Structural Hybrid Systems

Guilherme Henrique Caçador Ramos

Thesis to obtain the Master of Science Degree in
Mathematics and Applications

Examination Committee
Chairperson: Prof. Maria Cristina de Sales Viana Serôdio Sernadas
Supervisor: Prof. António Pedro Rodrigues Aguiar
Co-supervisor: Prof. Jaime Arsénio de Brito Ramos
Members of the Committee: Prof. João Manuel Freitas Xavier

July 2013
Dedicated to my family and friends.
Acknowledgments

I would like to thank to the people whose support has been important in order to develop my work on this thesis.

First, and more important, I would like to thank to LARSyS (Laboratory for Robotics and Systems in Engineering and Science) that have gave me a scholarship to do research related with the project PEst-OE/EEI/LA0009/2011 FCT-Portugal in order to study and develop computational analysis of scalable techniques for large-scale complex systems, notably ones, which tackle design and decision making in a single framework.

Immediately after, I would like to express my exceptional gratitude to my advisor Professor António Pedro Aguiar, to my co-advisor Professor Jaime Ramos and to my mentor Sérgio Pequito for stimulating my knowledge, for believing in my work and for all the support, encouragement and guidance they have give me, not only in technical matters, but also in nontechnical matters.

Also, I want to acknowledge to all my friends in general, for all the good times and also for their support and encouragement. Namely, André Gomez, António Salgueiro, Carlos Ferreira, Carlos Silva, Carolina Varela, Catarina Dinis, Cátia Conceição, Diogo Fernandes, Diogo Poças, Diogo Raimundo, Filipe Casal, Frederico Horta, Isabel Carreira, Isabel Ferreira, Joana Ribeiro, João Carvalho, João Morence, José Amaral, José Horta, José Ricardo, Lubélia Ribeiro, Luís Costa, Márcia Chicharro, Marta Matias, Rita Ricardo, Rui Barreia, Sérgio Pequito, Tiago Guerreiro, Tiago Lúcio and Tiago Ponte.

In particular for this work, I want to acknowledge to Sérgio Pequito, to Diogo Poças, to Filipe Casal and to Tiago Guerreiro, who have discussed with me several ideas about my work on this thesis.

Finally, but not less important, I would like to thank to my family for all the support they have give me in the most difficult situations that I have been through. Without them I would not have came this far in my academics.

Obrigado!
Resumo

O problema de quantas entradas de controlo (e onde) são necessários para actuar (controlar) um sistema linear tem sido um problema fundamental e desafiante em teoria de controlo, bem como em aplicações a sistemas de controlo. A teoria estrutural tem proporcionado um enquadramento eficiente para responder a esta pergunta para uma classe de equivalência de sistemas, onde as propriedades são exploradas com base na localização de valores zero e não-zero. Nesta tese usaremos as técnicas desenvolvidas recentemente, sobre sistemas estruturais.

Esta tese introduz o conceito de sistemas estruturais híbridos para resolver, como um caso particular, o problema de verificação de modelos de sistemas comutados com tempo linear (possivelmente de grande escala). Dentro do cenário proposto, fornecemos condições necessárias para garantir propriedades tais como controlabilidade, em cada instante de tempo.

Mostramos que este modelo, para verificar propriedades de controlabilidade, pode ser implementado usando algoritmos eficientes (com complexidade polinomial). Um exemplo, com base no sistema elétrico 5-bus do IEEE, é apresentado onde se ilustra o algoritmo proposto para verificação de modelos, bem como os métodos de projecto do modelo.

As principais contribuições desta dissertação são as seguintes:

(i) a introdução do novo conceito de sistemas estruturais híbridos;

(ii) uma nova demonstração, baseada num critério recente para controlabilidade, do resultado que afirma que um sistema linear é estruturalmente controlável se e só se o seu grafo dirigido for gerado por uma união disjunta de input cacti;

(iii) a redução do problema de controlabilidade minimal ao problema de cobertura de conjuntos, o que fornece uma forma eficiente de aproximar soluções do problema de controlabilidade minimal baseadas em algoritmos que calculam soluções aproximadas para o problema de cobertura de conjuntos.

Palavras-chave: Sistemas comutados; Sistemas estruturais; Teoria de grafos; Sistemas híbridos dinâmicos; Modelação de sistemas; Verificação de sistemas.
Abstract

The question of how many inputs are needed and where to place them in order to actuate (control) a linear system has been a fundamental and challenging problem in control theory and applications of control systems. Structural theory has provided an efficient framework to answer this question for an equivalent class of systems, where properties are explored based on the location of zero and non-zero values. In this thesis we will use recent techniques, about structural systems.

This thesis introduces the concept of structural hybrid systems to address, as a particular case, the model checking problem of switching (possible large scale) linear time invariant systems. Within the proposed setup, we provide necessary conditions to ensure properties such as controllability, at each time.

We show that such model checking controllability properties can be implemented using efficient algorithms (with polynomial complexity). An example, based on the IEEE 5-bus power system, is presented which illustrates the proposed model checking and design methodologies.

The main contributions of the thesis are the following:

(i) the introduction of the new concept of structural hybrid systems;

(ii) a new proof, based on a recent criteria for controllability, of the result which states that a linear system is structurally controllable if and only if its associated directed graph is spanned by a disjoint union of input cacti;

(iii) the reduction of the minimal controllability problem to the set covering problem, providing an efficient way of approximating solutions to the minimal controllability problem based on algorithms that compute approximated solutions to the the set covering problem.

Keywords: Switching systems; Structural systems; Graph theory; Hybrid dynamic systems; Model systems; Verifying system.
# Contents

Acknowledgments .................................................. v
Resumo ............................................................ vii
Abstract ........................................................... ix
List of Tables ......................................................... xiii
List of Figures ........................................................ xvi

1 Introduction ......................................................... 1

2 Preliminaries ....................................................... 5
  2.1 Linear Control Systems ......................................... 5
  2.2 Structural Control Systems .................................... 11
  2.3 Control Design .................................................. 29
  2.4 Hybrid Dynamical Systems .................................... 32
  2.5 Hybrid Automata ................................................ 34

3 Structural Hybrid System ......................................... 36

4 A power electrical grid application ............................... 46
  4.1 5-Bus System ................................................... 46

5 Complexity of the Minimal Controllability Problem .......... 66

6 Conclusions ......................................................... 69
  6.1 Achievements ................................................... 69
  6.2 Future Work .................................................... 69

Bibliography .......................................................... 72
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Comparison between the structural hybrid system and hybrid automaton.</td>
<td>40</td>
</tr>
<tr>
<td>4.1</td>
<td>Steam turbine values.</td>
<td>52</td>
</tr>
<tr>
<td>4.2</td>
<td>Load values</td>
<td>52</td>
</tr>
</tbody>
</table>
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Graph and structural graph representation of an LTI system</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>A cycle with $n$ vertices.</td>
<td>17</td>
</tr>
<tr>
<td>2.3</td>
<td>Strongly connected graph</td>
<td>17</td>
</tr>
<tr>
<td>2.4</td>
<td>Non-top/non-bottom link strong connected components</td>
<td>18</td>
</tr>
<tr>
<td>2.5</td>
<td>State stem/elementary path</td>
<td>18</td>
</tr>
<tr>
<td>2.6</td>
<td>Input stem</td>
<td>19</td>
</tr>
<tr>
<td>2.7</td>
<td>Examples that are not state cacti</td>
<td>19</td>
</tr>
<tr>
<td>2.8</td>
<td>State cactus</td>
<td>20</td>
</tr>
<tr>
<td>2.9</td>
<td>Input cactus</td>
<td>20</td>
</tr>
<tr>
<td>2.10</td>
<td>Spanning subgraph</td>
<td>21</td>
</tr>
<tr>
<td>2.11</td>
<td>One input connected to two state vertices</td>
<td>24</td>
</tr>
<tr>
<td>2.12</td>
<td>One input connected to two state stems with the same origin.</td>
<td>25</td>
</tr>
<tr>
<td>2.13</td>
<td>Bipartite graph</td>
<td>26</td>
</tr>
<tr>
<td>2.14</td>
<td>Bipartite representation of a directed graph</td>
<td>26</td>
</tr>
<tr>
<td>2.15</td>
<td>Matching, alternating path and maximum matching</td>
<td>28</td>
</tr>
<tr>
<td>2.16</td>
<td>Decomposition in cycles and state stems</td>
<td>28</td>
</tr>
<tr>
<td>2.17</td>
<td>Illustrative example of Theorem 6</td>
<td>29</td>
</tr>
<tr>
<td>2.18</td>
<td>Bouncing ball example, height along the time</td>
<td>33</td>
</tr>
<tr>
<td>2.19</td>
<td>Thermostat examples, temperature along time</td>
<td>34</td>
</tr>
<tr>
<td>2.20</td>
<td>(a) Hybrid automata of the bouncing ball and thermostat examples</td>
<td>35</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Hybrid system with two states and two discrete jumps, where each sate represents a switching system with the state variable on a certain domain.</td>
<td>37</td>
</tr>
<tr>
<td>3.2</td>
<td>Illustration of the proof of Proposition 6</td>
<td>37</td>
</tr>
<tr>
<td>3.3</td>
<td>Abstraction of the hybrid automata representation depicted in Figure 3.1.</td>
<td>38</td>
</tr>
<tr>
<td>3.4</td>
<td>Equivalence (almost surely) between the hybrid system abstraction based on the pair of matrices $(A, B)$ of the linear systems and the the hybrid system abstraction based on the structural graph representation of the linear systems.</td>
<td>38</td>
</tr>
<tr>
<td>3.5</td>
<td>Abstraction of the hybrid automata representation of an evolution of a linear system, based on the structural controllability of the digraph representation of the pair of structural matrices of the system.</td>
<td>39</td>
</tr>
<tr>
<td>3.6</td>
<td>Structural hybrid system</td>
<td>41</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The analysis of controllability of physical systems has been the subject of much research in the last few decades. The problem of controlling a physical system consists in conveniently select the actuating signals in order to achieve a desired behavior. One problem before the control system design is the selection of actuators and sensors. When we design the inputs (actuators) there are mainly two aspects for measuring how good the actuators can control the system, its cost, for instance the number of actuator and its placement, and its performance, which may depend on the precision of the actuators or how fast can the actuators effectively control the system. This problem has many applications in several fields like power systems, wireless control systems, biological complex networks and formation control.

In this thesis we will focus on the problem of finding the minimum number of inputs (actuators) needed to control a linear system. Notice that for large scale linear systems, the verification that the system is controllable has some known numerical issues, see [20] for more details. In addition, the problem of finding the minimum number of inputs needed to control a linear system has very recently been shown to be an NP-hard problem [21]. To avoid the difficulty of this problem, we propose the use of structural systems theory [7], which briefly consists in studying an equivalent class of systems where its properties are explored based only on the location of zeroes and non-zeroes (i.e., the sparsity pattern) of the state space representation matrices. One advantage of this approach is that it allows us to cope with uncertainties about the parameters of the system and to have a directed graph representation where we can infer properties of the system, the details of this approach can be found in sections 2.1 and 2.2. Using structural systems, the analysis of the system provides results that hold for almost all values of the parameters (except for a manifold of zero Lebesgue-measure) [24].

The main motivation of this thesis is the lack of efficient and scalable methods to design and verify properties of hybrid dynamical systems, commonly referred to as model checking [9]. In particular, we focus on the important subclass corresponding to switching systems that can model the behavior of several physical phenomena, including circumstances where a control module has to switch [10, 8, 1], with examples ranging from simple thermostats to an electrical power grid.

Informally, model checking consists of two main steps: first, a possible design of the dynamic system of interest, i.e., the model is considered; second, we check if a desired property of the model holds or not, if not, the design step is revisited and a new model is constructed. The majority of the tools for model checking are compu-
tationally cumbersome and, in particular, the hybrid automata (i.e., a common tool used to do model checking for hybrid dynamical systems) is, in general, undecidable [10]. In fact, even approximation methods exhibit numerical instabilities and suffer from the curse of dimensionality, see [8, 1] and references therein.

Common properties of interest in model checking include (but are not restricted to) reachability and safety [3]. Here, we focus on controllability, i.e., the ability of driving the system state toward a goal, by proper selection of the system’s inputs. From the design point of view, the problem of finding the minimum number of inputs (actuators) to ensure system’s controllability has recently been shown to be an NP-hard problem [21], which implies that designing the minimum number of inputs that ensure a switching system to be controllable is at least as difficult. Therefore, alternative design approaches have been proposed, see, for example, [23].

In this thesis, using structured systems theory [7], we obtain necessary conditions to ensure controllability for switching linear systems. Such property is of fundamental interest in several critical systems, such as the electric power grid, where it is critical to ensure that the system state stays within imposed standards in the face of dynamic mode changes often triggered by uncertain events like transmission line failures or faults.

To achieve this, we propose the concept of a structural hybrid system, hinged on the graph theoretic representation of a structural switching system [18], in which each mode system operating corresponds to a directed graph (digraph), and mode transitions are captured by switching between distinct digraph representations. We demonstrate that the above concept enables efficient model checking for switching systems and, in particular, provides easy-to-verify necessary conditions that ensure controllability.

Formally, we would like to address the following problem:

\[ P_1 \]

Given \( A(\sigma(t)) \), where \( A \in \mathbb{R}^{n \times n} \) and \( \sigma(t) : [0, +\infty) \rightarrow \mathbb{R} \) is a piecewise switching signal, that may only switch at most once in a given dwell-time \([t, t+\varepsilon], \varepsilon > 0\), for all \( t \geq 0 \), we are interested in obtaining \( B(\sigma(t)) \in \mathbb{R}^{n \times p} \) such that

\[
\dot{x}(t) = A(\sigma(t))x(t) + B(\sigma(t))u(t),
\]

is controllable for all \( t \geq 0 \) and \( B(\sigma(t)) \) comprises at most one non-zero entry for each column, i.e., uses dedicated inputs only.

Given that this problem is hard to solve, hereafter we focus on addressing its structural reformulation \( P_2 \), which provides necessary conditions for the all-time controllability requirement in \( P_1 \).

**Problem Statement**

\[ P_2 \]

Given structural matrices \( \bar{A}(\sigma(t)) \), with \( \bar{A} \in \{0, \star\}^{n \times n} \), where \( \star \) stands for a non-zero entry, and \( \sigma(t) : [0, +\infty) \rightarrow \mathbb{R} \) is a piecewise constant switching signal satisfying the dwell-time property for some \( \varepsilon > 0 \), find \( \bar{B}(\sigma(t)) \in \{0, \star\}^{n \times p} \) such that the (structural) dynamical system

\[
\dot{x}(t) = \bar{A}(\sigma(t))x(t) + \bar{B}(\sigma(t))u(t),
\]

is structurally controllable (to be defined precisely) for all \( t \geq 0 \) and \( \bar{B}(\sigma(t)) \) comprises at most one non-zero entry for each column, i.e., uses dedicated inputs only.
In addition, in this thesis, we restrict the analysis of problem $P_2$ to the case where $\bar{A}$ satisfies a specific structural constraint, to be made precise in Assumption $A_1$, that is consistent with several practical physical systems, such as the electric power grid as modeled in [13]. Such structural treatment of dynamical properties of switching systems (i.e. switching systems where only its structure is considered) and, in particular, obtaining necessary conditions to ensure structural controllability, has been considered in prior work, see [6, 17], for instance. In [6] a (vector) dynamical system was modeled as a multi-agent network with each agent corresponding to a single scalar state variable of the dynamical system, and the design goal was to obtain the minimal placement of sensors such that system structural observability (the dual of structural controllability) is retained in the face of arbitrary agent departures (such departure events correspond to mode changes and are captured by deleting all the edges in the nominal system digraph incident to the departed agent). Similarly, in [17], in a multi-agent networked system setting, mode changes consisted of potential removal of bi-directional edges between physically coupled agents (states), and the minimal placement of actuators, necessary to retain structural controllability, was sought. However, both the approaches were limited to systems in which the digraph representation had the special structure of being the disjoint union of strongly connected components, i.e., with no edge between the components. In contrast, the design and verification methodologies introduced in this thesis are more general, include as instances the scenarios studied in [6], [17], and, in particular, applicable to systems in which the digraph representation may consist of several strongly connected components with directed edges between them.

**Main Results**

The main contributions of this thesis are threefold.

1. we introduce the concept of structural hybrid system and provide an efficient model checker (i.e., with polynomial time complexity) to ensure the structural hybrid system’s structural controllability at all times;

2. we re-derived\(^1\) the necessary and sufficient conditions for structural controllability, using the Popov-Belevitch-Hautus controllability test, making the proof more intuitive and less extensive;

3. we show that the minimal controllability problem is actually NP-complete, by deducing it to the set cover problem (see chapter 5).

**Outline of the Thesis**

The thesis is organized as follows. In chapter 2 we introduce our framework. We describe some important concepts of linear systems, control theory and its relation with some graph theory concepts. We also study the relationship between structural controllability of a linear system and the properties of the directed graph representation of a linear system, presenting a new proof of a result which states this relationship, using a recent criteria for controllability.

In chapter 3 we introduce the concept of hybrid dynamic system and a tool used to verify properties about this class of systems. These ideas are the base of a new concept that we introduce, called hybrid structural system. We compare these two concepts, pointing out some advantages/disadvantages of them and we provide a way to verify and recover the property of structural controllability in the hybrid structural system paradigm.

\(^1\)Later, we found that the some approach has been partially explored by Fátima Leite in [15].
In chapter 4 we present some application examples of the new concept of hybrid dynamic system, comparing the obtained results with the theoretical ones.

In the conclusion we gauge our work and what remains to be done, presenting some questions for further research.
Chapter 2

Preliminaries

In this section we review some important concepts of linear systems, control theory and its relationship with some graph theory concepts. This concepts were introduced in [16].

2.1 Linear Control Systems

Briefly, a control problem is the mathematical study of manipulating signals that affect the behavior of a system in order to achieve a specified goal.

In what follows we are interested in time invariant switching system, i.e., systems that can be represented by

\[
\dot{x}(t) = A(\sigma(t))x(t) + B(\sigma(t))u(t), \quad x(0) = x_0
\]

(2.1)

where \(x(t) \in \mathbb{R}^n\) denotes the state of the system, \(A(t) \in \mathbb{R}^{n \times n}\) represents the autonomous dynamics and \(B(t) \in \mathbb{R}^{n \times p}\) is the so-called input matrix that represents which variables are manipulated by the input vector \(u(t) \in \mathbb{R}^p\) and \(\sigma(t) : [0, +\infty[ \rightarrow \mathbb{R}\) is a piecewise switching signal that may only switch at most once in a given dwell-time \([t, t + \varepsilon[\). Additionally, we denote by \(x_i(t)\) the \(i\)-th component of the state vector and \(u_j(t)\) denotes the \(j\)-th component of the input or, alternatively, the input \(j\).

Now, consider the particular case where \(\sigma\) is constant, in other words we have a linear time-invariant system (LTI) described by

\[
\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0,
\]

(2.2)

\[ t \in [t_i, t_f], \]

for the continuous case and described by

\[
x(t + 1) = Ax(t) + Bu(t), \quad x(0) = x_0,
\]

(2.3)

\[ t \in \{t_i, t_{i+1}, \ldots, t_f\}, \]

on the discrete case.
Lemma 1. [Solution for the system (2.2)] The solution of (2.2) is given by

\[ x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau) \, d\tau. \]  

(2.4)

Proof. Consider (2.4) with both sides multiplied by the integrating factor \( e^{-At} \), we obtain

\[ e^{-At} \dot{x}(t) = e^{-At}Ax(t) + e^{-At}Bu(t), \]  

(2.5)

\[ e^{-At}Bu(t) = e^{-At}\dot{x}(t) - e^{-At}Ax(t). \]  

(2.6)

Observe that the right hand side of (2.6) is in the derivative of \( e^{-At}x(t) \), so we obtain

\[ e^{-At}Bu(t) = \frac{d}{dt}e^{-At}x(t) \]  

(2.7)

Now, integrating on both sides and consider the initial value of (2.7) leads to

\[ \int_0^t e^{-A\tau}Bu(\tau) \, d\tau = \int_0^t \frac{d}{d\tau}e^{-A\tau}x(\tau) \, d\tau, \]  

(2.8)

\[ \int_0^t e^{-A\tau}Bu(\tau) \, d\tau = e^{-At}x(t) - x(0), \]  

(2.9)

which, multiplying both sides by \( e^{At} \) and rearranging the terms, we obtain

\[ x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau) \, d\tau. \]  

(2.10)

\[ \Box \]

Alternatively, consider the linear time invariant in discrete time. The following result holds.

Lemma 2. [Solution of the system (2.3)] The solution for (2.3) is described by

\[ x(t) = A^tx(0) + \sum_{\tau=0}^{t-1} A^{t-1-\tau}Bu(\tau). \]

\[ \Box \]

Proof. Follows immediately by induction.  

\[ \Box \]

We say that the input \( u : \{0, \ldots, t\} \to \mathbb{R}^m \) transfers the state from \( x(0) = x_0 \) to \( x(t) = x_t \) on \( t \) instants of time.

Considering Lemma 1 and Lemma 2, we can characterize the set of points that the systems can reach within \( t \) instants of time, which is described by the following.

Definition 1 (Set of reachable points in \( t \) instants of time). The set of reachable points for system (2.2) in \( t \) instants
of time is given by:
\[ R_t = \left\{ e^{At}x_0 + \int_0^t e^{A(t-\tau)} Bu(\tau) \, d\tau \mid u : [0, t] \to \mathbb{R}^m \right\}, \]
and for the system (2.3) is given by
\[ R_t = \left\{ A^t x(0) + \sum_{\tau=0}^{t-1} A^{t-\tau-1} Bu(\tau) \mid u(0), \ldots, u(t-1) \in \mathbb{R}^m \right\}. \tag{2.11} \]

The set \( R_t \) has the following properties:

- is a subspace of \( \mathbb{R}^n \);
- \( R_t \subseteq R_s \), if \( t \leq s \).

In order to better understand the concepts associated with the set of points that are reachable from a starting point, we now focus on LTI systems with discrete time as described in (2.3). Assume that we do not have any constraint on time and that \( x(0) = 0 \). Then the set of points that can be achieved from a specified initial condition is given as follows.

**Definition 2** (Reachable set). The set of reachable point for the LTI discrete time system is given by the union of all possible reachable sets in \( t \) instants of time, for all \( t \).

\[ \mathcal{R} = \bigcup_{t \geq 0} R_t, \tag{2.12} \]

where \( R_t \) is defined as in (2.11).

Now rewriting \( x(t) \) as:
\[ x(t) = C_t \begin{bmatrix} u(t-1) \\ \vdots \\ u(0) \end{bmatrix}, \]
where \( C_t = \left[ \begin{array}{cccc} B & AB & A^2 B & \cdots & A^{t-1} B \end{array} \right] \) and \( u \in \mathbb{R}^{np} \). Therefore \( u \) can be understood as coefficients in a linear combination of the matrices in \( C_t \).

This provides us with an alternative way to express the set of reachable points in \( t \) instants of time as
\[ \mathcal{R}_t = \text{Range}(C_t), \]
where, given \( M \in \mathbb{R}^{k \times l} \) we denote the range of \( M \) as \( \text{Range}(M) = \{ b \mid b = Mx, x \in \mathbb{R}^l \} \).

To express the reachable set \( \mathcal{R} \) associated with the discrete time system (2.3) in a more compact way, we require the following theorem.

**Theorem 1.** [Cayley-Hamilton] Given a matrix \( M \in \mathbb{R}^{n \times n} \). Let \( I_n \in \mathbb{R}^{n \times n} \) be the identity matrix with dimension \( n \) and \( p(\lambda) = \det(\lambda I_n - M) \) the characteristic polynomial with respect to \( M \), where \( \det \) is the determinant, then it follows that by replacing \( \lambda \) by the matrix \( A \) in the characteristic polynomial \( p(M) = 0 \).
Theorem 1 states that a matrix satisfies its own characteristic polynomial and moreover that we can express the \( n \)-th power of a matrix \( M \in \mathbb{R}^{n \times n} \) by a linear combination of \( M^0, \ldots, M^{n-1} \). As a consequence of this fact, this implies that the \( \text{Range}(C_t) = \text{Range}(C_n) \), for all \( t \geq n \) and therefore the set of reachable points in \( t \) instants of time can be described as

\[
\mathcal{R}_t = \begin{cases} 
\text{Range}(C_t), & t < n \\
\text{Range}(C), & t \geq n 
\end{cases}
\]

where \( C = C_n \).

A similar characterization for \( \mathcal{R}_t \), for the case of LTI systems in continuous-time can be done by approximating (2.2) using the Euler rule.

**Proposition 1.** The controllability matrix for the continuous time system \( \dot{x}(t) = Ax(t) + Bu(t) \), with \( x \in \mathbb{R}^n \), \( u \in \mathbb{R}^p \), is also \( C \).

**Proof.** Consider the solution for the continuous time LTI (2.2) and without loss of generality let \( x(0) = 0 \),

\[
x(t) = \int_0^t e^{(t-\tau)A} Bu(\tau) d\tau.
\]

Now consider a change of variable \( T = t - \tau \). We can rewrite (2.13) as

\[
x(t) = -\int_{-t}^0 e^{TA} Bu(t - T) dT
\]

Recalling the definition of exponential of a matrix, we have that

\[
e^{TA} = \sum_{k=0}^{+\infty} \frac{(TA)^k}{k!} = I + TA + \frac{T^2 A^2}{2!} + \ldots + \frac{T^n A^n}{n!} + \ldots,
\]

and by Theorem 1 we can now rewrite \( A^k \), for \( k \geq n \) as a linear combination of \( A^0, \ldots, A^{n-1} \), hence

\[
e^{TA} = \sum_{k=0}^{n-1} \alpha_k(T) A^k,
\]
where \( \alpha_k(T) \) is given as a function of \( T \). Rewrite (2.14) as

\[
x(t) = \int_0^t e^{TA}Bu(t-T) \, dT
\]

\[
= \int_0^t \left( \sum_{k=0}^{n-1} \alpha_k(T)A^kB \right)u(t-T) \, dT
\]

\[
= \sum_{k=0}^{n-1} A^kB \int_0^t \alpha_k(T)u(t-T) \, dT
\]

\[
= Cz(t),
\]

where \( C = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix} \) and \( z(t) \) is a vector in \( \mathbb{R}^{np} \) and its \( i \)-th component is given by

\[
z_i(t) = \int_0^t \alpha_k(T)u(t-T) \, dT.
\]

Hence, we conclude that \( x(t) \) is a linear combination of the columns of \( C \) and so \( x(t) \in \text{Range}(C) \).

**Definition 3.** [Controllability matrix] Given a linear system of the form (2.2) or (2.3), the controllability matrix associated to the system is

\[
C = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix}.
\]

\[\Box\]

This leads to an alternative and simpler definition of the reachable set of points for \( t \geq n \) instants of time.

Given Proposition 1, we can now characterize if a system is controllable.

**Definition 4.** [Controllable system] A linear system (2.2) or (2.3), sometimes denoted by a pair \( (A, B) \), is said to be controllable if all the states are reachable, i.e.

\[
\mathcal{R} = \text{Range}(C) = \mathbb{R}^n.
\]

Therefore, we have a simple way to state whether a system is controllable or not.

**Corollary 1.** A linear system given by (2.2) or (2.3) is controllable if and only if \( \text{rank}(C) = n \).

\[\Box\]

The above result provides necessary and sufficient conditions to verify if a LTI system is controllable. Nevertheless, alternative equivalent criteria exists which may be more suitable depending on the goal.

**Theorem 2.** [Popov-Belevitch-Hautus test for controllability [11]] The system (2.2) or (2.3) is controllable if and only if

\[
\text{rank} \left( \begin{bmatrix} A - \lambda I_n & B \end{bmatrix} \right) = n, \quad \forall \lambda \in \mathbb{C}.
\]

\[\Box\]

The following example illustrates the different criteria for checking if a system is controllable.
Example 1. Consider the following LTI system

\[
\dot{x}(t) = \begin{bmatrix} 0 & 0 \\ a_{1,2} & 0 \end{bmatrix} x(t) + \begin{bmatrix} b_1 \\ 0 \end{bmatrix} u(t), \quad x \in \mathbb{R}^2, \quad u \in \mathbb{R},
\]

with \( a_{1,2}, b_1 \neq 0 \). The controllability matrix is given by

\[
C = \begin{bmatrix} B & AB \end{bmatrix} = b_1 \begin{bmatrix} 1 & 0 \\ 0 & a_{1,2} \end{bmatrix},
\]

which has full rank. Therefore, the system is controllable. If we consider instead the result of Lemma 2 we have to check the rank of the matrix

\[
\begin{bmatrix} A - \lambda I & B \end{bmatrix} = \begin{bmatrix} -\lambda & 0 \\ a_{1,2} & \lambda \end{bmatrix}, \quad \text{for any } \lambda \in \mathbb{C}
\]

which, as expected, has full rank for any \( \lambda \in \mathbb{C} \), meaning that the system is controllable, as we already knew. \( \diamond \)

As we said, an LTI system (2.2) is controllable if starting from any initial state we can drive the system to any desired final state in finite time. The following proposition illustrates that fact.

Proposition 2. Let \( x(t) = e^{AT}x(0) + \int_0^T e^{A(T-\tau)} Bu(\tau) \, d\tau \) be the solution of the system (2.2) and let

\[
u(\tau) = B^T e^{A(T-\tau)} W^{-1}(T) \left[ x(T) - e^{AT} x(0) \right], \quad (2.15)\]

where

\[
W(T) = \int_0^T e^{A(T-\tau)} BB^T e^{A(T-\tau)} \, d\tau
\]

is assumed to be invertible. The input \( u \) drives the state from \( x(0) \) to \( x(T) \). \( \diamond \)

Proof. Replacing \( u \) in the expression for \( x \), we have

\[
x(T) = e^{AT}x(0) + \int_0^T e^{A(T-\tau)} B \left[ B^T e^{A(T-\tau)} W^{-1}(T) \left( x(T) - e^{AT} x(0) \right) \right] \, d\tau
\]

\[
= e^{AT}x(0) + \int_0^T e^{A(T-\tau)} BB^T e^{A(T-\tau)} \, d\tau \left[ W^{-1}(T) \left[ x(T) - e^{AT} x(0) \right] \right]
\]

\[
= e^{AT}x(0) + W(T) W^{-1}(T) \left[ x(T) - e^{AT} x(0) \right]
\]

\[
= x(T),
\]

hence, the input \( u \) drives the state from \( x(0) \) to \( x(T) \). \( \square \)

It is even possible to show that the input \( u \) of Proposition 2 is the only input that drives the system from state \( x(0) \) to \( x(T) \). Moreover, \( W \) is the so-called Controllability Gramian.

Theorem 3 (Lyapunov test for controllability [11]). Considering the LTI system (2.2) or (2.3) and assuming \( A \)
is a matrix stable\(^1\). The LTI system is controllable if and only if there is a unique positive-definite solution \(W\) of the following Lyapunov equation

\[
AW + AW^T = -BB^T / \quad AW^T - W = -BB^T.
\]

Moreover, the unique solution is

\[
W = \int_0^\infty e^{A\tau}BB^T e^{A^\tau T} d\tau / \quad W = \sum_{\tau=0}^{\infty} A^\tau BB^T(A^\tau)^T.
\]

The Lyapunov criteria provides additional information about how easy or difficult is to drive a system from an initial condition to a specified goal. In summary, because \(W\) is a positive definite solution its eigenvalues are positive, and each of these are associated with eigenvectors which represent linear combinations of the states of the dynamic system. If an eigenvalue is small it means that it is easy to drive the system toward a state lying close to the direction of such eigenvector, and difficult if a state lies close to the eigenvector associated with the highest eigenvalue. The notion easy/difficult is in close relation with the notion of energy that consists of the weighted norm with \(W\) between the final state and the initial state, i.e. \(E(x_T) = (x_T - x_0)^TW(x_T - x_0)\). One way of studying this notion of easy/difficult is to analyze the principal components and the singular value decomposition of the Gramian associated with the linear system. The main idea is that the singular values of the controllability Gramian correspond to the amount of energy which has to be put into the system in order to move the corresponding states. The singular values of the Gramian provide a measure for the importance of the states, the state with the largest singular value is the one that is mostly affected by control moves. For model reduction, the states less sensitive to the input behavior can be eliminated and, in this way, the reduced system keep the best possible approximation to the full-order system. Many of this ideas are used and extended to analyze controllability of large scale systems using reduced order models, see for example the work in [19].

### 2.2 Structural Control Systems

Given a linear system (2.2) or (2.3), we can represent it as a directed weighted graph with vertices representing the state variables \(x \in \mathbb{R}^n\) and the control variables \(u \in \mathbb{R}^p\), edges representing which state variables affect each other and which control variables affect the state variables. The edge \((u, v)\) have weight \(A_{v,u}\), if \(u, v\) are state variables, or \(B_{v,u}\) if \(u\) is a control variable and \(v\) is a state variable.

**Definition 5.** Given a linear system (2.2) or (2.3), its weighted directed graph representation is given by

\[
\mathcal{D}(A, B) = (\mathcal{X} \cup \mathcal{U}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \cup \mathcal{E}_{\mathcal{U}, \mathcal{X}}),
\]

\(^1\)A matrix \(A\) of (2.2) is said to be stable if all its eigenvalues have non-positive real part. On the other hand, a matrix \(A\) of (2.3) is said to be stable if all its eigenvalues lie within the unitary circle.
with the vertices given by $X = \{x_1, \ldots, x_n\}$ and $U = \{u_1, \ldots, u_p\}$, and the edges given by

$$E_{X,X} = \{(x_i, x_j, A_{j+1,i+1}) | A_{j+1,i+1} \neq 0, 0 \leq i,j \leq n-1\}$$

and $E_{U,X} = \{(u_i, x_j, B_{j+1,i+1}) | B_{j+1,i+1} \neq 0, 0 \leq i \leq n-1, 0 \leq j \leq p-1\}$, where $(x, y, a)$ means that there is an edge from vertex $x$ to $y$ with weight $a$.

**Example 2.** [Weighted directed graph representation of an LTI system] Let us consider the following LTI system with continuous time:

$$\dot{x}(t) = \begin{bmatrix} 8 & 0 & 5 \\ 0 & 2 & 7 \\ 9 & 3 & 2 \end{bmatrix} x(t) + \begin{bmatrix} 3 & 0 \\ 2 & 1 \\ 0 & 1 \end{bmatrix} u(t), \quad x(t) \in \mathbb{R}^3, \ u(t) \in \mathbb{R}^2. \quad (2.16)$$

The weighted directed graph representation of this system is depicted on the left hand side of Figure 2.1.

![Figure 2.1](image.png)

Figure 2.1: (a) depicts the weighted directed graph representation, and (b) depicts the structural directed graph representation. The edges in green belong to $E_{U,X}$ and the one in blue to $E_{X,X}$, the vertices in red are input vertices and the blue ones state vertices. Finally, the blue numbers are the weights of each edge.

Conversely, if we have a weighted directed graph of the form $\mathcal{D} = \langle X \cup U, E_{X,X} \cup E_{U,X} \rangle$, then we can reconstruct the original linear system, once we know that its dynamic is continuous or discrete, by reconstructing the matrices $A$ and $B$.

Instead of a weighted directed graph we will look at the directed graph representation of the system or, in other words, we will only look at the structure of the matrices $A$ and $B$, i.e., the locations of zero or non-zero entries.

**Definition 6.** Given a matrix $M \in \mathbb{R}^{n \times m}$ we define $\bar{M} \in \{0, \star\}^{n \times m}$ as the zero/non-zero matrix that represents the structure of $M$ such that $\bar{M}_{i,j} = \star$ iff $M_{i,j} \neq 0$.

Recall that given $(A, B)$ we say that the system is controllable if $\text{rank}(C) = n$. When considering the structure of this pair of matrices, $(\bar{A}, \bar{B})$, since we no longer have access to the specific parameters of the matrix $A$ (i.e. the weights in the digraph representation) we need to introduce the concept of structural controllability which intuitively states that there exists a pair $(A, B)$ with the same structural pattern as $(\bar{A}, \bar{B})$ that is controllable.
Definition 7. [Structural Controllability] Consider a pair of matrices \((A_0, B_0)\), \(A_0 \in \{0, \ast\}^{n \times n}\), \(B_0 \in \{0, \ast\}^{n \times p}\), we say that the pair is structurally controllable if and only if for all \(\varepsilon \geq 0\) there exist controllable pairs \((A_1, B_1)\) and \((A_2, B_2)\), \(A_1, A_2 \in \mathbb{R}^{n \times n}\), \(B_1, B_2 \in \mathbb{R}^{n \times p}\) such that:

- \(\|A_1 - A_0\| > \varepsilon\) and \(\|B_1 - B_0\| > \varepsilon\);
- \(\|A_2 - A_0\| < \varepsilon\) and \(\|B_2 - B_0\| < \varepsilon\).

Definition 8. [Structural System] Given a linear system as in (2.2) or (2.3), the corresponding structural system is obtained by replacing the matrices \(A\) and \(B\) by their structure \(\tilde{A}\) and \(\tilde{B}\), respectively.

Structural controllability can be verified using the criterion based on generic rank of a matrix, defined next.

Definition 9. [Generic Rank of a Matrix] Considering the equivalence class \(\bar{M} = \{M \in \mathbb{R}^{m \times n} | M_{i,j} = 0\text{ iff }\bar{M}_{i,j} = 0, 1 \leq i \leq n, 1 \leq j \leq m\}\), the generic rank of \(\bar{M}\) is defined as

\[\text{grank}(\bar{M}) \equiv \max_{A \in [\bar{M}]} \text{rank}(A)\]

Note that, if a matrix \(\bar{M}\) verifies \(\text{grank}(\bar{M}) = n\), then the set \(\{A \mid A \in [\bar{M}]\text{ and rank}(A) < n\}\) has zero Lebesgue measure [24].

The seminal work by [7] verifies structural controllability by checking the generic rank of the controllability matrix. Hereafter, we derive the results obtained in [25] by simply verifying the generalized rank of the PHB criteria (Lemma 2).

Lemma 3. [Structural Controllability] Given a pair of matrices \((A, B)\), \(A \in \mathbb{R}^{n \times n}\), \(B \in \mathbb{R}^{n \times p}\) the pair is structurally controllable if and only if

\[\text{grank} \left( \begin{bmatrix} \bar{A} - \lambda I_n & B \end{bmatrix} \right) = n, \quad \forall \lambda \in \mathbb{C}.\]

Proof. A pair of matrices \((A, B)\), \(A \in \mathbb{R}^{n \times n}\), \(B \in \mathbb{R}^{n \times p}\) is structurally controllable if and only if there is an instance of values such that the pair \((A, B)\) is controllable iff this pair \((A, B)\) fulfil the conditions of the Lemma 2, and therefore there is an instance \((A, B)\) with the same sparseness of \((\bar{A}, \bar{B})\), hence the result follows.

One important property about structural matrices that we will use latter is given in the next result.

Proposition 3. Let \(\preceq\) be a relation between equivalence classes of structural matrices given by

\([\bar{M}] \preceq [\bar{M}']\), whenever if \(\bar{M}'_{i,j} = 0\), then \(\bar{M}_{i,j} = 0\).

The relation \(\preceq\) between equivalence classes of structural matrices is a partial order.
Proof. The proof consists in verifying that the relation \(\preceq\) is antisymmetric, transitive and reflexive.

- Reflexivity: \([\bar{M}] \preceq [\bar{M}]\) for all \([\bar{M}]\) follows by definition;
- Transitivity: \([\bar{M}_1] \preceq [\bar{M}_2]\) and \([\bar{M}_2] \preceq [\bar{M}_3]\) then whenever if \(\bar{M}_{1,i,j} = 0\), then \(\bar{M}_{3,i,j} = 0\), hence \([\bar{M}_1] \preceq [\bar{M}_3]\);
- Antisymmetry: \([\bar{M}_1] \preceq [\bar{M}_2]\) and \([\bar{M}_2] \preceq [\bar{M}_1]\) that implies the condition has to be an equivalence, hence follows the equality \([\bar{M}_1] = [\bar{M}_2]\).

\[\blacksquare\]

Lemma 4.

If \([\bar{M}] \preceq [\bar{M}']\) then \(\text{grank}(\bar{M}) \leq \text{grank}(\bar{M}')\).

Proof. To show that this property holds without loss of generality (due to the transitivity of \(\preceq\)), we only need to show that: if \(\bar{M}'\) has only one more non-zero entry than \(\bar{M}\) and \(\text{grank}(\bar{M}) = k\) then \(\text{grank}(\bar{M}') \geq k\). To this end, suppose that the dimension of \(\bar{M}\) is \(l \times m\) and denote by

\[
\bar{S} = \begin{bmatrix}
\bar{v}_1 & \cdots & \bar{v}_k \\
\vdots & \ddots & \vdots \\
\bar{v}_1 & \cdots & \bar{v}_k \\
\end{bmatrix}_{(l \times k)}
\]

a subset of \(k \leq m\) columns of \(\bar{M}\) that lead to \(\text{grank}(\bar{S}) = k\). This implies, by definition of grank, that there exists a numerical realisation \(\bar{S}\) that leads to a subset of vectors

\[
\bar{S} = \begin{bmatrix}
\bar{v}_1 & \cdots & \bar{v}_k \\
\vdots & \ddots & \vdots \\
\bar{v}_1 & \cdots & \bar{v}_k \\
\end{bmatrix}_{(l \times k)}
\]

linearly independent. Now suppose we replace one zero entry of a column vector in \(\bar{S}\) with a non-zero entry (if the change is outside this set of columns, then \(\text{grank}(\bar{M}') \geq k\)), that is \(\bar{v}'_i = \bar{v}_i + \bar{v}, \; \bar{v} = \begin{bmatrix} 0 & \cdots & 0 & * & 0 & \cdots & 0 \end{bmatrix}^T\)

where the location of the non-zero entry is a location on a zero entry in \(\bar{v}_i\). We can consider a permutation matrix \(P\), such that \(\bar{S}' = \bar{S}P\), \(\text{grank}(\bar{S}) = \text{grank}(\bar{S}'P)\) and the zero entry that is changed is on the first row and first column. Thus, we can write

\[
\bar{S}'P = \begin{bmatrix}
\bar{S}'_{1(1 \times k)} \\
\bar{S}'_{2(l-1 \times k)} \\
\end{bmatrix} = \begin{bmatrix}
\bar{s}_1 & \cdots & \bar{s}_k \\
\vdots & \ddots & \vdots \\
\bar{s}_1 & \cdots & \bar{s}_k \\
\end{bmatrix}_{(l \times k)} , \; \bar{S}'_{2(l-1 \times k)} = \begin{bmatrix}
\bar{r}_1 & \cdots & \bar{r}_k \\
\vdots & \ddots & \vdots \\
\bar{r}_1 & \cdots & \bar{r}_k \\
\end{bmatrix}_{((l-1) \times k)}
\]

where the \(1 \times k\) matrix \(\bar{S}'_{1(1 \times k)}\) is a row where the first entry is the new non-zero entry. Now, there are two cases that need to be considered:
1. \( \text{grank}(\bar{S}_{P}^{2((l-1)\times k)}) = k \) and therefore \( \text{grank}(\bar{M}') = k \);

2. \( \text{grank}(\bar{S}_{P}^{2((l-1)\times k)}) < k \). Considering the numerical realization of \( \bar{S}_{P} \) obtained from \( S \) and denoting it by \( S_{P} \),

\[
S_{P} = \begin{bmatrix} S_{P}^{1(1\times k)} \\ S_{P}^{2((l-1)\times k)} \end{bmatrix} = \begin{bmatrix} s_1 & \cdots & s_k \\ \vdots & \ddots & \vdots \\ r_1 & \cdots & r_k \end{bmatrix}_{(l \times k)},
\]

where the \( 1 \times k \) matrix \( S_{P}^{1(1\times k)} \) is the first row of the matrix \( S_{P} \). There are \( \alpha_1, \ldots, \alpha_k \) such that at least one is non-zero and

\[\alpha_1 r_1 + \ldots + \alpha_k r_k = 0, \text{ Moreover, any other solution of different } \alpha_1, \ldots, \alpha_k \text{ is } \omega \alpha_1, \ldots, \omega \alpha_k, \omega \in \mathbb{R} \backslash \{0\}. \]

Again, there are two cases:

(i) \( \alpha_1 = 0 \), that is, the change on the matrix \( \bar{M} \) does not lead to a lower generic rank and thus \( \text{grank}(\bar{M}') \geq k \);

(ii) \( \alpha_1 \neq 0 \), in this case by setting the first entry of \( s_1 \) different from the first entry of the vector

\[
\frac{-\alpha_2 s_2 + \ldots + \alpha_k s_k}{\alpha_1},
\]

it follows that the set of columns \( S_{P} \) is now linearly independent, thus \( \text{rank}(M') \geq k \) which implies \( \text{grank}(\bar{M}') \geq k \).

\[\square\]

Similar to the weighted digraph representation, we can now introduce the structural digraph representation, i.e., the digraph representation of the system structure provided by \((\bar{A}, \bar{B})\).

**Definition 10.** Given a system of the form (2.2) or (2.3), we represent its structural directed graph representation as

\[
\mathcal{D}(\bar{A}, \bar{B}) = \langle X \cup U, \mathcal{E}_{X,X} \cup \mathcal{E}_{U,X} \rangle,
\]

with the vertices given by \( X = \{x_1, \ldots, x_n\} \) and \( U = \{u_1, \ldots, u_p\} \), and the edges given by

\[
\mathcal{E}_{X,X} = \{(x_i, x_j) \mid A_{j+1,i+1} \neq 0, 0 \leq i \leq n - 1, 0 \leq j \leq n - 1\}
\]

and

\[
\mathcal{E}_{U,X} = \{(u_i, x_j) \mid B_{j+1,i+1} \neq 0, 0 \leq i \leq n - 1, 0 \leq j \leq p - 1\}.
\]

\[\diamond\]

Note that \( \mathcal{D}(\bar{A}, \bar{B}) \) is no more than the weighted directed graph \( \mathcal{D}(A, B) \), where we ignore the edges’ weight.

**Example 3.** [**Structural directed graph representation of an LTI system**] Consider the LTI system of Example 2.16, the structural matrices are given by
\[
\bar{A} = \begin{bmatrix}
\star & 0 & \star \\
0 & \star & \star \\
\star & \star & \star \\
\end{bmatrix}, \quad \bar{B} = \begin{bmatrix}
\star & 0 \\
\star & \star \\
0 & \star \\
\end{bmatrix}.
\]

The structural directed graph representation of the system is the one depicted on the right hand side of Figure 2.1.

Up to this point, we have introduced some algebraic characterizations of structural controllability. With that framework, we now address a possible more interesting question: is there a graph criteria that allow us to infer the existence of structural controllability? If so, can such identification be done efficiently, i.e., using polynomial complexity algorithms? As we will see in the sequel, the answers to both questions are positive, but before we focus on this, we have to introduce some fundamental concepts and constructions from graph theory.

**Definition 11.** [Directed Graph] A directed graph (digraph) is an ordered pair \(D = (V, E)\), where \(V\) is a set of vertices and \(E \subseteq V \times V\) is a set of edges, connecting elements in \(V\).

If a vertex belongs to an edge we say that the edge is incident to that vertex. Another useful concepts are the notion of directed subgraph of a directed graph and union of such directed graphs.

**Definition 12.** [Directed Subgraph] Let \(D = (V, E)\) be a directed graph. A directed graph \(D_s = (V_s, E_s)\) is a subgraph of \(D\) if \(V_s \subseteq V\) and \(E_s \subseteq E\).

**Definition 13.** [Union of Directed Graphs] Given a collection of digraphs \(D_i = (X_i, E_{X_i}), i \in \{1, \ldots, m\}\), the union digraph of these digraphs is given by \(D = \bigcup_{i=1}^{m} X_i, \bigcup_{i=1}^{m} E_{X_i}\).

**Definition 14.** [Path] Let \(D = (V, E)\), where \(V\) is the set of vertices and \(E\) is the set of edges. A path from a vertex \(x_0\) to a vertex \(x_n\) in \(D\) is a sequence of vertices \(x_0, x_1, \ldots, x_n\) such that \((x_i, x_{i+1}) \in E\) where \(0 \leq i < n\), i.e., there is an edge between any pair of consecutive vertices. \(x_0\) is called the root of the path and \(x_n\) is called the tip of the path.

This definition is important in order to state what is an elementary path and a cycle.

**Definition 15.** [Elementary Path] An elementary path is a path that does not repeat vertices.

An example of an elementary path is depicted in Figure 2.5.

**Definition 16.** [Cycle] A cycle is a path \(x_0, x_1, \ldots, x_n\) that only repeats the first and the last vertices, that is, \(x_0 = x_n\), or a single vertex \(x_0\) with the edge \((x_0, x_0)\).

An example of a cycle is depicted in Figure 2.2.

Another definitions that will be latter important are the one of a strongly connected directed graph and of direct acyclic graph.

**Definition 17.** [Strongly Connected Directed Graph] A directed graph \(D = (V, E)\) is said to be strongly connected if there is a path from each vertex \(v \in V\) in the graph to every other vertex \(u \neq v \in V\).

An example of a strongly connected directed graph is depicted in Figure 2.3.
Figure 2.2: A cycle with n vertices.

Figure 2.3: A strongly connected directed graph with 5 vertices.

Definition 18. [Direct Acyclic Graph] A directed graph $D = \langle V, E \rangle$ is said to be a directed acyclic graph (DAG) if it does not contain cycles.

Definition 19. [Strongly Connected Component] Let $D = \langle V, E \rangle$ be a directed graph. A subgraph $D_s$ is a strongly connected component (SCC) of $D$ if $D_s$ is a strongly connected directed graph.

Definition 20. [Non-top/Non-bottom Linked Strongly Connected Component] Let $D$ be a directed graph and $N$ one of its strongly connected components. The SCC $N$ is said to be a non-top/non-bottom linked strongly connected component if $N$ has no incoming/outgoing edges from any other strongly connected component.

Given a directed graph $D$, we can create a DAG by visualizing each SCC as a virtual node, where there is a directed edge between vertices belonging to two SCCs if and only if there exists a directed edge connecting the corresponding SCCs in the digraph $D$, the original digraph. The DAG associated with $D$ can be computed efficiently in $O(|V| + |E|)$ [5].

An example of a digraph with its strongly connected components, some of them non-top linked and non-bottom linked is depicted in Figure 2.4. This example also depicts the construction of the DAG representation of the digraph.

Additionally, consider the following notions:

Definition 21. [State Stem] Let $D = \langle X, E_X, X \rangle$ be a directed graph representation of a linear system. A state stem is an elementary path composed exclusively by state vertices, or a single state vertex.

An example of a state stem is depicted in Figure 2.5.

Definition 22. [Input Stem] An input stem is an elementary path composed of an input vertex (the root) connected to the root of a state stem.

An example of an input stem is depicted in Figure 2.6.

Definition 23. [State Cactus] A state cactus has the following inductive definition: A state stem is a state cactus. A state cactus connected from any vertex, except the tip, to a vertex in a cycle (not belonging to the state cactus) is also a state cactus.

An example of a state cactus is depicted in Figure 2.8 and, for disambiguation, two examples that are not state cactus are depicted in Figure 2.7.
Figure 2.4: The strongly connected components of a directed graph, the red/green vertices correspond to non-top/non-bottom linked strongly connected components. Visualizing each SCC as a virtual node (represented by the gray boxes), we have the DAG representation of the directed graph.

\[ x_0 \rightarrow x_1 \rightarrow \ldots \rightarrow x_{n-2} \rightarrow x_{n-1} \]

Figure 2.5: An elementary path (and also a state stem) with \( n \) (state) vertices, the \( x_i, 0 \leq i \leq n-1 \).

**Definition 24.** [Input Cactus] An input cactus has the following inductive definition: An input stem with at least one state vertex is an input cactus. An input cactus connected from any vertex, except the tip, to a vertex in a cycle (not belonging to the input cactus) is also a state cactus.

By analogy to the nature plant cactus, a stem is an elementary path and a cactus is a stem with leaves represented by cycles. An example of an input cactus is depicted in Figure 2.9.

Definitions 23 and Definition 24 are examples of subgraphs that the original digraph should contain, more precisely, we need to consider the notion of a spanning subgraph introduced next.

**Definition 25.** [Spanning Subgraph of a Graph] Let \( \mathcal{D} = (X, E_{X,X}) \) be a graph with vertices \( X \) and edges \( E_{X,X} \). We say that the graph \( \mathcal{D}' = (X', E'_{X',X'}) \) is a spanning subgraph of (or that spans) \( (X, E_{X,X}) \) if \( E'_{X',X'} \subseteq E_{X,X} \) and \( X' = X \).

Note that, by definition, the subgraph of a graph which is only constituted by its isolated vertices is always a spanning subgraph. An example of a graph and two spanning subgraphs are depicted in Figure 2.10.

At this point we can now show that an input stem and an input cactus correspond to digraph representations that are structurally controllable. Then, we will show that any digraph representation that is spanned by an input stem or an input cactus is also structurally controllable. Finally we will demonstrate that a system is structurally...
Figure 2.6: An input stem with $n$ state vertices, the $x_i$, $0 \leq i \leq n - 1$, the red vertex ($u_0$) corresponds to the input vertex.

Figure 2.7: (a) and (b) depict two examples of directed graphs which are not state cacti.

Controllable if its digraph representation is spanned by a disjoint union of input cacti. Although this last result is not new, we provide an alternative proof, using Lemma 2.

**Proposition 4.** An input stem is structurally controllable.

**Proof.** Suppose, without loss of generality, that the input stem is given by the elementary path $u_0, x_0, x_1, \ldots, x_{n-2}, x_{n-1}$, as depicted in Figure 2.6. This corresponds to the pair of matrices $\bar{A} \in \{0, \star\}^{n \times n}, \bar{B} \in \{0, \star\}^{n \times 1}$

$$
\bar{A} = \begin{bmatrix}
0 & 0 & \cdots & \cdots & 0 \\
\star & 0 & \cdots & \cdots & 0 \\
0 & \star & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & \cdots & \cdots & 0
\end{bmatrix}, \quad \bar{B} = \begin{bmatrix}
\star \\
0 \\
\vdots \\
0
\end{bmatrix}
$$

Using the Lemma 2 we have that

$$
[\bar{A} - \lambda I \quad \bar{B}] = \begin{bmatrix}
-\lambda & 0 & \cdots & \cdots & 0 & \star \\
\star & -\lambda & 0 & \cdots & 0 & 0 \\
0 & \star & \ddots & \ddots & \vdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
0 & \cdots & \cdots & \cdots & -\lambda & 0
\end{bmatrix}.
$$
To compute its rank we consider two cases:

(a) $\lambda \neq 0$, in this case, looking only at the first $n$ columns, $[\bar{A} - \lambda I]$, we have $\det ([\bar{A} - \lambda I]) = (-\lambda)^n \neq 0$, and therefore

$$\text{grank} \left( \begin{bmatrix} \bar{A} - \lambda I & \bar{B} \end{bmatrix} \right) = n;$$

(b) $\lambda = 0$, in this case $[\bar{A} - \lambda I \bar{B}] = [\bar{A} \bar{B}]$ and each row have a non-zero entry on different columns. Therefore

$$\text{grank} \left( \begin{bmatrix} \bar{A} - \lambda I & \bar{B} \end{bmatrix} \right) = n.$$

Now, consider the case of an input cactus.

**Proposition 5.** An input cactus is structurally controllable.

**Proof.** Suppose that the input cactus is constituted by $k$ cycles and, without loss of generality, with its stem constituted by $u_0, x_0, \ldots, x_{n_0 - 1}$ and its cycles by $x_{n_i}, \ldots, x_{n_{i+1} - 1}$, $x_{n_{i+1} - 1}, x_{n_i}$ with $0 \leq i \leq k - 1$ and $n = n_0 + \ldots + n_k$. The stem represented by the structural matrix $S_{n_0}$ and a cycle represented by the structural matrices $C_{n_i}$ corresponds to blocks of size $n_0 \times n_0$ and $n_i \times n_i$, respectively and given by

$$S_{n_0} = \begin{bmatrix} 0 & 0 & \ldots & \ldots & \ldots & \ldots & 0 \\ * & 0 & \ldots & \ldots & \ldots & \ldots & 0 \\ 0 & * & \cdot & \cdot & \cdot & \cdot & \cdot \\ \vdots & \vdots & \cdot & \cdot & \cdot & \cdot & \cdot \\ \vdots & \vdots & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \ldots & \ldots & 0 & * \\ \end{bmatrix}, \quad C_{n_i} = \begin{bmatrix} 0 & 0 & \ldots & \ldots & \ldots & \ldots & 0 & * \\ * & 0 & \ldots & \ldots & \ldots & \ldots & 0 & \end{bmatrix}. \quad \square$$
Figure 2.10: (a) depicts a graph with 5 vertices and (b) depicts two spanning subgraphs of it (one in red and the other in green).

Therefore, a state cactus is given by

\[
A = \begin{bmatrix}
[S_{n_0}] & 0 & \cdots & \cdots & 0 \\
\vdots & \ddots & \cdots & \cdots & \vdots \\
\vdots & \ddots & L & \cdots & 0 \\
\vdots & \cdots & \cdots & [C_{n_k}] \\
\end{bmatrix}.
\]

The lower block triangular part of $A$, $L$, has exactly one non-zero entry per each row-block intersected with each column-block. Using Lemma 2 we have that

\[
[A - \lambda I B] = \begin{bmatrix}
[S_{n_0}] - \lambda I_{n_0} & 0 & \cdots & \cdots & 0 \\
\vdots & \ddots & \cdots & \cdots & \vdots \\
\vdots & \ddots & L & \cdots & 0 \\
\vdots & \cdots & \cdots & [C_{n_k}] - \lambda I_{n_k} \\
\end{bmatrix} 
\]

where we can conclude that the first $n_0$ rows lead to a generic rank $n_0$, by similar arguments used in the proof of Proposition 5. Now we need to prove that each $i$-th row-block, $2 \leq i \leq k$ have grank $n_i$ for any $\lambda \in \mathbb{C}$, then grank \( \left[ A - \lambda I B \right] \) = $n$ for any $\lambda \in \mathbb{C}$. For each row-block containing
\[ [C_{n_i}] - \lambda I_{n_i} = \begin{bmatrix}
-\lambda & 0 & \cdots & \cdots & 0 & \ast \\
\ast & -\lambda & 0 & \cdots & 0 & 0 \\
0 & \ast & \ddots & \ddots & \vdots & \vdots \\
\vdots & \ddots & \ddots & 0 & \vdots & \vdots \\
\vdots & \ddots & \ddots & 0 & \ast & -\lambda \\
0 & \cdots & \cdots & 0 & \ast & -\lambda 
\end{bmatrix}, \]

we have two cases:

(a) \( \lambda = 0 \), in this case \([C_{n_i}] - \lambda I_{n_i} = [C_{n_i}]\) have exactly one non-zero entry per row on different columns, and therefore

\[ \text{grank} \left( [C_{n_i}] - \lambda I_{n_i} \right) = n_i; \]

(b) \( \lambda \neq 0 \), and we have again two cases

(i) \((-\lambda)^{n_i}\) is different from the product of the non-zero entries \(\ast\) which implies that

\[ \det \left( [C_{n_i}] - \lambda I_{n_i} \right) \neq 0, \]

therefore

\[ \text{rank} \left( [C_{n_i}] - \lambda I_{n_i} \right) = n_i, \]

hence,

\[ \text{grank} \left( [C_{n_i}] - \lambda I_{n_i} \right) = n_i; \]

(ii) \((-\lambda)^{n_i}\) is equal to the product of the non-zero entries \(\ast\) which implies that

\[ \det \left( [C_{n_i}] - \lambda I_{n_i} \right) = 0. \]

Considering that the corresponding row-block of the lower block triangular part, \(L\), has exactly one non-zero entry we can permute this column, say \(r\), with the one in \([C_{n_i}]\) having \(-\lambda\) in the same line as \(r\) has the non-zero entry. Using Cramer’s rule, the determinant is \(\ast(-\lambda)^{n_i-1} \neq 0\). Thus,

\[ \text{rank} \left( [C_{n_i}] - \lambda I_{n_i} \right) = n_i, \]

hence

\[ \text{grank} \left( [C_{n_i}] - \lambda I_{n_i} \right) = n_i; \]

Thus, the rank of the complete matrix is the sum of the ranks of each row block, \(n_0 + n_1 + \ldots + n_k = n\). Remark that by the definition of input cactus it may have an input connected to several state vertices, which corresponds to having more \(\ast\) on row of \(\tilde{B}\), but by Lemma 4 the generic rank will also be \(n\). \(\square\)

From the above we now have an idea of which graph structures are well behaved with respect to structural
controllability. The next step is to characterize, if possible, all the directed graphs that correspond to structural systems (which translates directly to the case of the linear systems where we only consider the structure of the matrices $A$ and $B$) that are structurally controllable. The next theorem states precisely this.

**Theorem 4 (Structurally Controllable Systems [25]).** Given a structural system with matrices $(\bar{A}, \bar{B})$, the following assertions are equivalent:

(i) The structural system $(\bar{A}, \bar{B})$ is structurally controllable.

(ii) $\mathcal{D}(\bar{A}, \bar{B})$ is spanned by a disjoint union of input cacti.

To proof this theorem we will need to use, once again, Proposition 5.

**Proof.**

• $(i) \implies (ii)$

Suppose, by contradiction, that the structural system $(\bar{A}, \bar{B})$ is structurally controllable and $\mathcal{D}(\bar{A}, \bar{B})$ is not spanned by a disjoint union of input cacti. Since the structural system is structurally controllable we have that

$$\text{grank} \left(\begin{bmatrix} \bar{A} - \lambda I_n & \bar{B} \end{bmatrix}\right) = n \quad \text{for any } \lambda \in \mathbb{C}. \quad (2.18)$$

Now we can consider a permutation matrix $P$ that does not affect the grank and which allows us to rewrite

$$\begin{bmatrix} \bar{A} - \lambda I & \bar{B} \end{bmatrix}$$

as

$$\begin{bmatrix} L_0 & P \bar{B} \\ \vdots & \vdots \\ L_{k-1} \\ \end{bmatrix} = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \\ \end{bmatrix},$$

where $L_i$ is a row-block, $0 \leq i \leq k - 1$. Suppose, without loss of generality, that $L_0, \ldots, L_{k-2}$ correspond to a maximal disjoint union of input cacti, i.e. with the maximum number of vertices, and the row-block $L_{k-1}$ does not correspond to a cacti. Assuming that $L_{k-1}$ has $r$ rows, we have that $\text{grank}(L_{k-1}) = r$, due to (2.18). Therefore, there exists at least $r$ non-zero entries in $L_{k-1}$ on different rows and columns. Let us write $L_{k-1}$ as

$$\begin{bmatrix} E_{(r \times (n-r))} & D_{(r \times r)} & B'_{(r \times p)} \end{bmatrix},$$

where the structure of the matrix $D_{r \times r}$ contains at least the following non-zero entries

$$\begin{bmatrix} 0 & 0 & \cdots & \cdots & \cdots & 0 \\ \ast & 0 & \cdots & \cdots & \cdots & \vdots \\ 0 & \ast & \cdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \ast & 0 \end{bmatrix}_{(r \times r)}.$$
and $E_{(r \times (n-r))}$ corresponds to the left block of the matrix $L_{k-1}$. This structure has $r-1$ non-zero entries, hence there exists another column in $L_{k-1}$ that contains the structure $\bar{v} = \begin{bmatrix} \star & 0 & \cdots & 0 \end{bmatrix}^T$. We now analyze the possibilities for the location of $\bar{v}$.

- $\bar{v}$ is in $D_{r \times r}$, which implies that $D_{r \times r}$ contains at least the following non-zero entries

\[
\begin{bmatrix}
0 & 0 & \cdots & \cdots & 0 & \star \\
\star & 0 & \cdots & \cdots & 0 & 0 \\
0 & \star & \cdots & \cdots & 0 & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
0 & \cdots & \cdots & 0 & \star & 0
\end{bmatrix}_{(r \times r)}
\]

Therefore does not have grank $= r$ for all $\lambda \in \mathbb{C}$ since the determinant of $D$ is a polynomial of degree $r$ in $\lambda$, by the Algebra’s Fundamental Theorem, it has $r$ roots in $\mathbb{C}$. This implies that there is another non-zero entry either located in $E$ or in $B'$, but in both cases this implies that $L_{k-1}$ corresponds to an extension of a cactus in $L_i$ for some $0 \leq i \leq k-2$, which contradicts the assumptions;

- $\bar{v}$ is not in $D$ and in this case we consider the three possibilities

  i) $\bar{v}$ is in $B'$, on a column of $B$ that only has one non-zero entry, which implies that $L_{k-1}$ corresponds to an input stem contradicting the assumptions;

  ii) $\bar{v}$ is in $B'$, on a column of $B$ that has more than one non-zero entries, which implies that $D(\bar{A}, \bar{B})$ contains the subgraph depicted in Figure 2.11, corresponding to

\[
\begin{bmatrix}
\bar{A}' - \lambda I_2 & \bar{B}'
\end{bmatrix} = \begin{bmatrix}
-\lambda & 0 & \star \\
0 & -\lambda & \star
\end{bmatrix},
\]

and for $\lambda = 0$ it does not have full grank, that contradicts the assumption that grank$(L_{k-1}) = r$.

![Figure 2.11: One input (red vertex) connected to two state vertices.](image)

iii) $\bar{v}$ is in $E$, which implies that $D(\bar{A}, \bar{B})$ contains the subgraph depicted in Figure 2.12, corresponding to

\[
\begin{bmatrix}
\bar{A}'' - \lambda I_3 & \bar{B}''
\end{bmatrix} = \begin{bmatrix}
-\lambda & 0 & 0 & \star \\
\star & -\lambda & 0 & 0 \\
\star & 0 & -\lambda & 0
\end{bmatrix},
\]

and for $\lambda = 0$ it does not have full grank, which contradicts the assumption that (2.18).
Figure 2.12: One input (red vertex) connected to two state stems with the same origin.

• \( (i) \iff (ii) \)

Suppose that \( D(\bar{A}, \bar{B}) \) is spanned by a disjoint union of input cacti with the disjoint \( r \) cactus \( K_{n_0}, \ldots, K_{n_r-1} \), with sizes \( n_0, \ldots, n_{r-1} \) and \( n = n_0 + \ldots + n_{r-1} \), respectively (the structure of an input cactus is as on the proof of the Proposition 5). Then, the the matrix \( \begin{bmatrix} \bar{A} & \bar{B} \end{bmatrix} \) contains the structure of the following matrix

\[
\begin{bmatrix}
K_{n_0} & 0 & \ldots & \ldots & 0 \\
0 & K_{n_1} & \ddots & \ddots & \ddots \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
\vdots & \ddots & \ddots & \ddots & 0 \\
0 & \ldots & \ldots & 0 & K_{n_{r-1}} \\
\end{bmatrix}
\]

where \( *_j = (0, \ldots, 0, *, 0, \ldots, 0) \) with the * in the \( j \)-th position. For each set of rows corresponding to \( K_{n_i} \), denoting it by \( S_{n_i} \), we have that \( \text{grank}(S_i) = n_i \), by Proposition 5. Therefore, it is easy to check that

\[
\text{grank}(\begin{bmatrix} \bar{A}' & \bar{B}' \end{bmatrix}) = n_0 + \ldots + n_{r-1} = n.
\]

Using Lemma 4 we have that

\[
\text{grank}(\begin{bmatrix} \bar{A}' & \bar{B}' \end{bmatrix}) \leq \text{grank}(\begin{bmatrix} \bar{A} & \bar{B} \end{bmatrix}) \leq n,
\]

Thus \( \text{grank}(\begin{bmatrix} \bar{A} & \bar{B} \end{bmatrix}) = n \).

Given the graph characterization of properties such as structurally controllability, we can raise two different questions:

Q1 Given the structural system matrix \( \bar{A} \), which variables should we actuate to achieve structural controllability? In other words, where should we assign controls, described by \( \bar{B} \) such that \( (\bar{A}, \bar{B}) \) is structurally controllable?

Q2 Given a specific realization of \( (A, B) \) with the same sparseness of \( (\bar{A}, \bar{B}) \), how should we design the control \( u \) such that (2.2) and/or (2.3) achieve a specified goal?

First we address question Q1. To this effect, we first need to introduce the concept of a bipartite graph.
Definition 26. [Bipartite Graph] A graph is a bipartite graph denoted by \( B(V_1, V_2, E_{V_1, V_2}) \) if the vertices can be divided into two disjoint sets \( V_1 \) and \( V_2 \), and \( E_{V_1, V_2} \subseteq \{(v_1, v_2) : v_1 \in V_1 \text{ and } v_2 \in V_2\} \).

Two examples of bipartite graphs are depicted in Figure 2.13.

![Figure 2.13: Two examples of bipartite graphs. (a) depicts a bipartite graph with the set of vertices \( V_1 = \{x_0, x_1, x_2, x_3\} \) and \( V_2 = \{x_4, x_5, x_6, x_7\} \). (b) depicts a bipartite graph with the set of vertices \( V_1 = \{x_0, x_1, x_2, x_3, x_4\} \) and \( V_2 = \{x_5, x_6, x_7\} \).](image)

Given a directed graph \( D = \langle X, E_X, X \rangle \), with vertices \( X \) and edges \( E_X, X \), the bipartite representation is denoted by \( B(\tilde{X}, X, E_{\tilde{X}, X}) \), and due to its graphical representation (for instance, as depicted in Figure 2.13) we say that the first set \( \tilde{X} \) represents the left vertices and the second set \( X \) corresponds to a copy of the first one, and represents the right vertices.

An example of a directed graph and its bipartite representation is depicted in Figure 2.14.

![Figure 2.14: (a) a directed graph; (b) its bipartite representation, the left set of vertices is circumscribed in blue and the right set of vertices is circumscribed in purple.](image)

Given a bipartite graph, additional terminology and some properties upon bipartite graphs can be introduced.

Definition 27. [Matching] Let \( B(V_1, V_2, E_{V_1, V_2}) \) be a bipartite graph, a matching on this graph is a subset of edges \( M \subseteq E_{V_1, V_2} \) such that at most one edge is incident to a vertex.
We say that a vertex \( v \in V_1 \cup V_2 \) is matched if it is incident to some edge in \( M \), otherwise we say that the vertex is free with respect to the matching \( M \). Similarly we say that an edge \( e \in E_{V_1,V_2} \) is matched if \( e \in M \).

**Definition 28.** [Alternating Path] Let \( B \equiv B(V_1, V_2, E_{V_1,V_2}) \) be a bipartite graph and \( M \) be a matching on \( B \). An alternating path on \( B \) is an elementary path such that the edges belong alternatively to the matching \( M \) and not to the matching \( M \).

Given Definition 28 we have the following notion.

**Definition 29.** [Augmenting Path] Let \( B \equiv B(V_1, V_2, E_{V_1,V_2}) \) be a bipartite graph and \( M \) be a matching on \( B \). An augmenting path on \( B \) is an alternating path that starts from a free vertex and ends on a free vertex.

Now, we can define what is a maximum matching and a perfect matching.

**Definition 30.** [Maximum Matching] Let \( B \equiv B(V_1, V_2, E_{V_1,V_2}) \) be a bipartite graph and \( M \) be a matching on \( B \). \( M \) is a maximum matching on \( B \) if for any other matching \( M' \), we have \(|M'| \leq |M|\), where \(|M|\) is the number of edges in the matching \( M \).

In other words, we say that a matching is maximum if it has the maximum possible size.

**Remark 1.** We can compute a maximum matching of a bipartite graph \( B(V_1, V_2, E_{V_1,V_2}) \) with time complexity \( O(|E_{V_1,V_2}|\sqrt{|V_1| + |V_2|}) \) using the Hopcroft–Karp algorithm [5].

**Definition 31.** [Perfect Matching] Let \( B \equiv B(V_1, V_2, E_{V_1,V_2}) \) be a bipartite graph, with \(|V_1| = |V_2|\) and \( M^* \) a maximum matching on \( B \). \( M^* \) is a perfect match if all the vertices are matched.

The importance of the concept of augmenting path is expressed in the following.

**Theorem 5** (Berge [5]). Let \( B \equiv B(V_1, V_2, E_{V_1,V_2}) \) be a bipartite graph and \( M \) be a matching on \( B \). The matching \( M \) is a maximum matching if and only if there is no augmenting path with respect to \( M \).

Finally, we introduce the notion of right/left-unmatched vertices with respect to a maximum matching, that will be handful in the sequence.

**Definition 32.** [Left/Right-Unmatched Vertices] Let \( B \equiv B(V_1, V_2, E_{V_1,V_2}) \) be a bipartite graph and \( M \) be a matching on \( B \). The left/right-unmatched vertices are the free vertices with respect to \( M \) on the set \( V_1 / V_2 \).

Examples of a matching and the corresponding left and right unmatched vertices, an alternating path that is an augmenting path and a maximum matching that is a perfect match are depicted in Figure 2.15.

These concepts are important to state the following Lemma.

**Lemma 5** (Maximum Matching Decomposition [23]). Consider a digraph \( D(\bar{A}) = (\mathcal{X}, E_{\mathcal{X}}, \mathcal{X}) \) and let \( M^* \) be a maximum matching associated with the bipartite graph \( B(\mathcal{X}, \mathcal{X}, E_{\mathcal{X},\mathcal{X}}) \). Suppose \( M^* \) consists of a non-empty set of right-unmatched vertices. Then, \( M^* \) (more precisely, the edges in \( M^* \)), together with the set of isolated vertices, constitutes a disjoint union of cycles and state stems (with roots in the right-unmatched vertices and tips in the left-unmatched vertices) that span \( D(\bar{A}) \). Moreover, such a decomposition is minimal, in the sense that, no other spanning subgraph decomposition of \( D(\bar{A}) \) into state stems and cycles contains strictly fewer number of state stems.
Figure 2.15: (a) depicts in green the edges belonging to a matching associated with the bipartite graph $B(X_1 \equiv \{ x_0, x_1, x_2 \}, X_2 \equiv \{ x_3, x_4, x_5 \}, E_{X_1, X_2}$), and the set of left-unmatched vertices are depicted by black vertices whereas the set of right unmatched vertices is depicted by yellow. (b) depicts an augmenting path with respect to the matching in (a) (red edges), the edges are dashed/filled if they belong/don’t belong to the matching in (a). (c) depicts a maximum matching (also a perfect matching) obtained from the matching in (a) and the augmenting path in (b) (yellow edges).

From the above result we can conclude that given $D(\bar{A})$ and its bipartite representation, the maximum matching problem leads to two different kinds of matched edge sequences in $M^*$, the ones in $M^*$ starting in right-unmatched state vertices, and the remaining sequences of edges that start and end in a matched vertices. These edge sequences represent state stems and cycles, respectively. An illustrative example of this decomposition is depicted in Figure 2.16.

As a particular case of the previous Lemma, with $D(\bar{A}) = (\mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$ and $M^*$ a maximum matching associated with $B(\mathcal{X}, \mathcal{X}', \mathcal{E}_{\mathcal{X}, \mathcal{X}'})$, if $M^*$ is a perfect match, then the edges in $M^*$ correspond to a disjoint union of cycles in $D(\bar{A})$.

Figure 2.16: (a) is the bipartite representation of the directed graph in (b), with a maximum matching (green edges), the set of left-unmatched vertices (black vertices) and the set of right-unmatched vertices (yellow vertices). (b) is the original directed graph with a decomposition, as stated in Lemma 5, (red and green edges), which corresponds to two stems (the edge in red and the stem constituted only by $x_0$) and a cycle (represented by the green edges). The vertices in yellow and black are the origin of stems (right-unmatched vertices) and the terminus of stems (left-unmatched vertices).($x_0$ corresponds to a right and left-unmatched vertex).

Remark that due to the equivalence between the system digraph and the pair of matrices ($\bar{A}, \bar{B}$) of the system (2.2) or (2.3) we have the following. Let $U, E_{U, \mathcal{X}}$ denote the dedicated inputs and its corresponding assignments. Given a state digraph $D(\bar{A}) = (\mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$, we say that a subset of state variables $\mathcal{S}_u \subset \mathcal{X}$ is a feasible dedicated
input configuration (FDIC) if, by assigning dedicated inputs\(^2\) to the state variables in \(S_u\), we have that \(D(\bar{A}, \bar{B}) = (X \cup U, \mathcal{E}_X \cup \mathcal{E}_U)\) corresponds to the digraph representation of a structurally controllable system associated with \((\bar{A}, \bar{B})\). The following result may be used to characterize minimal FDICs [22].

**Theorem 6 (Minimal FDIC [23]).** Let \(D(\bar{A}) = (X, \mathcal{E}_X)\) denote the system digraph and \(B = B(X, X, \mathcal{E}_X, X)\) its bipartite representation. A set \(S_u \subset X\) is a minimal feasible dedicated input configuration if and only if there exist two disjoint subsets \(U_R\) and \(A^c_u\) such that \(S_u = U_R \cup A^c_u\), \(U_R\) corresponds to the set of right-unmatched vertices of some maximum matching of \(B\) with maximum number of right-unmatched vertices in different SCCs, and \(A^c_u\) comprising only one state variable from each non-top linked SCC of \(D(\bar{A})\) without a right-unmatched vertex from \(U_R\).

Remark that any FDIC contains a set of right-unmatched vertices with respect to (w.r.t.) a maximum matching of the state bipartite graph \(B\) and at least one state variable from each non-top linked SCC. To illustrate Theorem 6 we have the example depicted in Figure 2.17.

Figure 2.17: (a) represents \(D(\bar{A}) = (X, \mathcal{E}_X, X)\) and its SCCs represented by the dashed rectangles, where the red edges correspond to \(M^*\), a maximum matching associated with \(B = B(X, X, \mathcal{E}_X, X)\) and the red vertices the corresponding \(U_R\) set of right-unmatched vertices. (b) depicts the set of right-unmatched vertices of some maximum matching of \(B\) with maximum number of right-unmatched vertices in different SCCs, the vertices and edges in red, respectively. (c) reproduces \(D(\bar{A}, \bar{B})\), where the green vertices are the input variables connected with the green edges to the set \(S_u\) of Theorem 6.

With this results we are now able to design the structure of the matrix \(\bar{B}\) associated with the matrix \(B\) in the LTI system (2.3) or (2.2). In particular, we can compute the minimum number of inputs that we need and where to assign them in order to guarantee that the system is structurally controllable. This topic will be developed in the next sections but first we will address Q2, i.e., how to design the control signal \(u\) such that some specified goal is achieved. This is the focus of the next section.

### 2.3 Control Design

In this section we address Q2, i.e., we want to find an optimal control that minimizes a quadratic cost functional associated with a linear system; this control is then designated as optimal control with respect to that cost.

---

\(^2\)a dedicated input is an input which is assigned to a single state variable
A simple solution that leads to an optimal control, consists in considering a quadratic cost function, in addition to
the LTI system (2.2) or (2.3), such that the final goal is the state to be identically zero. Such problem is commonly
referred as the linear quadratic regulator (LQR), which is formalized next.

Different LQR problems can be posed upon the fact that we are using discrete/continuous LTI systems and the
duration available to perform the control of the system. This duration is commonly referred as time window and
can be finite or infinite, where its terminal time is the so-called horizon. Alternatively, we can consider different
windows of time as time evolves, leading to the receding LQR and to be used in the sequence.

For simplicity, let us start by considering the discrete time and finite horizon LQR. Formally, we are interested
in solving the following problem:

\[
\begin{align*}
\text{Find} & \quad u^* = \arg \min_{u \in \mathbb{R}^p} J(u) \\
\text{subject to:} & \quad x(t + 1) = Ax(t) + Bu(t), \quad t = 0, \ldots, N - 1
\end{align*}
\]

where

\[
J(u) = \sum_{\tau = 0}^{N-1} (x(\tau)^TQx(\tau) + u(\tau)^TRu(\tau)) + x(N)^TQ_fx(N), \quad u = (u(0), \ldots, u(N - 1)),
\]

\[Q = Q^T \geq 0, \quad Q_f = Q_f^T \geq 0, \quad R = R^T > 0,\]

where for a given square matrix \(A\), \(A > 0\) means that \(A\) is definite positive and \(A \geq 0\) means that \(A\) is semi-definite positive.

In order to solve this problem we will introduce the notion of dynamic programming. The main idea is to
solve smaller independent problems of the same kind. In our case we assume that we know the cost of going from
\(x(t) = z\) to the final state, therefore we would like to solve the following optimization problem

\[
V_t(z) = \min_{u(t), \ldots, u(N-1)} \sum_{\tau = t}^{N-1} (x(\tau)^TQx(\tau) + u(\tau)^TRu(\tau)) + x(N)^TQ_fx(N)
\]

subject to \(x(t) = z, x(\tau + 1) = Ax(\tau) + Bu(\tau), \tau = t, \ldots, N.\)

By assuming that \(V_{t+1}(z) = z^TP_{t+1}z\), with \(P_{t+1} = P_{t+1}^T \geq 0\), we can verify that \(V_t\) has a similar quadratic
form. Therefore, we can rewrite \(V_t(z)\) as

\[
V_t(z) = z^TQz + \min_w (w^TRw + (Az + Bw)^TP_{t+1}(Az + Bw))
\]

which can be solved by setting its derivative to zero

\[
\frac{d}{dw}(w^TRw + (Az + Bw)^TP_{t+1}(Az + Bw)) = 0
\]

\[2w^TR + 2(Az + Bw)^TP_{t+1}B = 0\]

\[w^T(R + B^TP_{t+1}B) = -(Az)^TP_{t+1}B\]

\[(R + B^TP_{t+1}B)^Tw = -B^TP_{t+1}^T Az\]

\[w = -(R + B^TP_{t+1}B)^{-1} B^TP_{t+1}^T Az\]

\[w^* = -(R + B^TP_{t+1}B)^{-1} B^TP_{t+1} Az.\]
Therefore we have

\[
V_t(z) = z^\tau Q z + w^\tau R w^* + (Az + Bw^*)^\tau P_{t+1}(Az + Bw^*) \\
= z^\tau (Q + A^\tau P_{t+1}A - A^\tau P_{t+1}B(R + B^\tau P_{t+1}B)^{-1}B^\tau P_{t+1}A)z \\
= z^\tau P_t z,
\]

where \(P_t = Q + A^\tau P_{t+1}A - A^\tau P_{t+1}B(R + B^\tau P_{t+1}B)^{-1}B^\tau P_{t+1}A\), is a symmetric and semi-definite positive.

The previous result, allows us to describe an iterative algorithm to find an optimal control with respect to the discrete and finite time LQR. Briefly, first we compute the quadratic form for all the instance of time (described by \(P_t\)) in a backward fashion, i.e., from the ending time to the initial time. Afterward, based in these quadratic forms we can compute an explicit optimal control that is a function of \(P_t\) and the actual state of the system \(x(t)\). The detailed procedure is as follows:

1. set \(P_N := Q_f\)
2. for \(t = N, \ldots, 1\)
   \[
P_{t-1} := Q + A^\tau P_t A - A^\tau P_t B(R + B^\tau P_t B)^{-1}B^\tau P_t A
   \]
3. for \(t = 0, \ldots, N - 1\), \(K_t := -(R + B^\tau P_{t+1}B)^{-1}B^\tau P_{t+1}A\)
4. for \(t = 0, \ldots, N - 1\), \(u(t) := K_t x(t)\)

Now, consider the discrete case with infinite horizon. In this case we have a similar optimization problem with

\[
J(u) = \sum_{\tau=0}^{\infty} (x(\tau)^\tau Q x(\tau) + u(\tau)^\tau R u(\tau)) x(\tau), \quad u = (u(0), u(1), \ldots),
\]

where \(Q = Q^\tau \geq 0\), \(R = R^\tau > 0\), and where the optimal control sequence is given by

\[
u(t) = -K x(t),
\]

where \(K = (R + B^\tau P B)^{-1}B^\tau P A\) and \(P\) is the unique positive definite solution to the discrete time algebraic Riccati equation [14]:

\[
P = Q + A^\tau (P - PB(R + B^\tau P B)^{-1}B^\tau P)A.
\]

On the other hand, if we consider the continuous time with finite horizon, the optimization problem is similar but constrained to the continuous dynamic system and the cost function given by

\[
J(u) = \int_0^T (x(\tau)^\tau Q x(\tau) + u(\tau)^\tau R u(\tau)) \, d\tau + x(T)^\tau Q_f x(T),
\]

where \(Q = Q^\tau \geq 0\), \(Q_f = Q_f^\tau \geq 0\) and \(R = R^\tau > 0\).
The feedback control law that minimizes the value of the cost is \( u(t) = K_t x(t) \), where \( K_t = R^{-1} B^T P_t \) and \( P \) is found by solving the continuous time algebraic Riccati equation [14]

\[
-\dot{P}_t = A^T P_t + P_t A - P_t B R^{-1} B^T P_t + Q, \quad P_f = Q_f.
\]

Finally, consider the continuous case with infinite horizon. In this case we have the same optimization problem constrained to the continuous dynamic system and the functional cost

\[
J(u) = \int_0^{\infty} (x(\tau)^T Q x(\tau) + u(\tau)^T R u(\tau)) \, d\tau,
\]

\( Q = Q^T \geq 0, \quad R = R^T > 0 \). The feedback control law that minimizes the value of the cost is \( u(t) = K x(t) \), where \( K = R^{-1} B^T P \) and \( P \) is found by solving the continuous time algebraic Riccati equation (ARE)

\[
A^T P + PA - PBR^{-1} B^T P + Q = 0.
\]

Similarly to the discrete and finite time system, there is an iterative procedure to design the control function \( u(t) \) for the discrete and the continuous case.

### 2.4 Hybrid Dynamical Systems

In addition to the previous section, we now introduce the notion of hybrid dynamical systems, which is an abstraction that models several physical phenomena and incurs in well-known problems from the application point of view. Mainly, the simulation of such systems imposes strong constraints on numerical approximations and scalability of the known systems as the system increases in its state space. Briefly, we want to introduce the basic notions and give emphasis to some of the complexity issues and afterwards introduce the main results of this thesis, i.e., an abstraction of hybrid dynamic systems which will be called structural hybrid systems.

A hybrid dynamic system is a dynamical system that exhibits both continuous and discrete variables. The evolution of a hybrid dynamical system is given by equations of motion that generally depend on all the variables that model a physical phenomenon. These equations contain a combination of logic, discrete (digital) dynamics, and continuous (analog) dynamics. The discrete dynamics of the system is governed by jump relations and the continuous dynamics is generally given by differential equations. There are several systems which are modeled by hybrid dynamic systems. Physical and biological systems are examples of this.

To illustrate the concept of hybrid system consider the following examples:

**Example 4. [The Bouncing Ball]**

The ball (viewed as a point-mass) is dropped from an initial height and bounces off the ground, dissipating part of its energy with each bounce, see Figure 2.18. It exhibits continuous dynamic between each bounce, however, as the ball impacts the ground, its velocity undergoes a discrete change modeled after an inelastic collision. In the simpler case, let \( h \) be the height of the ball and \( v \) the velocity, assuming that the ball can only have vertical oscillations, then its mathematically description can be done as follows:

\[
32
\]
• When \((h, v) \in \mathcal{C} = \{h > 0\}\), the flow is governed by \(\dot{h} = v, \dot{v} = -g\), where \(g\) is the acceleration due to gravity. This means when the ball is above ground, it is being drawn to the ground by gravity.

• When \((h, v) \in \mathcal{D} = \{h = 0\}\), the jumps are governed by \(h^+ = h, v^+ = -\gamma v\), where \(0 < \gamma < 1\) is a dissipation factor. This is saying that when the ball hits the ground, the velocity is reversed and decreased by a factor of \(\gamma\).

Figure 2.18: Ball’s height \(h\) evolution along the time \(t\).

Example 5. [Thermostat] Consider a room being heated by a radiator controlled by a thermostat. Assume that when the radiator is off, the temperature, \(x \in \mathbb{R}\), of the room decreases exponentially towards 0 degrees according to the differential equation

\[
\dot{x} = -ax,
\]

for some \(a > 0\).

When the thermostat turns the heater on the temperature increases exponentially towards 30 degrees, according to the differential equation

\[
\dot{x} = a(x - 30).
\]

Let \(x\) denotes the temperature in the room and \(s\) denotes the state of the thermostat, \(s = 1\) if it is ON and \(s = 0\) if it is OFF. A hybrid dynamic system describing the thermostat is as follows:

• When \((x, s) \in \mathcal{C} = \{(x, s) \mid 0 < x < 30, s \in \{0, 1\}\}\), the flow is governed by \(\dot{x} = -ax\), if \(s = 0\) and \(\dot{x} = a(x - 30)\), if \(s = 1\).

• When \((x, s) \in \mathcal{D} = \{(x, s) \mid x = 0 \lor x = 30, s \in \{0, 1\}\}\), the jumps are governed by \(s := 1 - s\), that is, the thermostat changes its state when the temperature \(x\) reaches either the lower bound or the upper bound.
Given a hybrid dynamic system, there are some properties about the system that we would like to check. For instance we could verify if the height of the bouncing ball is always non-negative or we could ask if the temperature of the room always stays between a certain range of values.

In order to verify properties about a hybrid dynamic system we have to have some formalism to model the system and check the desired properties. There are different formalisms, hereafter we present the hybrid automaton, that has demonstrated successful results.

2.5 Hybrid Automata

A hybrid automaton is a mathematical model for precisely describing systems in which digital computational processes interact with analog physical processes [10]. A hybrid automaton is a finite state machine with a finite set of continuous variables whose values are described by a set of ordinary differential equations. This combines specifications of discrete and continuous behaviors which allows to model and analyze dynamical systems that comprise both digital and analog components. Therefore, hybrid automata are used to do model checking of hybrid dynamical systems.

Definition 33. [Hybrid Automaton] A hybrid automaton is constituted by:

- **continuous state space** \( \mathbb{R}^n \);
- **finite directed graph**: vertices \( \mathcal{Q} \) (modes), edges \( \mathcal{E} \) (control switches);
- **flows** \( \varphi_q \), where \( \varphi_q(t; x) \in \mathbb{R}^n \) is the state reached after staying in mode \( q \in \mathcal{Q} \) for time \( t \geq 0 \) when continuous evolution starts in state \( x \in \mathbb{R}^n \);
- **evolution domain constraints** \( \text{inv}_q \subseteq \mathbb{R}^n \) for \( q \in \mathcal{Q} \);
- **jump relations** \( \text{jump}_e \subseteq \mathbb{R}^n \times \mathbb{R}^n \) for edges \( e \in \mathcal{E} \) usually comprising guard on current state and reset relations.
The hybrid automata representation of the bouncing ball and thermostat presented in the Example 4 and Example 5 are depicted in Figure 2.20(a) and Figure 2.20(b), respectively.

Figure 2.20: A hybrid Automaton representing the bouncing ball, and (b) a hybrid Automaton representing the thermostat.

These examples are particularly simple, but in general, the state space is much larger and the task of verifying properties on its automata representation can be, in general, quite complex.

There are some important classes of properties that are usually used to characterize hybrid dynamical systems. For instance, in the safety verification problem for hybrid systems [3], a hybrid system model $M$, a set of initial states $Q_0$ of $M$, and a set $S$ of “safe” states of $M$ are given, and the goal is to verify whether every execution of $M$ starting in an initial state always stays within the set of safe states and, if not, a counter example should be provided. Another example is the reachability problem, where in this case, a set of initial states $Q_0$ of $M$, and a set $B$ of “bad” states of $M$ are given, and the goal is to verify if every execution of $M$ starting in an initial state does not reach a state in $B$. If it does, again a counter example should be provided.

It is important to stress that the above problems cannot be solved efficiently with the exception of classes that severely restrict the dynamics of real-valued variables. A possible tool to verify safety and reachability properties is the symbolic representation approach that tries to compute the set $R$ of reachable states of a hybrid system $M$ using an iterative procedure. The algorithm verifies at each iteration if $R \subseteq S$. If not, it provides a counter-example.

Several approaches consist on approximation schemes, given that the safety and verification problems are in general undecidable [10]. This is essentially due to the infinite domains and the possibility of unbounded invariants and the evolution domain constraints. In fact, even approximation methods exhibit numerical instabilities and suffer from the curse of dimensionality, see [8, 1] and references therein. Besides the adversities, there are several verification tools for hybrid systems, based in hybrid automata, that can deal with complex and large-scale systems, see for instance [26].

35
Chapter 3

Structural Hybrid System

In order to overcome some of the problems described in the previous sections, we introduce a new concept, named structural hybrid system. With this concept we will provide sufficient conditions to verify a relevant class of hybrid systems: linear time invariant switching systems. We now recall the problem statement:

\[ P_2 \]

Given structural matrices \( \bar{A}(\sigma(t)) \), with \( \bar{A} \in \{0, \star\}^{n \times n} \), where \( \star \) stands for a non-zero entry, and \( \sigma(t) : [0, +\infty[ \rightarrow \mathbb{R} \) is a piecewise constant switching signal satisfying the dwell-time property for some \( \varepsilon > 0 \), find \( \bar{B}(\sigma(t)) \in \{0, \star\}^{n \times p} \) such that the (structural) dynamical system

\[ \dot{x}(t) = \bar{A}(\sigma(t))x(t) + \bar{B}(\sigma(t))u(t), \]

is structurally controllable for all \( t \geq 0 \) and \( \bar{B}(\sigma(t)) \) comprises at most one non-zero entry for each column, i.e., uses dedicated inputs only.

In this chapter, the main contribution consists in the development of a framework to address verification of properties such as controllability, with a polynomial complexity. Note also that properties like reachability and safety can also be inferred. To illustrate the new concepts we are going to consider systems with only two states and latter we will introduce the general concept.

A switching system corresponds to allow the entries of the matrices \( A \) and \( B \) to change along the time. These evolutions along the time impose changes on the domain of the system, for instance if we have a linear system with continuous time \( \dot{x}(t) = A_1x(t) + B_1u(t),\ x \in \Omega_1 \) it can evolve to another linear system \( \dot{x}(t) = A_2x(t) + B_2u(t),\ x \in \Omega_2 \), with the transition of the system given by a guard on the state variable \( x(t) \). These evolutions of the system are depicted in a hybrid automaton in Figure 3.1.

**Proposition 6.** Consider a linear system \( \dot{x}(t) = Ax(t) + Bu(t), \) let \( \Omega \) be an open and connected set, if the pair of matrices \( (A, B) \) is controllable then we can ensure that \( x(t) \in \Omega \).

**Proof.** If the pair of matrices \( (A, B) \) of the linear system \( \dot{x}(t) = Ax(t) + Bu(t) \) is controllable, then given any final state \( x(T) \) we can transfer the initial state \( x(0) \) to it. In other words, the set of reachable states is
Figure 3.1: Hybrid system with two states and two discrete jumps, where each state represents a switching system with the state variable on a certain domain.

\[ \mathcal{R}_t = \mathbb{R}^n \]. Therefore, given an initial state \( x(0) \in \Omega \) and a final state \( x(T) \in \Omega \) we can transfer any \( x(t) \in \Omega \) to \( x(t + \varepsilon) \in \Omega \) and incrementally transfer \( x(0) \) to \( x(T) \) without leaving the set \( \Omega \).

The construction provided in the proof of Proposition 6 is illustrated in Figure 3.2.

Figure 3.2: Illustration of the proof of Proposition 6, (a) shows the initial state \( x_0 \) and the final state \( x_T \) and (b) and (c) illustrate the first steps of the incremental transference of the initial state to the final state.

The next results follows immediately.

**Corollary 2.** Given a switching system (2.1), using Proposition 6 applied to each mode, the result holds for the switching entire system.

Observe that, if the structural hybrid system is structurally controllable, then, almost surely, using Corollary 2, it is controllable.

With Proposition 6 in mind we can abstract our previous model and think about the controllability of the pair of matrices \((A_1, B_1)\) and \((A_2, B_2)\) where the system transition is, once again, a guard on the state variable \( x(t) \).

We refer the reader to Figure 3.3 for a graphical representation of the new aforementioned abstraction.

Using the concepts introduced in Section 2.2, we can think about the structural controllability of the system. In other words, the structural controllability of the pair of matrices \((\bar{A}_1, \bar{B}_1)\) and \((\bar{A}_2, \bar{B}_2)\) where the system transitions correspond to changes on the structure of the pair of structural matrices of the system. Moreover, we can think on the directed graph representation, in which this transition corresponds to a transition from \( \mathcal{D}(\bar{A}_1, \bar{B}_1) \) to \( \mathcal{D}(\bar{A}_2, \bar{B}_2) \), and it consists in removing and/or adding edges on the directed graph (i.e, in changes on the structure of \( \bar{A}_i \) and/or \( \bar{B}_i \), \( i = 1, 2 \)). An example of this is depicted in Figure 3.5.
Now, consider the case illustrated by Figure 3.1 and the one illustrated by Figure 3.3. A natural question to ask is which properties can be inferred from the new abstraction?, or in other words, which properties can we verify in the switching system depicted in Figure 3.1 using the switching system depicted in Figure 3.3. The following result holds.

**Proposition 7.** Given an LTI system, the following assertions are equivalent except for a set with zero Lebesgue measure (that is, with probability 1):

- The LTI system is controllable.
- The structural system of the LTI system is structurally controllable.

Figure 3.4 illustrates this result applied to the corresponding hybrid systems’ abstractions.

In addition, the following propositions address the posed questions.

**Proposition 8.** If $\dot{x}(t) = Ax(t) + Bu(t)$ with $x(t) \in \Omega$ for all $t$, $\Omega$, an open and connected set, then it is not necessarily true that the pair of matrices $(A, B)$ is controllable.
Figure 3.5: Abstraction of the hybrid automata representation of an evolution of a linear system, based on the structural controllability of the digraph representation of the pair of structural matrices of the system.

Proof. In order to prove this we give a counterexample. Let us consider the following dynamical system:

\[
\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = 0
\]

with \( B = 0 \) and \( \Omega = B\varepsilon(0), \varepsilon > 0 \). The controllability matrix of the dynamical system is

\[
C = [B \ AB \ldots A^nB] = [0 \ 0 \ldots 0],
\]

and \( \text{rank}(C) = 0 \neq n \). Therefore the system is not controllable, that is, the pair of matrices \((A, B)\) is not controllable.

With this abstraction steps we can now define a new object based on the hybrid automata concept: the structural hybrid system. In this case, the main idea is to construct an object with nodes that are composed by structural directed graphs \( D(\bar{A}, \bar{B}) \) and the transitions from these states are given by changes on the structure of the pair \((\bar{A}, \bar{B})\), the changes on the edges of \( D(\bar{A}, \bar{B}) \) as we have mentioned. The new definition that we propose is, formally, the following:

**Definition 34.** [Structural Hybrid System] A Structural Hybrid System is constituted by:

- state space:
  
  set of structural directed graphs with \( n \) state vertices and \( p \) input vertices

  \[ D_n = \{D(\bar{A}, \bar{B}) : \bar{A} \in \{0, \star\}^{n \times n}, \bar{B} \in \{0, \star\}^{n \times p}\}; \]

- finite directed graph:
  
  vertices \( Q \) (modes, which are directed graphs), edges \( E \) (control switches);

- jump relations \( \text{jump}_e \subseteq D_n \times D_n \) for edges \( e \in E \):

  if \( q_1 = D(\bar{A}_1, \bar{B}_1) \in Q \) and \( q_2 = D(\bar{A}_2, \bar{B}_2) \in Q \), then \((q_1, q_2) \in \text{jump}_e\) if \((\bar{A}_1, \bar{B}_1)\) differs from \((\bar{A}_2, \bar{B}_2)\) in, at least, one entry.

For a comparison between the structural hybrid system and hybrid automaton, see Table 3.1.
### Structural Hybrid System

- Set of structural directed graphs with \( n \) state vertices and \( p \) input vertices \( D_n = \{ D(A, B) : A \in \{0, \star\}^{n \times n}, B \in \{0, \star\}^{n \times p} \} \)
- finite directed graph: vertices \( Q \) (modes, which are directed graphs), edges \( E \) (control switches)

### Hybrid Automaton

- continuous state space \( \mathbb{R}^n \)
- finite directed graph: vertices \( Q \) (modes), edges \( E \) (control switches)

<table>
<thead>
<tr>
<th>Jump relations ( \text{jump}_e \subseteq D_N \times D_N ) for edges ( e \in E ): if ( q_1 = D(A_1, B_1) \in Q ) and ( q_2 = D(A_2, B_2) \in Q ), then ((q_1, q_2) \in \text{jump}_e) if ((A_1, B_1)) differs from ((A_2, B_2)) in, at least, one entry.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jump relations ( \text{jump}_e \subseteq \mathbb{R}^n \times \mathbb{R}^n ) for edges ( e \in E ) usually comprising guard on current state and reset relations.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Evolution domain constraints ( \text{inv}_q \subseteq \mathbb{R}^n ) for ( q \in Q )</th>
</tr>
</thead>
<tbody>
<tr>
<td>jump relations ( \text{jump}_e \subseteq \mathbb{R}^n \times \mathbb{R}^n ) for edges ( e \in E ) usually comprising guard on current state and reset relations.</td>
</tr>
</tbody>
</table>

**Table 3.1:** Comparison between the structural hybrid system and hybrid automaton.

Note that, two reasons that are the source of the undecidability problems on the hybrid automata theory, the continuous state space (infinite precision semantics) and the possibility of unbounded invariants, no longer exist on the definition of structural hybrid systems, which motivates the use of such concept.

Figure 3.6 illustrates an example of a structural hybrid system. In this case, the idea is to model the situation where the transitions to represent losing edges of the original structural directed graph and, if needed, changes of the placement of the inputs to ensure that all states are structurally controllable. Figure 3.6 starts from the structural directed graph \( D(A, B) \), where

\[
\tilde{A} = \begin{bmatrix}
0 & \star & 0 & 0 \\
\star & 0 & \star & \star \\
0 & \star & 0 & 0 \\
0 & \star & 0 & 0
\end{bmatrix}, \quad \tilde{B} = \begin{bmatrix}
\star & 0 \\
0 & 0 \\
0 & \star \\
0 & 0
\end{bmatrix},
\]

which is structurally controllable.

We now provide some results about the controllability and structural controllability of switching systems. In what follows we assume that the switching signal \( \sigma \) satisfies the dwell time property. First, if a system is controllable in each state, then it follows that it is controllable, formally, we have:

**Proposition 9 ([12]).** Given a switching system (3), if each mode of the switching system is controllable, then the switching system is controllable.

The next result relates Proposition 9 with structural controllability, and follows by definition of structural controllability.
Figure 3.6: Structural hybrid system representation of some discrete transitions from the initial structural directed graph (the graph in the center) to other structural directed graphs where each transition corresponds to losing one edge between state vertices. The edges in red of each directed graph correspond to a decomposition on this graph in cycles and state stems and the green edges correspond to the input placement.

**Corollary 3.** Given a switching systems (3), if each mode of the switching system is structurally controllable, then the switching system is structurally controllable.

An alternative interpretation may be obtained as follows:

**Proposition 10 ([18]).** A switching linear system (3) with state digraphs $D_i$, $i \in \{1, \ldots, m\}$ is structurally controllable if its union graph $D$ is structurally controllable.

Therefore, if a mode in the structural switching system is structurally controllable, it follows that the union of the digraphs corresponding to the states of the structural switching system is structurally controllable. Once again, recall that our goal consists in ensuring that each state of the structural switching system is structurally controllable, hence the system is structurally controllable at all times.

A more interesting question to pose (from a system designer point of view) is: What should be $\bar{B}$ such that at each mode, the system digraph associated with $(\bar{A}, \bar{B})$ is structurally controllable. In other words, what is the solution to problem $P_2$? We address this problem in a restricted setting, more precisely, we only consider systems whose associated DAG representations always consist of the same non-top linked SCC across mode changes, and the remaining SCCs satisfy an additional constraint. Otherwise, the edge set of each SCC may vary with mode (structural) transitions, caused, for instance, by failures or natural switches.

Specifically, from this point onwards we consider the following additional assumption:
A1. The DAG representation of $D(\bar{A})$ has only one non-top linked SCC and all the other SCCs originate\(^1\) a perfect matching.

There are several physical systems with this property, in particular, electric power grids as modeled in [13], which we explore later. Also, note that requiring an SCC to originate a perfect matching is not a very restrictive assumption, since, in practice, most of the diagonal entries in the system matrix are non-zero, which correspond to self-loops in the system’s digraph and may contribute edges to the maximum matching. In particular, if all states in an SCC have self-loops, it readily follows that such an SCC originates a perfect matching.

Under assumption A1, we now obtain a set of results that provide an understanding of why the solution to $P_2$ can be restricted to the analysis of a single (non-top linked) SCC and a maximum matching problem on the bipartite graph, associated with the system digraph. Such understanding will constitute the basis of the design procedure in our model checker, stated in Theorem 2.

**Proposition 11.** Consider a digraph $D(\bar{A})$ with its DAG representation, constituted by $\{N_i\}_{i=0}^{n-1}$ SCCs, where $N_i = (X_i, E_{X_i, X_i})$. Let $N_0$ be the only non-top linked SCC and $B_i = B(X_i, X, E_{X_i, X})$ the bipartite graph associated with $N_i$, $i = 0, \ldots, n$. If $M_i^*$ is a maximum matching associated with $B_i$ ($i = 0, \ldots, n - 1$) and $M_i^*$ is a perfect matching for $i = 1, \ldots, n - 1$, then $M^* = \bigcup_{i=0}^{n-1} M_i^*$ is a maximum matching of $B(X, X, E_{X_i, X})$. \(\diamond\)

The previous Proposition follows immediately by noticing that if the matching $M^*$ was not maximum, then some of the $M_i^*$ was not maximum, which leads to a contradiction. Consequently, we obtain the following result.

**Corollary 4.** Under the same assumptions of Proposition 11, to compute the set of right-unmatched vertices we only need to compute a maximum matching $M_0^*$ of $B_0$, since we can always extend $M_0^*$ to a maximum matching of $B(X, X, E_{X_i, X})$. \(\diamond\)

Corollary 4 states that if a system fulfills the conditions of Proposition 11, then, to construct $\bar{B}$, we just need to consider the design of dedicated inputs restricted to the non-top linked SCC. Where the system become structural controllable, by noticing that set of right-unmatched vertices is a FDIC, as stated in Theorem 6.

Now, suppose we consider a transition between modes in the structural hybrid system. If a structural change in the digraph occurs, two scenarios are possible: 1) the placement of inputs previously considered ensures structural controllable; or 2) a new placement of inputs needs to be considered, i.e., the system must be redesigned to ensure structural controllability. Next, we explore the implications of the structure change in the system’s dynamics, through the edges present in the non-top linked SCC, and corresponding maximum matching.

**Proposition 12.** Let $\mathcal{G}$ be a structural hybrid system with at least two modes, one associated with the directed graph $D(X, E_{X,X})$ and the other to $D(X, E_{X,X} \setminus \{(u, v)\})$, where $(u, v) \in E_{X,X}$ and let $B(X, X, E_{X,X})$ and $B(X, X, E_{X,X} \setminus \{(u, v)\})$ be their bipartite representations, respectively. Additionally let $M \subseteq E_{X,X}$ be a set of edges corresponding to a maximum matching on $B(X, X, E_{X,X})$ and $U_R$ the set of its associated right-unmatched vertices. If $(u, v) \notin M$, then $M$ is also a maximum matching with respect to $B(X, X, E_{X,X} \setminus \{(u, v)\})$ and, consequently, the set of its associated right-unmatched vertices is also $U_R$. \(\diamond\)

\(^1\)We say that a subgraph $D_S = (X_S, E_{X_S,X_S})$ originates a perfect match if the maximum matching associated with the bipartite graph $B(X_S, X_S, E_{X_S,X_S})$ has no right-unmatched vertices.
Proof. Suppose, by contradiction, that $M$ is not a maximum matching associated with $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$, then, once it is a matching (at most one edge is incident to a vertex), there exists an augmenting path $A$ with respect to $M$ and $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$. However this implies that $A$ is also a augmenting path with respect to $M$ and $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$, which contradicts the fact that $M$ is a maximum matching associated with $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$. Therefore we conclude that $M$ is also a maximum matching associated with $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$. \qed

As an immediate consequence we have the following result.

**Corollary 5.** Let $\mathcal{G}$ be a structural hybrid system with at least two modes, associated with the strongly connected directed graphs $\mathcal{D}(\mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$ and $\mathcal{D}(\mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$, where $(u, v) \in \mathcal{E}_{\mathcal{X}, \mathcal{X}}$ and let $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$ and $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$ be their bipartite representations, respectively. Additionally let $M \subseteq \mathcal{E}_{\mathcal{X}, \mathcal{X}}$ be a set of edges corresponding to a maximum matching on $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$ and $U_R$ the set of its associated right-unmatched vertices. If $(u, v) \notin M$, then $\mathcal{D}(\mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$ is also structurally controllable with the same set of inputs $U_R$. \hfill \Diamond

If the eliminated edge belongs to a specific maximum matching of the non-top linked SCC, we obtain the following result.

**Proposition 13.** Let $\mathcal{G}$ be a structural hybrid system with at least two modes, one associated with the directed graph $\mathcal{D}(\mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$ and the other to $\mathcal{D}(\mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$, where $(u, v) \in \mathcal{E}_{\mathcal{X}, \mathcal{X}}$ and let $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$ and $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$ be their bipartite representations, respectively. Additionally let $M \subseteq \mathcal{E}_{\mathcal{X}, \mathcal{X}}$ be a set of edges corresponding to a maximum matching on $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}})$ and $U_R$ the set of its associated right-unmatched vertices. If $(u, v) \in M$, then there exists a maximum matching $\hat{M}$ with respect to $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$ such that $U_R \subseteq \hat{U}_R$ (set of right unmatched vertices with respect to $\hat{M}$) if and only if $\hat{M} = M \setminus \{(u, v)\}$ is a maximum matching for $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$ or there exists an augmenting path in $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$, with respect to $M \setminus \{(u, v)\}$ ending in $v$. \hfill \Diamond

**Proof.** ($\Leftarrow$) There are two cases:

(a) $\hat{M} = M \setminus \{(u, v)\}$ is a new maximum matching of $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$, and therefore the set of right unmatched vertices is $\hat{U}_R = U_R \cup \{v\}$ with respect to $\hat{M}$;

(b) $\hat{M} = M \setminus \{(u, v)\}$ is not a maximum matching of $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$ implies that there exists an augmenting path in $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$, with respect to $M \setminus \{(u, v)\}$, in particular starting in a left unmatched vertex and ending in $v$. Remark that only the vertices belonging to the matched edges are used on the augmenting path, otherwise it contradicts the assumption that $M$ was a maximum matching. Thus, $\hat{M} = M \cup \{(w, v)\}$ for $(w, v) \in \mathcal{E}_{\mathcal{X}, \mathcal{X}}$, where $w$ is a previous matched vertex on the left set of vertices, by construction of the augmenting path, hence $\hat{U}_R = U_R$ is the set of right unmatched vertices with respect to $\hat{M}$.

($\Rightarrow$) Suppose that we lose an edge $(u, v) \in M$ and that there exists a corresponding maximum matching $\hat{M}$ with respect to $B(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X}, \mathcal{X}} \setminus \{(u, v)\})$ such that the associated right unmatched vertices set $\hat{U}_R \supseteq U_R$. There are two cases, the first one is when we still have a maximum matching with the same cardinality as $M$, i.e.
there is an edge that didn’t belong to the previous maximum matching ending in \(v\). The second case is when \(M \setminus \{(u, v)\}\) is a maximum matching for \(B(X, X, E_{X,X} \setminus \{(u, v)\})\). Formally,

(a) \(\hat{U}_R = U_R\), where \(\hat{U}_R\) is the set of right unmatched vertices with respect to \(\hat{M}\) and \(U_R\) with respect to \(M\), meaning that there exists an augmenting path in \(B(X, X, E_{X,X} \setminus \{(u, v)\})\), with respect to \(M \setminus \{(u, v)\}\), ending in \(v\).

(b) \(U_R \subseteq \hat{U}_R\), where \(\hat{U}_R\) is the set of right unmatched vertices with respect to \(\hat{M}\) and \(U_R\) with respect to \(M\), and therefore there is exactly one more right unmatched vertex which has to be \(v\), hence \(\hat{M} = M \setminus \{(u, v)\}\).

\[\square\]

Now, recall that the complexity of finding an augmenting path has complexity \(O(|X| + |E_{X,X}|)\), which implies an efficient method to verify which edges jeopardize the structural controllability.

As a particular case of the previous Proposition 3 we have the following result.

**Corollary 6.** Let \(\Theta\) be a structural hybrid system with at least two modes, associated with the strongly connected directed graphs \(D(X, E_{X,X})\) and \(D(X, E_{X,X} \setminus \{(u, v)\})\), where \((u, v) \in E_{X,X}\) and let \(B(X, X, E_{X,X})\) and \(B(X, X, E_{X,X} \setminus \{(u, v)\})\) be their bipartite representations, respectively. Additionally, let \(M \subseteq E_{X,X}\) be a set of edges corresponding to a maximum matching on \(B(X, X, E_{X,X})\) and \(U_R\) the set of its associated right-unmatched vertices. If \((u, v) \in M\) and there exists an augmenting path with respect to \(D(X, E_{X,X} \setminus \{(u, v)\})\) ending in \(v\), then \(D(X, E_{X,X} \setminus \{(u, v)\})\) is also structurally controllable with the same set of inputs \(U_R\).

\[\diamond\]

Notice that verifying if a graph is strongly connected has complexity \(O(|X| + |E_{X,X}|)\) using the Tarjan’s strongly connected components algorithm [5]. The next results provides a solution to problem \(P_2\).

**Theorem 7.** Let \(\hat{A}(\sigma(I_l))\) represent the structure of \(A(\sigma(I_l))\) on the time interval \(I_l = [t_l, t_{l+1}]\), \((\hat{A}(\sigma(I_l)), \hat{B}(\sigma(I_l))\)) the \(l\)-th state of the structural hybrid system, \(D(\hat{A}) = (X, E_{X,X})\) and \(\Sigma = \{\sigma(I_l), l = 0, 1, \ldots\}\) such that \(|\Sigma| \leq p(|X|)\), where \(p\) is a polynomial on the number of state variables \(|X|\). The solution of \(P_2\) can be found with the following procedure: First, consider the initial state of the structural hybrid system and find \(\hat{B}(\sigma(I_0))\) that ensures structural controllability by proceeding as follows:

1. Create the DAG representation of \(D(\hat{A})\);

2. Compute the maximum matching \(M^*\) for the non-top linked SCC;

3. Construct \(\hat{B}(\sigma(I_0))\) by assigning dedicated inputs to the right-unmatched vertices associated with \(M^*\);

For each possible state corresponding to the system with \(\hat{A}(\sigma(I_{l+1}))\) determine \(\hat{B}(\sigma(I_{l+1}))\) as follows:

1. Detect the set of removed edges from \((\hat{A}(\sigma(I_l)), \hat{B}(\sigma(I_l)))\), and denote by \(E_i\);

2. For each \(e \in E_i\) use the results of Proposition 13 and 12 to design \(\hat{B}(\sigma(I_{l+1}))\).

The complexity of the overall procedure is \(O((\sqrt{|X||E_{X,X}|} + E_{X,X}|E_{X,X}|)p(|X|))\).

\[\diamond\]
In the above, we have only considered the case where one edge in the directed graph fails, however, the results can be readily extended to the case with multiple edge failures. In the latter case, if \( n \) edges fail simultaneously, for analysis purposes, the failures may be viewed as happening sequentially one at a time and the previous results are applicable. Specifically, under the assumptions of Proposition 13, if we lose a set of edges \( E \subseteq M \), if there is a maximum matching with respect to \( D(\mathcal{X}, \mathcal{E}_{\mathcal{X} \setminus X} \setminus E) \) with the same right unmatched vertices, then it is structurally controllable with the same set of inputs \( U_R \), otherwise we have a new set of right unmatched vertices \( \hat{U}_R \) with \( |\hat{U}_R| \leq |U_R| \).
Chapter 4

A power electrical grid application

In this section we provide an illustrative example of the structural hybrid systems in the context of power electrical grids, where link failures, may occur due to fatigue and over heat of transmission lines.

4.1 5-Bus System

The IEEE 5-bus power system, depicted in Figure 4.1, is a standard benchmark model used as proof-of-concept for different methodologies suggested in power systems. It corresponds to a electric power grid composed by 5 buses (depicted by black rectangles), interconnected through transmission lines (depicted by solid lines), and which represents the network topology. Here, we consider three power generators, denoted by $G_i$ ($i = 1, 2, 3$) and two power loads $L_i$ ($i = 1, 2$), coupled through the network topology. Additionally, we adopt the cyber-physical modeling of the generators (as Steam-Turbine-Generators) and loads (as induction machines), similar to the proposed approach in [2], where the linear system is obtained by linearization. Each electrical component has its own dynamics described in more details hereafter.

![Figure 4.1: Graph representation of a 5-bus system.](image-url)
The physical process of a generator G-T-G (governor-turbine-generator) (depicted in Figure 4.2) is defined as a closed-loop continuous time dynamic model as follows:

\[
J\dot{\omega}_G + D\omega_G = P_T + e_t a - P_G
\]

(4.1)

\[
T_u \dot{P}_T = -P_T + K_t a
\]

(4.2)

\[
T_g \dot{a} = -r_a - \omega_G + \omega_G^{ref}
\]

(4.3)

where the variable \( e_t \) is the coefficient of the valve position, \( P_T \) and \( P_G \) are the mechanical and electrical power of the turbine and generator, respectively. The constants \( J, D, T_u, \) and \( T_g \) are, respectively, the moment of inertia of the generator, its damping coefficient, the time constant of the turbine and the time constant of the generator. The state variable \( \omega_G \) corresponds to the generator’s frequency and \( a \) to the valve opening. \( \omega_G^{ref} \) is the set point value of the governor. Equation (4.1) represents the dynamics of the generator characterized by its local state variable \( \omega_G \) and controlled by the mechanical power of turbine \( P_T \), which, by its turn, is controlled by the governor adjusting the fuel valve position \( a \), in response to the locally sensed deviations of \( \omega_G \) from the reference value \( \omega_G^{ref} \), as shown in (4.2) and (4.3), respectively. The G-T-G internal state variables are denoted as \( x_G = \begin{bmatrix} \omega_G & P_T & a \end{bmatrix}^T \). These equations can be stated in a standard state space form as:

\[
\dot{x}_G = \begin{bmatrix}
-\frac{D}{T} & \frac{1}{J} & \frac{e_t}{J} \\
0 & -\frac{1}{T_u} & \frac{K_t}{T_u} \\
-\frac{1}{T_g} & 0 & \frac{T_g}{T_g}
\end{bmatrix} x_G + \begin{bmatrix}
-\frac{1}{J} \\
0 \\
0
\end{bmatrix} P_G,
\]

(4.4)

where \( A_G \) represents the system power plant and it can be seen from (4.4) that the local dynamics \( x_G \) of each G-T-G set are coupled to the network only through the electrical power \( P_G \) sent to the rest of the system.

---

**Figure 4.2: G-T-G set module (Figure from [13]).**
The load is also represented as a module, more precisely, as an induction machine. The load module is as depicted in Figure 4.3, where $L$ is the real energy consumed by the load between two samples ($t$ and $(t + 1)$), $P_L$ is the electrical energy that arrives to the lead, $J_L$ and $D_L$ are parameters of a postulated lead model, whose local physical state variable is $\omega_L$, the frequency measured at the load location. The dynamics of the load module are governed by:

$$J_L \ddot{\omega}_L + D_L \omega_L = -P_L - L,$$

(4.5)

where $J_L$ and $D_L$ denote the effective moment of inertia and the damping coefficient of the aggregated system, respectively. For more details, the reader is referred to [13]. Denoting the state variable of the load dynamics as $x_L = \begin{bmatrix} \omega_L & L \end{bmatrix}^T$ and discretizing the (4.5), the dynamic model of a load module yields

$$x_{L,t+1} = \begin{bmatrix} 1 - \Delta T \frac{D_L}{J_L} & -\frac{1}{J_L}E_L \\ 0 & 0 \end{bmatrix} x_{L,t} + \begin{bmatrix} -\frac{1}{J_L} \\ 0 \end{bmatrix} P_{L,t} + \begin{bmatrix} 0 \\ E_L^T \end{bmatrix} w_L,$$

(4.6)

where $\Delta T$ is the sampling period and $E_L = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$. 

![Load Module Cyber-Physical Dynamics](image)

**Figure 4.3:** Load module (Figure from [13]).

Considering $m$ generators and $n$ loads, given a transmission network incident matrix $M$, the load vector of all interaction variables out of the generators $P_G$ and the interaction variables into the load modules $P_L$ are related to the vector of power flows in lines $p$ as (4.5)
The linearized power flows in lossless transmission lines can be expressed as

\[ p = BM^T \theta, \]  
(4.8)

where the matrix \( B \) is a diagonal matrix whose terms are \( V_i V_j \hat{B}_{i,j} \) for the line connecting nodes \( i \) and \( j \) with respective voltages \( V_i \) and \( V_j \) and \( \hat{B}_{i,j} = \frac{1}{X_{i,j}} \), where \( X_{i,j} \) is the reactance of an inductive transmission line.

Combining the two previous relations defines constraints on the interaction variables and the nodal phase angles \( \theta \) as

\[ \begin{bmatrix} P_G \\ P_L \end{bmatrix} = H \theta, \]  
(4.9)

where \( H = MBM^T \) is the susceptance matrix, that is diagonally dominant.

Subtracting Equation (4.9) for two consecutive samples \( [t] \) and \( [t+1] \) and approximating \( (\theta[t+1] - \theta[t]) \approx \omega[t] \), and combining with the discrete-time models of all modules results in a discrete-time state space model of the energy system as follows:

\[
\begin{bmatrix}
  x_{G,t+1} \\
  x_{L,t+1} \\
  P_{G,t+1} \\
  P_{L,t+1}
\end{bmatrix} =
\begin{bmatrix}
  I + A_G & 0 & C_G & 0 \\
  0 & A_L & 0 & C_L \\
  H_{GG} E_1 & H_{GL} E_2 & I & 0 \\
  H_{LG} E_1 & H_{LL} E_2 & 0 & I
\end{bmatrix}
\begin{bmatrix}
  x_{G,t} \\
  x_{L,t} \\
  P_{G,t} \\
  P_{L,t}
\end{bmatrix} +
\begin{bmatrix}
  0 \\
  0 \\
  E_3 \\
  0
\end{bmatrix}
\begin{bmatrix}
  w_L
\end{bmatrix},
\]  
(4.10)

where

\[
\begin{align*}
x_G & = \begin{bmatrix} x_{G1}^1 & \cdots & x_{Gm}^1 \end{bmatrix}^T, \\
x_L & = \begin{bmatrix} x_{L1}^1 & \cdots & x_{Ln}^1 \end{bmatrix}^T, \\
P_G & = \begin{bmatrix} P_{G1}^1 & \cdots & P_{Gm}^1 \end{bmatrix}^T, \\
P_L & = \begin{bmatrix} P_{L1}^1 & \cdots & P_{Ln}^1 \end{bmatrix}^T, \\
A_G & = \text{blockdiag}(A_{G1}^1, \ldots, A_{Gm}^1), \\
A_L & = \text{blockdiag}(A_{L1}^1, \ldots, A_{Ln}^1), \\
C_G & = \text{blockdiag}(C_{G1}^1, \ldots, C_{Gm}^1), \\
C_L & = \text{blockdiag}(C_{L1}^1, \ldots, C_{Ln}^1), \\
w_L & = \text{diag}(w_{L1}, \ldots, w_{Ln}), \\
E_1 & = \begin{bmatrix} E_{11}^1 & \cdots & E_{1m}^1 \end{bmatrix}, \\
E_2 & = \begin{bmatrix} E_{21}^1 & \cdots & E_{2n}^1 \end{bmatrix}, \\
E_3 & = \begin{bmatrix} E_{31}^1 & \cdots & E_{3n}^1 \end{bmatrix},
\end{align*}
\]
and

\[
E^1_i = \begin{bmatrix}
| & 0 & 0 \\
| & e_i & \vdots \\
| & 0 & 0 \\
\end{bmatrix},
\]

\[
E^2_i = \begin{bmatrix}
| & 0 \\
| & e_i & \vdots \\
| & 0 \\
\end{bmatrix},
\]

\[
E^3_i = \begin{bmatrix}
- & e_{2i-1} & - \\
\end{bmatrix}^T,
\]

with \(e_i\) the \(i\)-th canonical vector of \(\mathbb{R}^m\) for \(E_1\), of \(\mathbb{R}^n\) for \(E_2\) and of \(\mathbb{R}^{2n}\) for \(E_3\). The matrix

\[
\begin{bmatrix}
H_{GG} & H_{GL} \\
H_{LG} & H_{LL}
\end{bmatrix}
\]

corresponds to a block partition of the load flow Jacobian matrix.

Now, we show the feasibility of assumption \(A_1\) in the electrical power grid, as modeled in [13], where a single loss of a transmission line corresponds to the loss of two edges on the system digraph, and where the DAG representation of the system remains the same.

**Proposition 14.** For any power electrical grid modeled as in [13], using Steam-Turbine-Generator, and for any single transmission line failure, we have that the system digraph is composed of several SCCs, where only one is a non-top linked SCC and the others SCCs originate perfect matchings. \(\diamond\)

**Proof.** First, we observe that the digraph representation of a generator dynamics is an SCC, let us denote it by \(SCC_{G_i}, i = 1,\ldots, m\). For the load module, the digraph representation corresponds to an SCC connected to another SCC, which is a state variable with a self-loop, let us denote them by \(SCC_{L_1}^1\) and \(SCC_{L_i}^2\), \(i = 1,\ldots, n\), respectively. When we connect these modules to the buses, since each transmission line of the bus system is bidirectional, and the coupling variables are \(\omega_{G_i}\) and \(\omega_{L_i}\) for the generators and loads, respectively, we obtain a non-top linked SCC connected to the SCCs \(SCC_{L_i}^2\) and each \(SCC_{L_i}^2\) originates a perfect matching. When a transmission line is lost, it corresponds to loosing two edges on the system digraph. Once each component, generator/load, have two transmission lines, if only one transmission line is lost, then the system is still composed by an SCC connected to the same SCCs \(SCC_{L_i}^2\). \(\square\)

**Example**

In this example, we consider a 5-bus system with three Steam-Turbine-Generators given by a common model LTI system, which structural system representation is depicted in Figure 4.4 (as well as its decomposition in cycles and state stems).

The objective is to solve \(P_2\), assuming that only a single transmission line failure can occur. First, notice that assumption \(A_1\) holds for the original system digraph, as well as for the system digraph where one transmission line fails, accordingly with Proposition 14. For illustrative purpose, consider Figure 4.4 and Figure 4.5, that depict the system's original digraph and under the failure of transmission line \(l_1\).
Consider now the design/selection of the dedicated inputs ensuring the structural controllability of the system. To this end, we can use the procedure of Theorem 7 in order so design the matrix \( \bar{B} \). Since for each possible connection line failure the SCCs of the system digraph still have a perfect matching, by Theorem 6, we only need to assign one input to any state variable belonging to the non-top linked SCC in order to ensure structural controllability. However, from the physical point of view, we can only actuate the variables \( \omega_{G_i} \), \( i = 1, 2, 3 \) of the generators. Hence, we can design \( \bar{B} \) as one of the canonical vectors in \( \{ e_2, e_5, e_8 \} \), where \( e_i \in \{ 0, \ast \}^{18} \) is a vector with \( \ast \) in the \( i \)-th position and zero elsewhere. Remark that any subset of \( \{ e_2, e_5, e_8 \} \), with at least one element, also corresponds to a \( \bar{B} \) that ensures structural controllability.

Next, we will design an optimal control law for \( u \) considering 3, 2 and 1 inputs and compare its performance with respect to the convergence and cost of the inputs. The choice of the cost matrices of the LQR was made in order to satisfy the frequency standards of power systems, detailed in the sequence. Later, we provide a discussion of the achieved results.

**Considering 3 inputs**

We start by considering the initial system given in (4.10), where the input matrix is given by

\[
B = \begin{bmatrix}
e_2 & e_5 & e_8 \\
e_2 & e_5 & e_8 \\
e_2 & e_5 & e_8
\end{bmatrix}.
\]
The values of the system variables $A$ of (4.10) are the ones in Table 4.1 and 4.2. Now, in order to design the control function $u(t)$ for the original system and for each possible link failure we will use the discrete finite horizon LQR. Our goal is to keep the frequencies of each connection within the range $[55, 65]$ Hz (the $\omega_{G_i}$ value, $i = 1, 2, 3$), which corresponds to the US requirement standard for power electric grids.

\[
\begin{align*}
  & \begin{array}{|c|c|c|}
    \hline
    J & 1.26 & \gamma & 0.05 \\
    D & 2.0 & T_g & 0.25 \\
    \epsilon_t & 0.15 & r & 0.05 \\
    T_u & 0.2 & \gamma & 0.05 \\
    \hline
  \end{array} & \quad & \begin{array}{|c|c|c|}
    \hline
    D_L & 2 \\
    J_L & 1 \\
    \phi_L & 0 \\
    \hline
  \end{array}
\end{align*}
\]

Table 4.1: Steam turbine values. Table 4.2: Load values.

First we compute the control function $u(t) \in \mathbb{R}^3$ for the entire system, where

\[
\begin{bmatrix}
   H_{GG} & H_{GL} \\
   H_{LG} & H_{LL}
\end{bmatrix} =
\begin{bmatrix}
   4.12143 & 2.07746 & 0. \\
   2.07746 & 5.95121 & 1.96342 \\
   0. & 1.96342 & 4.00198 \\
   2.04397 & 1.91032 & 0. \\
   0. & 0. & 2.03856 \\
   5.95351 & 1.99922 & 4.03777
\end{bmatrix},
\]

and considering as initial condition

\[
x_0 =
\begin{bmatrix}
   59.9241 \\
   0.701406 \\
   -0.477371 \\
   59.0049 \\
   -0.913923 \\
   2.47971 \\
   60.6721 \\
   -0.21701 \\
   0.723808 \\
   0.4984 \\
   -1.29131 \\
   0.768001 \\
   0.900757 \\
   1.86266 \\
   -1.7825 \\
   0.516668 \\
   0.475219 \\
   1.36882
\end{bmatrix}
\]

and $Q = Q_f = 1000I_{18}$, $R = 1000I_3$. with $t \in [0, 40]$.

The LQR solution is depicted on Figure 4.6.

Assuming that a link fails, all the system’s parameters are the same except the following ones,

\[
\begin{bmatrix}
   H_{GG} & H_{GL} \\
   H_{LG} & H_{LL}
\end{bmatrix}
\]

(4.11)

If the link $(i,j)$ fails, the entries of the matrix 4.11 $(i,j)$ and $(j,i)$ are set to zero $(i,j = 1, \ldots, 5)$.

Figures 4.6-4.13 illustrate that in this situation it is still possible to keep the frequency value between the
Figure 4.6: (a) time evolution of $\omega_{G_i}$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when there is no link failure, considering 3 inputs.

Figure 4.7: (a) time evolution of $\omega_{G_i}$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when there is no link failure, considering 3 inputs and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.

desired range even when considering gaussian noise, however the initial point has to be very close to the desired one in order to attain convergence on the LQR method. This is explained by the fact that the matrix of the system $A$ is not well-conditioned ($\kappa(G) = 15.3528$, not close to 1) and also not stable ($\lambda_{max} = 5.78295 \geq 1$).
Figure 4.8: Zoom window of Figure 4.6 (a) and Figure 4.7 (a), respectively depicted in (a) and (b).

Figure 4.9: (a) time evolution of $\omega_G^i$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_1$ fails, considering 3 inputs.

Figure 4.10: (a) time evolution of $\omega_G^i$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_1$ fails, considering 3 inputs and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.
Figure 4.11: Zoom window of Figure 4.9 (a) and Figure 4.10 (a), respectively depicted in (a) and (b).

Figure 4.12: (a) time evolution of $\omega_{G_i}$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_4$ fails, considering 3 inputs.

Figure 4.13: (a) time evolution of $\omega_{G_i}$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_4$ fails, considering 3 inputs and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$. 
Figure 4.14: Zoom window of Figure 4.12 (a) and Figure 4.13 (a), respectively depicted in (a) and (b).
**Considering 2 inputs**

Now we will use only two inputs to control the system, with the matrix

\[ B = \begin{bmatrix} e_2 & e_5 \end{bmatrix}, \]

where \( e_i \) is the \( i \)-th canonical vector of \( \mathbb{R}^{18} \). Starting with the initial point

\[ x_0 = \begin{bmatrix} 60.512005 \\ 0.422562 \\ -0.423566 \\ 60.860023 \\ 0.890687 \\ -0.012781 \\ 59.48275 \\ -0.358725 \\ -0.92605 \\ -0.133518 \\ -1.29605 \\ 0.0861733 \\ 1.18366 \\ -1.05655 \\ 1.39057 \\ -0.0786169 \\ 0.604204 \\ -0.308496 \end{bmatrix} \]

and with the same other parameters, we obtain the following LQR solutions to the control input \( u(t) \) for the entire system and for the possible link failures depicted in Figures 4.15-4.22

![Graph](image)

**Figure 4.15:** (a) time evolution of \( \omega_{G_i} \), \( i = 1, 2, 3 \), and (b) of the control signal, obtained with LQR when there is no link failure, considering 2 inputs.

With two inputs, we have a similar situation as when using three inputs, that is, even with gaussian noise, we are able to keep the frequency values between the desired range.
Figure 4.16: (a) time evolution of $\omega_{G_i}, i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when there is no link failure, considering 2 inputs and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.

Figure 4.17: Zoom window of Figure 4.15 (a) and Figure 4.16 (a), respectively depicted in (a) and (b).

Figure 4.18: (a) time evolution of $\omega_{G_i}, i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_1$ fails, considering 2 inputs.
Figure 4.19: (a) time evolution of $\omega_{G_i}, i = 1, 2, 3,$ and (b) of the control signal, obtained with LQR when link $l_1$ fails, considering 2 inputs and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.

Figure 4.20: Zoom window of Figure 4.18 (a) and Figure 4.19 (a), respectively depicted in (a) and (b).

Figure 4.21: (b) time evolution of $\omega_{G_i}, i = 1, 2, 3,$ and (b) of the control signal, obtained with LQR when link $l_5$ fails, considering 2 inputs.
Figure 4.22: (a) time evolution of $\omega_{Gi}, i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_5$ fails, considering 3 inputs and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.

Figure 4.23: Zoom window of Figure 4.21 (b) and Figure 4.22 (a), respectively depicted in (a) and (b).
Considering 1 input

Now we will use only one input to control the system, with the matrix

\[ B = \begin{bmatrix} e_2 & e_5 \end{bmatrix}, \]

where \( e_i \) is the \( i \)-th canonical vector of \( \mathbb{R}^{18} \). Starting with the initial point

\[ x_0 = \begin{bmatrix} 59.6385 \\ 0.41905 \\ -0.832706 \\ 60.5322 \\ 1.29718 \\ 0.919765 \\ 60.1698 \\ 0.768294 \\ 1.48421 \\ -0.245211 \\ -0.757001 \\ 1.15799 \\ 0.129548 \\ -0.162835 \\ 0.45347 \\ 0.746103 \\ -0.289579 \\ 0.358177 \end{bmatrix} \]

and \( Q = Q_f = 1000I_{18}, \quad R = 2000I_1, \)

and with the same other parameters, we obtain the following LQR solutions to the control input \( u(t) \) for the entire system and for the possible link failures depicted in Figures 4.24-4.23

Figure 4.24: (a) time evolution of \( \omega_{G_i}, \ i = 1, 2, 3, \) and (b) of the control signal, obtained with LQR when there is no link failure, considering 1 input.

Also, with one input, we have a similar situation as when using three or two inputs, that is, even with gaussian noise, we are able to keep the frequency values between the desired range. This confirms the theory which says that this system is controllable with only one input.
Figure 4.25: (a) time evolution of $\omega_{G_i}$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when there is no link failure, considering 1 input and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.

Figure 4.26: Zoom window of Figure 4.24 (a) and Figure 4.25 (a), respectively depicted in (a) and (b).

Figure 4.27: (a) time evolution of $\omega_{G_i}$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_1$ fails, considering 1 input.
Figure 4.28: (a) time evolution of $\omega_{G_i}, i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_1$ fails, considering 1 input and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.

Figure 4.29: Zoom window of Figure 4.27 (a) and Figure 4.28 (a), respectively depicted in (a) and (b).

Figure 4.30: (a) time evolution of $\omega_{G_i}, i = 1, 2, 3$, and (a) of the control signal, obtained with LQR when link $l_3$ fails, considering 1 input.
Figure 4.31: (a) time evolution of $\omega_{G_i}$, $i = 1, 2, 3$, and (b) of the control signal, obtained with LQR when link $l_3$ fails, considering 1 input and gaussian noise with standard deviation equal to 5% of the maximum amplitude of the evolution of $x(t)$.

Figure 4.32: Zoom window of Figure 4.30 (a) and Figure 4.31 (a), respectively depicted in (a) and (b).
Discussion of Results

In each of these cases we provide a brief analysis of the implementation results. In order to keep the frequencies of the power system under the standard range, as we decrease the number of inputs, we have to increase the LQR cost functions, in particular the $Q_f$ cost matrix. This would be expected, since when we use a smaller number of inputs it is expected to have a larger control effort in order to control the system. Moreover, the initial state in all the cases, for the LQR, has to be close to the desired value, which is a consequence of the linearization of the physical system.
Chapter 5

Complexity of the Minimal Controllability Problem

In this chapter, we show that the minimal controllability problem is NP-complete, extending the result in [21]. First, we introduce the set covering problem [5] and the minimal controllability problem [21]. Finally, we provide a reduction from the minimal controllability to the set covering problem.

The set covering problem is a classical NP-Hard problem in computer science and complexity theory and is described as follows.

Set Covering Problem

Given a universe of size \( m \) (set of \( m \) elements) \( U = \{1, 2, \ldots, m\} \) and a set of \( n \) sets \( S = \{S_1, \ldots, S_n\} \) such that

\[
\bigcup_{i=1}^{n} S_i = U,
\]

we would like to find a set of indices \( \mathcal{I}^* \subseteq \{1, 2, \ldots, n\} \), where \( \mathcal{I}^* \) is given by

\[
\mathcal{I}^* = \arg \min_{\mathcal{I} \subseteq \{1, 2, \ldots, n\}} |\mathcal{I}|
\]

subject to: \( U \subseteq \bigcup_{i \in \mathcal{I}} S_i \).

The Minimal Controllability Problem

The minimal controllability problem is a problem of control theory and is described as follows.

Given a continuous time linear system

\[
\dot{x}(t) = Ax(t), \quad A \in \mathbb{R}^{n \times n}, \quad x \in \mathbb{R}^n,
\]
we would like to find the sparsest vector $b$ (with the largest number of zero entries) such that the linear system
\[ \dot{x}(t) = Ax(t) + bu(t), \quad A \in \mathbb{R}^{n \times n}, \quad x, b \in \mathbb{R}^n, \quad u \in \mathbb{R} \]
is controllable.

Next, we show that the minimal controllability problem is reducible to the set covering problem. In order to do that we first introduce the concept of Turing reduction. In [4], Cook introduced the polynomial time reduction, although this reduction had only be formally defined one year later by Karp. A problem $P_1$ is polynomial reducible to a problem $P_2$, denoted by
\[ P_1 \leq^P P_2, \]
if $P_1$ can be transformed, in polynomial time, to $P_2$ and the answer to $P_2$ provides an answer to $P_1$.

**The Minimal Controllability Problem is reducible to The set Covering Problem**

**Proposition 15.** Consider a linear system of the form $\dot{x}(t) = Ax(t)$, $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times n}$ whose eigenvectors are $v_1, \ldots, v_n$ and for which we want to find a vector $b \in \mathbb{R}^n$, a solution to the minimal controllability problem ($P_2$) associated with the linear system. Denoting the location of the zero and non-zero entries of the eigenvectors by $\bar{v}_i$, $i = 1, \ldots, n$, the set covering problem ($P_1$) defined by
\[ S_i = \{ j \mid \bar{v}_j[i] \neq 0, \quad 1 \leq i \leq n \}, \quad 1 \leq j \leq n, \]
\[ S = \{ S_1, \ldots, S_n \} \]
and
\[ \bigcup_{i=1}^{n} S_i = U, \]
is such that $I$ is a solution for this set covering problem if and only if the vector $\bar{b}$ defined as an $n$ dimensional vector such that its $i$-th entry is 0 when $i \notin I$ ($b[i] = 0$ if and only if $i \notin I$) is a solution for the minimal controllability problem ($P_2$) associated to the linear system $\dot{x}(t) = Ax(t)$.

**Proof.** Suppose we have the described set covering problem and minimal controllability problem, both associated with the linear system $\dot{x}(t) = Ax(t)$. Then we have the following equivalences
\[ U = \bigcup_{i=1}^{n} S_i, \]
which is equivalent to
\[ \forall k \exists j \in I \text{ and } k \in S_j, \]
this also means that
\[ \forall k \exists j \ b[j] \neq 0 \text{ and } \bar{v}_k[j] \neq 0, \]
and we can rewrite this as
\[ \forall k \exists j \ b[j] \bar{v}_k[j] \neq 0 \]
and therefore
\[ \forall k \bar{v}_k \cdot \bar{b} \neq 0. \]

The last equivalence, when considering the \( v_i \) (instead of \( \bar{v}_i \)) and a numerical realisation of \( \bar{b} \), denoted by \( b \), is true except for a set with zero Lebesgue-measure since it is possible that for some \( k \in \{1, \ldots, n\} \)
\[ b[j]v_k[j] = -\sum_{l \neq j} b[l]v_k[l]. \]

However we can always find a vector \( b \) with the same location of the zero and non-zero entries such that \( v_i \cdot b \neq 0 \) for \( i = 1, \ldots, n \). This may be obtained by solving a linear system with \( n \) equations and \( n \) unknowns. \( \square \)
Chapter 6

Conclusions

6.1 Achievements

In this thesis we introduced the concept of structural hybrid system and provided a systematic method with polynomial complexity (in the number of the state variables) to obtain the input matrices of the structural hybrid system that ensure structural controllability, for all time. By duality, these results readily extend to the structural observability and corresponding output design. We have also proved the equivalence between an LTI system being structurally controllable and its digraph representation being spanned by a disjoint union of input cacti, using a recent criteria for controllability and, finally, we presented a polynomial turing reduction between the minimal controllability problem and the set covering problem, which allows us to use polynomial known approximation algorithms for the set covering problem in order to obtain approximated solutions for the minimal controllability problem.

6.2 Future Work

As part of future research, interesting open questions consist in integrating fault detection and isolation for the detection of a jump in the structural hybrid system, as well as its implications on the quantitative performance of the physical system. Another case than can be studied is the use of LQR with constraints for the design of the inputs.
Bibliography


