ANS*, $|V| = 18$.

	$d = \sqrt{\log \lvert V \rvert}$		$d = \log \lvert V \rvert$	
	G	$G^{\mathbb{A}}$	G	$G^{\mathbb{A}}$
μ	**0**	**1**	**0**	**1**
$\lvert\mathbb{P}\rvert$	12	13	25	104
$\lvert E \rvert$	20	23	20	40
δ	1	2	1	3
	$d, \lvert S \rvert, \lvert T \rvert = 2$		$d, \lvert S \rvert, \lvert T \rvert = 3$	

Table 6.8. *Peer1*, $|V| = 16$.

Random graphs

We run measurements on random graphs on few nodes (5, 8 and 10). After generating the graphs G and then computing $G^{\mathtt{A}}$ with `Agrid` for 50, 100 or 500 times, we count the fraction of cases where μ in the case of $G^{\mathtt{A}}$ is increasing or remaining the same (it is never strictly less) and what is the maximal increment of μ reached in a pair $(G, G^{\mathtt{A}})$. Monitors on G and $G^{\mathtt{A}}$ are again placed according to MDMP heuristic. On the *rows* of Tables 6.9 and 6.10 there are the numbers of generated graphs, while on the *columns*, for each of the three cases of 5, 8 and 10 nodes, we separate cases where $\mu(G^{\mathtt{A}}) > \mu(G)$, from where $\mu(G^{\mathtt{A}}) = \mu(G)$. In the square bracket there is the information of the maximal value of $\mu(G^{\mathtt{A}}) - \mu(G)$ obtained in the tested pairs $(G, G^{\mathtt{A}})$.

	5		8		10	
	$>$	$=$	$>$	$=$	$>$	$=$
50	[2]8%	92%	[2]40%	60%	[1]16%	84%
100	[2]18%	82%	[2]39%	61%	[2]18%	82%
500	[2]14%	86%	[2]34%	66%		

Table 6.9. *Case* $d = \log n$.

	5		8		10	
	$>$	$=$	$>$	$=$	$>$	$=$
50	[2]8%	92%	[2]40%	60%	[1]16%	84%
100	[2]18%	82%	[2]39%	61%	[2]18%	82%
500	[2]14%	86%	[2]34%	66%		

Table 6.10. *Case* $d = \log n$.

Random monitors

MDMP is a simple heuristic for monitor placement. However the lower bound of Theorem 4.2.1 holds for any monitor placement. We try to give some evidence that `Agrid` is a good strategy for boosting maximal identifiability independently of where monitors are placed. In the following Tables, we collect results for percentage of values of $\mu(G)$ on 20 random placements of monitors both in G and $G^{\mathtt{A}}$. Tables 6.11, 6.12, 6.13, 6.14, 6.15 and 6.16 show that moving to $G^{\mathtt{A}}$ gives an improvement in the

maximal identifiability, independently of the monitor placement. Also in these cases the data are computed only for the most significant case of $\log N$.

$G \setminus \mu$	**0**	**1**	**2**
G	20%	80%	0%
$G^{\mathtt{A}}$	0%	0%	100%

Table 6.11. *Claranet,* $|V| = 15$, $S, T, d = 3$.

$G \setminus \mu$	**0**	**1**	**2**
G	100%	0%	0%
$G^{\mathtt{A}}$	0%	80%	20%

Table 6.12. *EuNetworks,* $|V| = 14$, $S, T, d = 3$.

$G \setminus \mu$	**0**	**1**	**2**
G	0%	100%	0%
$G^{\mathtt{A}}$	0%	10%	90%

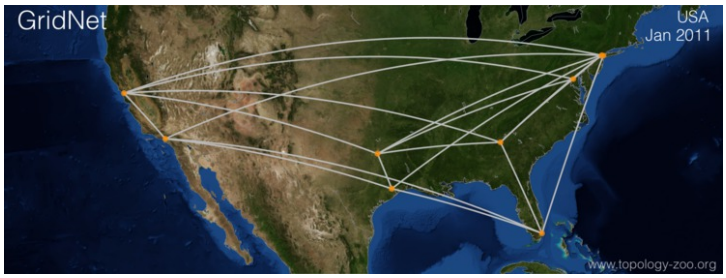
Table 6.13. *GetNet,* $|V| = 7$, $S, T, d = 3$.

$G \setminus \mu$	**0**	**1**	**2**
G	0%	90%	30%
$G^{\mathtt{A}}$	0%	35%	65%

Table 6.14. *Epoch,* $|V| = 6$, $m, M, d = 3$.

$G \setminus \mu$	**0**	**1**	**2**
G	0%	85%	15%
$G^{\mathtt{A}}$	0%	5%	95%

Table 6.15. *DataXchange,* $|V| = 6$, $m, M, d = 3$.



$G \setminus \mu$	**0**	**1**	**2**
G	0%	100%	0%
$G^{\mathtt{A}}$	0%	10%	90%

Table 6.16. *GridNet,* $|V| = 9$, $S, T, d = 3$.

Feasibility of `Agrid`. Adding a link in some cases may require local or physical access to nodes, access which might not be feasible. However there are examples of networks where adding links may not require a local intervention (or requires a limited one) and hence our approach is reasonable. we consider two cases of static networks and dynamic networks.

Static networks. Static networks are grounded on a fixed topology which does not change in the time. In such cases it makes sense to analyze the economical feasibility of running `Agrid`. A way to reduce both the costs and the physical access to nodes and links of the network is an approach suggested for nodes in [32].

On-demand link placement. Similarly to what is done for nodes in [32], we can think to employ as additional links in $G^{\mathtt{A}}$ *temporary links*, which only participate in taking measurements (hence built upon very simple hardware and protocols to transmit simple data packet) and not in other more complex functions. This would reduce the cost of adding links and it might simplify the type of physical access to the network.

Dynamic networks. In dynamic networks the topology is changing in the time according to some rules (but they can be even unpredictable). They are specified by a sequence of graphs $\{G_t\}_{t\in T}$ where T is a set of times. For example some cases of wireless networks are dynamical networks where the underlying topology changes at each given time. Nodes are supposed to have a built-in mechanism $\mathcal{M}$ to set new links among the nodes in-sight. In such cases we can think to modify `Agrid` in such a way that links to one node u are added randomly choosing the other nodes among the nodes reachable from u according to mechanism $\mathcal{M}$. The approach of temporary links would be particularly suited in the case of dynamic networks, where we can think of adding new edges at each time for each network G_t.

6.3 MLE Method and Experiments

Let $\nu_{n,m,\vec{\lambda}}(u) = \Pr[u \in \mathsf{SEP}_k(\mathbb{P})]$. Assume to have a real set of M paths $\hat{\mathbb{P}}$ over N nodes. We extract from $\hat{\mathbb{P}}$ the $\hat{\lambda}_i$ for all $i \in V = [N]$ and we then estimate $|\mathsf{SEP}_k(\hat{\mathbb{P}})|$ as $\chi(\hat{\mathbb{P}}, k, \hat{\lambda}) = \sum_{u\in[N]} \nu_{N,M,\hat{\lambda}}(u)$, using the closed formula in Theorem 5.4.2.

In figures 6.1, 6.2 and 6.3 we consider three graphs from the Internet Topology Zoo (ClaraNet, BTEurope and Bridge Networks) and we consider set of measurement paths $\hat{\mathbb{P}}$ obtained from these networks by taking all the different paths starting from a source node and ending at a target node (green nodes are sources and red nodes are targets). In the second table in each Figure we compare the real values of $|\mathsf{SEP}_k(\hat{\mathbb{P}})|$ with the values of $\chi(\hat{\mathbb{P}}, k, \hat{\lambda})$ for all these paths, obtaining results very tight to the real values.

Notice that to compute $\nu_{N,M,\hat{\lambda}}(u)$ we need to compute $\Pr[(u,W)\,\mathsf{GOOD}]$ for all $W \in S(k)$[1]. We consider another estimate of $|\,\mathsf{SEP}_k(\hat{\mathbb{P}})|$ obtained from $\chi(\hat{\mathbb{P}},k,\hat{\lambda})$ by having only one value for all the λ_i's. We consider the significant case $\chi_2(\hat{\mathbb{P}},k,\hat{\lambda}_{\max})$, where $\hat{\lambda}_{\max} = \max_i \lambda_i$[2]. Notice that in these cases we do not have to have available all the λ_w for all $W \subseteq V = [N]$ of size at most k and the computation can be made much less expensive since $\chi_2(\hat{\mathbb{P}},k,\hat{\lambda}_{\max}) = \prod_{j\in[k]}(1-(1-\lambda_u(1-\lambda_{\max})^j)^m)^{\binom{n-1}{j}}$ and we can use methods to approximate $\binom{n-1}{j}$. Notice that the estimate χ_2 is already very good in all these examples.

In each table we also scatter the estimates $\hat{\lambda}_i$ coming from the MLE method.

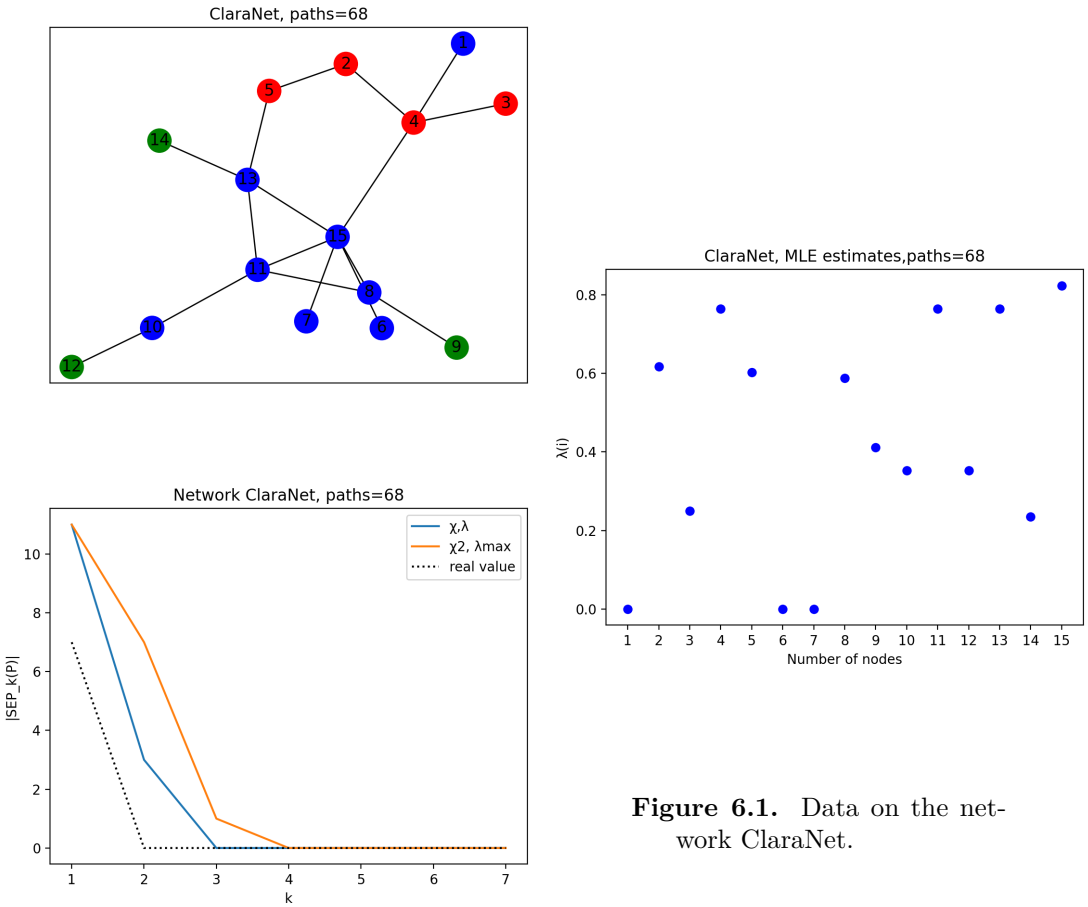


Figure 6.1. Data on the network ClaraNet.

[1] This is because we need to use the λ_w for all $w \in W$.

[2] It is easy to see that $\chi(\hat{\mathbb{P}},k,\hat{\lambda}) \geq \chi_2(\hat{\mathbb{P}},k,\hat{\lambda}_{\max})$.

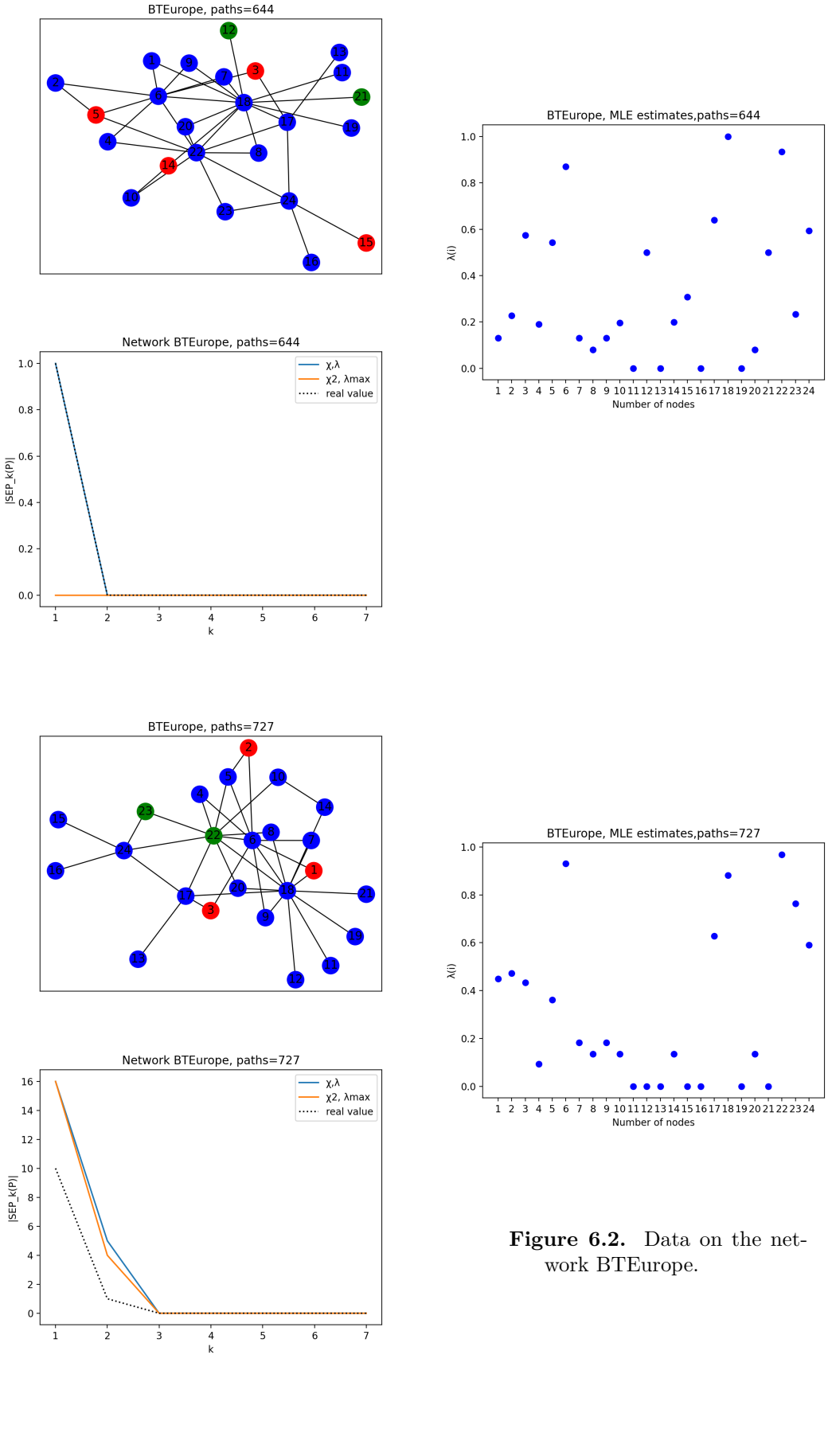


Figure 6.2. Data on the network BTEurope.

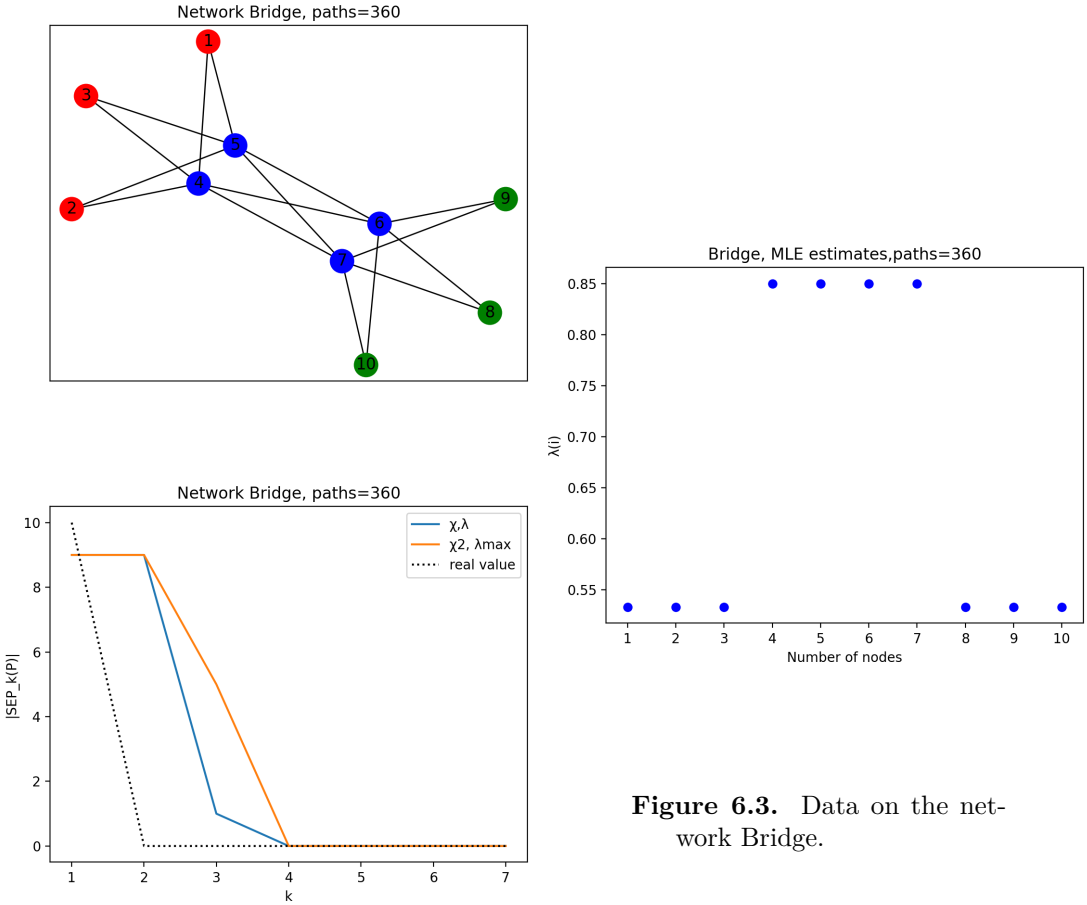


Figure 6.3. Data on the network Bridge.

6.4 Complexity of k-Identifiability and The Minimum Hitting Set

Minimum hitting set problem

Consider the optimization problem *Minimum Hitting Set*, MHS, that given a hypergraph (a set-system) $\mathcal{H} = (V, E)$, where $E \in \binom{V}{2}$, asks to find the smallest $V' \subseteq V$ such that for all $e \in E$, $V' \cap e \neq \emptyset$. MHS is a notorious NP-complete problem [2, 28] extending vertex cover.

We show how to use MHS to find the minimal k such that u is not k-ID in $\mathbb{P}$, i.e. there exits a set of nodes W of size $|W| \leq k$, such that $\mathbb{P}(u) \subseteq \mathbb{P}(W)$.

Theorem 6.4.1. *Assume* MHS *is solvable in polynomial time, then deciding whether u is not k-*SEP *in $\mathbb{P}$ is solvable in polynomial time.*

Proof. Consider the subset $T(u)$ of $V = [n]$ of those nodes touching at least a path in $\mathbb{P}(u)$. Let Y be the vector of dimension $|\mathbb{P}(u)|$ defined in the j-th coordinate as following:

$$Y[j] = \bigvee_{v \in T(u)} \mathbb{P}[j, v] \qquad j \in \mathbb{P}(u)$$

Y has no 0-coordinate for otherwise there is a path in $\mathbb{P}$ only touching u. Hence Y has all 1-coordinates. We consider the set-system $\mathcal{H}$ obtained from $\mathbb{P}$ by restricting the columns to $T(u)$ and the rows to $\mathbb{P}(u)$. Let W be the smallest subset of $T(u)$ provided by MHS and covering all $\mathbb{P}(u)$. Hence u is not $|W|$-SEP, since $\mathbb{P}(u) \subseteq \mathbb{P}(W)$.

The optimality of the bound is an immediate consequence of the optimality of MHS. There is no subset Z of $V = [n]$ smaller than W such that $\mathbb{P}(u) \subseteq \mathbb{P}(Z)$, since of course $Z \subseteq T(u)$ and, by optimality of MHS, Z cannot be smaller than W. □

The problem of finding a *minimal transversal* in a hypergraph is a simplification of MHS (see below) which can be decided efficiently. Our reduction hence suggests to implement an algorithm on concrete examples of paths where we find the minimal transversal instead of the minimum hitting set.

Let us recall the following definitions from hypergraph transversal problem [20].

Definition 6.4.2 (Transversal)**.** *Let $\mathcal{H} = (V, E)$ be a hypergraph. A set $T \subseteq V$ is called a* transversal *of $\mathcal{H}$ if it meets all the edges of $\mathcal{H}$, i.e. if $\forall e \in E : T \cap e \neq \emptyset$. A transversal T is called* minimal *if no proper subset T' of T is a transversal.* *Transversal*

It is possible to find in time $O(|V||E|)$ a minimal transversal of $\mathcal{H}$ by the following algorithm (see also [20]). If $E = \emptyset$, then every subset of V is a transversal of $\mathcal{H}$, hence the minimal one is $\emptyset$. If $E \neq \emptyset$, let $V = \{v_1, \ldots, v_n\}$. Then define:

$$V_0 = V$$
$$V_{i+1} = \begin{cases} V_i & V_i \setminus \{v_i\} \text{ is not a transversal of } \mathcal{H} \\ V_i \setminus \{v_i\} & V_i \setminus \{v_i\} \text{ is a transversal of } \mathcal{H} \end{cases} \qquad 0 \leq i < n$$

Hence V_n is a minimal transversal of $\mathcal{H}$. However notice that V_n is not necessarily the smallest (by cardinality) transversal of $\mathcal{H}$. In fact this last problem is the MHS problem which is NP-hard.

Let us call HT be a procedure that implements the previous algorithm on a given $\mathcal{H}(V, E)$, and a given order on V, and outputs a minimal transversal of $\mathcal{H}$.

The proof of Theorem 6.4.1 suggests an algorithm to compute an upper bound on the k-separability of a node u in $\mathbb{P}$, where instead of computing the minimum hitting set we compute a minimal transversal using HT on any order of the variables.

Algorithm 2 Algorithm Simple-SEP

Data: $\mathbb{P}, u$
Result: (W, s) s.t. $\mathbb{P}(u) \subseteq \mathbb{P}(W)$ and $|W| = s$
1: $W = \texttt{HT}([n], \mathbb{P}(u))$
2: **return** $(W, |W|)$;

However we can think of a slightly different algorithm which is computing HT not only once on $([n], \mathbb{P}(u))$ but several times on a sequence of hypergraphs of decreasing complexity.

Consider the following sets: for all $i \in \mathbb{P}(u)$, let $Z(v) = \{i \in \mathbb{P} \mid \mathbb{P}[i, v] = 0\}$ for all $v \in [n]$ and $V_i = \{v \in [n] \mid |\mathbb{P}(u) \cap Z(v)| = i\}$. Let $I = \{i_1, \dots i_N\} \subseteq \mathbb{P}(u)$ be the set of indices of the $V_{i_j} \neq \emptyset$. We say that $V_{i_N}, \dots, V_{i_1}$ is a 0-decreasing sequence since, by definition, $Z(v) > Z(w)$ whenever $v \in V_i$, $w \in V_j$ and $i < j$.

Algorithm 3 Decr-SEP

Data: $\mathbb{P}, u$
Result: (W, s) s.t. $\mathbb{P}(u) \subseteq \mathbb{P}(W)$ and $|W| = s$
Compute all V_i's
Compute I
1: **for** $l = 0, \dots, N$ **do**
2: $k = N - l$
3: **for** $j \in \mathbb{P}(u)$ **do**
4: **if** $j \in \mathbb{P}_l[u]$ **then**
5: $\vec{Y}_{i_k}[j] = \bigvee_{v \in V_{i_k}} \mathbb{P}_l[j, v]$
6: **else**
7: $\vec{Y}_{i_k}[j] = 0$
8: $\hat{V}_{i_k} = \texttt{HT}(V_{i_k}, \mathbb{P}_l[u])$
9: $Z_{i_k} =$ 0-coordinates of Y_{i_k}
10: $\mathbb{P}_{l+1}(u) = \mathbb{P}_l(u) \cap Z_{i_k}$
11: $\mathcal{Y} = \texttt{HT}(I, \bigcup_{i \in I} \vec{Y}_i)$
12: $W = \bigcup_{i \in \mathcal{Y}} \hat{V}_i$
13: **return** $(W, |W|)$

The algorithm starts by computing the set V_i and the set of indices I of such sets which are not empty. The main observations on the algorithm are the following:

- that $V_{i_N}, \ldots, V_{i_1}$ is a 0-decreasing sequence. At each step we try to cover only the paths in $\mathbb{P}(u)$ not already covered before. This is the reason why in line 9 we restrict only to 0-coordinates in Z_{i_k}. The vectors Y_i are also defined accordingly. Only the coordinates in $\mathbb{P}_l(u)$ are important since the rest are already covered by some previous V_i. That is the reason why in line 7 we define the Y_i vector to be 0 in all the coordinates not in $\mathbb{P}_l(u)$.

- another observation is that at each step l we want to save the minimal set of nodes $\hat{V}_{i_{N-l}}$ sufficient to cover all the 1's in $\mathbb{P}_l[u]$. This is the meaning of the call `HT` in line 8.

- finally, when we are done with analyzing the family of all the sets V_i's, $\mathbb{P}(u)$ is covered by the union of the Y_i vectors (this is by an argument similar to that of Theorem 6.4.1). However it is sufficient to have the minimal subset of this family for covering all $\mathbb{P}(u)$. To this end we perform a final call `HT` on the input set-system, $(I, \bigcup_{i \in I} \vec{Y}_i)$ in line 11.

NP-Completeness Consider the following optimization problem MIN-NOT-SEP (MNS):
Input: A Boolean $m \times n$ matrix $\mathbb{P}$, an element $u \in V$;
Output: k such that u is not k-SEP and u is k'-SEP for all $k' < k$.

Theorem 6.4.3. MNS *is* NP*-complete.*

Proof. To see that MNS is in NP we can use the reduction in Theorem 6.4.1 which is in fact proving that MNS $\leq_p$ MHS. Since MHS $\in$ NP [2], then MNS $\in$ NP.

To prove the NP-hardness of MNS we show the opposite reduction, i.e. that MHS $\leq_p$ MNS. Hence the result follows by the NP-hardenss of MHS [2]. Let $\mathcal{H} = (V, E)$ be an instance of MHS. We define an instance of MNS as following:

- The set of nodes of $\mathbb{P}$ is $V \cup \{u\}$;

- The set of paths of $\mathbb{P}$ is E;

- $\mathbb{P}(u) = E$;

Since a minimal hitting set W is touching all edges in E, that means that $\mathbb{P}(W) = E = \mathbb{P}(u)$. Hence u is not $|W|$-SEP. Moreover since it is minimal, then for any subset $W'$of V of size smaller than $|W|$, there is an edge $e \in E$ not in W'. That means that $e \in \mathbb{P}(u) \setminus \mathbb{P}(W')$, that is u is k'-SEP in $\mathbb{P}$ for any $k' < |W|$.

On the opposite direction, assume that $W \subseteq V \setminus \{u\}$ is witnessing that u is not $|W|$-SEP but it is k'-SEP for any $k' < |W|$, then W is clearly a minimal hitting set in $\mathcal{H}$. □

6.5 Algorithm for Counting k-SEP Nodes

We show how to use previous results to localize and upper bound the number of k-identifiable nodes on real sets of measurement paths. The estimate will depend on what set $\mathbb{W}(u)$ we consider for any node u and also on what set of paths $\mathcal{P}$ we are going to test the path not distinguishability. However once we have fixed $\mathbb{W}$ and $\mathcal{P}$, the algorithm we run is always the same and reflects the discussions in the previous chapter (See Algorithm lb-DIS$_k$).

Algorithm 4 lb-DIS$_k$: Counting k-SEP nodes

Data: $\mathbb{P}$
Result: number of k-SEP nodes
1: **for** $u \in [n]$ **do**
2: Compute $\mathbb{W}(u)$
3: **for** $w \in \mathbb{W}(u)$ **do**
4: Compute $\mathcal{P}(u, w)$
5: $V = [n], i = 1, \tau = 0$
6: **while** $i \leq k$ **do**
7: Compute $E_{V,i}[\mathbb{W}, \mathcal{P}]$
8: $\tau = \tau + |E_{V,i}[\mathbb{W}, \mathcal{P}]|$
9: $V = V - E_{V,i}[\mathbb{W}, \mathcal{P}]$
10: $i = i + 1$
11: **return** $n - \tau$

Algorithm Complexity. This is straightforward to see that the above algorithm has the complexity of $O(n^2)$ (it has quadratic complexity).

Our method can be applied to a network given as a graph once we have decided the set of measurement paths. Every possible way of choosing $\mathbb{W}$,

and $\mathcal{P}$ is giving a way to count nodes which are not distinguishable. We can therefore think of applying the method restricting for each node u the nodes we are checking to be not distinguishable and the effective paths we are going to consider. For instance we can consider here the following three potential examples:

Neighbours For any given $u \in V = [n]$, let $\mathbb{W}_k(u) = \binom{N(u)}{\leq k}$, where $N(u)$ is the set of neighbours of u and consider for all $v \in N(u)$, the $\mathcal{P}^{N(u)}$ of the paths touching both u and its neighbours v.

Nodes at a fixed distance d For any given $u \in V = [n]$, let $N_d(u) = \{v \in V : d(u,v) = d, d \geq 1\}$ and $\mathbb{W}_k(u) = \binom{N_d(u)}{\leq k}$. For all $v \in N_d(u)$, consider the $\mathcal{P}^d$ of the paths touching both u and v.

Shortest paths In this case we consider set $\mathbb{W}_k(u) = \binom{V \setminus \{u\}}{\leq k}$, and for all $v \in V \setminus \{u\}$, $\mathcal{P}$ is the set of shortest paths from u to v or vice versa.

Bibliography

[1] D. Angluin, and L. G. Valiant, Fast Probabilistic Algorithms for Hamiltonian Circuits and Matchings, *Journal of Computers and Systems Sciences*, 18:155–193, 1979.

[2] G. Ausiello, A. D'Atri and M. Protasi, Structure Preserving Reductions among Convex Optimization Problems, *Journal of Computers and Systems Sciences*, 21(1): 136–153, 1980.

[3] N. Bartolini, T. He, and H. Khamfroush, Fundamental limits of failure identifiability by boolean network tomography, in: *IEEE International Conference on Computer Communications, INFOCOM 2017*. IEEE, 2017.

[4] N. Bartolini, T. He, V. Arrigoni, A. Massini and H. Khamfroush, On fundamental bounds of failure identifiability by boolean network tomography, IIEEE/ACM Transactions on Networking. 28(2):588–601, 2020.

[5] P. Berman, A. Bhattacharyya, E. Grigorescu, S. Raskhodnikova, D. P. Woodruff and G. Yaroslavtsev. Steiner transitive-closure spanners of low-dimensional posets, *Combinatorica*, 34(3):255–277, 2014.

[6] B. Bollobás, A probabilistic proof of an asymptotic formula for the number of labelled regular graphs, *European Journal of Combinatorics*, 1:311–316, 1980.

[7] B. Bollobás, T. I. Fenner, and A. M. Frieze, An algorithm for finding Hamilton paths and cycles in random graphs, *Combinatorica*, 7:327–341, 1987.

[8] B. Bollobás, *Random Graphs*, Number 73 in Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2nd edition, 2001.

[9] T. Bu, N. G. Duffield, F. L. Presti and D. F. Towsley. Network tomography on general topologies. in: *International Conference on Measurement and Modeling of Computer Systems, ACM SIGMETRICS 2002 (Ed.), Proceedings of SIGMETRICS*, pages 21–30, 2002.

[10] R. Castro, M. Coates, G. Liang, R. Nowak and B. Yu, Network tomography: Recent developments, *Statistical Science*, 19(3):499–517, 2004.

[11] M. Cheraghchi, A. Karbasi, S. Mohajer and V. Saligrama, Graph-constrained group testing, *IEEE Transactions on Information Theory*, 58(1):248–262, 2012

[12] M. Coates, A. O. Hero, R. Nowak and B. Yu, Internet tomography, *IEEE Signal Processing Magazine*, 19:47–65, 2002.

[13] A. Czumaj and X. Wang, Communication problems in random line-of-sight ad-hoc radio networks, In *Stochastic Algorithms: Foundations and Applications, 4th International Symposium, SAGA 2007, Munich, Switzerland, September 13-14, 2007, Proceedings*, pages 70–81, 2007.

[14] L. Devroye and L. Farczadi, Connectivity for line-of-sight networks in higher dimensions, *Discrete Mathematics and Theoretical Computer Science*, 15(2):71–86, 2013.

[15] D.-M. Du and F. K. Hwang, Combinatorial Group Testing and Its Applications, World Scientific, 2000.

[16] N. G. Duffield, Simple network performance tomography, in: *Proceedings of the 3rd ACM SIGCOMM Internet Measurement Conference, IMC 2003, Miami Beach, FL, USA, October 27-29, 2003*, pages 210–215. ACM, 2003.

[17] N. G. Duffield, Network tomography of binary network performance characteristics, *IEEE Transactions on Information Theory*, 52(12):5373–5388, 2006.

[18] N. G. Duffield and F. L. Presti, Network tomography from measured end-to-end delay covariance, *IEEE/ACM Transactions on Networking*, 12(6):978–992, 2004.

[19] B. Dushnik and E. W. M. Source, Partially ordered sets, *American Journal of Mathematics,*, 63(3):600–610, 1941.

[20] T. Eiter and G. Gottlob, Identifying the Minimal Transversals of a Hypergraph and Related Problems, *SIAM Journal on Computing*, 24(6): 1278–1304, 1995.

[21] M. Ettehad, N. Duffield and G -.Berkolaiko, Optimizing Consistent Merging and Pruning of Subgraphs in Network Tomography, *CoRR abs/1908.03519*, arXiv:1908.03519.

[22] P. Frankl and Z. Füredi, Union-free families of sets and Equations over field, *Journal of Number Theory*, 23: 210–218, 1986.

[23] A. M. Frieze, J. M. Kleinberg, R. Ravi, and W. Debany, Line-of-sight networks, In Nikhil Bansal, Kirk Pruhs, and Clifford Stein, editors, *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 968–977. SIAM, 2007.

[24] N. Galesi and F. Ranjbar, Tight bounds for maximal identifiability of failure nodes in boolean network tomography, in: *38th IEEE International Conference on Distributed Computing Systems, ICDCS 2018, Vienna, Austria, July 2-6, 2018, IEEE Computer Society*, pages 212–222, 2018.

[25] N. Galesi and F. Ranjbar, Tight bounds to localize failure nodes on trees, grids and through embeddings under boolean network tomography, *Submitted*, 2020. At the link: (http://wwwusers.di.uniroma1.it/~galesi/TCS2.pdf)

[26] N. Galesi, F. Ranjbar and M. Zito, Vertex-connectivity for node failure identification in boolean network tomography, in: *F. Dressler, C. Scheideler (Eds.), Algorithms for Sensor Systems - 15th International Symposium on Algorithms and Experiments for Wireless Sensor Networks, ALGOSENSORS 2019, Munich, Germany, September 12-13, 2019, Revised Selected Papers, Vol. 11931 of Lecture Notes in Computer Science, Springer*, pages 79–95, 2019.

[27] N. Galesi and F. Ranjbar, Counting and localizing defective nodes by boolean network tomography, *Submitted*, 2020. arXiv:2101.04403 (https://arxiv.org/abs/2101.04403).

[28] M. R. Garey and D. S. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, 22 ed., W. H. Freeman and Company, New York, 2000.

[29] D. Ghita, C. Karakus, K. Argyraki and P. Thiran, Shifting network tomography toward a practical goal, in: *Proceedings of the Seventh Conference on Emerging Networking Experiments and Technologies*, CoNEXT '11, pages 24:1–24:12, New York, NY, USA, 2011. ACM.

[30] F. Harary, *Graph Theory*, Addison-Wesley, Reading, MA, 1969.

[31] T. He, N. Bartolini, H. Khamfroush, I. Kim, L. Ma and T. L. Porta. Service placement for detecting and localizing failures using end-to-end observations, In *36th IEEE International Conference on Distributed Computing Systems, ICDCS 2016, Nara, Japan, June 27-30, 2016*, pages 560–569. IEEE Computer Society, 2016.

[32] T. He, A. Gkelias, L. Ma, K. K. Leung, A. Swami and D. Towsley. Robust and efficient monitor placement for network tomography in dynamic networks. *IEEE/ACM Transactions on Networking*, 25(3):1732–1745, 2017.

[33] Internet Topology Zoo. University of Adelaide, http://www.topology-zoo.org/ 2018.

[34] T. Kloks and D. Kratsch, Finding All Minimal Separators of a Graph, STACS 94, 11th Annual Symposium on Theoretical Aspects of Computer Science, *Lecture Notes in Computer Science* Vol. 775, pp. 759–768, 1994.

[35] B. Liouville, Sur la connectivité des produits de graphes, *C. R. Acad. Sci. Paris Sér. A-B*, 296:A363–A365, 1978.

[36] L. Ma, T. He, K. K. Leung, A. Swami, and D. Towsley, Inferring link metrics from end-to-end path measurements: Identifiability and monitor placement, *IEEE/ACM Transactions on Networking*, 22(4):1351–1368, 2014.

[37] L. Ma, T. He, A. Swami, D. Towsley, and K. K. Leung, On optimal monitor placement for localizing node failures via network tomography *Performance Evaluation*, 91:16–37, 2015.

[38] L. Ma, T. He, A. Swami, D. Towsley, and K. K. Leung, Network capability in localizing node failures via end-to-end path measurements, *IEEE/ACM Transactions on Networking*, 25(1):434–450, 2017.

[39] L. Ma, T. He, A. Swami, D. Towsley, K. K. Leung, and J. Lowe, Node failure localization via network tomography, in: *Proceedings of the*

2014 Internet Measurement Conference, IMC 2014, Vancouver, BC, Canada, November 5-7, 2014, pages 195–208. ACM, 2014.

[40] L. Pósa, Hamiltonian circuits in random graphs, *Discrete Mathematics*, 14:350–364, 1976.

[41] W. Ren and W. Dong, Robust network tomography: K-identifiability and monitor assignment, In *35th Annual IEEE International Conference on Computer Communications, INFOCOM 2016, San Francisco, CA, USA, April 10-14, 2016*, pages 1–9. IEEE, 2016.

[42] P. Sangha, P.W.H. Wong and M. Zito, Independent Sets in Restricted Line of Sight Networks, In A. Fernández Anta et al., editors, *The 13th International Symposium on Algorithms and Experiments for Wireless Networks, ALGOSENSORS 2017*, volume 10718 of *Lecture Notes in Computer Science*, pages 1–12. Springer Verlag, 2017.

[43] P. Sangha and M. Zito, Finding large independent sets in line of sight networks, In D. Gaur and N. S. Narayanaswamy, editors, *Algorithms and Discrete Applied Mathematics; Third International Conference CALDAM 2017*, volume 10156 of *Lecture Notes in Computer Science*, pages 332–343. Springer Verlag, 2017.

[44] W. Schnyder, Embedding planar graphs on the grid, in: *Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '90, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA. pages 138–148, 1990.

[45] B. W. Schröder, Ordered set: an introduction, *Birkahüser*, 1966.

[46] C. Shangguan and I. Tamo, New Turán exponents for two extremal hypergraph problems, *SIAM Journal on Discrete Mathematics*, 2020.

[47] W. T. Trotter, Combinatorics and Partially Ordered Sets: Dimension Theory, *The John Hopkins University Press*, 1992.

[48] W. T. Trotter and J. I. Moore Jr, The dimension of planar posets, *Journal of Combinatorial Theory, Ser. B*, 22(1):54–67, 1977.

[49] Y. Vardi, Network tomography: Estimating source-destination traffic intensities from link data, *Journal of the American Statistical Association*, 91(433):365–377, 1996.

[50] J. Yáñez and J. Montero, A poset dimension algorithm, *Journal of Algorithms*, 30(1):185–208, 1999.

[51] M. Yannakakis, The complexity of the partial order dimension problem, *SIAM Journal on Algebraic Discrete Methods*, 3(3):351–358, 1982.

Appendix A

A.1 Proof of The Theorem 4.1.4

First we prove that $\mathcal{T}_n$ is not 2-identifiable. Notice that we assume the tree to be line-free (LF) otherwise $\mu(\mathcal{T}_n|\chi) = 0$. Now we have to find two sets W and U of cardinality at most two such that $\mathbb{P}(U)\triangle\mathbb{P}(W) = \emptyset$. Let u be a node in $\mathcal{T}_n$. Since $\mathcal{T}_n$ is monitor-balanced under χ, there exists a neighbour of u, $w \in N(u)$ such that $T^{(u,w)}(w)$ is an input tree. There also exists another neighbour of u, $z \in N(u)$, with $z \neq w$ such that $T^{(u,z)}(z)$ is an output tree. Fix $U = \{w\}$, $W = \{u, w\}$. $\mathbb{P}(U) \subseteq \mathbb{P}(W)$. Moreover each path passing through u and z is also touching w, hence $\mathbb{P}(\{u\}) \subseteq \mathbb{P}(\{w\})$. Therefore $\mathbb{P}(W) \subseteq \mathbb{P}(U)$. Hence $\mathbb{P}(U) = \mathbb{P}(W)$ and $\mathbb{P}(U)\triangle\mathbb{P}(W) = \emptyset$. Thus $\mu(\mathcal{T}_n|\chi) \leq 1$.

Now for the lower bound let u and w be two distinct nodes in $\mathcal{T}_n$. Fix $U = \{u\}$ and $W = \{w\}$. If w and u lie on different (end-to-end) paths, the claim is trivially proved. If w and u lie on the same path p and say that p meets w before u. Let p_w be the subpath of p truncated at node w. Since $\mathcal{T}_n$ is LF, then $\deg(w) \geq 3$. Let $w_1 \in N_{\mathsf{o}}(w)$ be the neighbour of w lying on p. $T^{(w,w_1)}(w_1)$ is obviously an output tree. It is easy to argue that by monitor-balanced assumption there is necessarily another node $w_2 \neq w_1, w_2 \in N_{\mathsf{o}}(w)$ and $T^{(w,w_2)}(w_2)$ is also an output tree. Therefore in $\mathcal{T}_n$ there is a path q from w_2 to an output monitor. This implies that from each node there are at least two different (end-to-end) paths passing through. In other words, the concatenation of p_w with q is a path touching w but not u. Hence $\mathbb{P}(U)\triangle\mathbb{P}(W) \neq \emptyset$ and $\mu(\mathcal{T}_n|\chi) \geq 1$.

□

A.2 Proof of The Theorem 4.2.1

Here we prove the Theorem 4.2.1 for the general case $d > 2$. The upper bound follows from Lemma 3.2.1.

For the lower bound we first consider the following claim:

Claim A.2.1. *Let $z_1 = (i_1, \ldots, i_d), z_2 = (j_1, \ldots, j_d), z_3 = (k_3, \ldots, k_d)$ be three nodes in $\mathcal{H}_{n,d}$ such that z_1 and z_3 are distinct nodes. There exists a simple path from z_1 to z_3 touching z_2.*

Proof. First we consider a d-dimensional hypercube (d-cube) in $\mathcal{H}_{n,d}$ such that all these three nodes are lying on some edges of this d-cube. Then we start from the node that we want to be the origin of our path and move along the edges towards our second node that we want to be touched by our path. After reaching the second node we continue moving along the edges which will lead to the third node that our path terminates at. We then build a path from z_1 to z_3 touching z_2. (Notice that there may exist more than one such path since we have d directions for each node in the corners and more directions for other nodes. Therefore when we are starting our path from the origin node, we can pick any direction and move along the edges towards the second node and then the same to reach the third node. We just need to pick the best direction which would lead to a short simple path). □

Now we have to prove that independently of what nodes form S and T, for any $U, W \subseteq V$ with $U \triangle W \neq \emptyset$ such that $|U|, |W| \leq d-1$, then $\mathbb{P}(U) \triangle \mathbb{P}(W) \neq \emptyset$. Since $U \neq W$, $|U|, |W| \leq d-1$ and $|S|, |T| = d$, then there is at least one $S \notin W$, at least one $T \notin W$ and wlog there exists at least one node $u \in U \setminus W$. By Claim A.2.1 we get a simple path p from S to T passing through u. If this path touches W (say it is touching the nodes $w_1, \ldots, w_i$), then we can avoid it. If the nodes $w_1, \ldots, w_i$ are all internal nodes (not on the borders), in order to avoid them we remove these nodes and all the edges linked to them. Then we have some holes in our grid. By previous observation, after removing $w_1, \ldots, w_i$, at least one node in S and one node in T are in the remaining network and they must be different since we we do not have loops or DLP. By previous claim applied to S, T and u we have an $S - T$ path in $\mathcal{H}_{n,d}$ touching U but not W. Notice that if some parts of the d-cube that we are considering in Claim A.2.1 intersect with our holes then we can move along the borders of our holes as you have seen in the case $d = 2$ (Figure 4.4). If all or some of the nodes $w_1, \ldots, w_i$

are on the borders but u and all other nodes in U are internal nodes, then by the same argument as above we can touch W and avoid U. If $w_1, \ldots, w_i$ and u (maybe some other nodes of U as well) are on the same borders and one of them say u is isolated by W, S and T (for example the node u is at the corner), then we remove u and the edges linked to it and again by the same argument as above we have a path $S - T$ touching W but not U (Notice here that W can not be isolated by the rest of the elements of U since we have at least d directions for each node and $|U| \leq d - 1$. Moreover one node u is already isolated. So for each node in W, $S \notin U$ and $T \notin U$ we have at least 2 directions free and not touched by U. For a better understanding refer to the case $d = 2$ (Figure 4.5)). □