ABSTRACT

Title of dissertation: OPACITY AND STRUCTURAL RESILIENCE IN CYBERPHYSICAL SYSTEMS

Bhaskar Ramasubramanian
Doctor of Philosophy, 2018

Dissertation directed by: Professor Steven I. Marcus
Department of Electrical and Computer Engineering

Cyberphysical systems (CPSs) integrate communication, control, and computing with physical processes. Examples include power systems, water distribution networks, and on a smaller scale, medical devices and home control systems. Since these systems are often controlled over a network, the sharing of information among systems and across geographies makes them vulnerable to attacks carried out (possibly remotely) by malicious adversaries. An attack could be carried out on the physical system, on the computer(s) controlling the system, or on the communication links between the system and the computer. Thus, significant material damage can be caused by an attacker who is able to gain access to the system, and such attacks will often have the consequence of causing widespread disruption to everyday life. Therefore, ensuring the safety of information critical to nominal operation of the system is of utmost importance. This dissertation addresses two problems in the broad area of the Control and Security of Cyberphysical Systems.

First, we present a framework for opacity in CPSs modeled as a discrete-time

linear time-invariant (DT-LTI) system. The current state-of-the-art in this field studies opacity for discrete event systems (DESs) described by regular languages. However, the states in a DES are discrete; in many practical systems, it is common for states (and other system variables) to take continuous values. We define a notion of opacity called *k-initial state opacity (k-ISO)* for such systems. A set of secret states is said to be k-ISO with respect to a set of nonsecret states if the outputs at time k of every trajectory starting from the set of secret states is indistinguishable from the output at time k of some trajectory starting from the set of nonsecret states. Necessary and sufficient conditions to achieve k-ISO are presented in terms of sets of reachable states. Opacity of a given DT-LTI system is shown to be equivalent to the output controllability of a system obeying the same dynamics, but with different initial conditions.

We then study the case where there is more than one adversarial observer, and define several notions of decentralized opacity. These notions of decentralized opacity will depend on whether there is a centralized coordinator or not, and the presence or absence of collusion among the adversaries. We establish conditions for decentralized opacity in terms of sets of reachable states. In the case of colluding adversaries, we derive a condition for non-opacity in terms of the structure of the communication graph.

We extend this work to formulate notions of opacity for discrete-time switched linear systems. A switched system consists of a finite number of subsystems and a rule that orchestrates switching among them. We distinguish between cases when the secret is specified as a set of initial modes, a set of initial states, or a combination

of the two. The novelty of our schemes is in the fact that we place restrictions on: i) the allowed transitions between modes (specified by a directed graph), ii) the number of allowed changes of modes (specified by lengths of paths in the directed graph), and iii) the dwell times in each mode. Each notion of opacity is characterized in terms of allowed switching sequences and sets of reachable states and/ or modes. Finally, we present algorithmic procedures to verify these notions, and provide bounds on their computational complexity.

Second, we study the resilience of CPSs to denial-of-service (DoS) and integrity attacks. The CPS is modeled as a linear structured system, and its resilience to an attack is interpreted in a graph-theoretic framework. The structural systems approach presumes knowledge of only the positions of zero and nonzero entries in the system matrices to infer system properties. This approach is attractive due to the fact that these properties will hold for almost every admissible numerical realization of the system. The structural resilience of the system is characterized in terms of unmatched vertices in maximum matchings of the bipartite graph and connected components of directed graph representations of the system under attack. Further, we establish a condition based on the zero structure of an input matrix that will ensure that the system is structurally resilient to a state feedback integrity attack if it is also resilient to a DoS attack.

Finally, we formulate an extension to the case of switched linear systems, and derive conditions for such systems to be structurally resilient to a DoS attack.

OPACITY AND STRUCTURAL RESILIENCE IN CYBERPHYSICAL SYSTEMS

by

Bhaskar Ramasubramanian

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2018

Advisory Committee:
Professor Steven I. Marcus, Chair/Advisor
Professor Rance Cleaveland
Professor Michael Fu
Professor Charalampos (Babis) Papamanthou
Professor Yasser Shoukry

Acknowledgments

This dissertation marks the culmination of an effort that has lasted the better part of seven years. Many people have contributed towards making this an enjoyable and rewarding journey- this acknowledgment is a small (yet verbose) way of saying, "Thank You."

My advisor, Steve Marcus, gave me the freedom and provided financial support to explore different areas of research at the start of my Ph.D. He encouraged me when I came up with something that looked promising, softened the blow when I ran into a brick wall, and his encyclopedic knowledge of the literature prevented me from taking roads already traveled. When I thought I had settled on a research problem, his probing questions helped refine the problem statement; his attention to detail ensured that I made myself clear while explaining my ideas. He promptly gave feedback on drafts of papers, which meant I did not have to worry about submitting them minutes before a deadline. Steve has not only been a great research advisor, he is also a very good teacher. I got the opportunity to be the TA for an undergraduate Control Systems class that he gave. His ability to explain seemingly difficult concepts at a level that most students were able to follow particularly struck me. The amount of time and effort that he put in to preparing and presenting every lecture, and his responsiveness to student concerns are qualities that I will strive to emulate should I have to teach a course in the future.

Rance Cleaveland has co-directed this dissertation. Several topics in this work have come about as a refinement of his thinking aloud during our meetings. His emphasis on clear exposition has (hopefully) made me better at being able to explain my work to an audience that might not necessarily come from a similar background or share my research interests. He also gave an interesting course in Model Checking. Although this dissertation does not use those tools and techniques, the integration of Formal Methods with Control and Security is a promising area of research that I will be interested in pursuing in the future.

Steve and Rance have been extremely generous with their time during the last six years. Our weekly meetings comprised a healthy combination of reports on research progress and anecdotes from 'back in the day'[1]. They have written several letters of recommendation aiding me in my search for a postdoctoral position, for which I am grateful. Most importantly, they have been very strong voices of reason, encouragement, and support, especially over the past year.

I thank the other members of my committee– Professors Michael Fu, Charalampos (Babis) Papamanthou, and Yasser Shoukry– for reading this dissertation and providing feedback. Profs. Fu and Papamanthou were on my thesis proposal committee as well, and the comments they gave at that stage were especially useful in shaping parts of this document. Prof. Fu also wrote letters of recommendation for me, for which I am grateful. A conversation with Prof. Shoukry before I started applying for a postdoc was useful in terms of providing perspective into how the 'system' works.

I am grateful to Professor John Baras for his mentorship. I was the TA for an intense and really interesting graduate course in Linear Systems that he gave. He

[1] When some of those stories started being retold, it was a sign that it was time to graduate!

exhorted me to apply to postdoc positions, and was extremely helpful in suggesting the names of people and places to apply. I would not have got in touch with as many research groups as I did if it was not for his insistence.

Dr. M. A. Rajan, Dr. M. Girishchandra, and Dr. P. Balamuralidhar hosted me at Tata Consultancy Services Innovation Labs during Summer 2015, which was, research wise, an extremely productive two months. The structural resilience part of this document came out of the time that I spent there.

The competence, helpful nature, and cheerful disposition of the ECE Administrative Staff ensured that my need to focus on administrivia was reduced to a bare minimum. I've lost count of the number of times that I have walked into the offices of Melanie Prange, Bill Churma, Heather Stewart, and Emily Irwin to ask questions about various things– they never turned me away, and answered even the most innocuous of questions with a smile. Vivian Lu ensured that appointment letters at the start of every term were signed and processed in time; Jesse and Carla in the ECE and ISR Business Offices were responsible for the timely processing of travel reimbursements.

While the aforementioned were largely responsible for creating an environment conducive to research, the last few years would not have been as much fun if it were not for friends.

Manish Purohit, Amit Chavan, Kartik Nayak, and Ramakrishna Padmanabhan transformed *5002, Cheyenne Pl* from a house to a home. We made it a habit to have dinner together everyday, and our dinner (and tea-time) conversations covered almost every topic under the sun– including sports, entertainment, technology, science fiction, GMOs, and the scope of the human species! They have been beer aficionados, squash partners, *gyaan* givers, and good listeners (to some really bad jokes and bizarre ideas!). Anshul Sawant was a 'pseudo' member of the household at *5002*, and was privy to many of the conversations above.

Ninad Kelkar and I shared a healthy disdain for several cricket teams and players; the Indian cricket team not winning matches was, for a period of time, a frequent source of mirth.

Sudha Rao, Meethu Malu, Pallabi Ghosh, and Manaswi Saha (together, the *Ladies of Laguna*) have been fellow 'culture vultures'. They have been willing accomplices in dining out, and attending plays and concerts, and have generally been wonderful company.

James Ferlez's friendship has been invaluable. We had similar opinions on what we found funny (Yes Minister, Seinfeld; he introduced me to The Thick of It and Curb Your Enthusiasm) and what was not (certain smells and sounds). On more than one occasion, he volunteered to present at our weekly meetings with Steve and Rance when I felt I did not have much to talk about. His 'two minutes per slide' heuristic has been particularly helpful while preparing slides for a talk.

Praneeth Boda, Devanarayanan P.E., and Prashanth L. A. partook in wide-ranging conversations over long lunches. Praneeth and I have been through the trials and tribulations of graduate school together– he has been a sounding board for some of the topics presented in this document.

Alborz Alavian, James, Tingyue Gan, Sam Huang, Kun Lin, Van Sy Mai, and

Waseem Malik lit up the windowless world of AVW 3182; I will cherish the many conversations, academic and otherwise, that we had.

Marie Chau, Karamatou Y.D., and Kun introduced me to the various culinary delights of the D.C. Area early on in graduate school. Restaurant hopping in D.C. is an exercise I have sustained since then!

Zamira Daw, Peter Fontana, Sam, and Christoph Schulze– Rance's postdoc and students– made for good travel companions to the several annual CMACS project (and other) meetings.

Debdipta Goswami, Dipankar Maity, and Van Sy Mai were amenable to skipping the occasional conference session to explore the streets and alleys of Boston, Seattle, and Melbourne. Van Sy and Dipankar were also part of a fruitful collaboration that started as a course project, and (along with Michael Rotkowitz) eventually became a conference paper.

Quiz sessions every other Friday with Kartik Abinav, Ankan Bansal, Sarthak Chandra, Varun Manjunatha, and Yogarshi Vyas was something to look forward to during the last one year.

Many other friends have made my time in College Park memorable– Kunvar Chokshi, Soham De, Biswadip Dey, Gokul Iyer, Udit Halder, Shashikant Koul, Evripidis Paraskevas, Aneesh Raghavan, Vidya Raju, Saurabh Sahu, Avni Sawant, Neeti Sawant, Sohil Shah, Priyanka Shende, Ganesh Sivaraman, Raviteja Vemulapalli, and Jayanand Vijayan deserve special mention.

The work carried out in this dissertation was supported in part by the National Science Foundation under Grants $CNS-1446665$ and $CMMI-1362023$, and by the Air Force Office of Scientific Research under Grant $FA9550-15-10050$. Support was also provided by the A. James Clark School of Engineering through a Distinguished Graduate Fellowship, and the Electrical and Computer Engineering Department via (multiple) Teaching Assistantships and TA Training and Development Fellowships. Travel support to conferences to present published research was provided in part by the University of Maryland Graduate School through the Goldhaber Travel Award, the Institute for Systems Research through a Graduate Student Travel Award, and the IEEE Control Systems Society via a Student Travel Award. All the above sources of financial support are gratefully acknowledged.

My parents, Chandra and Ramasubramanian, and my brother, Deepak have been my strongest supporters, and an endless source of love and motivation. This dissertation is dedicated to them.

Contents

List of Tables

List of Figures

Chapter 1: Introduction

Cyberphysical systems (CPSs) are complex systems in which the functioning of the physical system is governed by computers that communicate instructions and operational protocols over a network. The presence of a network, which may be wired or wireless, is indicative of the fact that computational resources and bandwidth can also affect the operation of the CPS. CPSs are ubiquitous; examples include power systems, water distribution networks, and on a smaller scale (but no less complex), medical devices and home control systems [1]. While computer-controlled systems are more efficient, the sharing of information among devices and across geographies makes the system vulnerable to attacks. An attack could be carried out on the physical system, on the computer(s) controlling the system, or on the communication links between the system and the computer. Moreover, these attacks could be carried out remotely. Thus, significant material damage can be caused by an attacker who is able to gain access to the system remotely, and such attacks will often have the consequence of causing widespread disruption to everyday life.

The following two examples serve to illustrate the potential damage and disruption that can be caused by an attack on a CPS:

1. In December 2015, an attack was carried out on the power grid in Ukraine, where attackers remotely gained access to circuit breakers which brought several substations offline. They also remotely disabled backup power supplies, and flooded call centers with fake calls, to prevent affected customers from reporting complaints. This left more than $200,000$ people without electricity for several hours. The possible impact of a similar attack on the United States power grid is examined in [2].

2. In an experiment reported in [3], the authors carried out an attack on the Antilock Braking System (ABS) of a vehicle. They developed a spoofer that would inject a spurious magnetic field in order to tamper with measurements of speed sensors located on the wheels of the vehicle. The result was that the ABS did not work as intended because of the incorrect speed reported to it. Further, this particular attack was completely noninvasive, in the sense that it did not require tampering with sensors on the original system.

Several other instances of attacks on CPSs have been documented in the literature [4], [5]. A compilation of potential challenges in securing these systems to such attacks is tabled in [6].

It is very difficult to ensure that a CPS will be immune to every possible attack. However, it is important that all stakeholders actively work towards ensuring that the system is resilient to a large class of attacks, and further, formulate techniques and develop tools in order to make it difficult for an attacker to carry out an attack. The requirements of a system to address security concerns can be given by the *CIA*

Triad, as enumerated in [7]. We restate them here:

1. *Confidentiality* is the ability to keep information from falling into the wrong hands. A lack of confidentiality will result in disclosure of sensitive data. An example of a system where maintaining confidentiality is of interest is in smart grids, where individual users supply data about their energy usage to a utility company, which then decides on the price per unit of electricity and the amount of electricity to be generated, among other things. However, the users would want to keep their individual usage hidden from an eavesdropper who might be able to use this data to determine, for instance, whether the user is at home or not, based on their electricity consumption patterns.

2. *Integrity* involves maintaining accuracy and trustworthiness of data. Measures to ensure integrity include user access controls and file permissions. A lack of integrity will result in *deception*. For a CPS, maintaining integrity will enable it to be resilient to deception attacks carried out on the sensors and actuators.

3. *Availability* ensures that trusted parties will have access to information on demand. Some means to ensure availability of data include safeguards against interruptions when data is being accessed, and backups to ensure redundancy. A lack of availability will result in *denial of service*. A denial of service attack could lead to the blocking of sensing and actuating signals, resulting in a loss of controllability or stability of the system.

If one chooses to focus on the flow of information from the CPS to the attacker [8, 9], to gain illicit access to a CPS (or any other system), a prospective attacker

must be able to extract useful information pertaining to the system, which can then be used by him or her to subvert the operation of the system. Thus, information critical to nominal operation should be safeguarded in a well designed system. This motivation has led researchers to develop approaches for analyzing how *opaque* the system behavior is to an adversary. Opacity is a property that captures whether an intruder, modeled as an adversarial observer, can infer a 'secret' of a system based on its observation of the system behavior. The current state-of-the-art in this area studies opacity within the framework of discrete event systems (DESs) described by regular languages [10, 11]. Techniques from supervisory control have been used to enforce opacity on a system [12, 13] that was not opaque initially. In other words, it was shown that a controller could be designed to disable actions that would lead to the leaking of the secret.

Although this theory is quite rich, a shortcoming is that it only studies the case when the states are discrete (like in a DES). In many practical systems, it is common for the system variables to take values in a continuous domain. This is indeed the case in CPSs like power systems and water distribution networks. To address this, in this dissertation, we model the CPS as a discrete-time linear time-invariant (DT-LTI) system [14] (thus, while time steps are discrete, the state, control, and output variables are real valued). We will use tools from control theory to study opacity for such systems.

A second shortcoming is that a large part of the current literature that studies the security of CPSs assumes complete knowledge of the system parameters, and analyzes the consequences of attacks on these systems. Parameters in CPSs with a

large number of state and measured variables are prone to variations. Conventional methods of analysis based on these models for every possible numerical realization of the system variables might therefore be computationally infeasible. The structural systems approach, introduced by Lin in [15], offers a way out of this conundrum. This technique presumes knowledge of just the zero structures (that is, the *positions* of zero and nonzero entries) of the system matrices to infer system properties. This approach is attractive since these properties will hold for almost every valid numerical realization.

1.1 Related Work

In this section, we summarize prior work in the literature that is relevant to the two broad topics that is the focus of this dissertation.

1.1.1 Literature Review: Opacity

Opacity was first presented as a tool to study cryptographic protocols in [16]. The intruder was modeled as a passive observer who could read messages exchanged between two parties, but could not modify, block, or send a message. The aim of the parties was to exchange secret information without making it obvious to the intruder. A theory of supervisory control for DESs represented by finite state automata (FSA) and regular languages was formulated in [17, 18]. This framework spawned research in many areas including fault diagnosis [19], hybrid systems [20], and robotics [21].

DESs were used to study opacity in [10], which assumed multiple intruders

with different observation maps. Under the assumption that the supervisor could control all events, it was shown that there existed an optimal control that enforced opacity. In the DES framework, the secret could have been specified as a subset of states or a sublanguage of the system. A notion of opacity was formulated for each instance accordingly. Verification of the opacity of a secret specified as a language was presented in [12, 22], while [11, 23, 24] studied the same for secrets specified as states. Language and state based notions of opacity were shown to be equivalent in [25], where algorithms (that were polynomial in the number of states) to transform one notion of opacity to the other were presented. Opacity was compared with detectability and diagnosability of DESs, and other privacy properties like secrecy and anonymity in [26]. A subsequent paper [27] defined opacity for DESs in a decentralized framework with multiple adversaries, each carrying out its own observation of the system.

The enforcement of opacity using techniques from supervisory control was studied in [12, 13]. The authors of [28] formulated an alternate method of opacity enforcement using insertion functions, which are entities that modify the output behavior of the system in order to maintain a secret. A notion of joint opacity was also proposed in this paper, in which a system could have been observed by multiple adversarial observers who share their observations with a coordinator, which then verifies opacity.

1.1.2 Literature Review: The Structural Approach

System- and graph-theoretic conditions were formulated and proved in [14,29] for an attack on a cyberphysical system (modeled as a linear descriptor system subject to unkown inputs) to be undetectable and unidentifiable by monitors. In [30], for a wireless control network modeled as a discrete-time linear time-invariant system, under the assumption that (A, B) was stabilizable and (A, C) was detectable, the authors presented methods to determine a subset of columns $B_I \subset B$, and a subset of rows, $C_J \subset C$ such that (A, B_I) was stabilizable and (A, C_J) was detectable[1].

The success of different kinds of attacks on linear time-invariant (LTI) systems in terms of the ability to ensure or disrupt controllability of a suitably modified LTI system was characterized in [31]. It is this approach that we wish to extend to structured linear systems. Interpreting security properties within this framework will allow for a characterization of resilience to attacks for general classes of CPSs. We note that [31] also modeled classes of attacks using notions from game theory, but we do not provide an analogue in this work.

The structural design of large scale systems was studied in [32]. The input and output matrices were designed to select the smallest number of actuated and sensed variables to ensure structural controllability and observability. The state feedback matrix was then designed to ensure the minimum number of input-output interconnections and such that the closed loop system had no structural fixed modes[2].

[1] Here, A, B, and C are the system, input, and output matrices of a linear time invariant system: $\dot{x} + Ax + Bu$; $y = Cx$.

[2] For the purposes of this dissertation, it is sufficient to understand that the absence of structural

Control selection problems have attracted a lot of attention of late. For an LTI system, given the system matrix A, the *minimal controllability problem* aims to find the sparsest input matrix B, that will ensure that the system described by (A, B) is controllable. In the unconstrained case, this problem was shown to be $NP-$hard in [33]. Interestingly, the authors of [32] showed that the *minimal structural controllability problem* was polynomially solvable. The minimal controllability problem for single input structural systems was studied in [34], which showed that this problem was solvable when a rank condition was satisfied; in the case when no structure was imposed on the system matrix, the problem was solvable with a single nonzero entry in the input matrix. Further, the authors of [35] showed that the *minimum constrained input selection problem* was $NP-$hard. In this problem, given the structures of the system and input matrices, the goal was to determine a minimal set of indices of columns of the input matrix to ensure structural controllability. They also showed that if the system matrix had a certain structure, the minimum dedicated input selection problem was polynomially solvable.

In [36], given the costs of actuating each state, the *minimum cost structural controllability problem* was shown to be polynomially solvable. This work was extended to the constrained case by the authors of [37], and the *minimum cost constrained structural controllability problem* was shown to be $NP-$hard by deriving a reduction from the constrained minimum input selection problem. This problem was polynomially solvable when the system matrix was irreducible or, equivalently, the directed graph of the system was strongly connected. We note that most of the

fixed modes will allow arbitrary placement of the closed loop poles of the system.

recent work only deals with determining the smallest subset of the $[B]$ matrix to ensure controllability of the system. However, the structural controllability of the system can also be influenced by changing the number of connections from controls to states.

The structural controllability of switched linear systems was studied in [38], where the authors used union graphs and colored union graphs to determine conditions that would ensure structural controllability. In particular, a switched system can be controllable even when each of its individual modes is not controllable. The problem of determining the smallest subset of actuators needed to ensure structural controllability of a switched system was studied in [39]. The authors also presented a polynomial algorithm to determine such a subset of actuators. However, the problem of selecting a minimum collection of modes from among a sequence of modes to ensure that the switched system is structurally controllable was shown to be $NP-$hard.

In this dissertation, we will formulate conditions to ensure the structural resilience of a system to an attack in relation to the structural controllability of the system after a subset of its inputs are 'disconnected'.

1.2 Contributions of this Dissertation

1.2.1 Developing a Unified Framework for Opacity in Cyberphysical Systems

The contributions as a result of this work are listed below:

1. For CPSs represented as a discrete-time linear time-invariant system with a single adversarial observer, we define a notion of opacity at a time k called *k-initial state opacity (k-ISO)* [40]. A set of secret states is said to be k-ISO with respect to a set of nonsecret states if the outputs at time k of every trajectory starting from the set of secret states cannot be distinguished from the output at time k of some trajectory starting from the set of nonsecret states. Necessary and sufficient conditions to achieve k-ISO are presented in terms of sets of reachable states. Opacity of a given DT-LTI system is shown to be equivalent to the output controllability of a system obeying the same dynamics, but with different initial conditions.

2. We extend this to the case when there is more than one adversarial observer [41], where we define several notions of decentralized opacity. These notions of decentralized opacity will depend on whether there is a centralized coordinator or not, and the presence or absence of collusion among the adversaries. Conditions for decentralized opacity will be established in terms of sets of reachable states. In the case of colluding adversaries, we derive a condition for *nonopacity* in terms of the structure of the communication graph.

3. Finally, we formulate notions of opacity for switched linear systems (SLSs) [42]. An SLS consists of a finite number of linear subsystems (called modes) and a rule that governs the switching among them. Many practical systems can be modeled as operating in one of several modes, often switching from one mode

of operation to another[3]. We will distinguish between the cases when the secret is specified as an initial mode, an initial state, or a combination of the two, and whether the adversary will observe a mode, a function of the state, or a combination of the two. Constraints will be placed on the modes that the system will be allowed to transition into from a given mode and we will impose bounds on the dwell times in each mode. Moreover, constraints will be imposed on the number of changes of modes before the adversary makes its observation in our definitions of opacity for SLSs. In each case, we will present conditions that will establish that particular notion of opacity. We will also enumerate algorithmic procedures that provide conservative upper bounds on the computational complexity to verify these notions of opacity.

This body of work follows a natural progression, in that we will start by formulating notions of opacity for linear time-invariant systems with a single adversary, extend this to the case of multiple adversaries, and finally combine the DES framework with ours to establish notions of opacity for switched linear systems [46].

1.2.2 Characterizing the Structural Resilience of Cyberphysical Systems to Attacks

For the structural resilience problem, the CPS is modeled as a linear structured system, and structural conditions for an attack to be successful, in terms of

[3]Further, it has been shown that switching control strategies can achieve better control performance than nonswitching strategies. The reader is referred to [43], [44], [45] for an introduction to the design and control of switched systems.

disrupting or obtaining controllability of a (modified) linear structured system are provided. The structural resilience of the system to denial of service (DoS) attacks and integrity attacks is characterized in terms of the structural controllability of an associated linear structured system. Specifically, the contribution in this area is threefold:

1. First, we characterize the structural resilience of the system in terms of unmatched vertices in maximum matchings of the bipartite graph and connected components of the directed graph representations of the system under attack [47].

2. Next, we present conditions under which a system that is already structurally resilient to a DoS attack will also be structurally resilient to a type of integrity attack called a state feedback integrity attack.

3. Finally, we provide extensions to the case of switched linear systems. Switched linear systems are systems that can operate in one of several *modes*, each of which is a linear system, and can switch from one mode of operation to another [48]. We derive graph-theoretic conditions for the structural resilience of such systems to DoS attacks.

1.3 Outline of Dissertation

The remainder of this dissertation is structured as follows:

Chapter 2 gives a review of the notions of opacity studied for discrete event systems, and an introduction to structured linear systems and graph theory. We

will define several terms that will be needed to understand the results presented in subsequent chapters.

Notions of opacity for continuous state systems is the subject of Chapters 3, 4, and 5, where we focus on linear time-invariant systems with a single adversarial observer, linear time-invariant systems with multiple adversarial observers, and switched linear systems respectively. The various notions of opacity are characterized in terms of sets of reachable states.

Chapter 6 presents a characterization of the resilience of a cyberphysical system (CPS) modeled as a linear structured system to denial-of-service (DoS) attacks. The resilience of the system to an attack is interpreted in terms of unmatched vertices in maximum matchings of bipartite graph, and connected components of directed graph representations of the system under attack.

We conclude this dissertation by presenting future directions of research in Chapter 7.

Chapter 2: Preliminaries

This chapter presents a review of the notions of opacity studied for discrete event systems, and an introduction to structured linear systems and graph theory, and defines several terms that will be needed to understand the results in this dissertation.

Notions of opacity for discrete event systems are presented in Section 2.1. A DES is typically modeled as a finite state automaton, and the secret can be specified as a subset of states or a sublanguage of this automaton. The reader is referred to [11, 25, 26] for a more detailed exposition on opacity for discrete event systems.

Section 2.2 provides an introduction to structured linear systems. The structural approach presumes knowledge of only the zero structures of the matrices in a linear time-invariant model of the CPS under consideration. This characterization can be thought of as a representation of how the variables (state, input, and output variables, as the case may be) of the system influence one another. Furthermore, this approach naturally lends itself to representations of the system as directed and bipartite graphs. Section 2.3 is a primer on graph theory. The interested reader is referred to the survey paper [49] for a more detailed exposition and references to

prior work in this area.

2.1 Opacity for Discrete Event Systems

2.1.1 Languages and Automata

Let Σ be an alphabet, and let Σ^* be the set of all strings of elements from Σ of finite length, including the empty string ϵ. A language $\mathcal{L}$ is a subset of the strings of finite length in Σ^*. Let $G = (X, \Sigma, f, X_0)$ be an finite state automaton, where X is a nonempty set of states, $X_0 \subseteq X$ is a nonempty set of initial states, and Σ represents the set of events. $f : X \times \Sigma \to X$ is the (partial) state transition function: given $x, y \in X$ and $\sigma \in \Sigma$, $f(x, \sigma) = y$ if the execution of σ from x takes the system to y. We write $f(x, \sigma)!$ if $f(x, \sigma)$ is a valid transition. The transition function is extended to $f : X \times \Sigma^* \to X$ in the usual recursive way:

$$f(x, \epsilon) := x$$

$$f(x, se) := f(f(x, s), e) \text{ for } s \in \Sigma^*, e \in \Sigma.$$

The language generated by G is $\mathcal{L}(G) := \{s \in \Sigma^* : f(x, s)!\}$, and describes all possible trajectories of the system. Let K_1 and K_2 be sublanguages of $\mathcal{L}(G)$.

Let $P : \Sigma^* \to \Sigma^*$ be a projection map. Then, if a string of events s occurs in the system, an external agent would see $P(s)$. P can be extended from strings to languages as follows: for languages $L, J \subseteq \Sigma^*$, define

$$P(L) = \{t \in \Sigma^* : (\exists s \in L) t = P(s)\}$$

$$P^{-1}(J) = \{t \in \Sigma^* : P(t) \in J\}$$

2.1.2 Notions of Opacity

A secret specification (states, or language) will be *opaque* with respect to a nonsecret specification if every secret execution (will be made clear subsequently) is indistinguishable from a nonsecret execution. The notion of opacity under consideration will depend on how the secret is specified (sublanguage, set of initial states, or set of current states).

Definition 2.1. *K_1 is* strongly language based opaque (LBO) *with respect to K_2 and P if for every trajectory in K_1, there exists a trajectory in K_2 that 'looks' the same under P, i.e. $K_1 \subseteq P^{-1}(P(K_2))$.*

Definition 2.2. *K_1 is* weakly LBO *with respect to K_2 and P if there exists a trajectory in K_1 that is confused with some trajectory in K_2, under P, i.e. $K_1 \cap P^{-1}(P(K_2)) \neq \phi$.*

Definition 2.3. *Given G with $X_s, X_{ns} \subseteq X_0$, and P, X_s is* initial state opaque (ISO) *with respect to X_{ns} and P if for every $i \in X_s$ and every $t \in L(G, i)$ such that $f(i, t)$ is defined, there exists $j \in X_{ns}$ and $t' \in L(G, j)$ such that $f(j, t')$ is defined and $P(t) = P(t')$.*

Definition 2.4. *Given G with $X_s, X_{ns} \subseteq X$, and P, X_s is* current state opaque (CSO) *w.r.t. X_{ns} and P if for every $i \in X_0$ and $t \in L(G)$ such that $f(i, t) \in X_s$, there exists $j \in X_0$ and $t' \in L(G)$ such that $f(j, t') \in X_{ns}$ and $P(t) = P(t')$.*

These state-based and language-based definitions are essentially equivalent, since it has been shown that there exist algorithms polynomial in the number of

states that relate any pair of the notions of opacity [25].

2.1.3 Examples

Example 2.5. *[25] Consider the FSA G shown in Figure (2.1). Let the set of observable events be given by $\Sigma_o = \{a, b, c\}$. It is language based opaque when $L_s = \{abd\}$ and $L_{ns} = \{abcc^*d, adb\}$ because whenever the intruder sees $P(L_s) = \{ab\}$, it is not sure whether the string abd or the string adb has been executed. Notice that this system is not LBO if $L_s = \{abcd\}$ and $L_{ns} = \{adb\}$. No string in L_{ns} appears the same as the secret string abcd.*

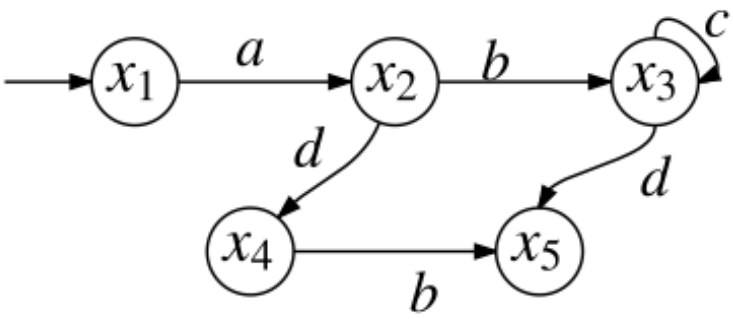


Figure 2.1: Language Based Opacity

Example 2.6. *[23]*

Consider the FSA G in Figure (2.2), with $\Sigma_o = \{a, b\}$, $X_s = \{x_3\}$ and $X_{ns} = X \setminus X_s$. X_s is intial state opaque with respect to X_{ns} because for every string s starting from x_3, there is another string ϵs starting from x_1, that looks the same. However, ISO does not hold if $X_s = \{x_1\}$. In this case, whenever the intruder sees the string aa, it is sure that the system started from X_s (i.e., no other initial state

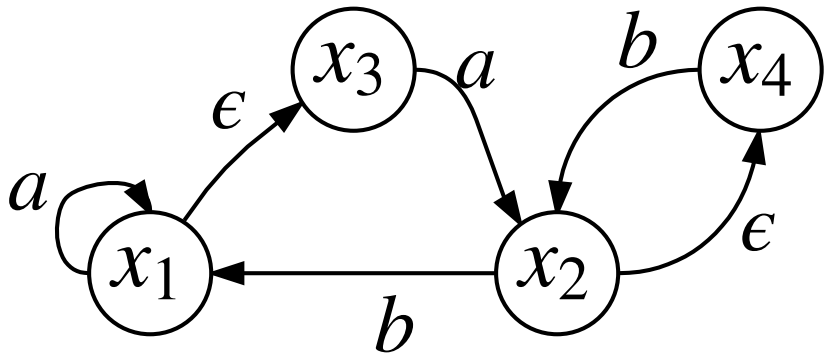


Figure 2.2: Initial State Opacity

can generate a string that appears the same as aa).

2.2 Structured Linear Systems

Consider the linear time-invariant system

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^p$, $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times p}$.

Definition 2.7. *The system in Equation (2.1) is said to be* controllable *if for every initial state $x(0) = x_0$ and final state $x(t_f) = x_f$, there exists an input $u(\cdot)$ on $[0, t_f]$ that transfers the system from x_0 to x_f.*

Verifying the controllability of an LTI system of the form in Equation (2.1) where the states and inputs are defined on finite dimensional vector spaces is equivalent to checking a matrix rank condition, as stated in the following result.

Theorem 2.8. *[50] The system in Equation (2.1) is controllable if and only if* $rank(\begin{pmatrix} B & AB & \dots & A^{n-1}B \end{pmatrix}) = n$.

The notions of structured linear systems and structural controllability were introduced by Lin in [15]. This framework assumes knowledge of only the zero stuctures, $[A] \in \{0, *\}^{n \times n}$ and $[B] \in \{0, *\}^{n \times p}$, of A and B respectively. That is, every entry in $[A]$ and $[B]$ is either a *fixed* zero or a *free* parameter. $[A]$ and $[B]$ are called *structured matrices*.

The rows and columns of $[A]$ indicate how the states of the system influence one another. A nonzero entry $a_{ij} \in [A]$ indicates that the j^{th} component of the state vector, x_j, influences changes in the i^{th} component, x_i (the j^{th} and i^{th} entries in the state vector of dimension n). The rows and columns of $[B]$ indicate how inputs to the system influence the states. In this case, a nonzero entry $b_{ij} \in [B]$ indicates that a change in x_i is influenced by the input u_j (the j^{th} entry in the input vector of dimension p). A zero entry in either case would imply the lack of an interconnection between corresponding state and/ or input variables.

The reader is encouraged to think of the structured representation of a system in the following way:

Example 2.9. *Consider a symmetric structured matrix* $[H] \in \{0, *\}^{n \times n}$ *that is representative of a power system. The dimension of* $[H]$, n, *is indicative of the number of components in the system (generators, transformers, loads). A free parameter* $h_{ij} = *$ *signifies that there is a wire connecting components* i *and* j, *with the direction of current through the wire from* j *to* i. *Likewise, a fixed zero entry in* $[H]$

corresponds to the absence of a wire between the respective components. $h_{ij} = *$ *is an indicator of the fact that changes in the numerical value of a parameter associated with component* j *influences changes in the numerical value of a parameter associated with component* i*. This parameter could be the current flowing through the component, or the voltage drop across the component, and is not precluded from being set to (the numerical value) zero. For example, when two purely resistive loads are connected to each other,* $h_{ij} = h_{ji} = *$*; this free parameter can be assumed to take the numerical value* 0 *when both these loads are isolated from a source.*

Remark 2.10. *In the sequel, the components of the* state (input) vectors *will correspond to* state (input) vertices *in a directed graph. As we will describe in the next part of this section, the edges in this graph will be determined by the* $[A]$ *and* $[B]$ *matrices.*

A matrix $H \in \mathbb{R}^{m \times n}$ with the same zero structure as the structured matrix $[H] \in \{0, *\}^{m \times n}$ is called an admissible numerical realization of $[H]$.

Definition 2.11. $([A], [B])$ *is* structurally controllable *if there exists an admissible numerical realization* (A, B) *that is controllable.*

Remark 2.12. *If* $([A], [B])$ *is structurally controllable, then almost every admissible numerical realization will be controllable*[1]*.*

[1]Some authors refer to such a system as *generically controllable* [49]

2.3 Graph Theory

Directed graphs (digraphs) provide an elegant means to represent linear structured systems [32]. Properties of the system such as controllability and observability can be inferred from a digraph associated with the system, and independently of numerical values of parameters. This makes it an attractive tool to study large scale, complex systems, on which performing computations using numerical values of variables will invariably be costly. Consider the linear structured system

$$\dot{x}(t) = [A]x(t) + [B]u(t) \tag{2.2}$$

where, $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^p$, $[A] \in \{0, *\}^{n \times n}$ and $[B] \in \{0, *\}^{n \times p}$.

The *directed graph* of the structured system is $\mathcal{D} = (\mathcal{V}, \mathcal{E})$, where:

- $\mathcal{V} = \{u_1, \ldots, u_m, x_1, \ldots, x_n\} := \{\mathcal{U}, \mathcal{X}\}$;
- $\mathcal{E} = \mathcal{E}_A \cup \mathcal{E}_B$, where $\mathcal{E}_A = \{(x_j, x_i) | [A]_{ij} \neq 0\}$, $\mathcal{E}_B = \{(u_j, x_i) | [B]_{ij} \neq 0\}$.

A sequence of directed edges $\{(v_1, v_2), (v_2, v_3), \ldots, (v_{k-1}, v_k)\}$ is a *simple path* from v_1 to v_k if the vertices $v_1, \ldots, v_k$ are all distinct. The simple path described above, with an additional edge, (v_k, v_1), or a vertex with a self loop, is called a *cycle*. A vertex w_2 is *reachable* from another vertex w_1 if there exists a simple path from w_1 to w_2. Let $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathcal{V}$. Two paths from $\mathcal{V}_1$ to $\mathcal{V}_2$ are *disjoint* if they consist of disjoint sets of vertices. A set of v mutually disjoint and simple paths from $\mathcal{V}_1$ to $\mathcal{V}_2$ is a *linking* of size v from $\mathcal{V}_1$ to $\mathcal{V}_2$. A *cycle family* is a set of mutually disjoint cycles. A $\mathcal{U}-$*rooted path* is a simple path with source vertex in $\mathcal{U}$. A $\mathcal{U}-$*rooted path family* is a set of mutually disjoint $\mathcal{U}-$rooted paths.

A digraph $\mathcal{D}_s = (\mathcal{V}_s, \mathcal{E}_s)$ is a *subgraph* of $\mathcal{D}$ if $\mathcal{V}_s \subseteq \mathcal{V}$ and $\mathcal{E}_s \subseteq \mathcal{E}$. If $\mathcal{V}_s = \mathcal{V}$, then $\mathcal{D}_s$ is said to *span* $\mathcal{D}$. A subgraph $\mathcal{D}_s$ satisfying a property P is *maximal* if there is no other subgraph $\mathcal{D}_{s'}$ such that $\mathcal{D}_s$ is a strict subgraph[2] of $\mathcal{D}_{s'}$ and property P holds for $\mathcal{D}_{s'}$.

$\mathcal{D}$ is *strongly connected* if there is a simple path from each vertex to every other vertex in the graph. A *strongly connected component (SCC)* is a maximal subgraph $\mathcal{D}_S$, of $\mathcal{D}$, such that $\mathcal{D}_S$ is strongly connected. With SCCs as supernodes (an agglomeration of vertices of the graph), one can generate a *directed acyclic graph (DAG)* in which each supernode corresponds to an SCC, and there exists a directed edge between two SCCs if and only if there exists a directed edge connecting vertices in the SCCs in the original digraph. An SCC is *linked* if it has at least one incoming (outgoing) edge to (from) its vertices from (to) vertices of another SCC. An SCC is *non top (bottom) linked* if it has no incoming (outgoing) edges to (from) its vertices from (to) vertices of another SCC[3].

A *bipartite graph* is a graph whose vertices can be divided into two disjoint sets $\mathcal{V}_1$ and $\mathcal{V}_2$ such that every edge in the graph is from a vertex in $\mathcal{V}_1$ to a vertex in $\mathcal{V}_2$, or from a vertex in $\mathcal{V}_2$ to a vertex in $\mathcal{V}_1$. The bipartite graph is denoted $\mathcal{B}(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E}_{\mathcal{V}_1,\mathcal{V}_2})$. In this dissertation, we will restrict our discussion to bipartite graphs in which all edges are directed from $\mathcal{V}_1$ to $\mathcal{V}_2$, that is, $\mathcal{E}_{\mathcal{V}_1,\mathcal{V}_2} \subset \{(v_1, v_2) | v_1 \in \mathcal{V}_1, v_2 \in \mathcal{V}_2\}$. $\mathcal{B}(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E}_{\mathcal{V}_1,\mathcal{V}_2})$ can also be associated with a matrix H with $|\mathcal{V}_1|$ columns and $|\mathcal{V}_2|$

[2] A subgraph is *strict* if it is the case that at least one of $\mathcal{V}_s \subset \mathcal{V}$ or $\mathcal{E}_s \subset \mathcal{E}$ holds.

[3] Non top (bottom) linked SCCs are called *source (sink)* SCCs in the graph theory literature. In this document, however, we will use the terminology from [32].

rows, with $\mathcal{E}_{\mathcal{V}_1,\mathcal{V}_2} = \{(v_{1_j}, v_{2_i}) : [H]_{ij} \neq 0\}$. Given $\mathcal{B}(\mathcal{V}_1, \mathcal{V}_2, \mathcal{E}_{\mathcal{V}_1,\mathcal{V}_2})$, a *matching* is a subset of edges that do not share vertices. A *maximum matching* is a matching that has the largest number of edges. Vertices not belonging to a maximum matching are called *unmatched*. An unmatched vertex $v_2 \in \mathcal{V}_2$ (respectively, $v_1 \in \mathcal{V}_1$) is called a *right unmatched vertex* (*left unmatched vertex*). A *perfect matching* is a maximum matching with no unmatched vertices.

The *bipartite graph associated with a directed graph* $\mathcal{D}(\mathcal{V}, \mathcal{E})$ is constructed in the following way [51]: to each $v_i \in \mathcal{V}$, we associate two vertices u_i and w_i. There is a directed edge from u_i to w_j in the new graph if and only if there is an edge from v_i to v_j in $\mathcal{D}(\mathcal{V}, \mathcal{E})$. We abuse notation by using $\mathcal{B}(\mathcal{V}, \mathcal{V}, \mathcal{E})$ to denote the bipartite graph associated with $\mathcal{D}(\mathcal{V}, \mathcal{E})$.

The following example will help in making the preceding discussion clear.

Example 2.13. *Figure (2.3) shows the directed graph and bipartite graph representations corresponding to the system matrix* $[A]$ *given below:*

$$[A] = \begin{bmatrix} 0 & 0 & * & 0 & 0 & 0 & 0 \\ * & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & * & 0 & 0 & 0 & 0 & 0 \\ 0 & * & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & * & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * & * & 0 \end{bmatrix}$$

The strongly connected components of the directed graph, $\mathcal{D}([A])$*, are the ver-*

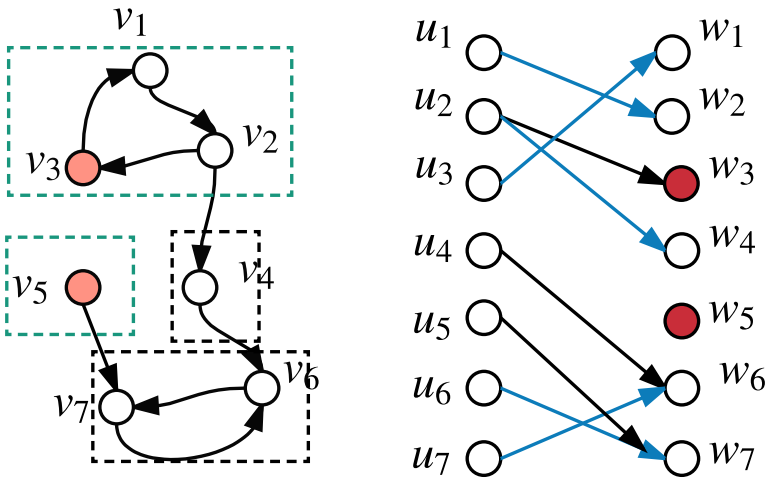


Figure 2.3: Structured system of Example 2.13 as a graph

tices within each dotted box. The dotted boxes in green (comprising the vertex (v_5) *and the vertices* (v_1, v_2, v_3)*) represent the non top-linked SCCs. The bipartite graph representation,* $\mathcal{B}([A])$ *is got by duplicating each vertex of the directed graph, and the edges are determined by the edges in* $\mathcal{D}([A])$*. The edges of* $\mathcal{B}([A])$ *in blue form a maximum matching. Removing the vertices that are incident on edges in the maximum matching, we see that* w_3 *and* w_5 *are right unmatched vertices. Notice that this maximum matching is not unique. Another maximum matching could be got by removing the edge* $(u_2 \to w_4)$ *from the previous maximum matching and adding the edge* $(u_2 \to w_3)$*. The right unmatched vertices of this maximum matching will be* w_4 *and* w_5*.*

2.4 Scope of this Dissertation

This chapter was a means to bring the reader up to speed on the background material relevant to the susbequent chapters. It is evident that the treatment of opacity in the literature only considers a very restricted class of systems, namely, discrete event systems. One of the goals of this dissertation is to define and analyze notions of opacity that might be applicable to larger classes of systems that might comprise general cyberphysical systems. The second goal is to leverage the tools and techniques of the structural approach to characterize the resilience of a cyberphysical system to denial-of-service attacks. We will then extend these results to other types of attacks, and more general structured systems.

Chapter 3: Opacity for Linear Systems: The Single Adversary Case

Although the presentation of opacity for discrete event systems is well-motivated and elegant, a shortcoming of the framework is that it only addresses the case when the states of the system are discrete. The states in CPSs like power systems and water networks are typically real valued. It is this gap that we seek to bridge in this dissertation, by formulating notions of opacity for continuous state systems. The system is modeled as a discrete-time linear time-invariant system. Therefore, while the time steps are discrete, the state, input and output variables are real valued. We use tools from control theory to study opacity for such systems.

We define a notion of opacity at a time k called *k-initial state opacity (k-ISO)* in Section 3.1. A set of secret states is said to be k-ISO with respect to a set of nonsecret states if the outputs at time k of every trajectory starting from the set of secret states can not be distinguished from the output at time k of some trajectory starting from the set of nonsecret states. Necessary and sufficient conditions to achieve k-ISO are presented in terms of sets of reachable states in Section 3.2. Section 3.3 studies k-ISO under unions and intersections of sets of states. Opacity of a given DT-LTI system is shown to be equivalent to the output controllability of a system obeying the same dynamics, but with different initial conditions in Section

3.4. Finally, the necessity of indistinguishability of outputs in the definition of k-ISO is relaxed, and we define a notion of ϵ-opacity in Section 3.5.

3.1 Opacity for LTI Systems

Consider the system:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) \\ x(0) &= x_0 \in X_0 \\ y(t) &= Cx(t) \end{aligned} \tag{3.1}$$

where $x \in \mathbb{R}^n, u \in \mathbb{R}^m, y \in \mathbb{R}^p$, and A, B, C are matrices of appropriate dimensions containing real entries.

Let $\mathcal{K}$ be a set of positive integers, corresponding to the instants of time at which the adversary makes an observation of the system. The subscript s (ns), when appended to the states, inputs, and outputs, will correspond to trajectories that start from the set of initial secret (nonsecret) states. The adversary is assumed to have knowledge of the initial sets of secret and nonsecret states, X_s and X_{ns}, the system model (A, B), and its own observation map C. Further, we assume that it has unlimited computing power, in that it will be able to compute the sets of reachable states at time k. Its goal is to deduce, on the basis of observing the system at times $k \in \mathcal{K}$, whether the system started from a state in X_s or not.

Definition 3.1. *For the system (3.1), given $X_s, X_{ns} \subset X_0$ and $k \in \mathcal{K}$, X_s is* strongly k-initial state opaque (k-ISO) *with respect to X_{ns} if for every $x_s(0) \in X_s$*

and for every sequence of admissible controls $u_s(0), \ldots, u_s(k-1)$*, there exist an* $x_{ns}(0) \in X_{ns}$ *and a sequence of admissible controls* $u_{ns}(0), \ldots, u_{ns}(k-1)$ *such that* $y_s(k) = y_{ns}(k)$.

X_s *is* strongly $\mathcal{K}$-ISO *with respect to* X_{ns} *if* X_s *is strongly* k*-ISO with respect to* X_{ns} *for all* $k \in \mathcal{K}$.

This means that starting from any secret state and applying any sequence of k admissible controls (corresponding to the instants the adversary makes an observation), the system will reach a state whose observation to the adversary will be indistinguishable from the observation of a state that can be reached by the application of an admissible control sequence of length k, starting from some nonsecret state. While this notion calls for **every** state in the set of initial secret states to be indistinguishable (after some time k) from some state in the initial set of nonsecret states, the following definition relaxes this requirement.

Definition 3.2. *For the system (3.1), given* $X_s, X_{ns} \subset X_0$ *and* $k \in \mathcal{K}$, X_s *is* weakly k-ISO *with respect to* X_{ns} *if for some* $x_s(0) \in X_s$ *and for some sequence of admissible controls* $u_s(0), \ldots, u_s(k-1)$*, there exist an* $x_{ns}(0) \in X_{ns}$ *and a sequence of admissible controls* $u_{ns}(0), \ldots, u_{ns}(k-1)$ *such that* $y_s(k) = y_{ns}(k)$.

X_s *is* weakly $\mathcal{K}$-ISO *with respect to* X_{ns} *if* X_s *is weakly* k*-ISO with respect to* X_{ns} *for all* $k \in \mathcal{K}$.

These definitions of opacity for LTI systems is different from familiar definitions of observability. The observability problem aims to determine the initial state $x(0)$, given the entire output and control histories. Here, however, the adversary

aims to determine $x(0)$ via access to only snapshots of the output and the set of possible controls. The small number of observations of the system is motivated by the following:

1. the adversary might not want to reveal its presence.

2. the adversary might not have the resources to continuously monitor the system.

Throughout this document, the set $\mathcal{K}$ is arbitrary.

Our formulation is also different from definitions of opacity in the DES literature. In those cases, the observation of the entire secret trajectory had to coincide with that of a nonsecret trajectory. We only need that the secret and nonsecret outputs at time k coincide. k-ISO also differs from the notion of k-step opacity proposed in [52]. In their formulation, k-step opacity was achieved when the adversary did not know if the system entered a secret state in k previous steps. We require that the ambiguity exist only at time k. It will subsequently become evident that an additional requirement to our conditions for k-ISO will also establish k-step opacity.

Finally, k-ISO is also different from the notion of simulation relations between dynamical systems [53]. Simulation relations typically verify the 'equality' of two systems governed by different dynamics. In our framework, however, we try to identify equivalence classes of outputs at time k. Opacity is deemed to have been achieved if the system starting from two disjoint sets of states at time 0 reaches the same equivalence class of outputs at time k.

Example 3.3. *The illustration in Figure (3.1) will be useful to motivate k-ISO. Consider the problem of a bank needing to transfer money from its offices to an*

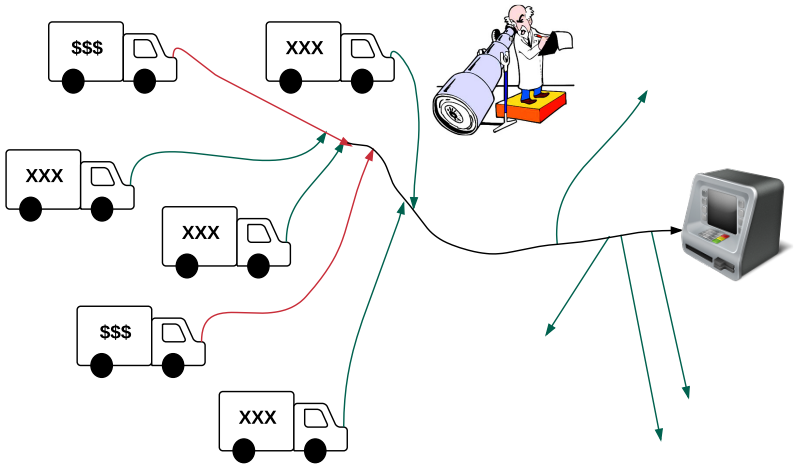


Figure 3.1: k-ISO Motivation: ATM Money Transfer

ATM machine.

One way for the bank to do this in a 'secure' manner would be to equip the truck with the best defenses that money can buy. However, this might not be a cost-effective solution, since customizations might be very expensive, and need to be continuously updated to stay ahead of potential attackers. An alternative approach would be for the bank to deploy several identical trucks, only some of which carry money. This is a reasonable strategy for the bank to adopt, under the assumption that the cost of carrying out an attack on a truck is very high. The motion of a truck can be represented as a state space equation with the position and velocity of the truck as the states, and acceleration as the input. Then, assuming unit mass,

and unit sampling interval, a discrete-time representation of the system is:

$$\begin{bmatrix} p(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} a(k)$$

$$y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} p(k) \\ v(k) \end{bmatrix}$$

The position of the truck at a time k is given by:

$$p(k) = p(0) + kv(0) + \sum_{i=0}^{k-1} (k - i - 0.5)a(i)$$

Let the locations at which the money is loaded (represented by $\$\$\$$) comprise the set of secret states (X_s), and the initial locations of the other trucks (represented by XXX) comprise the set of nonsecret states (X_{ns}). Then, if an adversary observes $p(k)$ at some time k, X_s will be k-ISO with respect to X_{ns} if it cannot determine whether the truck started from a location from where money was loaded into it. That is, for every possible location (at time 0) at which money was loaded, there is a corresponding location (at time 0) where there was no money loaded such that the positions of the trucks which start from these locations are the same at time k.

3.2 Opacity and Reachable Sets of States

The adversary has complete knowledge of the system model, and the sets of initial secret and nonsecret states. However, it does not know the exact control sequence applied in the time interval $[0, k]$; it only has knowledge of the sets of allowed inputs that can be applied. In this light, a possible means of checking that opacity holds is by relating it to reachability. In this section, we present necessary

and sufficient conditions to establish k-ISO in terms of sets of reachable states of the system.

Let $U_s^k := \{u_s(0), \ldots, u_s(k-1)\}$ and $U_{ns}^k := \{u_{ns}(0), \ldots, u_{ns}(k-1)\}$. Let $X_s(k)$ and $X_{ns}(k)$ denote the sets of states reachable in k steps, starting at time 0 from *nonempty* sets X_s and X_{ns} respectively. That is,

$$X_s(k) = \bigcup_{x_0 \in X_s} \bigcup_{U_s^k} \{x : x(i+1) = Ax(i) + Bu(i), \forall i < k\} \tag{3.2}$$

$$X_{ns}(k) = \bigcup_{x_0 \in X_{ns}} \bigcup_{U_{ns}^k} \{x : x(i+1) = Ax(i) + Bu(i), \forall i < k\} \tag{3.3}$$

Theorem 3.4. *The following hold:*

1. *X_s is strongly k-ISO with respect to X_{ns} if and only if $CX_s(k) \subseteq CX_{ns}(k)$.*

2. *X_s is strongly $\mathcal{K}$-ISO with respect to X_{ns} if and only if $CX_s(k) \subseteq CX_{ns}(k)$ for all $k \in \mathcal{K}$.*

Proof. First, let strong k-ISO hold. Then, for all $x_s(0) \in X_s$, and all $\{u_s(\cdot)\}_0^{k-1}$, there exist $x_{ns}(0) \in X_{ns}$ and $\{u_{ns}(\cdot)\}_0^{k-1}$ such that $y_s(k) = y_{ns}(k)$. Now, starting from X_s (respectively X_{ns}), and applying k admissible controls, one reaches a state in $X_s(k)$ ($X_{ns}(k)$). Therefore, k-ISO ensures that for every $x_s(k) \in X_s(k)$, there exists $x_{ns}(k) \in X_{ns}(k)$, such that $y_s(k) = y_{ns}(k)$. This gives $CX_s(k) \subseteq CX_{ns}(k)$.

Next, let $CX_s(k) \subseteq CX_{ns}(k)$. This means for every $x_s(k) \in X_s(k)$, there exists $x_{ns}(k) \in X_{ns}(k)$, such that $y_s(k) = y_{ns}(k)$. Since $X_s(k)$ and $X_{ns}(k)$ are the sets of reachable states starting from X_s and X_{ns} respectively, the previous sentence translates to: for every $x_s(0) \in X_s$ and every $\{u_s(\cdot)\}_0^{k-1}$, there exist $x_{ns}(0) \in X_{ns}$ and $\{u_{ns}(\cdot)\}_0^{k-1}$ such that $y_s(k) = y_{ns}(k)$. This, by definition, is strong k-ISO.

The second statement of the theorem easily follows by extending the previous argument to all $k \in \mathcal{K}$. □

Remark 3.5. *This result can be easily extended to verify k-step opacity if* $CX_s(k) \subseteq CX_{ns}(k)$ *for all* $k \in \mathcal{K} := \{m, m-1, \ldots, m-k+1\}$ *for some positive integer* m.

Remark 3.6. $X_s(k) \subseteq X_{ns}(k)$ *is only a sufficient condition for* X_s *to be strongly k-ISO with respect to* X_{ns}. *To see that this condition is not necessary, let* $C = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$, *and* $X_s(k) = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T$ *and* $X_{ns}(k) = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T$. *Then,* $CX_s(k) = CX_{ns}(k)$, *establishing k-ISO, even though* $X_s(k) \not\subseteq X_{ns}(k)$.

Similar results hold for weak k-ISO. The proofs follow identically.

Theorem 3.7. *The following hold:*

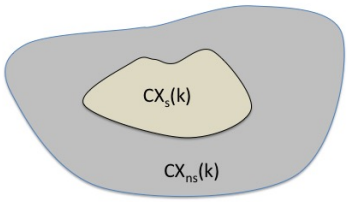
1. X_s *is weakly k-ISO with respect to* X_{ns} *if and only if* $CX_s(k) \cap CX_{ns}(k) \neq \phi$.
2. X_s *is weakly* $\mathcal{K}$*-ISO with respect to* X_{ns} *if and only if* $CX_s(k) \cap CX_{ns}(k) \neq \phi$ *for all* $k \in \mathcal{K}$.

Remark 3.8. $X_s(k) \cap X_{ns}(k) \neq \phi$ *is a sufficient condition for* X_s *to be weakly k-ISO with respect to* X_{ns}.

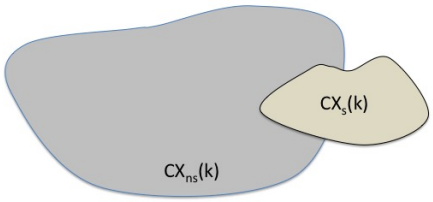
Figures 3.2a, 3.2b, and 3.2c respectively show the representations of strong k-ISO, weak k-ISO, and non-opacity in terms of sets of reachable states at time k.

3.3 k-ISO Under Set Operations

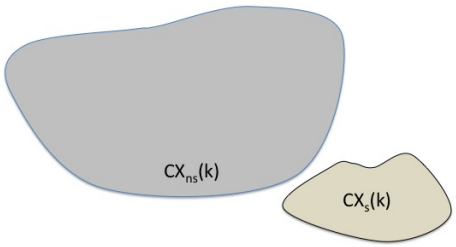
Properties of k-ISO are studied under unions and intersections. The properties verified will be for strong k-ISO, unless otherwise mentioned. Let X denote the set



(a)



(b)



(c)

Figure 3.2: Representations of strong, weak, and non-opacity in terms of sets of reachable states

of initial states, and $X(k)$ be the set of states reachable in k steps, starting from X at time 0.

We first study the effect of the set union operation on k-ISO. Lemmas 3.9 and 3.10 establish basic results on sets of reachable states and outputs under set union.

Lemma 3.9. *Given sets of initial states $X_1, X_2, \cdots \subseteq X$, the reachable set in k steps of their union is equal to the union of the reachable sets in k steps of each set of initial states. That is, $(\bigcup_i X_i)(k) = \bigcup_i X_i(k)$.*

Proof.

$$
\begin{aligned}
& x \in (\bigcup_i X_i)(k) \\
\Leftrightarrow & \exists x_0 \in (\bigcup_i X_i), \exists \{u(\cdot)\}, \ (3.1) \text{ holds } \forall i < k, x(k) = x \\
\Leftrightarrow & [(\exists x_0 \in X_1 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_1(k))] \vee \\
& [(\exists x_0 \in X_2 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_2(k))] \vee \ldots \\
\Leftrightarrow & x \in \bigcup_i X_i(k)
\end{aligned}
$$

□

Lemma 3.10. *Given $X_1, X_2, \cdots \subseteq X$ and $C : \mathbb{R}^n \to \mathbb{R}^m$, $C(\bigcup_i X_i)(k) = \bigcup_i CX_i(k)$.*

Proof.

$$
\begin{aligned}
& y \in C(\bigcup_i X_i)(k) \\
\Leftrightarrow & \exists x \in (\bigcup_i X_i)(k) \text{ such that } y = Cx \\
\Leftrightarrow & \exists x \in \bigcup_i X_i(k) \text{ such that } y = Cx \\
\Leftrightarrow & (y = Cx \wedge x \in X_1(k)) \vee (y = Cx \wedge x \in X_2(k)) \vee \ldots \\
\Leftrightarrow & (y \in CX_1(k)) \vee (y \in CX_2(k)) \vee \ldots \\
\Leftrightarrow & y \in \bigcup_i CX_i(k)
\end{aligned}
$$

□

Lemmas 3.9 and 3.10 are used to study k-ISO under set union, as stated in Theorems 3.11 and 3.12.

Theorem 3.11. *If X_{s_i} is k-ISO with respect to X_{ns} for each i, then $\bigcup_i X_{s_i}$ is k-ISO with respect to X_{ns}.*

Proof.

$$
\begin{aligned}
& X_{s_i} \; k - \text{ISO w.r.t. } X_{ns} \forall i \\
\Leftrightarrow & CX_{s_i}(k) \subseteq CX_{ns}(k) \forall i \\
\Leftrightarrow & \bigcup_i CX_{s_i}(k) \subseteq CX_{ns}(k) \\
\Leftrightarrow & C(\bigcup_i X_{s_i}(k)) \subseteq CX_{ns}(k) \\
\Leftrightarrow & \bigcup_i X_{s_i} \text{ is } k - \text{ISO w.r.t. } X_{ns}
\end{aligned}
$$

□

Theorem 3.12. *If X_s is k-ISO w.r.t. X_{ns_i} for each i, then X_s is k-ISO w.r.t. $\bigcup_i X_{ns_i}$*

Proof.

$$X_s \ k-\text{ISO w.r.t. } X_{ns_i} \forall i$$

$$\Leftrightarrow CX_s(k) \subseteq CX_{ns_i}(k) \forall i$$

$$\Leftrightarrow CX_s(k) \subseteq \bigcup_i CX_{ns_i}(k)$$

$$\Leftrightarrow CX_s(k) \subseteq C(\bigcup_i X_{ns_i}(k))$$

$$\Leftrightarrow X_s \ k-\text{ISO w.r.t. } \bigcup_i X_{ns_i}$$

□

We now turn our attention to k-ISO under set intersection. Lemmas 3.13 and 3.14 establish basic results on sets of reachable states and outputs under set intersection. These results will be useful while studying k-ISO under set intersection, as we will show in Theorems 3.16 and 3.17.

Lemma 3.13. *Given sets of initial states $X_1, X_2, \cdots \subseteq X$, the reachable set in k steps of the intersection of the sets of initial states is contained in the intersection of the reachable sets in k steps of each set of initial states. That is, $(\bigcap_i X_i)(k) \subseteq \bigcap_i X_i(k)$.*

Proof.

$$
\begin{aligned}
& x \in (\bigcap_i X_i)(k) \\
\Rightarrow & \exists x_0 \in (\bigcap_i X_i), \exists \{u(\cdot)\},\ (3.1) \text{ holds } \forall i < k, x(k) = x \\
\Rightarrow & [(\exists x_0 \in X_1 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_1(k))] \wedge \\
& [(\exists x_0 \in X_2 \wedge \exists \{u(\cdot)\}) \text{ s.t. } (x \in X_2(k))] \wedge \dots \\
\Leftrightarrow & x \in \bigcap_i X_i(k)
\end{aligned}
$$

□

Lemma 3.14. *Given* $X_1, X_2, \dots \subseteq X$ *and* $C : \mathbb{R}^n \to \mathbb{R}^m$, $C(\bigcap_i X_i)(k) \subseteq \bigcap_i CX_i(k)$.

Proof.

$$
\begin{aligned}
& y \in C(\bigcap_i X_i)(k) \\
\Leftrightarrow & \exists x \in (\bigcap_i X_i)(k) \text{ such that } y = Cx \\
\Rightarrow & \exists x \in \bigcap_i X_i(k) \text{ such that } y = Cx \\
\Leftrightarrow & (y = Cx \wedge x \in X_1(k)) \wedge (y = Cx \wedge x \in X_2(k)) \wedge \dots \\
\Leftrightarrow & (y \in CX_1(k)) \wedge (y \in CX_2(k)) \wedge \dots \\
\Leftrightarrow & y \in \bigcap_i CX_i(k)
\end{aligned}
$$

□

Remark 3.15. *The reverse inclusions need not hold in Lemmas 3.13 and 3.14. Let* $C = I$, $X_1 = X_s$ *and* $X_2 = X_{ns}$. $X_1 \cap X_2 = \emptyset$, *but* $X_1(k) \cap X_2(k)$ *need not be*

empty[1].

Theorem 3.16. *If X_{s_i} is k-ISO with respect to X_{ns} for each i, then $\bigcap_i X_{s_i}$ is k-ISO with respect to X_{ns}.*

Proof.

$$
\begin{aligned}
& X_{s_i}\ k-\text{ISO w.r.t. } X_{ns} \forall i \\
\Leftrightarrow & CX_{s_i}(k) \subseteq CX_{ns}(k) \forall i \\
\Rightarrow & \bigcap_i CX_{s_i}(k) \subseteq CX_{ns}(k) \\
\Rightarrow & C(\bigcap_i X_{s_i}(k)) \subseteq CX_{ns}(k) \\
\Leftrightarrow & \bigcap_i X_{s_i} \text{ is } k-\text{ISO w.r.t. } X_{ns}
\end{aligned}
$$

□

Theorem 3.17. *If X_s is k-ISO with respect to X_{ns_i} for each i, then $CX_s(k) \subseteq \bigcap_i CX_{ns_i}(k)$. However, in general, X_s is not k-ISO with respect to $\bigcap_i X_{ns_i}$.*

Proof.

$$
\begin{aligned}
& X_s\ k-\text{ISO w.r.t. } X_{ns_i} \forall i \\
\Leftrightarrow & CX_s(k) \subseteq CX_{ns_i}(k) \forall i \\
\Rightarrow & CX_s(k) \subseteq \bigcap_i CX_{ns_i}(k)
\end{aligned}
$$

However, we can have $\bigcap_i X_{ns_i} = \emptyset$, which means $C(\bigcap_i X_{ns_i})(k)$ is undefined. □

[1]Recall that the definition of the reachable set in k steps assumes a nonempty initial set of states.

Theorem 3.18 and Remark 3.19 are similar results for weak opacity.

Theorem 3.18. *If X_{s_i} is weakly k-ISO with respect to X_{ns} for each i, then $\bigcup_i X_{s_i}$ is weakly k-ISO with respect to X_{ns}.*

Proof.

$$
\begin{aligned}
& X_{s_i} \text{ weakly } k-\text{ISO w.r.t. } X_{ns} \forall i \\
\Leftrightarrow & CX_{s_i}(k) \bigcap CX_{ns}(k) \neq \emptyset \forall i \\
\Rightarrow & \bigcup_i CX_{s_i}(k) \bigcap CX_{ns}(k) \neq \emptyset \\
\Rightarrow & C(\bigcup_i X_{s_i}(k)) \bigcap CX_{ns}(k) \neq \emptyset \\
\Leftrightarrow & \bigcup_i X_{s_i} \text{ is weakly } k-\text{ISO w.r.t. } X_{ns}
\end{aligned}
$$

□

Remark 3.19. *If X_{s_i} is weakly k-ISO with respect to X_{ns} for each i, then $\bigcap_i X_{s_i}$ need not be weakly k-ISO with respect to X_{ns}. That is, given $CX_{s_i}(k) \bigcap CX_{ns_i}(k) \neq \emptyset \forall i$, if $\bigcap_i X_{s_i} = \emptyset$, then $C(\bigcap_i X_{s_i})(k) \bigcap CX_{ns}(k)$ will not be defined.*

3.4 Opacity and Output Controllability

A state of the system is said to be controllable if we can find an input that transfers the state to the origin in finite time. While there are several interesting results in the literature that relate controllability of a dynamical system to other properties of interest, the notion of *output controllability* has been largely overlooked. Output controllability is the ability of transferring the state of the system

such that the output corresponding to the state at some finite time is zero. It is easy to see that while controllability implies output controllability, the reverse need not necessarily hold. In fact, output controllability will imply controllability if the matrix C has full rank. This section establishes an equivalence between k-ISO and output controllability.

Definition 3.20. *A state x of (3.1) is* controllable *on $[0, k_f]$ if there exists a control sequence $\{u(\cdot)\}$ that transfers the state of the system from $x(0) = x$ to $x(k_f) = 0$.*

The output of (3.1) at time k is given by:

$$y(k) = CA^k x(0) + \sum_{j=0}^{k-1} CA^{k-j-1} Bu(j)$$

Definition 3.21. *A state x of (3.1) is* output controllable *on $[0, k_f]$ if there exists a control sequence $\{u(\cdot)\}$ that transfers the system from $x(0) = x$ to $x(k_f)$, such that $y(k_f) = 0$.*

Theorem 3.22 indicates that X_s being k-ISO with respect to X_{ns} ensures that there exists a state that is output controllable. Theorem 3.23 establishes the converse result, under an additional assumption.

Theorem 3.22. *Let X_s be (strongly or weakly) k-ISO with respect to X_{ns}. Then there exists a state of (3.1) that is output controllable on $[0, k]$. Further, if k-ISO is established for the pair $(x_s(0), x_{ns}(0)) \in X_s \times X_{ns}$ (and appropriate control sequences $\{u_s(\cdot)\}$ and $\{u_{ns}(\cdot)\}$), then the control sequence $u(i) = u_s(i) - u_{ns}(i)$, $i = 0, 1, \ldots, k-1$, will achieve output controllability for the initial state $x(0) = x_s(0) - x_{ns}(0)$.*

Proof. k-ISO implies $y_s(k) = y_{ns}(k)$ for appropriate $x_s(0), \{u_s(\cdot)\}, x_{ns}(0)$ and $\{u_{ns}(\cdot)\}$. Setting $x(0) = x_s(0) - x_{ns}(0)$ and $u(i) = u_s(i) - u_{ns}(i)$, $i = 0, 1, \ldots, k-1$ in the dynamics of (3.1) ensures $y(k) = 0$, thus achieving output controllability of the state $x(0) = x_s(0) - x_{ns}(0)$. □

Theorem 3.23. *Let (3.1) be output controllable in k steps for a set of states $X_{oc}(0) \setminus \{0\}$ and controls $\{U(\cdot)\}$. Let X_1 and X_2 be sets such that every $x_1 \in X_1$ can be written as $x + x_2$, where $x \in X_{oc}(0) \setminus \{0\}$ and $x_2 \in X_2$. Then, X_1 is strongly k-ISO with respect to X_2.*

Proof. Output controllability ensures that:

$$y(k) = CA^k x(0) + \sum_{j=0}^{k-1} CA^{k-j-1} BU(j) = 0 \tag{3.4}$$

For any control sequence $\{u_1(\cdot)\}$, the output at time k, starting from any $x_1(0) \in X_1$ is:

$$y_1(k) = CA^k x_1(0) + \sum_{j=0}^{k-1} CA^{k-j-1} Bu_1(j)$$

The output at time k starting from $x_2(0) \in X_2$ with the control sequence $\{u_1(\cdot) - U(\cdot)\}$ is:

$$y_2(k) = CA^k x_2(0) + \sum_{j=0}^{k-1} CA^{k-j-1} B[u_1(j) - U(j)]$$

Using the assumption that every $x_1 \in X_1$ can be written as $x + x_2$, where $x \in X_{oc}(0) \setminus \{0\}$, $x_2 \in X_2$, and Equation (3.4), we get $y_1(k) = y_2(k)$.

Thus, for any $x_1 \in X_1$ and any control sequence starting from x_1, there exist $x_2 \in X_2$ and another control sequence such that the outputs after k steps are the same. This is strong k-ISO with $X_s = X_1$ and $X_{ns} = X_2$. □

3.5 ϵ-Opacity

The condition that the output at times $k \in \mathcal{K}$ starting from every state in X_s be equal to the output obtained by starting from some state in X_{ns} is quite strong. In this section, we postulate that (a form of) opacity will still hold if the outputs differ by a predefined amount. We only consider the single adversary case; the material can be easily extended to the decentralized notions of opacity in Sections 4.2 and 4.4. Defining $\epsilon-$opacity for the case in Section 4.3 will require more careful consideration.

Definition 3.24. *For system (3.1), given $X_s, X_{ns} \subseteq X_0$, $k \in \mathcal{K}$, and $\epsilon \geq 0$, X_s is* strongly $\epsilon - k$-ISO *with respect to X_{ns} if for all $x_s(0) \in X_s$ and for every sequence of admissible controls $u_s(0), \ldots, u_s(k-1)$, there exist an $x_{ns}(0) \in X_{ns}$, and a sequence of admissible controls $u_{ns}(0), \ldots, u_{ns}(k-1)$ such that $\|y_s(k) - y_{ns}(k)\|_2 \leq \epsilon$.*

X_s is strongly $\epsilon - \mathcal{K}$-ISO *with respect to X_{ns} if it is strongly $\epsilon - k$-ISO for all $k \in \mathcal{K}$.*

A couple of remarks are in order before we present the main result of this section. Notice that $\epsilon = 0$ corresponds to the definition of strong k-ISO seen earlier. Moreover, we can derive conditions that establish $\epsilon-$opacity in terms of sets of reachable states.

Let z be a point, and S be a set. Then, the distance of z from S is defined as $dist(z, S) := inf\{dist(z, s) | s \in S\}$.

Theorem 3.25. *The following hold:*

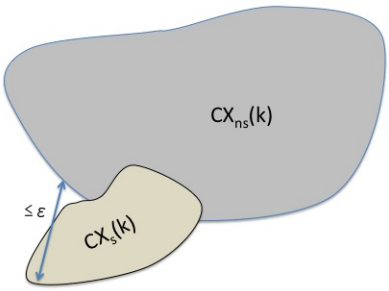


Figure 3.3: Representation of $\epsilon - k$-ISO

1. *X_s is strongly $\epsilon - k$-ISO with respect to X_{ns} if and only if :*

$$\max_{z \in CX_s(k)} dist(z, CX_{ns}(k)) \leq \epsilon \tag{3.5}$$

   That is, the farthest a point in $CX_s(k)$ can be from $CX_{ns}(k)$ is ϵ.

2. *X_s is strongly $\epsilon - \mathcal{K}$-ISO with respect to X_{ns} if and only if (3.5) holds for all $k \in \mathcal{K}$.*

Proof. The proof of this result follows from the definition of $\epsilon - k$-ISO and Theorem 3.4. □

Chapter 4: Opacity for Linear Systems: The Multiple Adversaries Case

In this chapter, we extend the framework for opacity presented in the previous chapter to the case when there is more than one adversarial observer, and define several notions of decentralized opacity. These notions of decentralized opacity will depend on whether there is a centralized coordinator or not, and the presence or absence of collusion among the adversaries. We establish conditions for decentralized opacity in terms of sets of reachable states. In the case of colluding adversaries, we derive a condition for *nonopacity* in terms of the structure of the communication graph.

Section 4.1 presents the system model. Sections 4.2, 4.3, and 4.4 defines and characterizes the three notions of decentralized opacity.

4.1 System Model

The system model is identical to that considered in the previous chapter, except that there are multiple adversaries, each seeing an output corresponding to its observation map C_i. As in the single adversary case, every adversary is assumed to have knowledge of the initial sets of secret and nonsecret states, X_s and X_{ns},

the system model (A, B), and its own observation map C_i, and is assumed to have unlimited computing power.

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) \\ x(0) &= x_0 \in X_0 \\ y_i(t) &= C_i x(t);\ i = 1, 2, \ldots, l \end{aligned} \tag{4.1}$$

where $x \in \mathbb{R}^n, u \in \mathbb{R}^m, y_i \in \mathbb{R}^{p_i}$, and A, B, C_i are matrices of appropriate dimensions containing real entries. In the sequel, we will assume that all of the adversaries observe the system at the same time instants in the set $\mathcal{K}$.

The presence or absence of coordination among the adversaries, and the presence or absence of a coordinator that aggregates information based on the adversaries' observations, is the distinguishing feature, and a definition of decentralized opacity is proposed in each case.

4.2 No Coordinator, No Coordination

The agents are assumed to not communicate with each other, and there is no centralized coordinator. Opacity of the secret will be achieved when it is simultaneously opaque with respect to every adversary.

Definition 4.1. *For system (4.1), given $X_s, X_{ns} \subseteq X_0$ and $k \in \mathcal{K}$, X_s is* strongly decentralized $k-$ISO *with respect to X_{ns} if for all $x_s(0) \in X_s$ and for every sequence of admissible controls $u_s(0), \ldots, u_s(k-1)$, there exist an $x_{ns}(0) \in X_{ns}$, and a se-*

quence of admissible controls $u_{ns}(0), \ldots, u_{ns}(k-1)$ *such that* $y_{s_i}(k) = y_{ns_i}(k)$ *for all* $i \in \{1, 2, \ldots, l\}$.

X_s *is* strongly decentralized $\mathcal{K}-$ISO *with respect to* X_{ns} *if it is strongly decentralized* $k-$*ISO for all* $k \in \mathcal{K}$.

As in the single adversary case, we have a necesary and sufficient condition for for decentralized opacity in terms of sets of reachable states in k steps.

Theorem 4.2. *The following hold:*

1. X_s *is strongly decentralized* $k-$*ISO with respect to* X_{ns} *if and only if* $C_i X_s(k) \subseteq C_i X_{ns}(k)$ *for all* $i \in \{1, 2, \ldots, l\}$.

2. X_s *is strongly decentralized* $\mathcal{K}-$*ISO with respect to* X_{ns} *if and only if* $C_i X_s(k) \subseteq C_i X_{ns}(k)$ *for all* $k \in \mathcal{K}$, *and for all* $i \in \{1, 2, \ldots, l\}$.

Proof. The proof of this result follows by applying Theorem 3.4 to every adversary $i = \{1, 2, \ldots, l\}$. □

The following result explores the relationship between decentralized $k-$ISO for a set of adversaries and $k-$ISO for a single adversary with an aggregated observation map.

Proposition 4.3. X_s *is strongly decentralized* $k-$*ISO with respect to* X_{ns} *and adversaries with observation maps* $C_1, \ldots, C_l$ *if* X_s *is strongly* $k-$*ISO with respect to* X_{ns} *for the single adversary with the aggregated observation map* $\bar{C} := \begin{pmatrix} C_1^T & C_2^T & \ldots & C_l^T \end{pmatrix}^T$.

Proof. X_s strongly $k-$ISO with respect to X_{ns} is equivalent to $\bar{C} X_s(k) \subseteq \bar{C} X_{ns}(k)$. This means that for every $x_s(k) \in X_s(k)$, there exists an $x_{ns}(k) \in X_{ns}(k)$ such that

$C_1 x_s(k) = C_1 x_{ns}(k)$, ..., $C_l x_s(k) = C_l x_{ns}(k)$. Thus, we have $C_i X_s(k) \subseteq C_i X_{ns}(k)$ for all $i \in \{1, \ldots, l\}$, which is equivalent to X_s being strongly decentralized $k-$ISO with respect to X_{ns}. □

It is to be noted that strong decentralized $k-$ISO need not necessarily ensure strong $k-$ISO with respect to an adversary with the aggregated observation map since, the nonsecret states in $X_{ns}(k)$ and the corresponding control sequences for each adversary might be different.

4.3 With Coordinator, No Coordination

Here, we assume that there is a coordinator, whose role is to poll the observations of each adversary, and decide on *co-opacity* according to some (predefined) rule. The coordinator does not have knowledge of the system model or the adversaries' observation maps. In fact, our model is such that the coordinator cannot do any better even if it knows the system model or the observation maps. It can be viewed as an agent whose role is to ensure that the whole is greater than the sum of its parts.

Formally, the coordinator communicates to the adversaries the time instants $\mathcal{K}$, at which the system needs to be observed. At each $k \in \mathcal{K}$, agent i observes $y_i(k) = C_i x(k)$. The agents communicate $\phi_i(y_i(k))$ to the coordinator, where $\phi_i : \mathbb{R}^{p_i} \to 2^{\mathbb{R}^n \times \mathbb{R}^n}$ is defined as:

$$\phi_i(y_i(k)) := \{(x^1, x^2) \in X_s(k) \times X_{ns}(k) : C_i x^1 = C_i x^2 = y_i(k)\}$$

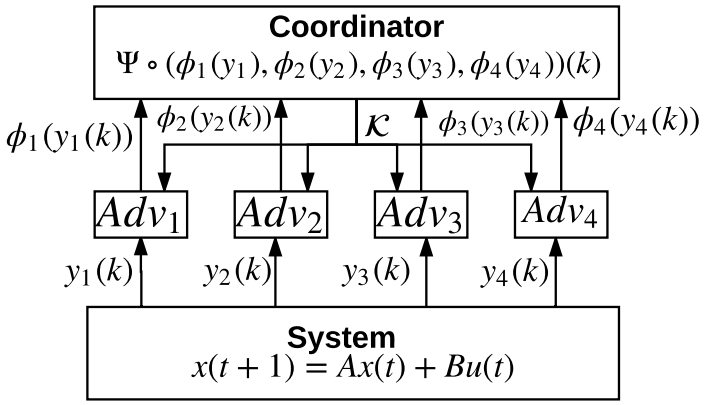


Figure 4.1: Coordinated Decentralized Opacity

Thus, $\phi_i(\cdot)$ returns secret-nonsecret state pairs that give the same output $y_i(k)$ at time k.

The coordinator then computes a function $\Psi(k) := \Psi(\phi_1(y_1(k)), \ldots, \phi_l(y_l(k)))$, where $\Psi : (2^{\mathbb{R}^n \times \mathbb{R}^n})^l \to 2^{\mathbb{R}^n \times \mathbb{R}^n}$. Thus, the coordinator plays the role of gathering the outputs of the observations of each adversary, and composing them to then decide on opacity. An example of a valid coordinator function is $\Psi(k) = \bigcup_i (\phi_i(C_i x(k)))$.

The scheme is shown in Figure (4.1) for the case of four adversaries.

Definition 4.4. *For system (4.1), given $X_s, X_{ns} \subseteq X_0$ and $k \in \mathcal{K}$, X_s is* strongly co-$k-$ISO *with respect to X_{ns} and Ψ if for all $x_s(0) \in X_s$ and for every sequence of admissible controls $u_s(0), \ldots, u_s(k-1)$, there exist an $x_{ns}(0) \in X_{ns}$, and a sequence of admissible controls $u_{ns}(0), \ldots, u_{ns}(k-1)$ such that $\Psi(k)$ is nonempty.*

X_s is strongly co-$\mathcal{K}-$ISO *with respect to X_{ns} and Ψ if it is strongly co-$k-$ISO for all $k \in \mathcal{K}$.*

Before presenting the main result of this section, we provide an alternative

characterization of strong $k-$ISO in terms of the map ϕ (the subscript on ϕ_i is dropped since we consider only a single adversary in this case). Further, it is important to note that the functions ϕ_i and Ψ return a set of pairs of states at time k. This information will need to be used to determine opacity of the initial set of secret states with respect to the initial set of nonsecret states.

We extend the definition of ϕ to sets of outputs at time k. Let $\phi(CX(k)) := \bigcup\{\phi(y(k)) : [y(k) = Cx(k)] \wedge [x(k) \in X(k)]\}$. For $(x_i^1, x_j^2) \in X_s(k) \times X_{ns}(k)$, in a slight abuse of notation, we treat each x_i^1 and x_j^2 as a set. This will allow us to define $\bigcup_{i,j}(x_i^1, x_j^2) := (\bigcup_i x_i^1, \bigcup_j x_j^2)$, where $\bigcup_i x_i^1 \subseteq X_s(k)$, and $\bigcup_j x_j^2 \subseteq X_{ns}(k)$.

Proposition 4.5. *X_s is strongly $k-$ISO with respect to X_{ns} if and only if $\phi(CX_s(k)) = (X_s(k), X'_{ns}(k))$, where $X'_{ns}(k) := \{x \in X_{ns}(k) : Cx \in CX_s(k)\}$.*

Proof. Let strong $k-$ISO hold. Then, $CX_s(k) \subseteq CX_{ns}(k)$ (Theorem 3.4), and $\phi(CX_s(k)) = (X_s(k), X'_{ns}(k))$, where $X'_{ns}(k)$ is as defined above.

If $\phi(CX_s(k)) = (X_s(k), X'_{ns}(k))$, then $\forall x^1 \in X_s(k)$, $\exists x^2 \in X'_{ns}(k) \subseteq X_{ns}(k)$ such that $Cx^1(k) = Cx^2(k)$. This gives $CX_s(k) \subseteq CX_{ns}(k)$, which implies strong $k-$ISO (Theorem 3.4). □

The above result says that strong $k-$ISO holds if and only if the first component of $\phi(\cdot)$ when acting on the set of secret outputs at time k is the entire set of reachable states at time k, starting from X_s. Further, it also determines the states in $X_{ns}(k)$ that ensure strong $k-$ISO.

Theorem 4.6. *X_s is strongly co-$k-$ISO with respect to X_{ns} and Ψ if and only if $\Psi(\phi_1(C_1X_s(k)), \ldots, \phi_l(C_lX_s(k))) = (X_s(k), X'_{ns}(k))$, where $X'_{ns}(k) \subseteq X_{ns}(k)$.*

Proof. The proof of this result follows from the previous result, and the definition of co$-k-$ISO. The major difference is that in this case, the first component of $\phi_i(C_i X_s(k))$ can be a subset of $X_s(k)$. However, the coordinator function Ψ must be such that its first component is $X_s(k)$. □

Thus, X_s can be strongly co$-k-$ISO with respect to X_{ns} though strong $k-$ISO might not hold for any single adversary.

4.4 No Coordinator, With Coordination

In this case, there is no coordinator, but the adversaries are assumed to communicate among themselves. The communication structure is represented by a directed graph $\mathcal{G}$, whose vertices are the adversaries, and $\mathcal{G}$ has an edge directed from i to j if adversary j can receive information from adversary i. The goal of the adversaries is to ensure, using the coordination structure, that X_s is not $k-$ISO with respect to X_{ns} for each of them. To this end, we introduce the following definitions:

Definition 4.7. *For the system (4.1), given $X_s, X_{ns} \subseteq X_0$ and $k \in \mathcal{K}$, X_s is* strongly not $k-$ISO *with respect to X_{ns} if X_s is not strongly $k-$ISO with respect to X_{ns} for every adversary.*

Definition 4.8. *Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ are the vertices of the graph and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ are edges, $D \subset \mathcal{V}$ is a* dominating set *if every vertex not in D has a neighbor in D.*

Given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $D \subset \mathcal{V}$ is a directed dominating set *(red vertices in figure (4.2)) if every vertex not in D has an incoming edge from some*

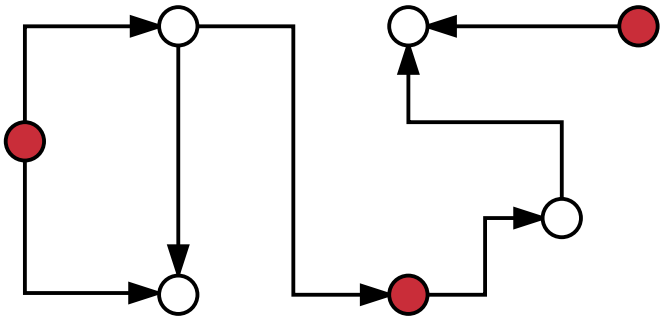


Figure 4.2: Vertices in red form a directed dominating set

vertex in D, that is, $[\forall u \in \mathcal{V} \setminus D, \exists v \in D$ such that $(v \to u) \in \mathcal{E}]$.

At each $k \in \mathcal{K}$, each adversary observes $y(k)$, determines if $k-$ISO holds or not, and communicates $(C_i, <k-\text{ISO status}>_i)$ to its neighbors in $\mathcal{G}$. If $<k-\text{ISO status}>_i = 0$, i.e. $k-$ISO does not hold for adversary i, then a neighbor j of i in $\mathcal{G}$ adopts C_i as its observation map if $<k-\text{ISO status}>_j \neq 0$. This scheme can be interpreted as a dynamic version of $k-$ISO, in which the adversaries change their observation maps at times $k \in \mathcal{K}$ depending on the $k-$ISO status of their neighbors in $\mathcal{G}$. A key assumption here is that the time required for the adversaries to communicate amongst themselves is much less than the time scale of the system. The following result provides a means to achieve strong non-opacity without requiring non-opacity with respect to every adversary using the communication scheme described above.

Theorem 4.9. *For the system (4.1), X_s is strongly not $k-$ISO with respect to X_{ns} if the set of adversaries for which X_s is not strongly $k-$ISO with respect to X_{ns} is a directed dominating set of $\mathcal{G}$.*

Proof. Each adversary communicates $(C_i, < k-\text{ISO status}>_i)$ to its neighbors in $\mathcal{G}$. Thus, if $k-$ISO does not hold for some adversary i, then its neighbors will also adopt the same C_i matrix at time k. The result then follows from the definition of a directed dominating set. □

Chapter 5: Opacity for Switched Linear Systems

In this chapter, we formulate notions of opacity for CPSs modeled as discrete-time switched linear systems (DT-SLSs). An SLS consists of a finite number of linear subsystems (called modes) and a rule that governs the switching among them. Many practical systems can be modeled as operating in one of several modes, often switching from one mode of operation to another. Further, it has been shown that switching control strategies can achieve better control performance than nonswitching strategies. The reader is referred to [43], [44], [45] for an introduction to the design and control of switched systems. We will assume that each subsystem is governed by linear, time-invariant dynamics.

We present the model of the system that will be studied in this chapter and underlying assumptions in Section 5.1. Sections 5.2, 5.3, and 5.4 present the main results of the chapter, wherein we formulate several notions of opacity for SLSs. We distinguish between the cases when the secret is specified as an initial mode, an initial state, or a combination of the two, and whether the adversary observes a mode, a function of the state, or a combination of the two. In each case, we present conditions that will establish that notion of opacity. We place constraints on the modes that the system will be allowed to transition into from a given mode

and impose bounds on the dwell times in each mode. Moreover, we constrain the number of changes of modes before the adversary can make its observation in our definitions of opacity for SLSs. Algorithmic procedures to verify these notions of opacity are given in Section 5.5, where we also provide conservative upper bounds on their computational complexity. Illustrative examples are presented in Section 5.6.

5.1 System Model and Assumptions

Consider a DT-SLS is of the form:

$$x(t+1) = A(\mathcal{M}_t)x(t) + B(\mathcal{M}_t)u(t) \tag{5.1}$$

$$x(0) = x_0 \in X_0$$

$$y(t) = Cx(t) \tag{5.2}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$, and $A(\cdot)$, $B(\cdot)$, C are matrices of appropriate dimensions containing real entries. $\mathcal{M}_t \in \{1, 2, \ldots, z\}$ denotes the mode active at time t. The solution to the state Equation (5.1) is given by Equation (5.3) (in Section 5.3). The system switches from a mode $\mathcal{M}'$ to $\mathcal{M}''$ at a time $t = t_s$, which is called a *switching time*. That is, $A(\mathcal{M}_{t_s-1}) = A(\mathcal{M}')$, while $A(\mathcal{M}_{t_s}) = A(\mathcal{M}'')$. The $B(\cdot)$ matrix switches similarly. A *switching sequence of length* N is a collection of N (possibly nonconsecutive) switching times $t_{s_1} < t_{s_2} < \cdots < t_{s_N}$. Let $\mathcal{K}$ be a set of positive integers corresponding to the instants of time the adversary makes an observation of the system. The subscript s (ns), when appended to the states, inputs, and outputs, will correspond to trajectories that start from the set of initial

secret (nonsecret) states.

The following assumptions will be needed to formalize notions of opacity for SLSs in this chapter.

Assumption 5.1. *The allowed transitions between modes is specified by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ whose vertices are the modes $\{1, \ldots, z\}$ and edges are the possible transitions between modes.*

Assumption 5.2. *The mode changes at every switching time t_s.*

Assumption 5.3. *The switching sequence does not depend on initial states and controls*[1].

Assumption 5.4. *(*Nonblocking Property*) It is possible for the system to switch to at least one other mode from every mode.*

Assumption 5.5. *(*Dwell constraints*) The system is allowed to remain in a mode a for a duration of time $\tau_d^a \in [\tau_{d_{min}}^a, \tau_{d_{min}}^a + 1, \ldots, \tau_{d_{max}}^a]$. Further, $\tau_{d_{min}}^a \geq 1 \forall a$.*

Assumption 5.6. *The adversary has knowledge of the initial secret and nonsecret specifications – sets of states and/ or modes, as the case may be – the $A(\cdot), B(\cdot)$ matrices, the observation map C, the graph $\mathcal{G}$, and the minimum and maximum dwell times in each mode.*

A path in $\mathcal{G}$ is an alternating sequence of vertices and edges of $\mathcal{G}$, $v_0 e_1 v_1 e_2 \ldots$. A path will always begin and end in a vertex of $\mathcal{G}$. Further, if the sequence $v_{i-1} e_i v_i$

[1] A standard assumption in the SLS literature is that there is only a finite number of switches in any finite time interval. This is needed to rule out the Zeno phenomenon in continuous time systems. It will not be needed here since we are dealing with discrete time systems.

appears in a path, then v_{i-1} and v_i are respectively the source and target vertices of edge e_i in $\mathcal{G}$. The *length* of a path is defined to be the number of vertices in the path. Let $\Theta_{[N]}$ be the set of paths of length $N+1$ in $\mathcal{G}$. This corresponds to N changes of modes. Knowledge of $\mathcal{G}$ will help eliminate transitions that are impossible. An element $\theta \in \Theta_{[N]}$ can be written as $v_0 e_1 v_1 e_2 \dots e_N v_N$, with $v_i \neq v_{i+1}$ (Assumption 5.2). We will represent $\theta \in \Theta_{[N]}$ as a sequence of vertices $v_0 v_1 \dots v_N$ when the edges representing transitions between vertices are obvious.

Assumption 5.7. *At a time $k \in \mathcal{K}$, the number of switches of modes, q, is strictly less than k.*

Assumption 5.8. *For given q and k, $t_{s_q} < k$. That is, the q mode changes occur before time k. Further, if k is a possible ($(q+1)^{st}$) switching time, and if the adversary is observing the mode of the system, its observation will be $\mathcal{M}_{k^-}$, that is, the mode the system is in just before the switching at time k*[2].

The notions of opacity developed in this chapter will be defined in terms of q and k. Informally, for a given number of mode changes q, the secret is said to be opaque at a time k if for every 'allowed' switching sequence of length q starting from the secret modes and/ or states, there is an 'allowed' switching sequence of length q starting from a nonsecret mode and/ or state, such that the observation at time k will be indistinguishable to the adversary. The 'allowed' switching sequences will be those that respect the dwell time constraints in the modes along paths of length

[2]This assumption is needed because the state, and consequently, the output of the system at a time t depends only on the modes of the system upto time $t-1$.

$q+1$. Throughout this paper, a switching sequence of length q will correspond to a path of length $q+1$ in $\mathcal{G}$.

5.2 Initial Mode Opacity

In this case, the secret is specified as a set of modes. Let $\mathcal{M}^s \subset \{1,\ldots,z\}$ and $\mathcal{M}^{ns} \subset \{1,\ldots,z\}$ be the sets of initial secret and nonsecret modes, with $\mathcal{M}^s \cap \mathcal{M}^{ns} = \emptyset$. The mode at time t, starting from a mode in $\mathcal{M}^s$ ($\mathcal{M}^{ns}$) at time 0 will be denoted $\mathcal{M}^s_t$ ($\mathcal{M}^{ns}_t$). The adversary observes the mode of the system at a time $k \in \mathcal{K}$. Its goal is to use this observation and other information that it has access to (Assumption 5.6) to deduce if the system started from a secret mode.

Definition 5.9. *Given $\mathcal{M}^s, \mathcal{M}^{ns}, k \in \mathcal{K}$, and $q < k$, $\mathcal{M}^s$ is (k,q)-**initial mode opaque** ((k,q)-**IMO**) with respect to $\mathcal{M}^{ns}$ if for all $\theta = v_0 \ldots v_q \in \Theta_{[q]}$ satisfying $v_0 \in \mathcal{M}^s$ and $\sum_\theta \tau_d = k$, there exists $\theta' = v'_0 \ldots v'_q \in \Theta_{[q]}$ that satisfies $v'_0 \in \mathcal{M}^{ns}$ and $\sum_{\theta'} \tau_d = k$, such that $\mathcal{M}^s_k = \mathcal{M}^{ns}_k$.*

The term $\sum_\theta \tau_d = k$ means that there exists a sequence of $q+1$ dwell times $\{\tau_d^{v_j} \in [\tau_{d_{min}}^{v_j}, \tau_{d_{max}}^{v_j}]\}, j = 0,\ldots,(q-1)$, and $1 \le \tau_d^{v_q} \le \tau_{d_{max}}^{v_q}$ along the path $\theta = v_0 \ldots v_q$, such that $\sum_{j=0}^{q-1} \tau_d^{v_j} + \tau_d^{v_q} = k$. Thus, we only consider paths of length $q+1$ in $\mathcal{G}$ for which the dwell times in modes along the path are 'sufficiently long'.

Theorem 5.10. *$\mathcal{M}_s$ is (k,q)-IMO with respect to $\mathcal{M}_{ns}$ iff*

$$\bigcup_{\substack{\theta=v_0\ldots v_q\in\Theta_{[q]}:\\ \sum_\theta \tau_d=k\wedge v_0\in\mathcal{M}_s}} \mathcal{M}^s_k \subseteq \bigcup_{\substack{\theta=v_0\ldots v_q\in\Theta_{[q]}:\\ \sum_\theta \tau_d=k\wedge v_0\in\mathcal{M}^{ns}}} \mathcal{M}^{ns}_k.$$

Proof. $\Rightarrow$: Follows from Definition 5.9.

$\Leftarrow$: Every mode the system can reach at time k starting from a secret mode at time 0 after q mode changes and respecting dwell time constraints along the path can also be reached by starting from a nonsecret mode at time 0, and an allowed switching sequence of length q. This establishes Definition 5.9. □

It is important that the system dwell in a mode for sufficiently long in order to meaningfully establish initial mode opacity.

Proposition 5.11. *If $\tau^a_{d_{max}} = 1$ for every mode $a \in \{1, \ldots, z\}$, then for any choice of $\mathcal{M}^s$ and $\mathcal{M}^{ns}$, and for every $q < k - 1$, $\mathcal{M}^s$ will not be (k, q)-IMO w.r.t. $\mathcal{M}^{ns}$.*

The main result of this section provides guarantees on (k', q')-IMO for $k' > k$, $q' > q$ if it has been established that (k, q)-IMO holds. In the sequel, we will write $\sum_q \tau_d$ to denote the sum of the dwell times in the modes along a path of length $q + 1$ in the directed graph.

Theorem 5.12. *If $\mathcal{M}^s$ is (k, q)-IMO with respect to $\mathcal{M}^{ns}$, then for every $Q > 0$, $\mathcal{M}^s$ is $(k + K, q + Q)$-IMO w.r.t. $\mathcal{M}^{ns}$ for all $K \in [\sum_q \tau_d - k + \sum_Q \tau_{d_{min}}, \sum_q \tau_d - k + \sum_Q \tau_{d_{max}}]$.*

Proof. Let $\Theta_{[q,Q]}$ denote the set of valid extensions of length Q to a switching sequence of length q. Then, every $\theta'' \in \Theta_{[q+Q]}$ can be written as $\theta.\theta_e$, where $\theta \in \Theta_{[q]}$ and $\theta_e \in \Theta_{[q,Q]}$, and . denotes the concatenation of the paths. (k, q)-IMO ensures that for every $\theta \in \Theta_{[q]}$ starting from a secret mode and satisfying the dwell time constraints along the path, there exists $\theta' \in \Theta_{[q]}$ starting from a nonsecret mode

and satisfying the dwell time constraints along θ', such that $\mathcal{M}_k^s = \mathcal{M}_k^{ns}$. Using Assumption 5.4, any extension of θ (θ') of length Q can be written as $\alpha = \theta.\theta_e$ ($\alpha' = \theta'.\theta_e$), where $\theta_e \in \Theta_{[q,Q]}$. This shows that $(k+K, q+Q)$-IMO holds for some $K \geq Q$.

The lower and upper bounds on K are obtained by considering the minimum and maximum dwell times along the extension of length Q to a path of length q, and noting that $\tau_{d_{min}}^a \geq 1$ for every mode (Assumption 5.5). Two cases need to be considered:

Case I: $k - (\sum_{j=0}^{q-1} \tau_d^{v_j}) \geq \tau_{d_{min}}^{v_q}$. In this case, the term $\sum_q \tau_d - k = 0$, and $K \in [\sum_Q \tau_{d_{min}}, \sum_Q \tau_{d_{max}}]$.

Case II: $k - (\sum_{j=0}^{q-1} \tau_d^{v_j}) < \tau_{d_{min}}^{v_q}$. Here, the $(q+1)^{st}$ change of mode can occur only after a time $\sum_{j=0}^{q-1} \tau_d^{v_j} + \tau_{d_{min}}^{v_q}$. Thus, we have $K \in [\sum_q \tau_d - k + \sum_Q \tau_{d_{min}}, \sum_q \tau_d - k + \sum_Q \tau_{d_{max}}]$ □

This formulation of (k,q)-IMO is reminiscent of 'state-based' notions of DES opacity [23]. However, unlike in the DES case, we do not insist that the entire secret trace be indistinguishable from the entire nonsecret trace; we require indistinguishability only at time k, with the caveat that there be only q changes of modes. (k,q)-IMO does not depend on the dynamics within each mode.

5.3 Initial Mode and State Opacity

In this case, the adversary observes $y(k)$ and $\mathcal{M}_k$ at a time $k \in \mathcal{K}$. This formulation is similar in flavor to pathwise observability (PWO) in [54]. However,

$$x(t+1) = A(\mathcal{M}_t)\dots A(\mathcal{M}_0)x(0)+\sum_{j=0}^{t-1}(A(\mathcal{M}_t)\dots A(\mathcal{M}_{j+1}))B(\mathcal{M}_j)u(j)+B(\mathcal{M}_t)u(t) \tag{5.3}$$

$$X_s(k,q) = \bigcup_{x_0\in X_s}\bigcup_{U_s^k}\bigcup_{\substack{\theta=v_0\dots v_q\in\Theta_{[q]}:\\ \sum_\theta \tau_d=k\wedge v_0\in\mathcal{M}^s}} \{x : x(i+1) = A(\mathcal{M}_i)x(i) + B(\mathcal{M}_i)u(i), \forall i < k\} \tag{5.4}$$

$$X_{ns}(k,q) = \bigcup_{x_0\in X_{ns}}\bigcup_{U_{ns}^k}\bigcup_{\substack{\theta=v_0\dots v_q\in\Theta_{[q]}:\\ \sum_\theta \tau_d=k\wedge v_0\in\mathcal{M}^{ns}}} \{x : x(i+1) = A(\mathcal{M}_i)x(i)+B(\mathcal{M}_i)u(i), \forall i < k\} \tag{5.5}$$

---

in their framework, the entire mode sequence and the output up to time k are available. Here, we only have snapshots of the output-mode pair at a time $k \in \mathcal{K}$. The secret in this case is specified as a state-mode pair.

Definition 5.13. *For the system 5.2, given $k \in \mathcal{K}, q < k, \bar{X}_s := (X_s; \mathcal{M}^s)$, and $\bar{X}_{ns} := (X_{ns}; \mathcal{M}^{ns})$, with $X_s, X_{ns} \subset X_0$, $\bar{X}_s$ is (k,q)-**initial mode and state opaque** ((k,q)-**IMSO**) with respect to $\bar{X}_{ns}$ if for every $x_s(0) \in X_s$, every sequence of admissible controls U_s^k, and every $\theta = v_0 \dots v_q \in \Theta_{[q]}$ satisfying $v_0 \in \mathcal{M}^s$ and $\sum_\theta \tau_d = k$, there exist an $x_{ns}(0) \in X_{ns}$, a sequence of admissible controls U_{ns}^k, and $\theta' = v'_0 \dots v'_q \in \Theta_{[q]}$, satisfying $v'_0 \in \mathcal{M}^{ns}$ and $\sum_{\theta'} \tau_d = k$ such that: i)$\mathcal{M}_k^s = \mathcal{M}_k^{ns}$, and ii)$y_s(k) = y_{ns}(k)$.*

That is, corresponding to every allowed switching sequence of length q starting from a secret mode and every valid control sequence of length k starting from a secret state, there is an allowed switching sequence of length q starting from a nonsecret mode, and a valid control sequence of length k starting from a nonsecret mode, such

that the mode and the output at time k will be indistinguishable to the adversary.

The set of reachable states at time k with q mode changes, starting from $\bar{X}_s$ ($\bar{X}_{ns}$), and applying k admissible controls and respecting the dwell constraints of the modes is given by Equation (5.4) (Equation (5.5)).

Theorem 5.14. *$\bar{X}_s$ is (k,q)-IMSO with respect to $\bar{X}_{ns}$ iff:*

$$\text{i)} \bigcup_{\substack{\theta=v_0\ldots v_q\in\Theta_{[q]}:\\ \sum_\theta \tau_d=k\wedge v_0\in\mathcal{M}^s}} \mathcal{M}_k^s \subseteq \bigcup_{\substack{\theta=v_0\ldots v_q\in\Theta_{[q]}:\\ \sum_\theta \tau_d=k\wedge v_0\in\mathcal{M}^{ns}}} \mathcal{M}_k^{ns}, \text{ and}$$

$$\text{ii)}\ CX_s(k,q) \subseteq CX_{ns}(k,q).$$

Proof. $\Rightarrow$: From the definition of (k,q)-IMSO, i) holds. $X_s(k,q)$ ($X_{ns}(k,q)$) is the set of states reachable at time k starting from states in X_s (X_{ns}) and modes in $\mathcal{M}^s$ ($\mathcal{M}^{ns}$), performing q changes of modes along the way, while respecting dwell constraints of each mode. Therefore, for each $x' \in X_s(k,q)$, there exists $x'' \in X_{ns}(k,q)$ such that $y_s(k) = Cx' = Cx'' = y_{ns}(k)$. This gives ii).

$\Leftarrow$: i) ensures that for every allowed switching sequence of length q starting from $\mathcal{M}^s$, there exists an allowed switching sequence starting from $\mathcal{M}^{ns}$ such that the mode at time k is indistinguishable. ii) ensures that for every $x' \in X_s(k,q)$, there exists an $x'' \in X_{ns}(k,q)$ such that $y_s(k) = Cx' = Cx'' = y_{ns}(k)$. From Equation (5.4) ((5.5)), x' (x'') is a state got by starting from an initial secret state and secret mode (nonsecret initial state and nonsecret mode), while satisfying dwell constraints and number of allowed changes of mode. This proves (k,q)-IMSO. □

5.4 Opacity for Unobserved Modes

In this case, the adversary observes the output $y(k)$ at a time $k \in \mathcal{K}$, and using only this information, it needs to determine if the system started from a secret state or mode. We consider two possible scenarios: when the secret is specified as a set of initial modes, and when the secret is specified as a set of initial states. This is like the unobservable mode case considered in [54], where they separately study the possibilities of recovering only the mode or only the state, after observing (at each time instant) the output. These notions are more general than (k, q)-IMSO in the sense that there is no constraint that the system start from a particular mode (Section 5.4.1) or a particular state (Section 5.4.2). Moreover, *the modes at time k corresponding to the secret and nonsecret trajectories need not be the same.* The subscript $\not{\mathcal{M}}$ will serve to indicate that the modes remain unobserved.

5.4.1 Initial Mode Opacity

The secret is specified as a set of modes, and the adversary has to deduce if the system started from a secret mode based on observing $y(k)$ at a time $k \in \mathcal{K}$.

Definition 5.15. *For system 5.2, given $X_0, \mathcal{M}_s, \mathcal{M}_{ns}, k \in \mathcal{K}$, and $q < k$, $\mathcal{M}^s$ is $(k, q)_{\not{\mathcal{M}}}$-**initial mode opaque** ($(k, q)_{\not{\mathcal{M}}}$-**IMO**) with respect to $\mathcal{M}^{ns}$ if for every initial state, every sequence of admissible controls, and every $\theta = v_0 \dots v_q \in \Theta_{[q]}$ satisfying $v_0 \in \mathcal{M}^s$ and $\sum_\theta \tau_d = k$, there exist an initial state, a sequence of admissible controls, and a $\theta' = v'_0 \dots v'_q \in \Theta_{[q]}$ that satisfies $v'_0 \in \mathcal{M}^{ns}$ and $\sum_{\theta'} \tau_d = k$,*

such that $y_s(k) = y_{ns}(k)$.

Let $X'_s(k,q)$ ($X'_{ns}(k,q)$) denote the set of states reachable at time k after q changes of modes starting from a secret mode (nonsecret mode) at time 0. That is, the condition $x_0 \in X_s$ ($x_0 \in X_{ns}$) in Equation (5.4) ((5.5)) is replaced by $x_0 \in X_0$, and the subscript on U_s^k (U_{ns}^k) is dropped.

Theorem 5.16. *$\mathcal{M}^s$ is $(k,q)_{\mathcal{M}}$-IMO with respect to $\mathcal{M}^{ns}$ if and only if $CX'_s(k,q) \subseteq CX'_{ns}(k,q)$.*

Proof. $\Rightarrow$: $X'_s(k,q)$ ($X'_{ns}(k,q)$) is the set of states reachable at time k starting from modes in $\mathcal{M}^s$ ($\mathcal{M}^{ns}$), performing q changes of modes along the way, while respecting dwell constraints of each mode. Therefore, for each $x' \in X'_s(k,q)$, there exists $x'' \in X'_{ns}(k,q)$ such that $y_s(k) = Cx' = Cx'' = y_{ns}(k)$. This gives $CX'_s(k,q) \subseteq CX'_{ns}(k,q)$.

$\Leftarrow$: Corresponding to every $x' \in X'_s(k,q)$, there exists an $x'' \in X'_{ns}(k,q)$ such that $y_s(k) = Cx' = Cx'' = y_{ns}(k)$. x' (x''), by definition, is a state that is got by starting from a secret mode (nonsecret mode), while satisfying dwell constraints and number of allowed changes of modes. Therefore, for every initial state starting from a secret mode and every allowed switching sequence of length q, there is an initial state starting from a nonsecret mode and a switching sequence of length q such that the outputs at time k are indistinguishable. This gives $(k,q)_{\mathcal{M}}$-IMO. □

5.4.2 Initial State Opacity

The adversary has to determine if the system started from a secret state, based on its observation $y(k)$ at time $k \in \mathcal{K}$. The underlying idea behind this notion is similar to k-ISO (Definition 3.1), with the difference that the system switches among several modes.

Definition 5.17. *For the system 5.2, given $X_s, X_{ns} \subset X_0$, $k \in \mathcal{K}$, and $q < k$, X_s is $(k,q)_{\mathcal{M}}$-**initial state opaque** ($(k,q)_{\mathcal{M}}$-**ISO**) with respect to X_{ns} if for every $x_s(0) \in X_s$, every sequence of admissible controls U_s^k, and every $\theta \in \Theta_{[q]}$ satisfying $\sum_\theta \tau_d = k$, there exist an $x_{ns}(0) \in X_{ns}$, a sequence of admissible controls U_{ns}^k, and a $\theta' \in \Theta_{[q]}$, satisfying $\sum_{\theta'} \tau_d = k$ such that $y_s(k) = y_{ns}(k)$.*

Let $X_s''(k,q)$ ($X_{ns}''(k,q)$) denote the set of states reachable at time k after q changes of modes starting from X_s (X_{ns}) at time 0, without restrictions on the initial modes. That is, the condition $v_0 \in \mathcal{M}^s$ ($v_0 \in \mathcal{M}^{ns}$) is removed from Equation (5.4) (Equation (5.5)).

Theorem 5.18. *X_s is $(k,q)_{\mathcal{M}}$-ISO with respect to X_{ns} if and only if $CX_s''(k,q) \subseteq CX_{ns}''(k,q)$.*

The proof of this result is similar to Theorems 5.14 and 5.16, and is omitted.

5.5 Computational Complexity

This section presents procedures to verify (k,q)-IMO and (k,q)-IMSO. Algorithm 1 depends on determining paths of length $q+1$ in $\mathcal{G}$, and determining a set of

$q+1$ numbers that sum to k. Algorithm 2 depends on determining sets of reachable states, and Algorithm 1.

The $SUBSETSUM$ problem asks the following question: given a set of non-negative integers S and a target number t, is there a subset of S whose elements sum to t? This problem is known to be NP-Complete [55][3]. The $rSUM$ problem asks: given a set of nonnegative integers S and numbers r and t, is there is a subset of S of size r whose elements sum to t? The brute-force algorithm for $rSUM$ runs in time $O(|S|^r)$. More recent results have significantly lowered this bound [56].

In our setting, given a set of $q+1$ lists $L_i := [\tau^i_{d_{min}}, \tau^i_{d_{min}}+1, \ldots, \tau^i_{d_{max}}], i \in \{1, 2, \ldots, q+1\}$, we ask if there is an assignment of nonzero numbers $\tau_1, \ldots, \tau_{q+1}$, with $\tau_i \in L_i$ such that $\sum_i \tau_i = k$. This is equivalent to the $rSUM$ problem, with $r = q+1$[4]. Let $\mathcal{C}_{sum}$ denote the number of operations needed to solve this problem.

Given the adjacency matrix $\mathcal{A}$ of $\mathcal{G}$ [5], the ij entry of $\mathcal{A}^q$ gives the number of paths in $\Theta_{[q]}$ with $v_0 = i$ and $v_q = j$. Let $\mathcal{C}_{pow}$ denote the complexity of computing $\mathcal{A}^q$. This typically takes $O(z^\omega log\ q)$ operations, where z is the number of vertices of $\mathcal{G}$ (modes of the system), and z^ω ($\omega < 3$) is the complexity of matrix multiplication. Let $\mathcal{C}_{path}$ be the maximum number of operations needed to determine a path of length q in $\mathcal{G}$. This problem is trivially solvable in $O(z^q)$, but can be more efficient,

[3] A decision problem is in class NP if all instances of the problem to which the answer is 'yes' can be efficiently verified by a deterministic Turing machine. It is NP-complete if, additionally, it is as hard as any problem in NP.

[4] An additional requirement is that the lists be of equal sizes. This can be achieved by padding them with zeros.

[5] $\mathcal{A}_{ij} = 1$ if there is an edge in $\mathcal{G}$ from v_i to v_j, and $\mathcal{A}_{ij} = 0$ otherwise.

depending on the structure $\mathcal{G}$. Let $b_i := \sum_j [\mathcal{A}^q]_{ij}$ denote the total number of paths in $\Theta_{[q]}$ from vertex i. Then, the number of operations to determine all $\theta \in \Theta_{[q]}$ such that $v_0 = i$ is at most $b_i \mathcal{C}_{path}$. Let $b := \sum_i b_i;\ i \in (\mathcal{M}^s \bigcup \mathcal{M}^{ns})$. Algorithm 2 invokes Algorithm 1, and further, computes sets of states that are reachable at time k after q mode changes. Let $\mathcal{C}_{reach}$ denote the complexity of computing the sets of reachable states. Under reasonable assumptions on the structures of the sets of controls and initial states, approximations of sets of reachable states can be calculated with arbitrary precision using procedures that are linear in the time horizon(k) and polynomial in the dimension of the state space(n) ([57], [58], [59]). Let $\mathcal{C}_{mult}$ denote the complexity of the matrix-vector multiplication Cx, $x \in X(\cdot, \cdot)$. This typically takes $O(pn)$ operations.

Proposition 5.19. *1.* $\mathcal{C}_{alg1} \leq k_1 b \mathcal{C}_{path} + k_2 \mathcal{C}_{sum} + k_3 \mathcal{C}_{pow}$;

2. $\mathcal{C}_{alg2} \leq k_1' \mathcal{C}_{alg1} + k_2' \mathcal{C}_{reach} + k_3' \mathcal{C}_{mult}$ *for constants* $k_1, k_2, k_3, k_1', k_2', k_3' > 0$.

5.6 Examples

Example 5.20. *Figure 5.1 shows the allowed mode transitions in a switched system with five modes and the minimum and maximum dwell times in each mode. Notice that the system is nonblocking. Let* $\mathcal{M}^s = \{1\}$ *and* $\mathcal{M}^{ns} = \{3, 5\}$. *Let* $q = 2$. *Then,* $\Theta_{[2]} \supset \{(1,2,3), (1,2,4), (1,5,2), (3,5,2), (5,2,3), (5,2,4)\}$ *(only considering paths starting from* $\mathcal{M}^s$ *or* $\mathcal{M}^{ns}$*). For* $k = 6$, $\mathcal{M}^s$ *is* $(6,2)$*-IMO w.r.t.* $\mathcal{M}^{ns}$, *since for all paths in Figure 5.1 of length* $q + 1 = 3$ *starting from mode* 1 *such that the dwell times in the modes along the path sum to* 6, *there is a corresponding path of length*

Algorithm 1 Verifying (k, q)-IMO

Input: $\mathcal{M}^s, \mathcal{M}^{ns}, k, q, [\tau^a_{d_{min}}, \tau^a_{d_{max}}]$ for each mode a, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ specifying allowed mode transitions.

Output: YES, if $\mathcal{M}^s$ is (k, q)-IMO w.r.t. $\mathcal{M}^{ns}$; NO, if not.

1: $\mathcal{M}^s_k = \mathcal{M}^{ns}_k = \emptyset$

2: $\Theta^{\mathcal{M}^s}_{[q]} := \{v_0 e_1 \dots v_q \in \mathcal{G}\colon\ v_0 \in \mathcal{M}^s\}$

3: $\Theta^{\mathcal{M}^{ns}}_{[q]} := \{v_0 e_1 \dots v_q \in \mathcal{G}\colon\ v_0 \in \mathcal{M}^{ns}\}$

4: **for** each $\theta \in \Theta^{\mathcal{M}^s}_{[q]}$ **do**

5: **if** $(\sum_\theta \tau_d = k)$ **then**

6: $\mathcal{M}^s_k = \mathcal{M}^s_k \bigcup v_q$

7: **end if**

8: **end for**

9: **for** each $\theta' \in \Theta^{\mathcal{M}^{ns}}_{[q]}$ **do**

10: **if** $(\sum'_\theta \tau_d = k)$ **then**

11: $\mathcal{M}^{ns}_k = \mathcal{M}^{ns}_k \bigcup v'_q$

12: **end if**

13: **end for**

14: **if** $\mathcal{M}^s_k \subseteq \mathcal{M}^{ns}_k$ **then** ▷ *Theorem 5.10*

15: return YES

16: **else**

17: return NO

18: **end if**

Algorithm 2 Verifying (k, q)-IMSO

Input: $\bar{X}_s = (X_s, \mathcal{M}^s), \bar{X}_{ns} = (X_{ns}, \mathcal{M}^{ns}), k, q, \mathcal{G} = (\mathcal{V}, \mathcal{E})$, $[\tau^a_{d_{min}}, \tau^a_{d_{max}}]$ for each mode a.

Output: YES, if $\bar{X}_s$ is (k, q)-IMSO w.r.t. $\bar{X}_{ns}$; NO, if not.

1: $ANS \leftarrow$ Result of Algorithm 1
2: **if** $ANS == YES$ **then**
3: Compute $X_s(k, q)$ using Equation (5.4)
4: Compute $X_{ns}(k, q)$ using Equation (5.5)
5: **if** $CX_s(k, q) \subseteq CX_{ns}(k, q)$ **then**
6: return YES ▷ *Theorem 5.14*
7: **else**
8: return NO
9: **end if**
10: **else**
11: return NO
12: **end if**

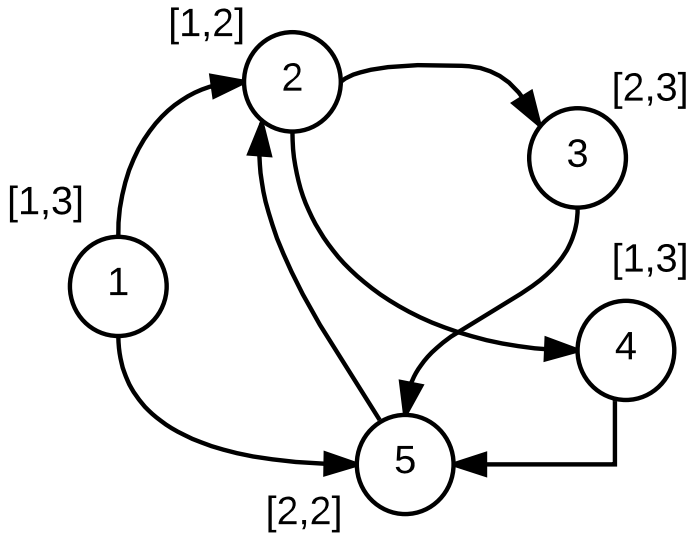


Figure 5.1: Switched system considered in Example 5.20

3 *starting from modes* 3 *or* 5 *also satisfying the dwell constraints along the path, such that* $\mathcal{M}_6^s = \mathcal{M}_6^{ns}$. *However,* $\mathcal{M}^s$ *is not* $(3,2)$*-IMO w.r.t.* $\mathcal{M}^{ns}$. *Consider the path* $(1,2,3)$ *starting from* $\mathcal{M}^s$ *with* $\tau_d^1 = 1, \tau_d^2 = 1, \tau_d^3 = 1$. *There does not exist a corresponding path starting from* $\mathcal{M}^{ns}$ *such that* $\mathcal{M}_3^{ns} = 3$.

Example 5.21. *Consider the system in Figure 5.1, with the following additional specifications:*

$$A(1) = A(3) = A(5) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \; A(2) = A(4) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

$$B(1) = B(2) = B(3) = \begin{pmatrix} 1 & 1 & 0 \end{pmatrix}^\intercal; \; B(4) = B(5) = \begin{pmatrix} 0 & 1 & 1 \end{pmatrix}^\intercal$$

Table 5.1: Example 5.21: $(4,1)$-IMSO

$\Theta_{[1]}$	$(\tau_d', \tau_d'') : \tau_d' + \tau_d'' = 4$	$\mathcal{M}_0\mathcal{M}_1\mathcal{M}_2\mathcal{M}_3$
1 2	2, 2	1 1 2 2
1 2	3, 1	1 1 1 2
5 2	2, 2	5 5 2 2
1 5	2, 2	1 1 5 5
1 5	3, 1	1 1 1 5
3 5	2, 2	1 1 5 5
3 5	3, 1	3 3 3 5

Let $\bar{X}_s = (X_s; \mathcal{M}_s) := \left(\begin{pmatrix}1 & 0 & 0\end{pmatrix}^\intercal, \begin{pmatrix}0 & 0 & 1\end{pmatrix}^\intercal; 1\right)$, $\bar{X}_{ns} = (X_{ns}; \mathcal{M}_{ns}) := \left(\mathbb{R}^3 \setminus X_s; 3, 5\right)$, *and* $C = \begin{pmatrix}1 & 1 & 1\end{pmatrix}$. *For* $q = 1$ *and* $k = 4$, *the set of switching sequences of length* 1 *starting from* $\mathcal{M}_s$ *or* $\mathcal{M}_{ns}$, *the dwell times along the path that sum to* k, *and the modes of the system is given in table 5.1. The output at time* 4 *for an initial state* $x(0)$ *and controls* $u(0), u(1), u(2), u(3)$ *for each of the mode sequences in table 5.1 is of the form:*

$$y(4) = \begin{pmatrix}1 & 1 & 1\end{pmatrix} x(0) + 2u(0) + 2u(1) + 2u(2) + 2u(3)$$

From Table 5.1, it is evident that for every initial secret mode, there is an initial nonsecret mode such that $\mathcal{M}_4^s = \mathcal{M}_4^{ns}$. *Further, in the cases when* $k = 4$ *happens to be a switching time, under Assumption 5.8, the adversary observes* $\mathcal{M}_{k^-}$. *Consider* $x'(0) = \begin{pmatrix}0 & 1 & 0\end{pmatrix}^\intercal \in X_{ns}$. *Then, for* $y_s(4) = y_{ns}(4)$, *the following condition will have to be satisfied: for every* $u_s(0), u_s(1), u_s(2), u_s(3)$, *there must exist*

$u_{ns}(0), u_{ns}(1), u_{ns}(2), u_{ns}(3)$, *such that:*

$$\sum_{j=0}^{3} u_s(j) = \sum_{j=0}^{3} u_{ns}(j)$$

If the controls are allowed to take arbitrary values in $\mathbb{R}$, *then* $\bar{X}_s$ *will be* $(4,1)$*-IMSO with respect to* $\bar{X}_{ns}$. *However, if* $X_{ns} := \{\begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}^\intercal : x_i \in \{0,1\},\ i = 1,2,3\}$ *and* $0 \leq u_s(j) \leq p_1$, $0 \leq u_{ns}(j) \leq p_2$ *for* $j = 0,1,2,3$, *and* $p_1 > p_2$, *then* $\bar{X}_s$ *will not be* $(4,1)$*-IMSO w.r.t.* $\bar{X}_{ns}$.

Example 5.22. *Consider the system in Figure 5.1, with the following additional specifications [60]:*

$$A(1) = A(2) = \begin{pmatrix} 18 & -4 \\ 25 & -10 \end{pmatrix};\ A(3) = A(4) = A(5) = \begin{pmatrix} -2 & 4 \\ 7 & -6 \end{pmatrix}$$

$$B(1) = B(2) = \begin{pmatrix} 16 & 24 \end{pmatrix}^\intercal;\ B(3) = B(4) = B(5) = \begin{pmatrix} 8 & 12 \end{pmatrix}^\intercal$$

Let $\bar{X}_s = (X_s; \mathcal{M}^s) := \big(\{\begin{pmatrix} 0 & 1 \end{pmatrix}^\intercal \cup \{\begin{pmatrix} x_1 & x_2 \end{pmatrix}^\intercal : x_1 = -1\}; 1\big)$, $\bar{X}_{ns} = (X_{ns}; \mathcal{M}^{ns}) := \big(\mathbb{R}^3 \setminus X_s; 3, 5\big)$, *and* $C = I_{2\times 2}$. *For* $q = 1$ *and* $k = 3$, *the set of switching sequences of length* 1 *starting from* $\mathcal{M}_s$ *or* $\mathcal{M}_{ns}$, *the dwell times along the path that sum to* k, *and the modes of the system is given in Table 5.2. If the controls are restricted in the following way:* $(2u_s(0), u_s(1), u_s(2)) = (u_{ns}(0), u_{ns}(1), u_{ns}(2)) = (u_0, u_1, u_2)$, *where* $u_0, u_1, u_2 \in \mathbb{R}$, *then for* $x(0) = \begin{pmatrix} 0 & 1 \end{pmatrix}^\intercal \in X_s$, *the state which ensures* $y_s(3) = y_{ns}(3)$ *is* $x'(0) = \begin{pmatrix} -1 & 0.23 \end{pmatrix}^\intercal$. *However,* $x'(0) \notin X_{ns}$ *(in fact,* $x'(0) \in X_s$*), which means that* $\bar{X}_s$ *is not* $(3,1)$*-IMSO w.r.t.* $\bar{X}_{ns}$.

Example 5.23. *Consider the system in Figure 5.1 with* $A(\cdot), B(\cdot), C$ *matrices as considered in example 5.21. Let* $X_s = (\begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^\intercal, \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^\intercal)$ *and* $X_{ns} = \mathbb{R}^3 \setminus$

Table 5.2: Example 5.22: $(3,1)$-IMSO

$\Theta_{[1]}$	$(\tau_d', \tau_d'') : \tau_d' + \tau_d'' = 3$	$\mathcal{M}_0\mathcal{M}_1\mathcal{M}_2\mathcal{M}_3$
1 2	(1, 2)	1 2 **2** 3
1 2	(2, 1)	1 1 **2** 3
1 2	(2, 1)	1 1 2 **2**
5 2	(2, 1)	5 5 **2** 3
5 2	(2, 1)	5 5 2 **2**
1 5	(1, 2)	1 5 **5** 2
1 5	(2, 1)	1 1 5 **5**
3 5	(2, 1)	3 3 5 **5**

X_s. The output at time 4 for any allowed switching sequence of length 1 has the same form as in example 5.21. If the control can take any value in $\mathbb{R}$, X_s will be $(4,1)_{\mathcal{M}}$-ISO with respect to X_{ns}.

Chapter 6: A Structured Systems Approach to Resilience to Denial-of-Service Attacks

This chapter presents a characterization of the resilience of a cyberphysical system (CPS) modeled as a linear structured system to denial-of-service (DoS) attacks. The resilience of the system to an attack is interpreted in terms of unmatched vertices in maximum matchings of bipartie graph, and connected components of directed graph representations of the system under attack. We further go on to establish conditions under which a system will be structurally resilient to a state feedback integrity attack if it is already structurally resilient to a DoS attack. We conclude the chapter by proposing a characterization of structural resilience for switched structured systems.

The problem to be solved is stated in Section 6.1, and Section 6.2 summarizes some of the results on structural controllability from the literature. The main results of the chapter are presented in Sections 6.3 and 6.4. The computational complexity of the results presented in this chapter only rely on determining maximum matchings in bipartite graphs and strongly connected components of directed graphs. Section 6.5 makes a note of this. Section 6.6 presents several examples to illustrate our results. We extend our work to characterize the structural resilience of switched

linear systems to DoS attacks in Section 6.7.

6.1 Problem Formulation

Consider the linear structured system

$$\dot{x}(t) = [A]x(t) + [B]u(t) \tag{6.1}$$

where, $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^p$, $[A] \in \{0, *\}^{n \times n}$ and $[B] \in \{0, *\}^{n \times p}$.

Removing the explicit dependence on t, and rewriting $u(t)$ in Equation (6.1) as $u = \begin{pmatrix} u_1 & \dots & u_d & u_{d+1} & \dots & u_p \end{pmatrix}^T$, we will use $u_{def} \in \mathbb{R}^d$ and $u_{att} \in \mathbb{R}^a$ (with $a := p - d$) to collectively denote the elements $\begin{pmatrix} u_1 & \dots & u_d \end{pmatrix}^T$ and $\begin{pmatrix} u_{d+1} & \dots & u_p \end{pmatrix}^T$ respectively. The sets u_{def} and u_{att} represent the input vertices accessible to the system (defender) and attacker respectively. The structural resilience of the system to the different types of attacks discussed in this chapter will depend, to a large extent, on the cardinality of the vertex sets u_{def} and u_{att} (that is, on d and a) *vis-à-vis* the number of unmatched state vertices.

The system model is now:

$$\dot{x}(t) = [A]x(t) + [B_{def}]u_{def}(t) + [B_{att}]u_{att}(t) \tag{6.2}$$

Assumption 6.1. *The sets of state vertices that can be directly connected to inputs controlled by the defender and attacker are disjoint. Let $\mathcal{X}_{def}$ and $\mathcal{X}_{att}$ denote these sets. That is, the inputs in u_{def} can only be connected to state vertices in $\mathcal{X}_{def}$ and inputs in u_{att} can only be connected to state vertices in $\mathcal{X}_{att}$.*

This is a reasonable assumption in that it means that the defender (system)

will have limited or restricted access to only a subset of the state vertices which it can 'directly' control ($\mathcal{X}_{def}$) in order to be resilient to an attack. Further, once the attacker has gained access to the system by manipulating a subset of the inputs, thereby influencing a set of states ($\mathcal{X}_{att}$), it can be assumed that it retains access to these states while the defender tries to ensure that the system is resilient to the attack by appropriately controlling the other states ($\mathcal{X}_{def}$).

In the structural setting, this would imply that $[B_{def}]$ will have fixed zeros in rows corresponding to $\mathcal{X}_{att}$, and $[B_{att}]$ will have fixed zeros in rows corresponding to $\mathcal{X}_{def}$.

The resilience of the CPS will be characterized in terms of the structural controllability of the system in the face of an attack. This will subsequently be shown to be equivalent to formulating conditions on the non-attacked nodes in the graph of the structured system. We shall assume that the sets $\mathcal{X}_{def}$ and $\mathcal{X}_{att}$ remain unchanged with time. The system will be structurally resilient to an attack if it is structurally controllable when it has 'access' to only some components of the state vector, while the remaining components of the state vector (those under 'attack') cannot be directly accessed by it.

We study the structural resilience of the system to two kinds of attacks:

1. During a *denial-of-service attack*, access to the inputs in u_{att} is blocked by the attacker. Our goal will be to formulate conditions for structural resilience assuming that only inputs in u_{def} are available.

2. An *integrity attack* corresponds to the situation when a state feedback strategy

(explained in greater detail in Section 6.4) is not implemented appropriately. That is, only some components of the input are faithfully reproduced, while the remaining are arbitrary.

At this point, we would like to point out two different ways of viewing a denial-of-service attack. In the computer science and cybersecurity literature, a denial-of-service attack typically occurs when an adversary 'floods' the system with spurious inputs or requests, thereby ensuring that the system cannot address 'genuine' service requests. In our framework, however, we view a denial-of-service attack in terms of ensuring the structural resilience of the system when certain inputs (corresponding to the attacker) are disregarded. A spurious input in our framework is assumed to not be of use, and is therefore set to zero. We then want to see if the system can satisfy certain properties in order to be structurally resilient to this attack in the absence of these inputs.

Problem 6.2. *Given the system (6.2) with* $([A], [B])$ *structurally controllable before an attack, characterize its structural resilience to denial-of-service (DoS) and integrity attacks.*

This problem for a given numerical realization, (A, B) was solved in [31] for several attack scenarios, and is stated below.

Problem 6.3. *Given a particular numerical realization of the structured system, (6.2), characterize its resilience to DoS and integrity attacks.*

Structural controllability of the system before the attack ensures that the cardinality of the input set is greater than the minimum number of inputs needed to

ensure structural controllability [61]. This will play an important role in determining the structural resilience of the system subsequent to an attack.

6.2 Structural Controllability

Before stating and proving our main results, we take a detour to present some intermediate results that will be needed in subsequent sections.

Definition 6.4. *Given the digraph representation of the system in (6.2), we define the following terms [32, 49]:*

- State stem*: simple path comprising only state vertices, or an isolated state vertex.*
- Input stem*: an input vertex linked to the root of a state stem.*
- *A* chain *is a single cycle or a group of disjoint cycles (composed of state vertices) connected to each other in a sequence.*
- *A* top assignable SCC *of* $\mathcal{D}([A]) = (\mathcal{X}, \mathcal{E}_A)$ *is a non-top-linked SCC which contains at least one right unmatched vertex in a maximum matching. Since a maximum matching is not unique, whether an SCC is top assignable will depend on the maximum matching under consideration.*
- *The* maximum top assignability index *of* $\mathcal{D}([A])$ *is the maximum number of top assignable SCCs among the maximum matchings associated with* $\mathcal{B}([A])$.
- *A* dedicated input *is an input that is connected to exactly one state. This corresponds to a column of* $[B]$ *having only one nonzero entry.*

The next three results present conditions for structural controllability, and lower bounds on the number of control inputs and the links from these inputs to unmatched state vertices and top assignable SCCs in order to ensure structural controllability. The reader is directed to the references cited for proofs of these results.

Theorem 6.5. *[32, 49] The following are equivalent:*

1. $([A], [B])$ *is structurally controllable.*
2. *Every state vertex is the end of a* $\mathcal{U}-$*rooted path and there exists a union of a* $\mathcal{U}-$*rooted path family and a cycle family containing all vertices in* $\mathcal{X}$*.*
3. *Every right unmatched vertex of a maximum matching of* $\mathcal{B}([A], [B])$ *is connected to a distinct input, and one state vertex from each non-top-linked SCC of* $\mathcal{D}([A])$ *is connected to some input.*

Theorem 6.6. *[61] Let* m *be the number of right unmatched vertices in a maximum matching of* $\mathcal{B}([A])$*. Then, the minimum number of inputs needed to ensure structural controllability is one, if* $m = 0$*, and* m*, otherwise.*

Theorem 6.7. *[32] Let* β *be the number of non-top-linked SCCs and* α *the maximum top assignability index in* $\mathcal{D}([A])$*. Then, the minimum number of input-state links needed to ensure structural controllability is* $m + \beta - \alpha$*.*

The following example shows why the cardinality of the input set and the number of links from the inputs to the states is important in ensuring structural controllability.

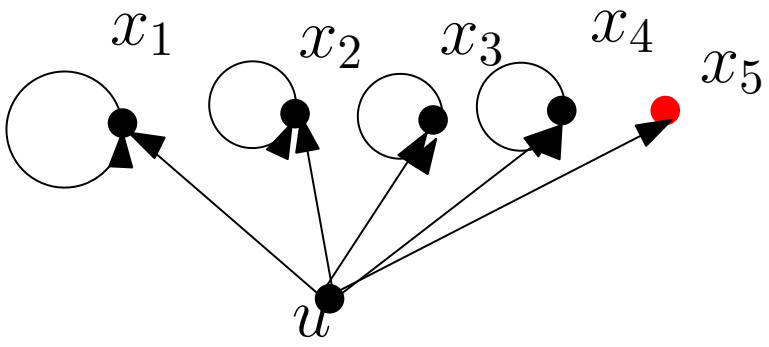


Figure 6.1: x_5 is unmatched. Each x_i is an SCC

Example 6.8. *Let $A = diag\{*, *, *, *, 0\}$, as shown in Figure (6.1). Then, $\mathcal{B}(A)$ has one right unmatched vertex, x_5, giving $m = 1$. Therefore, one input connected to x_5 is required. However, the system has five strongly connected components, giving $\beta = 5$. $\alpha = 1$ since only one of the SCCs is top assignable. Therefore, the minimum number of links from inputs to states to ensure structural controllability is $1+5-1 = 5$. Indeed, the input must be connected to all the states (to ensure that no state is unreachable).*

Example 6.9. *Let us turn our attention to the first maximum matching (that is, the edges in blue in Figure 2.3) of $\mathcal{B}([A])$ in Example 2.13. We see that w_3 and w_5 in $\mathcal{B}([A])$ correspond to v_3 and v_5 in $\mathcal{D}([A])$, which both belong to non-top linked SCCs, which makes these SCCs top-assignable.*

From the preceding discussion, it is evident that one way to reduce the number of input to state links needed to ensure structural controllability is to determine a maximum matching of $\mathcal{B}([A])$ in a way that as many right unmatched vertices belong to non-top linked SCCs. This will ensure that $\beta - \alpha$ is 'close' to zero, and

the minimum number of input to state links needed is 'close' to m, the number of right unmatched vertices.

We conclude this section by defining what it means for an attack to be structurally successful. The system post-attack is defined to be the configuration for which structural controllability has to be ensured when only vertices in $\mathcal{X}_{def}$ can be connected to inputs.

Definition 6.10. *An attack on the system is said to be* structurally successful *if the system post-attack is not structurally controllable.*

The system is structurally resilient *to the attack if the system post-attack is structurally controllable.*

6.3 Structural Resilience to DoS Attacks

During a DoS attack, the attacker blocks access to inputs in u_{att}. The system still has access to inputs in u_{def}. Structurally, this corresponds to designing $[B_{def}]$, with $[B_{att}] = 0$, to ensure structural resilience. The system model is:

$$\dot{x}(t) = [A]x(t) + [B_{def}]u_{def}(t) \tag{6.3}$$

We assume that the number of right unmatched vertices, m, in a maximum matching of $\mathcal{B}([A])$ is nonzero. Let m_{def} and m_{att} be the number of right unmatched vertices in $\mathcal{B}([A])$ corresponding to $\mathcal{X}_{def}$ and $\mathcal{X}_{att}$ (thus, $m_{def} + m_{att} = m$). Let $l(P \to Q)$ denote the set of links from P to Q. Lemma 6.11 provides a sufficient condition for a DoS attack to be successful.

Lemma 6.11. *A DoS attack on the system in (6.2) is structurally successful if:*

1. $p \geq m + \beta - \alpha$ OR

2. $p \geq m$ *and* $|l(u \to \mathcal{X})| \geq m + \beta - \alpha$

and $d < m_{def}$*, where* p *and* d *respectively denote the dimensions of* u *and* u_{def}*.*

Proof. $([A], [B])$ is assumed to be structurally controllable before an attack occurs. This means that there are at least m vertices in u and $m + \beta - \alpha$ links from u to $\mathcal{X}$, which gives the inequalities in 1) and 2). The last inequality is obtained from the fact that if, after an attack, the number of available inputs is less than the number of right unmatched vertices in $\mathcal{B}([A])$ corresponding to $\mathcal{X}_{def}$, then $([A], [B_{def}])$ will not be structurally controllable. Thus, the system will not be able to mitigate the effect of the attack. □

The conditions of Lemma 6.11 are not necessary, for an attack could be successful even in the case when $p \geq m + \beta - \alpha$ and $d \geq m_{def}$. Though the minimum input requirement is satisfied, the conditions to ensure structural controllability must be carefully checked.

Lemma 6.12. *If* $d \geq m_{def}$*, a DoS attack is structurally successful if:*

1. *There exists an unreachable state from the vertices of* u_{def}*. OR*

2. *There does not exist a disjoint union of* u_{def} *rooted path families and cycle families covering all the states. OR*

3. $|l(u_{def} \to \mathcal{X})| < m_{def} + \beta - \alpha$*. OR*

4. *Every maximum matching of* $\mathcal{B}([A])$ *has a right unmatched vertex in* $\mathcal{X}_{att}$. *OR*

5. *There is a non-top-linked SCC in* $\mathcal{D}([A])$ *comprising only vertices from* $\mathcal{X}_{att}$.

Proof. The first three conditions follow from Theorem 6.5 and Theorem 6.7. The last two follow from the fact that inputs from u_{def} cannot be assigned to vertices in $\mathcal{X}_{att}$. □

Gathering the results in Lemmas 6.11 and 6.12, we have the following result:

Theorem 6.13. *Given* $[A]$ *and the indices of* $[B]$ *corresponding to* $[B_{def}]$, *the system in (6.3) is* structurally resilient to a DoS attack *if and only if* $([A], [B_{def}])$ *is structurally controllable and:*

1. *there exists a maximum matching of* $\mathcal{B}([A])$ *that does not contain a right unmatched vertex in* $\mathcal{X}_{att}$;

2. $\mathcal{D}([A])$ *does not have a non-top linked SCC comprising vertices from only* $\mathcal{X}_{att}$.

Proof. If $([A], [B_{def}])$ is not structurally controllable, then at least one of the first two conditions of Lemma 6.12 will not be satisfied, and the system will not be structurally resilient to a DoS attack.

Now, let $([A], [B_{def}])$ be structurally controllable. Any right unmatched vertex in $\mathcal{X}_{att}$ or a non-top-linked SCC consisting of only vertices in $\mathcal{X}_{att}$ will have to be assigned to a control in u_{def}. This would violate the assumption that u_{def} can only be connected to states in $\mathcal{X}_{def}$. This means that the system will not be structurally resilient to a DoS attack. If $([A], [B_{def}])$ is structurally controllable, the absence of right unmatched vertices or non-top-linked SCCs comprised exclusively of vertices

from $\mathcal{X}_{att}$ corresponds to the existence of a control configuration such that $d \geq m_{def}$ and $|l(u_{def} \to \mathcal{X}_{def})| \geq m_{def} + \beta - \alpha$, which ensures structural resilience to a DoS attack. □

Remark 6.14. *This is different from the minimal controllability problem, where, given* $[A]$*, we need to find the sparsest* $[B]$ *such that* $([A], [B])$ *is structurally controllable. In our framework, if the number of columns of* $[B_{def}]$ *exceeds a certain threshold (*m*), then the only remaining task is to fill in the 'missing links' to ensure structural controllability. Conversely, structural controllability cannot be achieved if the number of columns of* $[B_{def}]$ *is below this threshold.*

6.4 Structural Resilience to Integrity Attacks

State feedback is a popular control strategy in which the closed-loop poles of a system can be appropriately placed in order to control the characteristics of the response of the system. The control $u(t)$ is a linear function of the state $x(t)$, that is, $u(t) = Kx(t)$, and state feedback corresponds to placing the eigenvalues of the modified system matrix $(A + BK)$ in order to achieve a desired response. If the system is controllable, these closed-loop poles can be arbitrarily placed.

During an integrity attack, only the control signals corresponding to the system maintain their integrity, while those of the attacker are arbitrary. That is, only the part of the input corresponding to u_{def} is faithfully reproduced, while the part corresponding to u_{att} is arbitrary. The attacker is successful if the system is structurally controllable without needing to connect inputs to $\mathcal{X}_{def}$.

Remark 6.15. *Notice that in contrast to Definition 6.10, this notion of resilience depends on the ability to connect inputs to $\mathcal{X}_{att}$, and not $\mathcal{X}_{def}$.*

With $[A_{def}] := ([A] + [B_{def}][K_{def}])$, the system model is:

$$\dot{x}(t) = [A_{def}]x(t) + [B_{att}]u_{att}(t) \tag{6.4}$$

The following result presents conditions for the system to be structurally resilient in the face of an integrity attack.

Theorem 6.16. *The system in (6.4) is* structurally resilient to an integrity attack *if and only if there is a right unmatched vertex in $\mathcal{X}_{def}$ in every maximum matching of $\mathcal{B}([A_{def}])$ or there exists a non-top-linked SCC of $\mathcal{D}([A_{def}])$ comprising exclusively vertices in $\mathcal{X}_{def}$.*

Proof. This follows from Assumption 6.1. The attacker will not be able to ensure structural controllability of (6.4) if some vertex in $\mathcal{X}_{def}$ has to be assigned to a control in u_{att}. □

Alternatively, through a measurement or other means (eg. changing a controller parameter), an attacker might gain access to a state. We label this scenario a *state feedback integrity attack*. In this case, $u_{att}(t) = K_{att}x(t)$, while u_{def} is arbitrary. For structural systems, this corresponds to designing $[B_{def}]$ to ensure structural controllability. With $[A_{att}] := ([A] + [B_{att}][K_{att}])$, we have:

$$\dot{x}(t) = [A_{att}]x(t) + [B_{def}]u_{def}(t) \tag{6.5}$$

Let m_A and $m_{A_{att}}$ denote the number of right unmatched vertices in a maximum matching of $\mathcal{B}([A])$ and $\mathcal{B}([A_{att}])$. Let $\mathcal{Z}(H)$ denote the *zero structure* of a

structured matrix H. A zero structure is therefore a particular configuration of 0s and $*$s. For structured matrices H and H' of the same dimension, we write $\mathcal{Z}(H') \subseteq \mathcal{Z}(H)$ whenever $h'_{ij} = 0$ in $[H']$ implies $h_{ij} = 0$ in $[H]$.

The main result of this section provides certain guarantees on the structural resilience of the system to a state feedback integrity attack depending on its resilience to a DoS attack.

Theorem 6.17. *If the system in (6.2) is structurally resilient to a DoS attack for some $[B_{def}]$ with zero structure $\mathcal{Z}(B_{def})$, then there exists a $[B'_{def}]$ which satisfies $\mathcal{Z}(B'_{def}) \subseteq \mathcal{Z}(B_{def})$ for which it will also be structurally resilient to a state feedback integrity attack. Moreover, if*

$$m_{A_{att}} + \beta_{A_{att}} - \alpha_{A_{att}} \leq m_A + \beta_A - \alpha_A \tag{6.6}$$

for some $[B_{def}]$ corresponding to the DoS case, then the same $[B_{def}]$ will ensure structural resilience to a state feedback integrity attack (here, m, β, and α are as in Theorem (6.7)).

Proof. Addition of edges corresponding to $[B_{att}][K_{att}]$ to $[A]$ will ensure that the number of right unmatched vertices in a maximum matching of $[A_{att}]$ can only be as many as the number of right unmatched vertices in a maximum matching of $[A]$. Therefore, $m_{A_{att}} \leq m_A$. From Theorem 6.6 and Equation (6.3), structural resilience to a DoS attack implies $d \geq m_A$ holds. This gives $d \geq m_{A_{att}}$.

If the inequality (6.6) holds, then $|l(u_{def} \to \mathcal{X})| \geq m_{A_{att}} + \beta_{A_{att}} - \alpha_{A_{att}}$, and no additional links between inputs and states will have to be added to ensure structural controllability, and $[B'_{def}] = [B_{def}]$. Additional links will have to be added if (6.6)

does not hold. This corresponds to adding free parameters to $[B_{def}]$, giving $[B'_{def}]$, which satisfies $\mathcal{Z}(B'_{def}) \subseteq \mathcal{Z}(B_{def})$. □

If the system is structurally resilient to DoS attacks and (6.6) holds, the same configuration, i.e. $[B_{def}]$, will automatically make it structurally resilient to state feedback integrity attacks. However, there might be a cost involved in 'turning on' controls to ensure structural controllability, and the system might want to be resilient with the lowest cost. This would entail choosing a subset of the columns of $[B_{def}]$, indexed by $\mathcal{I}$, to maintain structural controllability of $([A_{att}], [B_{def}(\mathcal{I})])$, while minimizing the cost of the control action.

It is important to note that structural resilience to DoS attacks guarantees structural resilience only to state feedback integrity attacks. It does not, in general, ensure structural resilience to arbitrary integrity attacks.

6.5 Computational Complexity

The computational complexity of determining the structural resilience of the system under both denial-of-service and integrity attacks depends on two factors:

1. the complexity of determining strongly connected components in a directed graph; and,

2. the complexity of determining a maximum matching in a bipartite graph.

Strongly connected components in a directed graph can be computed using Tarjan's algorithm [62]. This procedure uses depth-first search to push nodes onto

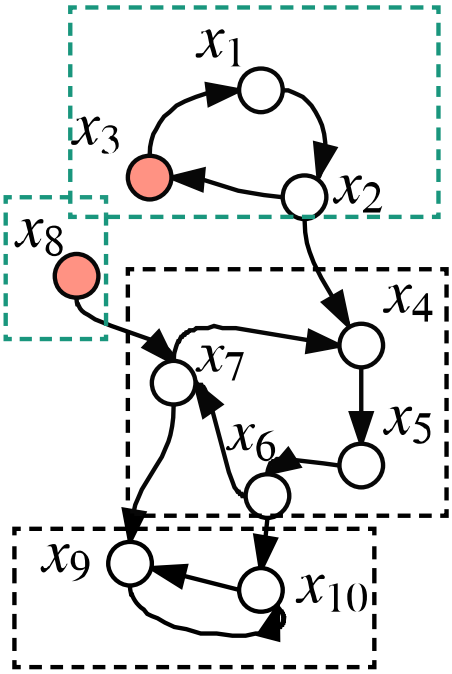
a stack, and a bookkeeping procedure to ensure that nodes are visited exactly once. The complexity of this procedure in the worst-case is linear in the cardinality of the vertices and edges of the directed graph, that is $\mathcal{O}(|\mathcal{V}| + |\mathcal{E}|)$.

A maximum matching of a bipartite graph can be determined by the Hopcraft-Karp algorithm [63]. Each phase of this algorithm comprises one breadth-first search and one depth-first search. Every phase increases the size of a partial matching by determining *augmented paths*, which are paths of the graph that start and end in vertices that do not belong to the matching, and in which the edges alternate between belonging to the matching and not belonging to it. The procedure terminates when the graph has no augmenting paths. The complexity of this procedure in the worst-case is $\mathcal{O}(\sqrt{|\mathcal{V}|}|\mathcal{E}|)$. An extension for determining maximum matchings in more general graphs with the same computational complexity was presented in [64].

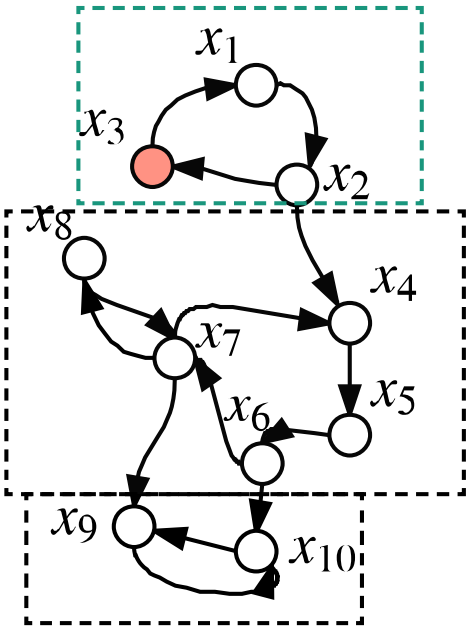
6.6 Examples

In this section, we present illustrative examples to illustrate the results in Sections 6.3 and 6.4. In all the examples, we will assume that $x_1, \ldots, x_6 \in \mathcal{X}_{def}$ and $x_7, \ldots, x_{10} \in \mathcal{X}_{att}$.

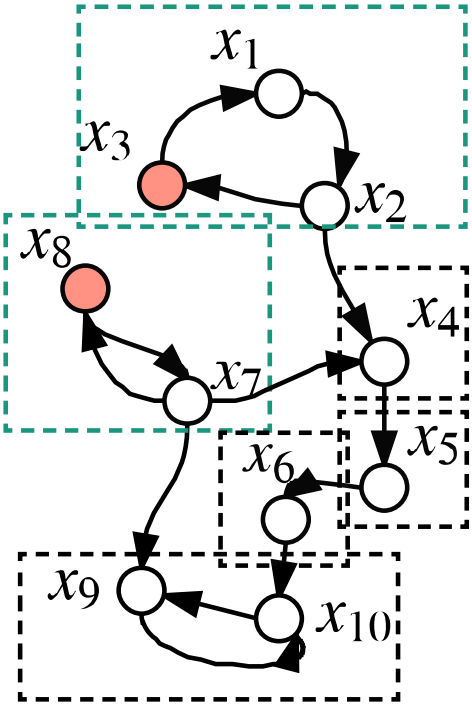
Example 6.18. *(**DoS Attack Resilience**) Figure (6.2a) shows the directed graph representation of a system, $\mathcal{D}([A])$. The SCCs are $(x_1, x_2, x_3), (x_8), (x_4, x_5, x_6, x_7)$, and (x_9, x_{10}). Inputs need to be asssigned to the first two SCCs, since they are not top linked. Every maximum matching of $\mathcal{D}([A])$ will have $x_8 \in \mathcal{X}_{att}$ as a right unmatched vertex. Thus, the system is not structurally resilient to a DoS attack.*



(a)



(b)



(c)

Figure 6.2: Structural Resilience to DoS Attack

Now, add the edge $x_7 \to x_8$ to the digraph as shown in Figure (6.2b). The SCCs are $(x_1, x_2, x_3), (x_4, x_5, x_6, x_7, x_8)$, and (x_9, x_{10}). Only the first SCC is not top linked, and there is only one right unmatched vertex in every maximum matching, and for some such matching, it is not in $\mathcal{X}_{att}$. Therefore, this system is structurally resilient to a DoS attack.

If the edge, $x_6 \to x_7$ is removed as shown in Figure (6.2c), then (x_7, x_8) becomes a non-top-linked SCC, which necessitates the assignment of a control to it, making the system vulnerable to a DoS attack.

Example 6.19. *(**State Feedback Integrity Attack Resilience**) In Figure (6.2a), if a state feedback adds edges $x_7 \to x_8$, $x_9 \to x_8$, or $x_{10} \to x_8$, then there exists a maximum matching of $\mathcal{D}([A_{att}])$ with no right unmatched vertices or non-top-linked SCCs in $\mathcal{X}_{att}$, ensuring structural resilience to a state feedback attack.*

For the system in Figure (6.2b), any state feedback $[K_{att}]x$ will add edges to the set $\{x_7, x_8, x_9, x_{10}\}$. We know that this graph does not have right unmatched vertices in $\mathcal{X}_{att}$. This ensures structural resilience with the same $[B_{def}]$ as in the DoS case.

It will be helpful to think of determining resilience to a state feedback integrity attck in the context of a power system. The term $[K_{att}]x$ can be thought of as 'adding wires' to the system between nodes in the system (generators, loads, etc.) in order to create redundancies. The addition of these wires will serve to reduce (or at least, not increase) the number of right unmatched vertices in a maximum matching of a bipartite graph representation of the system, which, as we have seen determines the resilience of the system to an attack. While it is difficult to ensure resilience to

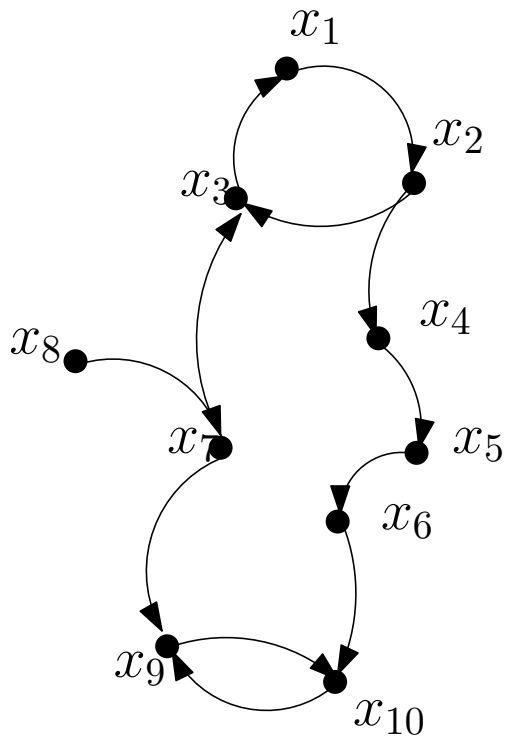


Figure 6.3: Structural Resilience to Integrity Attack

such attacks at *run-time*, it is also the case that such systems are subject to regular inspections. Such vulnerabilities that might become evident during an inspection might be fixed during a maintenance period.

Example 6.20. ***(Integrity Attack Resilience)*** *For $[A_{def}]$ given by Figures (6.2a, 6.2b, 6.2c), there is a non-top-linked SCC with vertices only in $\mathcal{X}_{def}$. Since controls in u_{att} cannot be assigned to vertices in $\mathcal{X}_{def}$, the systems are structurally resilient to an integrity attack.*

However, if $[A_{def}]$ is as in Figure (6.3), all maximum matchings will have x_8 as a right unmatched vertex, and x_8 will be the only non-top-linked SCC. An attacker can ensure structural controllability of the system by supplying an input to x_8, and the system will not be structurally resilient to an integrity attack.

6.7 Extension to Switched Systems

The material in the previous sections can be extended to the case of switched linear systems. In particular, it is important to note that a switched system can be controllable even when each of its individual modes is not controllable. In this section, we adopt the structured systems approach to study the resilience of switched linear systems to denial-of-service attacks. Structural controllability of switched linear systems was studied in [38], wherein the authors presented necessary and sufficient conditions for structural controllability of a switched linear system in terms of union graphs and colored union graphs. The problem of determining the smallest subset of actuators needed to ensure controllability of the switched system was studied in [39]. We leverage these results along with our results presented in Section 6.3 to establish conditions for a structured switched linear system to be resilient to DoS attacks.

6.7.1 Switched Linear Systems

A switched system is composed of a family of subsystems and a rule that governs switching among them. Consider the system:

$$\dot{x}(t) = A_{\sigma(t)}x(t) + B_{\sigma(t)}u(t) \tag{6.7}$$

where $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^p$. $\sigma : [0, \infty) \to \mathbb{M} := \{1, \ldots, z\}$ is a switching signal. $\mathbb{M}$ are the *modes* of the system, and $\sigma(t) = i$ implies that the i^{th} subsystem is active at time t. In this section, each subsystem will be an LTI system.

We will need several assumptions in the sequel.

Assumption 6.21. *The switching signal $\sigma(t)$ does not depend on initial states and controls.*

Assumption 6.22. *There is only a finite number of changes of mode in every finite time interval. This assumption is needed to rule out the Zeno phenomenon.*

Assumption 6.23. *All pairs of mode transitions are allowed. Further, there are no constraints on the duration of time the system must spend in each mode.*

The switched system is said to be controllable if for any initial state $x(0) = x_0$ and final state $x(t_f) = x_f$, there exists a switching signal $\sigma : [0, t_f) \to \mathbb{M}$ and an input $u : [0, t_f) \to \mathbb{R}^p$ that transfers the system from x_0 to x_f. Similar to the LTI case, a necessary and sufficient condition for the switched linear system to be controllable is given by a rank condition.

Theorem 6.24. *[43] The switched linear system in Equation (6.7) is controllable if and only if the matrix*

$[B_1, B_2, \dots B_z, A_1B_1, \dots, A_zB_1, A_1B_2, \dots, A_zB_2, \dots,$

$A_1B_z, \dots, A_zB_z, A_1^2B_1, A_2A_1B_1, \dots, A_zA_1B_1, A_1A_2B_1,$

$A_2^2B_1, \dots, A_zA_2B_1, \dots, A_1A_zB_z, A_2A_zB_z, \dots, A_z^2B_z, \dots,$

$A_1^{n-1}B_1, A_2A_1^{n-2}B_1, \dots, A_zA_1^{n-2}B_1, A - 1A_2A_1^{n-3}B_1,$

$A_2^2A_1^{n-3}B_1, \dots, A_1^{n-3}B_1, \dots, A_1A_z^{n-2}B_z, \dots A_z^{n-1}B_z]$

has full row rank n.

Let $[A_k]$ and $[B_k]$, $k \in \{1, \dots, z\}$ correspond to the structural realization of matrices A_k and B_k respectively. Therefore, $[A_k] \in \{0, *\}^{n \times n}$ and $[B_k] \in \{0, *\}^{n \times p}$.

We can associate a directed graph to each mode of the system. Let $\mathcal{D}_k = (\mathcal{V}_k, \mathcal{E}_k)$, where $\mathcal{V}_k = \mathcal{U}_k \cup \mathcal{X}_k$ and $\mathcal{E}_k = \mathcal{E}_{A_k} \cup \mathcal{E}_{B_k}$, where $\mathcal{E}_{A_k} = \{(x_j, x_i) | [A_k]_{ij} \neq 0\}$, $\mathcal{E}_{B_k} = \{(u_j, x_i) | [B_k]_{ij} \neq 0\}$, $k = \{1, \ldots, z\}$.

Definition 6.25. *The* union graph *of a collection of digraphs*

$$\mathcal{D}_k := \mathcal{D}([A_k], [B_k]) = (\mathcal{V}_k, \mathcal{E}_k), k = \{1, \ldots, z\}$$

is given by

$$\mathcal{D} := (\mathcal{V}_1 \cup \cdots \cup \mathcal{V}_z, \mathcal{E}_1 \cup \cdots \cup \mathcal{E}_z)$$

Remark 6.26. *Structurally, an edge e_{ij} in the union graph corresponds to a non zero entry in the (j, i) position in at least one of the $[A_k]$ (or $[B_k]$) matrices. The absence of an edge e_{ij} from vertex i to vertex j in the union graph corresponds to the case that the (j, i) entry in each of the $[A_k]$ and $[B_k]$ matrices is zero. Equivalently, the union graph is a representation of the structured system defined by* $([A_1] + \cdots + [A_z], [B_1] + \cdots + [B_z])$.

We will denote the union graph of structured matrices $[M_1]$ and $[M_2]$ by $\mathcal{D}([M_1] + [M_2])$. Following the notation used in [39], $[\ [M_1], [M_2]\ ]$ will denote the concatenation of the matrices $[M_1]$ and $[M_2]$.

The following is a necessary and sufficient condition for structural controllability of the switched linear system.

Theorem 6.27. *[39] A switched linear continuous time system is structurally controllable if and only if the following two conditions hold:*

1. *there exists an edge from an input in* $\mathcal{D}([A_1] + \cdots + [A_z], [B_1] + \cdots + [B_z])$ *to a state vertex in every non top linked SCC of* $\mathcal{D}([A_1] + \cdots + [A_z])$.

2. $\mathcal{B}([\ [A_1], \ldots, [A_z], [B_1], \ldots, [B_z]\ ])$ *has a maximum matching of size* n.

6.7.2 Structural Resilience

As in the single mode case, we assume that the input in Equation (6.7) is partitioned as

$$u = \begin{pmatrix} u_1 & \ldots & u_d & u_{d+1} & \ldots & u_p \end{pmatrix}^T$$

We shall use $u_{def} \in \mathbb{R}^d$ and $u_{att} \in \mathbb{R}^a$ (with $a := p - d$) to collectively denote the elements $\begin{pmatrix} u_1 & \ldots & u_d \end{pmatrix}^T$ and $\begin{pmatrix} u_{d+1} & \ldots & u_p \end{pmatrix}^T$ respectively.

Then, the structural equivalent of Equation (6.7) can be written as:

$$\dot{x}(t) = [A_{\sigma(t)}]x(t) + [B_{\sigma(t)_{def}}]u_{def}(t) + [B_{\sigma(t)_{att}}]u_{att}(t) \tag{6.8}$$

Like in the single mode case, if $\mathcal{X}_{def}$ and $\mathcal{X}_{att}$ denote the (disjoint) sets of state vertices that are accessible to the defender and attacker inputs respectively, then $[B_{k_{def}}]$ will have fixed zeros in the rows corresponding to $\mathcal{X}_{att}$ and $[B_{k_{att}}]$ will have fixed zeros in the rows corresponding to $\mathcal{X}_{def}$.

During a DoS attack, the system is denied access to certain inputs, effectively setting them to zero. In our model, this corresponds to the inputs in u_{att}. Structurally, this corresponds to setting every entry of $[B_{k_{att}}]$ to zero for every mode k.

Assumption 6.28. *The state vertices that the defender and attacker have accesss to remains the same irrespective of the mode of the system. That is, the column indices corresponding to* $[B_{k_{att}}]$ *is the same for every mode.*

We pose the following problem:

Problem 6.29. *Given that the system in Equation (6.8) is structurally controllable before an attack, characterize its structural resilience to a denial-of-service attack.*

As we will see, the solution to this problem will involve determining conditions for $([A_1], [B_{1_{def}}], \ldots, [A_z], [B_{z_{def}}])$ to be structurally controllable. Let m_{def} and m_{att} denote the number of right unmatched vertices in $\mathcal{B}([\,[A_1], \ldots, [A_z]\,])$ corresponding to $\mathcal{X}_{def}$ and $\mathcal{X}_{att}$ respectively.

Theorem 6.30. *The switched system is structurally resilient to a denial-of-service attack if and only if* $d \geq m_{def}$ *and:*

1. $\mathcal{D}([A_1]+\cdots+[A_z])$ *has no non-top linked SCC comprised exclusively of vertices from* $\mathcal{X}_{att}$.

2. *there exists a maximum matching of* $\mathcal{B}([[A_1], \ldots, [A_z]])$ *containing every vertex in* $\mathcal{X}_{att}$*, that is,* $m_{att} = 0$ *for some maximum matching.*

3. *every right unmatched vertex of* $\mathcal{B}([\,[A_1], \ldots, [A_z]\,])$ *in the maximum matching above is connected to a unique input in* u_{def}.

4. *every non-top linked SCC of* $\mathcal{D}([A_1]+\cdots+[A_z])$ *contains a vertex in* $\mathcal{X}_{def}$ *that is connected to some input in* u_{def}.

Proof. If $d < m_{def}$, then there is some vertex in $\mathcal{X}_{def}$ that does not have a 'dedicated input' needed to ensure structural controllability (Theorem 6.6).

Now consider the case when $d \geq m_{def}$, but $\mathcal{D}([A_1] + \cdots + [A_z])$ contains a non-top linked SCC comprised exclusively of vertices from $\mathcal{X}_{att}$ or if every maximum matching of $\mathcal{B}([\,[A_1], \ldots, [A_z]\,])$ contains some vertex in $\mathcal{X}_{att}$. This would mean that vertices in $\mathcal{X}_{att}$ would have to be connected to a control in u_{def}, which violates our assumption that controls in u_{def} can only be connected to states in $\mathcal{X}_{def}$.

The last two conditions are needed to ensure structural controllability of $([A_1], [B_{1_{def}}], \ldots, [A_z], [B_{z_{def}}])$.

Therefore, if any of the conditions are violated, the system will not be structurally resilient to a DoS attack. This proves necessity.

For sufficiency, it is clear that if all the conditions are met, there exists a control configuration which ensures structural controllability even when the system (defender) can control only a subset of the states (i.e., those in $\mathcal{X}_{def}$), and other states (i.e., those in $\mathcal{X}_{att}$) cannot be directly accessed. □

Chapter 7: Conclusion

This dissertation has addressed two problems in the broad area of the control and security of cyberphysical systems.

We proposed **a new framework for opacity** for systems in which the state and input variables take values from a continuous domain. Prior work in this field studied notions of opacity only for systems in which the states were discrete. The framework was built from the bottom-up in a natural way:

1. We started by formulating a notion of opacity for single-adversary LTI systems called *k-initial state opacity (k-ISO)* in Chapter 3. A set of secret states was defined to be k-ISO with respect to a set of nonsecret states if the outputs at time k of every trajectory starting from the set of secret states could not be distinguished from the output at time k of some trajectory starting from the set of nonsecret states. Necessary and sufficient conditions to achieve k-ISO were presented in terms of sets of reachable states. Opacity of a given DT-LTI system was shown to be equivalent to the output controllability of a system obeying the same dynamics, but with different initial conditions.

2. This was extended to the case when there was more than one adversarial observer in Chapter 4, where we defined several notions of decentralized opacity.

These notions of decentralized opacity depended on whether there was a centralized coordinator or not, and the presence or absence of collusion among the adversaries. We established conditions for decentralized opacity in terms of sets of reachable states. In the case of colluding adversaries, we derived a condition for *nonopacity* in terms of the structure of the communication graph.

3. Finally, in Chapter 5, we formulated notions of opacity for switched linear systems (SLSs). We distinguished between the cases when the secret was specified as an initial mode, an initial state, or a combination of the two, and whether the adversary observed a mode, a function of the state, or a combination of the two. Constraints were placed on the modes that the system was allowed to transition into from a given mode and we imposed bounds on the dwell times in each mode. Moreover, constraints were imposed on the number of changes of modes before the adversary made its observation in our definitions of opacity for SLSs. In each case, we presented conditions that established that particular notion of opacity. We also presented algorithmic procedures that gave conservative upper bounds on the computational complexity to verify these notions of opacity.

We then presented **a characterization of the structural resilience of cyber physical systems to denial of service and integrity attacks** using tools from linear structured systems and graph theory in Chapter 6. Conditions for the system to be resilient were established in terms of unmatched vertices of

bipartite graph and connected components of directed graph representations of the structured system. An extension to the linear structured switched systems case was studied and similar conditions needed to establish the resilience to denial of service attacks were presented.

7.1 Future Directions

This section presents a (non-exhaustive) summary of the directions in which the work presented in this dissertation can be extended. Specifically, we propose developing notions of opacity for nonlinear systems, means of computing approximations of reachable sets, quantifying opacity, and several extensions to the structural resilience problem.

7.1.1 Opacity for Nonlinear Systems

The nominal operation of many real-world systems relies on switching among a set of modes whose dynamics are nonlinear. Consider the discrete-time nonlinear system (DTNLS):

$$\begin{aligned} x(t+1) &= f(x(t), u(t)) \\ x(0) &= x_0 \in X_0 \\ y(t) &= h(x(t)) \end{aligned} \tag{7.1}$$

where $x \in \mathbb{R}^n, u \in \mathbb{R}^m, y \in \mathbb{R}^p$, and $f(\cdot,\cdot)$ and $h(\cdot)$ are sufficiently smooth functions with $h(0_n) = 0_p$, where 0_* is the $1 \times *$ vector of zeroes.

In a series of papers, the authors of [65], [66], and [67] derived conditions under which a DTNLS could be transformed into a discrete-time linear system via feedback. A geometric analysis of controllability of DTNLSs in terms of Lie algebras of vector fields was presented in [68]. A linear algebraic framework for the analysis of synthesis problems in DTNLSs was proposed in [69], where the notion of the rank of an analytic discrete time system was developed. In a more recent work [70], the authors studied input-to-state stability properties of DTNLSs, using well established notions of input-to-state stability from the continuous time version.

Remark 7.1. *[69] The analysis of continuous time nonlinear systems is largely focused on that class of systems that is affine in the input (that is, of the form* $\dot{x}(t) = f(x(t)) + g(x(t))u(t)$*). The advantages that such a model offers are twofold: i) the derivatives of the output depend polynomially on the inputs and their derivatives, and ii) the vector fields involved have a nice structure (a drift term and* m *control vector fields). Moreover, this class of systems is general enough to model many practical nonlinear systems. However, the class of discrete time nonlinear systems can also potentially include versions of continuous time systems that are sampled in time, which necessitates considering the more general form of the DTNLS in Equation (7.1).*

Formulating notions of opacity for such systems will contribute to the development of a comprehensive framework for opacity for general cyberphysical systems.

7.1.2 Opacity and Reachable Sets of States

Recall that our definitions of opacity were different from well known definitions of state observability and initial state estimation. Standard definitions of observability assume that the entire input and output sequences are known, and uses these to verify if an initial state can be uniquely determined. In our framework, however, we only have snapshots of the system output, and further, we do not exactly know the inputs; we only have knowledge of the set of potential inputs. This necessitated determining a reachable set of states at the time(s) the adversary made an observation, in order to conclude opacity.

Exactly computing the set of reachable states for a dynamical system is not easy. One method of exactly determining the set of reachable states at a time k is from the Minkowski sum [1] of the reachable states at time $k-1$ and the set of states at time k (obtained from the system dynamics). However, the size of the representation grows at each step, and the problem becomes intractable for large time horizons. This necessitates the use of approximate techniques to compute reachable sets.

Various techniques have been developed to compute over and under approximations of sets of reachable states depending on how the initial set of states is specified, including using support functions [71], zonotopes [58], and ellipsoids [57].

We present a brief survey of some of these techniques. Specifically, we are interested in the following: the use of support functions in determining over ap-

[1] The Minkowski sum of sets $S_1, S_2 \subseteq \mathbb{R}^d$ is $S_1 \oplus S_2 = \{s_1 + s_2 : \ s_1 \in S_1, \ s_2 \in S_2\}$.

proximations of reachable sets for arbitrary compact convex sets of initial states, and under approximations of the reachable set in the case when the initial set is represented as a zonotope. These approximations are tight in the sense that the approximate reachable set will touch the original reachable set at the points where inequalities defining the approximate sets (to be made clear subsequently) attain equality. The reader is referred to [72] for a succinct presentation of the techniques used in computing reachable sets.

Let $\Omega \subseteq \mathbb{R}^d$ be a compact convex set.

Definition 7.2. *[71] The* support function *of* Ω*, denoted* $\rho_\Omega : \mathbb{R}^d \to \mathbb{R}$*, is defined as* $\rho_\Omega(\ell) = \max_{x \in \Omega} \ell^T x$.

A support vector *of* Ω *in the direction* $\ell \in \mathbb{R}^d$ *is a vector (need not be unique)* $v_{\Omega,\ell} \in \mathbb{R}^d$ *such that* $v_{\Omega,\ell} \in \Omega$ *and* $\ell^T v_{\Omega,\ell} = \rho_\Omega(\ell)$.

The support function gives the distance of the supporting hyperplanes of Ω from the origin. A compact convex set is uniquely determined by its support function, since $\Omega = \bigcap_{\ell \in \mathbb{R}^d} \{x \in \mathbb{R}^d : \ell^T x \leq \rho_\Omega(\ell)\}$. Thus, the support function representation is (essentially) dual of the intersection of hyperplanes representation of a convex set.

A tight polyhedral overapproximation $\tilde{\Omega}$ of an arbitrary compact convex set can be got by 'sampling' its support function. The set Ω touches the faces of $\tilde{\Omega}$ at the points defined by the support vectors.

Proposition 7.3. *[71] For all matrices* A*, all compact convex sets* $S_1, S_2 \subseteq \mathbb{R}^p$*, and all nonzero vectors* $\ell \in \mathbb{R}^d$*, with* $hull(S_1, S_2)$ *denoting the convex hull of* S_1 *and*

S_2, the following properties hold:

$$\rho_{hull(S_1,S_2)}(\ell) = \max(\rho_{S_1}(\ell), \rho_{S_2}(\ell))$$

$$\rho_{S_1 \oplus S_2}(\ell) = \rho_{S_1}(\ell) + \rho_{S_2}(\ell)$$

$$\rho_{AS_1}(\ell) = \rho_{S_1}(A^T \ell)$$

For a DT-LTI system, $x(t+1) = Ax(t) + Bu(t),\ x_0 \in X_0,\ u(t) \in \mathcal{U}$, where $X_0 \subseteq \mathbb{R}^n$ and $\mathcal{U} \subseteq \mathbb{R}^m$ are compact convex sets, the set of reachable states at time k is given by $\Omega_k = A\Omega_{k-1} \oplus BU,\ \Omega_0 = X_0$. Given r arbitrarily chosen directions, $\ell_1, \ldots, \ell_r$, a tight polyhedral overapproximation of Ω_k, $\tilde{\Omega}_k$ can be computed as the intersection of halfspaces given by $\mathcal{H}_{k,i} = \{x : \ell_i^T x \leq \rho_{\Omega_k}(\ell_i)\},\ i = 1, \ldots, r$. Thus, computing the overapproximation is equivalent to evaluating the support function along r directions. The complexity of an algorithm to compute the support function presented in [71] is linear in the time horizon and polynomial in the dimension of the state space.

Over- and under-approximations of the reachable set of states can also be computed using *zonotopes*.

Definition 7.4. *[58] A* zonotope *is a subset of $\mathbb{R}^n$ represented by its center $u \in \mathbb{R}^n$ and its generators $v_1, \ldots, v_m \in \mathbb{R}^n$ as:*

$$(u, < v_1, \ldots, v_m >) := \{u + \Sigma_{j=1}^m \alpha_j v_j : \alpha_j \in [-1, 1], j = 1, 2, \ldots, m\}$$

A zonotope with m generators is said to have order $\frac{m}{n}$.

The attraction in this case is the fact that the Minkowski sum of zonotopes can be computed in $O(n)$, independently of the order of the operands. The authors

of [58] present an application of using polytopic under-approximations of zonotopes in control synthesis.

Depending on how the initial set of states is specified, one can choose an appropriate method to determine approximations of the set of reachable states at a future time k.

An interesting problem is to extend this to computations of approximations of the sets $CX_s(k)$ and $CX_{ns}(k)$ to verify if k-ISO holds. The question to be answered is how do set operations (susbet and intersection for strong and weak k-ISO respectively) affect the accuracy of the approximation.

7.1.3 Output Controllability and Opacity

An equivalence between k-ISO and output controllability was established in [40]. A notion of output controllability of a DT-SLS from a particular initial mode has been defined in [60]. Establishing a similar equivalence for switched systems under additional constraints on number of mode transitions and dwell times is more subtle. The possibility of different switches of modes yielding the same output at time k makes the analysis of comparing opacity with output controllability nontrivial. It remains an interesting problem to study, nonetheless.

7.1.4 Quantitative Approaches to Opacity

Another interesting problem is to model the scenario when the adversary incurs a cost to make an observation and has to decide on opacity by incurring as low a

cost as possible. The results in this dissertation have been qualitative in nature. An interesting topic to pursue is methods of quantifying opacity, and investigate its relation to the notion of differential privacy [73].

7.1.5 Structural Resilience

For the structural resilience problem, throughout this dissertation, we have assumed that the system and the attacker have access to disjoint sets of nodes. One direction of future research is to study structural resilience when there is a set of nodes accessible to both defender and attacker.

In the switched systems case, we assumed that the sets of states accessible to the defender and attacker remain the same for every mode. Future work will investigate the case when the states accessible to each is possibly different for each mode. Further, there were no restrictions on the allowed mode transitions or on the duration of time the system could spend in each mode. Extending our work to incorporate these restrictions is another area of interest. Alternatively, one could associate probabilities with the transitions from one mode to another, and use this to develop a notion of probabilistic structural resilience for switched systems.

Assuming that activating or disrupting an input-state link incurs a cost, there are two scenarios that can be envisaged. First, when the system successfully thwarts an attack, the attacker might want to ensure that the system incurs a high cost in maintaining resilience, while keeping its own cost of carrying out the attack low. A second problem of interest is to quantify the robustness of the system to the worst

possible attack. It would be interesting to see if these situations can be cast as optimization problems.

Bibliography

[1] Radhakisan Baheti and Helen Gill. Cyber-physical systems. *The Impact of Control Technology*, 12:161–166, 2011.

[2] Julia E Sullivan and Dmitriy Kamensky. How cyber-attacks in Ukraine show the vulnerability of the US power grid. *The Electricity Journal*, 30(3):30–35, 2017.

[3] Yasser Shoukry, Paul Martin, Paulo Tabuada, and Mani Srivastava. Non-invasive spoofing attacks for anti-lock braking systems. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 55–72. Springer, 2013.

[4] Jill Slay and Michael Miller. *Lessons learned from the Maroochy water breach*. Springer, 2008.

[5] James P Farwell and Rafal Rohozinski. Stuxnet and the future of cyber war. *Survival*, 53(1):23–40, 2011.

[6] Alvaro A. Cárdenas, Saurabh Amin, and Shankar Sastry. Research challenges for the security of control systems. In *HotSec*, 2008.

[7] Alvaro A. Cardenas, Saurabh Amin, and Shankar Sastry. Secure control: Towards survivable cyber-physical systems. In *28th Conference on Distributed Computing Systems Workshops*, pages 495–500. IEEE, 2008.

[8] Steve Schneider and Abraham Sidiropoulos. CSP and anonymity. In *Computer SecurityESORICS*, pages 198–218. Springer, 1996.

[9] Riccardo Focardi and Roberto Gorrieri. A taxonomy of trace-based security properties for CCS. In *Proceedings of Computer Security Foundations Workshop*, pages 126–136. IEEE, 1994.

[10] Eric Badouel, Marek Bednarczyk, Andrzej Borzyszkowski, Benoît Caillaud, and Philippe Darondeau. Concurrent secrets. *Discrete Event Dynamic Systems*, 17(4):425–446, 2007.

[11] Anooshiravan Saboori and Christoforos N. Hadjicostis. Notions of security and opacity in discrete event systems. In *46th IEEE Conference on Decision and Control*, 2007.

[12] Jérémy Dubreil, Philippe Darondeau, and Hervé Marchand. Supervisory control for opacity. *IEEE Trans. on Automatic Control*, 55(5):1089–1100, 2010.

[13] Anooshiravan Saboori and Christoforos N. Hadjicostis. Opacity-enforcing supervisory strategies via state estimator constructions. *IEEE Transactions on Automatic Control*, 57(5):1155–1165, 2012.

[14] Fabio Pasqualetti, Florian Dorfler, and Francesco Bullo. Control-theoretic methods for cyberphysical security: Geometric principles for optimal cross-layer resilient control systems. *IEEE Control Systems*, 35(1):110–127, 2015.

[15] Ching Tai Lin. Structural controllability. *IEEE Trans. on Automatic Control*, 19(3):201–208, 1974.

[16] Laurent Mazaré. Using unification for opacity properties. *Proceedings of the 4th IFIP WG1*, 7, 2004.

[17] Peter J. Ramadge and W. Murray Wonham. Supervisory control of a class of discrete event processes. *SIAM Journal on Control and Optimization*, 25(1):206–230, 1987.

[18] Peter J. G. Ramadge and W. Murray Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–98, 1989.

[19] Meera Sampath, Raja Sengupta, Stéphane Lafortune, Kasim Sinnamohideen, and Demosthenis C. Teneketzis. Failure diagnosis using discrete-event models. *IEEE Transactions on Control Systems Technology*, 4(2):105–124, 1996.

[20] Arjan J. Van Der Schaft and Johannes Maria Schumacher. *An introduction to hybrid dynamical systems*, volume 251. Springer London, 2000.

[21] Robert R. Burridge, Alfred A. Rizzi, and Daniel E. Koditschek. Sequential composition of dynamically dexterous robot behaviors. *The International Journal of Robotics Research*, 18(6):534–555, 1999.

[22] Franck Cassez, Jérémy Dubreil, and Hervé Marchand. Synthesis of opaque systems with static and dynamic masks. *Formal Methods in System Design*, 40(1):88–115, 2012.

[23] Anooshiravan Saboori and Christoforos N. Hadjicostis. Verification of initial-state opacity in security applications of DES. In *9th International Workshop on Discrete Event Systems*, pages 328–333. IEEE, 2008.

[24] Anooshiravan Saboori and Christoforos N. Hadjicostis. Verification of infinite-step opacity and complexity considerations. *IEEE Transactions on Automatic Control*, 57(5):1265–1269, 2012.

[25] Yi-Chin Wu and Stéphane Lafortune. Comparative analysis of related notions of opacity in centralized and coordinated architectures. *Discrete Event Dynamic Systems*, 23(3):307–339, 2013.

[26] Feng Lin. Opacity of DES and its applications. *Automatica*, 47(3):496–503, 2011.

[27] Andrea Paoli and Feng Lin. Decentralized opacity of discrete event systems. In *Proceedings of the American Control Conference*, pages 6083–6088. IEEE, 2012.

[28] Yi-Chin Wu and Stéphane Lafortune. Synthesis of insertion functions for enforcement of opacity security properties. *Automatica*, 50(5):1336–1348, 2014.

[29] Fabio Pasqualetti, Florian Dorfler, and Francesco Bullo. Attack detection and identification in cyber-physical systems. *IEEE Trans. on Automatic Control*, 58(11):2715–2729, 2013.

[30] Miroslav Pajic, Rahul Mangharam, George J. Pappas, and Suresh Sundaram. Topological conditions for in-network stabilization of dynamical systems. *IEEE Journal on Selected Areas in Communications*, 31(4):794–807, 2013.

[31] Carlos Barreto, Alvaro A. Cárdenas, and Nicanor Quijano. Controllability of dynamical systems: Threat models and reactive security. In *Decision and Game Theory for Security*, pages 45–64. Springer, 2013.

[32] Sergio Pequito, Soummya Kar, and A. Pedro Aguiar. A framework for structural input/output and control configuration selection in large-scale systems. *IEEE Transactions on Automatic Control*, 61(2):303–318, 2016.

[33] Alex Olshevsky. Minimal controllability problems. *IEEE Trans. on Control of Network Systems*, 1(3):249–258, 2014.

[34] Christian Commault and Jean-Michel Dion. The single-input minimal controllability problem for structured systems. *Systems & Control Letters*, 80:50–55, 2015.

[35] Sérgio Pequito, Soummya Kar, and A. Pedro Aguiar. On the complexity of the constrained input selection problem for structural linear systems. *Automatica*, 62:193–199, 2015.

[36] Sergio Pequito, Soummya Kar, and A. Pedro Aguiar. Minimum cost input/output design for large-scale linear structural systems. *Automatica*, 68:384–391, 2016.

[37] Sérgio Pequito, Soummya Kar, and George J. Pappas. Minimum cost constrained input-output and control configuration co-design problem: A structural systems approach. In *American Control Conference (ACC), 2015*, pages 4099–4105. IEEE, 2015.

[38] Xiaomeng Liu, Hai Lin, and Ben M. Chen. Structural controllability of switched linear systems. *Automatica*, 49(12):3531–3537, 2013.

[39] Sérgio Pequito and George J. Pappas. Structural minimum controllability problem for switched linear continuous-time systems. *Automatica*, 78:216–222, 2017.

[40] Bhaskar Ramasubramanian, Rance Cleaveland, and Steven I. Marcus. A framework for opacity in linear systems. In *Proceedings of the IEEE American Control Conference*, pages 6337–6344, 2016.

[41] Bhaskar Ramasubramanian, Rance Cleaveland, and Steven I. Marcus. A framework for decentralized opacity in linear systems. *Proceedings of the Annual Allerton Conference on Communications, Control, and Computing*, 2016.

[42] Bhaskar Ramasubramanian, Rance Cleaveland, and Steven I. Marcus. Opacity for switched linear systems: Notions and characterization. *Proceedings of the IEEE Conference on Decision and Control*, 2017.

[43] Zhendong Sun and Shunzi Ge. *Switched linear systems: Control and design*. Springer Science & Business Media, 2006.

[44] Hai Lin and Panos J. Antsaklis. Stability and stabilizability of switched linear systems: A survey of recent results. *IEEE Transactions on Automatic control*, 54(2):308–322, 2009.

[45] Daniel Liberzon. *Switching in systems and control*. Springer Science & Business Media, 2012.

[46] Bhaskar Ramasubramanian, Rance Cleaveland, and Steven I. Marcus. Notions of centralized and decentralized opacity in linear systems (in preparation).

[47] Bhaskar Ramasubramanian, M.A. Rajan, and M. Girish Chandra. Structural resilience of cyberphysical systems under attack. In *Proceedings of the IEEE American Control Conference*, pages 283–289, 2016.

[48] Bhaskar Ramasubramanian, M.A. Rajan, M. Girish Chandra, Rance Cleaveland, and Steven I. Marcus. Resilience to denial of service attacks: A structured systems approach. Submitted to a journal.

[49] Jean-Michel Dion, Christian Commault, and Jacob Van Der Woude. Generic properties and control of linear structured systems: a survey. *Automatica*, 39(7):1125–1144, 2003.

[50] Wilson J. Rugh. *Linear System Theory*. Prentice Hall, Upper Saddle River, NJ, 1996.

[51] Richard A. Brualdi, Frank Harary, and Zevi Miller. Bigraphs versus digraphs via matrices. *Journal of Graph Theory*, 4(1):51–73, 1980.

[52] Anooshiravan Saboori and Christoforos N. Hadjicostis. Verification of k-step opacity and analysis of its complexity. *IEEE Transactions on Automation Science and Engineering*, 8(3):549–559, 2011.

[53] Arjan Van Der Schaft. Equivalence of dynamical systems by bisimulation. *IEEE Transactions on Automatic Control*, 49(12):2160–2172, 2004.

[54] Mohamed Babaali and Magnus Egerstedt. Observability of switched linear systems. In *International Workshop on Hybrid Systems: Computation and Control*, pages 48–63. Springer, 2004.

[55] Thomas H. Cormen, Charles Eric Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms*. MIT Press Cambridge, 2001.

[56] Andrea Lincoln, Virginia Vassilevska Williams, Joshua R. Wang, and R. Ryan Williams. Deterministic time-space trade-offs for k-sum. In *43rd International Colloquium on Automata, Languages, and Programming*, pages 58:1–58:14, 2016.

[57] Alex A Kurzhanskiy and Pravin Varaiya. Ellipsoidal techniques for reachability analysis of discrete-time linear systems. *IEEE Transactions on Automatic Control*, 52(1):26–38, 2007.

[58] Antoine Girard, Colas Le Guernic, and Oded Maler. Efficient computation of reachable sets of linear time-invariant systems with inputs. In *International Workshop on Hybrid Systems: Computation and Control*, pages 257–271. Springer, 2006.

[59] Colas Le Guernic and Antoine Girard. Reachability analysis of linear systems using support functions. *Nonlinear Analysis: Hybrid Systems*, 4(2):250–262, 2010.

[60] Artur Babiarz, Adam Czornik, and Michał Niezabitowski. Output controllability of the discrete-time linear switched systems. *Nonlinear Analysis: Hybrid Systems*, 21:1–10, 2016.

[61] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. Controllability of complex networks. *Nature*, 473(7346):167–173, 2011.

[62] Robert Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

[63] John E. Hopcroft and Richard M. Karp. A $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. In *Switching and Automata Theory, 1971., 12th Annual Symposium on*, pages 122–125. IEEE, 1971.

[64] Silvio Micali and Vijay V. Vazirani. An $O(\sqrt{|V|}.|E|)$ algorithm for finding maximum matching in general graphs. In *Foundations of Computer Science, 1980., 21st Annual Symposium on*, pages 17–27. IEEE, 1980.

[65] Hong-Gi Lee and Steven I. Marcus. Approximate and local linearizability of non-linear discrete-time systems. *International Journal of Control*, 44(4):1103–1124, 1986.

[66] Hong-Gi Lee and Steven I. Marcus. On input-output linearization of discrete-time nonlinear systems. *Systems & Control Letters*, 8(3):249–259, 1987.

[67] H. G. Lee, A. Arapostathis, and S. I. Marcus. Linearization of discrete-time systems. *International Journal of Control*, 45(5):1803–1822, 1987.

[68] Bronislaw Jakubczyk and Eduardo D. Sontag. Controllability of nonlinear discrete-time systems: A Lie-algebraic approach. *SIAM Journal on Control and Optimization*, 28(1):1–33, 1990.

[69] J.W. Grizzle. A linear algebraic framework for the analysis of discrete-time nonlinear systems. *SIAM Journal on Control and Optimization*, 31(4):1026–1044, 1993.

[70] Zhong-Ping Jiang and Yuan Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37(6):857–869, 2001.

[71] Antoine Girard and Colas Le Guernic. Efficient reachability analysis for linear systems using support functions. *IFAC Proceedings Volumes*, 41(2):8966–8971, 2008.

[72] Oded Maler. Computing reachable sets: An introduction. *Technical Report, French National Center of Scientific Research*, 2008.

[73] Cynthia Dwork. Differential privacy. In *Encyclopedia of Cryptography and Security*, pages 338–340. Springer, 2011.