INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700  800/521-0600
DESCRIPTION, ANALYSIS AND CONTROL DESIGN OF DISCRETE STATE AND HYBRID SYSTEMS

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

by

Murat Doğrue, M.S.E.E., B.S.E.E.

The Ohio State University

1995

Dissertation Committee:
Professor Ümit Özgüner
Professor Kevin M. Passino
Professor Vadim I. Utkin

Approved by
Adviser
Department of Electrical Engineering
... to the People who

feel, need, seek, find, love, and establish

Truth.
ACKNOWLEDGMENTS

I sincerely thank my advisor Professor Ümit Özgüner for his concern and assistance in many ways through this work. I also thank Professor Kevin Passino, and Professor Vadim Utkin for their review of this thesis, and, Dr. Sergey Drakunov, Onur Toker and Hubert Chan for various technical discussions.

I would like to acknowledge that the full financial support for this work came from The Ministry of National Education of Turkey.
VITA

August 11, 1967 ........................ Born - İstanbul, Turkey

Sept. 1984 - Sept. 1988 .................. Electrical Engineering, BS Student
                                        İstanbul Technical University
                                        İstanbul, Turkey

Sept. 1988 - Feb. 1990 .................. Electrical Engineering, MS Student
                                        İstanbul Technical University
                                        İstanbul, Turkey

Sept. 1990 - June 1992 .................. Electrical Engineering, MS Student
                                        The Ohio State University
                                        Columbus, Ohio, USA

FIELDS OF STUDY

Major Field : Electrical Engineering

Major Area of Specialization : Control and Systems Theory

Minor Area of Specialization : Communications and Signal Processing
LIST OF PUBLICATIONS

Book Chapters:


Thesis and Proposal:


Journal Articles:


Workshop Presentations:


**Conference Proceedings:**


# Table of Contents

DEDICATION ................................................................. ii  
ACKNOWLEDGMENTS ...................................................... iii  
VITA ................................................................. iv  
LIST OF PUBLICATIONS .............................................. v  
LIST OF TABLES .......................................................... x  
LIST OF FIGURES ......................................................... xi  

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>I Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Historical Background</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Contents of the Thesis</td>
<td>7</td>
</tr>
<tr>
<td>II Dynamic Systems with Finite State Space</td>
<td>12</td>
</tr>
<tr>
<td>2.1 Automata</td>
<td>13</td>
</tr>
<tr>
<td>2.2 Discrete State Systems</td>
<td>15</td>
</tr>
<tr>
<td>III Hybrid Systems</td>
<td>21</td>
</tr>
<tr>
<td>3.1 Models of Hybrid Systems</td>
<td>21</td>
</tr>
<tr>
<td>3.1.1 Discrete Time Model</td>
<td>21</td>
</tr>
<tr>
<td>3.1.2 Continuous Time Model</td>
<td>25</td>
</tr>
<tr>
<td>3.2 Some Properties of Hybrid Systems</td>
<td>26</td>
</tr>
</tbody>
</table>
IV Discretization of Continuous State Systems .............................................. 28
  4.1 Discretization and Nondeterministic Discrete State Systems .... 28
  4.2 Neighbor-Transition Dynamic Systems with Finite State Space ... 31
  4.3 Multiresolutional Discrete State Systems ................................. 34

V Stability of Discrete State Systems and Automata .......................................... 36
  5.1 Reachability and Stabilizability in Automata ............................. 36
  5.2 Stability of Discrete State Systems ........................................... 39
  5.3 Refined Lagrange Stability Analysis ......................................... 44

VI Hybrid System State Space and Stability ................................................. 50
  6.1 The Hybrid System State Space .............................................. 50
  6.2 Stability of a Continuous State in Hybrid Systems ................. 56
  6.3 Hybrid State Stability ............................................................. 62
  6.4 Linear Hybrid Systems ............................................................. 67

VII Control Design ......................................................................................... 72
  7.1 Sliding Mode Control Design in Discrete State Systems ........... 72
  7.2 Sliding Mode Control Design in Hybrid Systems ...................... 78
  7.3 Interconnections of Hybrid Systems ......................................... 82
    7.3.1 Serial Connection ............................................................... 82
    7.3.2 Feedback ........................................................................... 84
  7.4 Lyapunov Based Control for Hybrid Systems ......................... 86

VIII Applications and Examples ...................................................................... 92
  8.1 A Robot in a Two Dimensional Work Space ............................. 92
  8.2 A Double Integrator Example ....................................................... 99
  8.3 A Robot with a Gripper ............................................................... 104
  8.4 Hybrid System Numerical Example .............................................. 111
  8.5 Modeling of a Car as a Hybrid System ....................................... 115

IX Conclusion ................................................................................................. 119
APPENDICES

A Stability of a Set of Matrices .................................................. 126
  A.1 Introduction ................................................................. 126
  A.2 The Problem Definition .................................................. 129
  A.3 Some Results for Stability of Sets ................................. 132
  A.4 Some Properties of USR and LSR ................................. 136
  A.5 Constructive Methods for Asymptotic Stability .............. 139
  A.6 Applications to Control ................................................ 145

BIBLIOGRAPHY ............................................................................. 150
## List of Tables

<table>
<thead>
<tr>
<th>TABLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$Q(X_2, U_2)$</td>
</tr>
<tr>
<td>2</td>
<td>$F_3(X_3, U_3)$</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The class of dynamical systems.</td>
</tr>
<tr>
<td>2</td>
<td>Discrete State System block diagram.</td>
</tr>
<tr>
<td>3</td>
<td>DSS, a network of automata.</td>
</tr>
<tr>
<td>4</td>
<td>Hybrid system block diagram.</td>
</tr>
<tr>
<td>5</td>
<td>Hybrid system configuration.</td>
</tr>
<tr>
<td>6</td>
<td>Discretization of a Continuous State System.</td>
</tr>
<tr>
<td>7</td>
<td>The Partition of the State Space.</td>
</tr>
<tr>
<td>8</td>
<td>A Discrete State System and a Lyapunov Function.</td>
</tr>
<tr>
<td>9</td>
<td>Hybrid System discrete state transition.</td>
</tr>
<tr>
<td>10</td>
<td>The uncertainty in hybrid systems.</td>
</tr>
<tr>
<td>11</td>
<td>Obtaining the partition of a hybrid system.</td>
</tr>
<tr>
<td>12</td>
<td>Unstable and stable hybrid systems.</td>
</tr>
<tr>
<td>13</td>
<td>A discrete state system with 3 states.</td>
</tr>
<tr>
<td>14</td>
<td>A discrete state system with 4 states.</td>
</tr>
<tr>
<td>15</td>
<td>The Sliding Mode Control approach in Automata.</td>
</tr>
<tr>
<td>16</td>
<td>Hybrid System.</td>
</tr>
<tr>
<td>17</td>
<td>A serial connection of two hybrid systems.</td>
</tr>
<tr>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Hybrid system representation of two serial connected hybrid systems.</td>
</tr>
<tr>
<td>19</td>
<td>A feedback hybrid system.</td>
</tr>
<tr>
<td>20</td>
<td>Hybrid system representation of feedback hybrid system.</td>
</tr>
<tr>
<td>21</td>
<td>A closed loop feedback hybrid system.</td>
</tr>
<tr>
<td>22</td>
<td>A closed loop hybrid system.</td>
</tr>
<tr>
<td>23</td>
<td>A hybrid feedback closed loop hybrid system.</td>
</tr>
<tr>
<td>24</td>
<td>A Robot in A Two Dimensional Work Space.</td>
</tr>
<tr>
<td>25</td>
<td>The Regions in The Two Dimensional Work Space.</td>
</tr>
<tr>
<td>26</td>
<td>Sliding Mode Control for the Robot.</td>
</tr>
<tr>
<td>27</td>
<td>The State Trajectory of the Robot.</td>
</tr>
<tr>
<td>28</td>
<td>The noise $d(k)$.</td>
</tr>
<tr>
<td>29</td>
<td>The State Trajectory of the Disturbed Robot.</td>
</tr>
<tr>
<td>30</td>
<td>A State Partition for the Double Integrator.</td>
</tr>
<tr>
<td>31</td>
<td>Discrete State Diagram for the Double Integrator</td>
</tr>
<tr>
<td>32</td>
<td>The Continuous State Trajectory of the Hybrid System.</td>
</tr>
<tr>
<td>33</td>
<td>The Continuous State Trajectory of the Hybrid System for a Different Control.</td>
</tr>
<tr>
<td>34</td>
<td>The Discrete State Trajectory of the Hybrid System for a Different Control.</td>
</tr>
<tr>
<td>35</td>
<td>A robot with a gripper.</td>
</tr>
<tr>
<td>36</td>
<td>Hybrid system model of the controlled robot with the gripper.</td>
</tr>
<tr>
<td>37</td>
<td>The outputs of the hybrid system when $a = 0.4$ with zero initial conditions.</td>
</tr>
<tr>
<td>38</td>
<td>Hybrid system continuous state regions.</td>
</tr>
</tbody>
</table>

xii
39 Hybrid system continuous state trajectories. .................................................. 113
40 A car on a road. ............................................................................................. 115
41 Closed Loop Hybrid System Model for the Car. ....................................... 116
42 Two stable trajectories (a and b), and, an unstable (c) and a stable (d) trajectories obtained by switching between the systems a and b. 127
CHAPTER I

Introduction

Mathematical models of physical systems can be obtained in many ways such that available analysis and design methods are applicable. Often two kinds of variables exist in a model. Some variables take on values which are real numbers, while others assume values from a finite set. In an automobile, for example the speed may be represented as a real number, and the position of the transmission as a member of the set of all possible (finite) positions of the transmission. Another example is a robot. The position of the robot in 3-dimensional space can be represented by a combination of three real numbers, while the task may be represented as an element of a finite set, such as “going left” or “holding object.” We shall call systems where the state variables can assume only a finite, discrete set of values Discrete State Systems (DSS) and systems with only continuous states Continuous State Systems (CSS). In this work we shall develop models that lead to the analysis of the combination mentioned above, which we call Hybrid Systems.
We use the term "Discrete State Systems" (DSS) as a model for a sub class of Discrete Event Systems (DES) which have been studied widely in the literature [56, 9, 63, 73]. The main advantage of using the DSS model is to analyze DES in a simpler and more familiar way. DSS can be seen as a pure automaton so that reachability, controllability and stabilizability properties can be studied [16, 18, 17]. The DSS model which we propose to develop, will be similar to state equations. Thus it plays a dual role connecting the theories of some DES and Continuous State Systems. Furthermore, as in continuous state systems, cascade and feedback connections of DSS can be analyzed easily and the resulting systems can be modeled as DSS [16].

Hybrid system model consists of discrete and continuous state systems. The combination of discrete and continuous state variables defines the hybrid state space. We consider a model of hybrid systems in which the discrete and the continuous state systems are directly connected. We do not use a separate interface in this model instead the interfacing is done at the entries of the discrete and continuous state parts. Since we have to deal with logical and differential-difference dynamics at the same time, some difficulty arises in the analysis and the design. We cannot use standard analysis methods in continuous or discrete state system analysis for hybrid systems since in the combination we see examples of different and interesting behavior which do not occur in classical system models. A continuous (or discrete) state system can also be thought of as a hybrid system with no discrete (or continuous) state system. There-
fore hybrid systems are a larger class of systems than both discrete and continuous state systems as shown in Figure 1.

1.1 Historical Background

In 1956 Kalman [42] and in 1960 Bellman [3] considered finite state approximations to continuous state systems for certain estimation and dynamic programming problems. The finite state control concept was developed in 1970’s [39]. VLSI technology has allowed using programmable microcontrollers since 1975. In 1982, Wimpey suggested a solution strategy for the finite state control problem [80]. Later in 1980’s, finite
state control design algorithms was developed and applied to some practical control problems. Johnson and Kaliski give the historical development in finite state control of continuous state processes in [40]. Here we outline recent developments in hybrid system theory. Using a DAC and an ADC, a Finite State Controller is connected to a continuous state system (plant) and shown to work in a desirable way in [41]. A more general model for controlling continuous state systems with finite events has been introduced by Göllü and Vareya in [29] and by Peleties and DeCarlo in [61]. Stiver and Antsaklis study these systems in which a plant is controlled by a DES Controller through an interface [1]. They show how to choose a state partitioning so that the continuous state system and interface part can be modeled as a deterministic DES [75]. Using similar terminology, Passino and Özbun introduced a hybrid system model which consists of a DES model, a continuous state system and an interface. They carefully define and study reachability and stabilizability properties [58]. Recently, Doğruel and Özbun proposed using just discrete and continuous state systems for modeling hybrid systems [16]. Passino, Michel, and Antsaklis introduce stability concepts for discrete event systems [57, 59]. They apply the definitions of classical stability to discrete event systems and provide analysis methods for various types of stability based on Lyapunov theory.

There are many other proposed ways to model hybrid systems. Holloway and Krogh consider a behavioral model and present the analysis of causality and time
monotonicity[35]. Nerode and Yakhnis use a game framework to obtain digital control programs for continuous state systems [53]. Grossman and Larson take a bialgebra approach to model hybrid systems [30]. They code the dynamics of a nonlinear control system and a finite automaton by a bialgebra of operators and the state space by an algebra of observations. Taking suitable products of these algebras models hybrid systems. Binns, Jackson and Vestal propose a layered architecture [6]. Each layer is described by using a particular formal model which consists of discrete event or continuous state variables. Then these layers together model hybrid systems. Kohn and Nerode present an architectural framework for modeling the interaction between continuous (evolution) and discrete (knowledge) components called a Multiple Agent Hybrid Control Architecture [44]. This model is especially suitable for systems involving multiple decision makers. Sobh, Bajcsy and James have considered implementing a DES observer for the execution of commands sent to a robotic arm [72]. By observing the sequence of events in the system, the state of the automaton is set so that a stage in the manipulation process is represented symbolically and the next actions are taken accordingly. There are many other modeling and design approaches for linguistic/numeric hierarchical intelligent control of systems, for example see [81, 70].

There is also an on-going efforts to study simulation and modeling of hybrid systems by using special computer languages. Benveniste and Le Guernic introduce a mathematical model along with a programming language SIGNAL for describing
and constructing hybrid systems which can model complex applications in nature [4]. Zeigler considers theory of modeling and simulation for both continuous state and discrete event systems [82, 83]. Guckenheimer and Nerode discuss issues for modeling and analyzing hybrid systems and using software tools as a means of validation of hybrid systems by exploring qualitative behavior [31]. A class of hybrid systems have been studied in control theory dealing with jump systems for many years [11, 78, 76]. Abrupt changes in system structures can be modeled by DSS. Thus the resulting systems inherently becomes hybrid. Although the theory needs more work, reasonable results for Markov and semi-Markov jump processes are obtained. Recent results on stability have been given [45] by Krtolica, Özgüner, Chan, et al.

There are also a few hybrid system control implementations. Connell uses a hybrid architecture applied to robot navigation called SSS which combines a servo-control layer, a subsumption layer, and a symbolic layer [13]. Summing up all the advantages of each technique results in the control of a robot which is able to map office building environments and navigate at a reasonable speed. Pomet et al. use a hybrid strategy for the feedback stabilization of nonholonomic mobile robots [62]. A special discontinuous control combines the advantages of time invariant and time-varying smooth controls and avoids their drawbacks. In [2], Balemi, Franklin, et al. apply supervisory control theory to a semiconductor manufacturing piece of equipment. The approach provides flexible design and reliable update of the process.
1.2 Contents of the Thesis

In Chapter II, we study dynamic systems with finite state space which include automata and discrete state systems. Since an important part of hybrid systems is the discrete state part, it is essential to understand the construction well. We can consider discrete state systems as a single automaton. In Section 2.1 we describe automata and introduce the adjacency matrix as a means of modeling it. In Section 2.2, discrete state systems are defined. Another aspect of a discrete state system is that we can consider it as a nonlinear discrete time system. Thus discrete state system construction provides us a bridge to work on both logical and differentiation-difference dynamics.

In Chapter III, we introduce hybrid systems and describe the models we use. A discrete time model is obtained by a discrete time discrete state system and either a discrete or continuous time continuous state system. In this case the discrete state can change only at certain points on the time axis. In continuous time models, however, the discrete states can change any time according to discrete inputs or continuous states. Some properties like reachability and stabilizability of a set of hybrid states are also defined in this chapter.

In Chapter IV, discretization of continuous state systems is considered. This is an important area of research for both hybrid systems and finite state control of continuous state systems. Discretization (or quantization) of continuous state
systems generates a nondeterministic discrete state system in general which we study in Section 4.1. The equivalent nondeterministic discrete state system is represented in the form of a deterministic discrete state system with a disturbance input. Discretized continuous state systems have also a special property that the next discretized states may only go to "neighboring" states. For this purpose neighbor-transition discrete state systems are defined in Section 4.2. Since the discretization is made by the help of an interface which discretizes and encodes (i.e. an A/D converter) the issue of resolution is important. The finer the resolution of this interface, the more states we will have in the automata model. Issues of stability, controllability and controller design will thus depend on this question of resolution. In Section 4.3 we address some issues related to having multiple resolution automata representations of systems.

In Chapter V, stability issues for discrete state systems and automata are considered. In Section 5.1, we study some of the properties of automata like reachability and stabilizability in the usual manner and provide theorems to test these properties using Boolean algebra and reachability matrices. An understanding of stability concepts in discrete state systems will be helpful in analysis of hybrid systems. In Section 5.2, we define classical stability for discrete state systems and provide theorems to check global asymptotic stability based on Lyapunov functions. In some cases classical stability is not adequate to directly apply to discrete state systems. Many concepts in classical stability are either lost or trivially satisfied for discrete
state systems. Thus we define and study a refined Lagrange stability concept to fill the gap in Section 5.3.

In Chapter VI, we study the hybrid system state space and consider various types of stability. We consider the continuous time model and try to understand the construction and behavior of hybrid systems. In Section 6.1 the continuous state space is divided into regions so that in every region depending on the discrete state of the hybrid system the continuous state system dynamics are established. Then the representative hybrid system is obtained from the original hybrid system. In Section 6.2, we define a classical stability concept of a continuous state in hybrid systems. Then using Lyapunov theory we provide some results on stability. A hybrid state stability concept which considers the stabilization of both the discrete and continuous states is defined and studied in Section 6.3. A special class of hybrid systems, such that the continuous state system portion is modeled as a linear discrete time system, called linear hybrid systems, is considered in Section 6.4. Some theorems for the global asymptotic stability of the origin of the continuous state space are provided by the help of a recent developed theory of stability of matrix sets.

In Chapter VII, control design issues for discrete state and hybrid systems are considered. In Section 7.1 the sliding mode control approach is developed for Discrete State Systems in a metric space. The utilization of sliding mode control is well known in continuous time systems [25]. The main feature of the standard sliding mode
control approach is the formulation as the problem of maintaining the system state on a prescribed manifold in the state space. A concept of trajectory continuity as introduced by neighbor-transition dynamic systems with finite state space in Section 4.2 allows the use of sliding mode design techniques similar to that in continuous state systems. We discuss the possibility of applying sliding mode control to discrete state systems and benefit from its nice properties. This approach is, then in Section 7.2, applied to a class of hybrid systems, which consist of a continuous plant, an interface and a discrete state controller. Interconnections of hybrid systems can also be modeled as hybrid systems in the framework considered in this thesis as given in Section 7.3. This allows us to use the same theory developed for hybrid systems for interconnected hybrid systems like serial or feedback connections. In Section 7.4 some control strategies are developed for hybrid systems based on Lyapunov functions. The Lyapunov theory for hybrid systems developed in Chapter VI is applied to design a feedback control for a desired type of stability.

In Chapter VIII, some examples of discrete state and hybrid systems are provided. First in Section 8.1, a robot in a two dimensional work space is modeled as a discrete state system. The control found stabilizes the robot as desired even if there is a strong disturbance. Second, we consider a double integrator and model it by a nondeterministic discrete state system by discretization in Section 8.2. Then using a discrete state controller we stabilize the system as desired. Then in Section 8.3, a robot with
a gripper is modeled as a hybrid system and some properties of hybrid systems are considered. A discrete state controller is designed to obtain a desirable behavior. In Section 8.4 we consider a continuous time hybrid system numerical example and study stability concepts. Finally in Section 8.5, a car is modeled as a hybrid system and a feedback is designed to achieve the desired behavior. The closed loop hybrid system is investigated for stability analysis.

In Appendix A, the concepts of asymptotic stability and stabilizability of a set of matrices are defined and investigated. The upper and lower spectral radius of a set of matrices are defined to aid in the analysis. Necessary and sufficient conditions for asymptotic stability and stabilizability are provided leading to some methods using Lyapunov theory and linear matrix inequalities. Finally in Section A.6 some problems from different areas of control are considered such that the theory of stability of matrix sets may be helpful in the analysis and design.

Finally in Chapter IX, we provide the conclusions and discuss possible research directions.
CHAPTER II

Dynamic Systems with Finite State Space

Let \( \mathcal{X}, \rho \) be a metric space, where \( \mathcal{X} \) is a finite set \( \rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+ \cup \{0\} \) is a metric, and \( T = \{0, 1, 2, \ldots\} \) is a set of time instants. The standard definition of a dynamic system on \( \mathcal{X} \) assumes that it is a set of transformations \( \{D(k, k_0, \cdot) \mid k_0 \leq k; k_0, k \in T\} \)

\[
D(k, k_0, \cdot) : \mathcal{X} \to \mathcal{X},
\]

which satisfies a semigroup property:

\[
D(k, k_1, D(k_1, k_0, x)) = D(k, k_0, x),
\]

and

\[
D(k, k, x) \equiv x
\]

for all \( x \in \mathcal{X} \) and \( k_0 \leq k_1 \leq k \); \( k_0, k_1, k \in T \).

A property of interest is the continuity of \( D(k, k_0, x_0) \) relative to a metric \( \rho \), which prevents jumps of trajectories. Since in our case the set \( \mathcal{X} \) is finite, every function
$D: \mathcal{X} \to \mathcal{X}'$ is continuous in the standard sense. In fact, for finite $\mathcal{X}$ there exists a nonzero minimal distance

$$\rho_{\text{min}} \triangleq \min_{x_1 \neq x_2} \rho(x_1, x_2).$$

(2.4)

Hence we can pick $0 < \delta < \rho_{\text{min}}$ to satisfy the inequality $\rho(D(x_1), D(x_2)) < \varepsilon$ for every given $\varepsilon > 0$, if $\rho(x_1, x_2) < \delta$, since the open ball of radius $\delta$ contains no other points of $\mathcal{X}$, except its center.

The possibility of the dynamic system trajectories to jump from an initial to any point in the state-space, but still to be considered as continuous contradicts intuition, but is correct, as seen above.

### 2.1 Automata

The dynamic systems with finite state space described in the previous section will represent the autonomous system which can be obtained by applying feedback to the system with input and output. Finite state automata are known to be a general form of such systems.

In this section we will discuss the specification of automata in the metric space which has continuous trajectories in accordance with the concept described above. We also introduce the adjacency matrix as a means of modeling the automata.
Definition 2.1 A finite state automaton is a seven-tuple

\[ AT = (X, \rho, U, Y, f, g, x_0) \]  

where \((X, \rho)\) is a metric space satisfying 2.4, \(U\) and \(Y\) denote the fixed and finite set of inputs, and outputs respectively. Transition and output functions are given as

\[ f : X \times U \rightarrow X \]  
\[ g : X \times U \rightarrow Y \]

where \(x^i \in X\), \(u^i \in U\), \(y^i \in Y\). \(x_0\) is the initial state.

The above model can be represented as a digraph.

Define the set of matrices with elements from the two-element set \(B = \{0, 1\}:\)

\[ A^{p \times s} \triangleq \{ A \in B^{p \times s} \mid \text{any column of } A \text{ has no more than a single } 1 \} \]

where \(p\) and \(s\) are positive integers. If \(s = 1\) we use \(A^p\) instead of \(A^{p \times 1}\).

Let \(X \in A^N\), where \(N\) is the number of states in \(X\). When \(X\) has only a single 1 at the \(i\)th entry, it represents \(x^i\), the \(i\)th state in \(X\). If \(X\) has no 1's, then it represents the case "no possible state." Furthermore, let \(y \in A^M\), where \(M\) is the number of outputs in \(Y\), and \(u \in A^R\), where \(R\) is the number of inputs in \(U\). Again a 1 in the vectors \(y\) and \(u\) represents the corresponding output and input, respectively. If \(y\) is a zero vector, then it represents the case "no possible output" whereas \(u\) is not allowed to be a zero vector.
**Definition 2.2** The Adjacency matrix $A(u)$ is defined as

$$[A(u)]_{ij} \triangleq \begin{cases} 
1 & \text{if } x^i = f(x^j, u), \\
0 & \text{otherwise}.
\end{cases}$$

The output matrix $C(u)$ is defined as

$$[C(u)]_{ij} \triangleq \begin{cases} 
1 & \text{if } y^i = g(x^j, u), \\
0 & \text{otherwise}.
\end{cases}$$

Note that $A(u) \in \mathbb{A}^{N \times N}$ and $C(u) \in \mathbb{A}^{M \times N}$. Let $x(i)$, $u(i)$, and $y(i)$ denote the $i$th state, input and output in time. Then

$$x(i + 1) = A(u(i)) x(i),$$

$$y(i) = C(u(i)) x(i)$$

represent the system 2.6 completely. The system output $y(i)$ can be written as

$$y(i) = C(u(i)) A(u(i - 1)) A(u(i - 2)) \cdots A(u(0)) x(0).$$

Here the standard definitions of addition and multiplication were used. Boolean algebra could also be used in the usual manner (see chapter 9 in [67]).

### 2.2 Discrete State Systems

In many cases, it is convenient to consider automata in the "vector" form, which we shall call a Discrete State System (DSS) representation.
Let \((X_i, \rho_i)\), for \(i = 1, 2, \ldots, n\), be a set of the finite metric spaces, with \(\rho_{\min}\) defined as in 2.4. Such \(X_i\) are analogous to one-dimensional subspaces.

**Definition 2.3** A Discrete State System is an automaton representation of the form:

\[
\begin{align*}
    x(k+1) &= F(x(k), u(k)), \\
    y(k) &= G(x(k), u(k)), \quad k = 0,1,2,\ldots
\end{align*}
\]  

(2.9)

where

\[
\begin{bmatrix}
    x_1(k) \\
    x_2(k) \\
    \vdots \\
    x_n(k)
\end{bmatrix},
\begin{bmatrix}
    u_1(k) \\
    u_2(k) \\
    \vdots \\
    u_r(k)
\end{bmatrix},
\begin{bmatrix}
    y_1(k) \\
    y_2(k) \\
    \vdots \\
    y_m(k)
\end{bmatrix}
\]

(2.10)

where \(x_i(\cdot), u_i(\cdot),\) and \(y_i(\cdot)\) take values from corresponding finite sets of states \(X_i\), inputs \(U_i\), and outputs \(Y_i\).

![Figure 2: Discrete State System block diagram.](image)

Note that, there are \(n\) state, \(r\) input, and, \(m\) output variables. The state space
of 2.9 is

\[ \mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_n. \] (2.11)

The system can be shown as a block diagram as in Figure 2.

Since the \( i \)th state variable can be written as

\[ x_i(k+1) = F_i(x_1(k), x_2(k), \ldots, x_i(k), \ldots, x_n(k), u(k)), \] (2.12)

we can consider it as a state of the individual automaton with other variables as the inputs to that automaton. So, DSS can be considered as a network of automata as shown in Figure 3.

![Figure 3: DSS, a network of automata.](image)

We assume that 2.12 is an automaton in the sense of the above definition with the state space \( \mathcal{X}_i \) and the set of inputs

\[ \mathcal{X}_1 \times \ldots \times \mathcal{X}_{i-1} \times \mathcal{X}_{i+1} \times \ldots \times \mathcal{X}_n \times \mathcal{U}_1 \times \ldots \times \mathcal{U}_m \] (2.13)
Since each $\mathcal{X}_i$ is a metric space with corresponding metric $\rho_i$, the metric $\rho$ on $\mathcal{X}$ can be introduced as

$$\rho(x, y) = \max\{\rho_i(x_i, y_i)\}. \quad (2.14)$$

From 2.14 it follows that the automata 2.9 also satisfies 2.4.

The way DSS's have been defined allows us introduce the concept dimensionality in the finite state space, therefore, we have the possibility to consider manifolds within the original space. The sets $\mathcal{X}_i$, by their structure, are similar to one-dimensional spaces ($\text{dim}(\mathcal{X}_i) = 1$), as the set $\mathcal{X}$ is $n$-dimensional ($\text{dim}(\mathcal{X}) = n$).

The DSS 2.9 can also be considered as a single automaton. This can be done by defining a single state variable $\hat{x}(k)$ of the single automaton to be obtained, which represents the vector $x(k)$. For this representation we need a mapping from a vector set to a single variable set. This mapping can be defined in many ways, one of them is to choose

$$\hat{x}(k) = p(x(k)) \triangleq x_1(k) + l_{x_1}x_2(k) + \cdots + \left(\prod_{i=1}^{n-1} l_{x_i}\right)x_n(k), \quad (2.15)$$

where $l_{x_i}$ denotes the number of states in $\mathcal{X}_i$.

Note that $\hat{x}(k)$ is a single variable whereas $x(k)$ is a vector of variables. The mapping 2.15 is one-to-one and onto, that is $p^{-1}$ exists, since we can recover $x_i(k)$ using

$$x_i(k) = p_i^{-1}(\hat{x}(k)) = \left(\hat{x}(k) \text{ div } \prod_{j=1}^{i-1} l_{x_j}\right) \text{ mod } l_{x_i}, \quad (2.16)$$
where \( \text{div} \) is integer division and \( \text{mod} \) is the modulo operator, and,

\[
x(k) = p^{-1}(\hat{x}(k)) = \begin{bmatrix} p_1^{-1}(\hat{x}(k)) \\ p_2^{-1}(\hat{x}(k)) \\ \vdots \\ p_n^{-1}(\hat{x}(k)) \end{bmatrix}.
\]  

(2.17)

Similar mappings for the input \( \hat{u} \) and the output \( \hat{y} \) of the single automaton can be defined as

\[
\hat{u}(k) = s(u(k)) \triangleq u_1(k) + l_{u1} u_2(k) + \cdots + \left( \prod_{i=1}^{r-1} l_{ui} \right) u_r(k),
\]

(2.18)

\[
\hat{y}(k) = v(y(k)) \triangleq y_1(k) + l_{y1} y_2(k) + \cdots + \left( \prod_{i=1}^{m-1} l_{yi} \right) y_m(k),
\]

(2.19)

where \( l_{ui} \) and \( l_{yi} \) denote the number of input and output values in \( U_i \) and \( Y_i \) respectively.

The mappings 2.18 and 2.19 are also one-to-one and onto, that is, \( s^{-1} \) and \( v^{-1} \) exist. Now, by considering the DSS equations 2.9 and the mappings 2.15, 2.18, and 2.19, we can obtain a single automaton as

\[
\hat{x}(k+1) = f(\hat{x}(k), \hat{u}(k)),
\]

\[
\hat{y}(k) = g(\hat{x}(k), \hat{u}(k)), \quad k = 0, 1, 2, \ldots
\]

(2.20)

where \( f \) and \( g \) are the projection of the functions \( F \) and \( G \) defined as

\[
f(\hat{x}(k), \hat{u}(k)) = p(F(p^{-1}(\hat{x}(k)), s^{-1}(\hat{u}(k)))),
\]

(2.21)

\[
g(\hat{x}(k), \hat{u}(k)) = v(G(p^{-1}(\hat{x}(k)), s^{-1}(\hat{u}(k)))),
\]

(2.22)
As we see 2.20 represents an automaton with \(\prod_{i=1}^{n} s_i\) states. By considering the definitions of \(A(u)\) and \(C(u)\), we can write

\[
\begin{align*}
\hat{x}(k + 1) &= A(\hat{u}(k))\hat{x}(k), \\
\hat{y}(k) &= C(\hat{u}(k))\hat{x}(k).
\end{align*}
\] (2.23)

Note that, \(\hat{x}(k)\) and \(\hat{y}(k)\) now represent a vector of zeros and ones as defined earlier.

One may wonder if there is any difference between the representational power of 2.9 and 2.20. Since \(F\) and \(G\) may take any value according to the vectors \(x\) and \(u\), the representation 2.9 is no weaker than representation 2.20. This means that the analysis of 2.9 is no simpler than 2.20, but the model 2.9 is more useful. If \(F\) and \(G\) do not depend on some of the variables in the vectors \(x\) and \(u\), on the other hand, the analysis of 2.9 may be simpler than 2.20.

The systems with a serial or a parallel interconnection or a feedback are easier to deal with when we use the DSS representation. When we use the automata representation, the connected systems are more difficult to deal with. So the use of state variables aids in the analysis.
CHAPTER III

Hybrid Systems

3.1 Models of Hybrid Systems

We construct hybrid systems as a combination of Discrete State Systems and Continuous State Systems. Let $X(t)$ and $x(t)$ denote the discrete and the continuous state variables in a hybrid system. Similarly let $U(t)$ and $u(t)$ be the discrete and the continuous inputs, and, $Y(t)$ and $y(t)$ be the discrete and the continuous outputs at the time instant $t$.

3.1.1 Discrete Time Model

In this model discrete state system part is modeled as a discrete time system whereas the continuous state system can be either discrete or continuous time system. If we use a continuous time configuration for continuous state system we have the equations as
\[ X(k + 1) = F(X(k), s(kT), U(k)), \] (3.1)
\[ \frac{d}{dt}x(t) = f(x(t), S(k), u(t)), \] (3.2)
\[ Y(k) = G(X(k), s(kT), U(k)), \] (3.3)
\[ y(t) = g(x(t), S(k), u(t)), \] (3.4)

where

\[ s(t) = q(x(t), S(k), u(t)), \] (3.5)
\[ S(k) = Q(X(k), s(kT), U(k)), \] (3.6)

\[ t \geq 0, \quad k = \lceil t/T \rceil, \quad T \in \mathbb{R}^+ \text{ is known.} \]

\[
X = \begin{bmatrix}
X_1 \\
\vdots \\
X_N
\end{bmatrix} \in \mathcal{X}, \quad
U = \begin{bmatrix}
U_1 \\
\vdots \\
U_R
\end{bmatrix} \in \mathcal{U}, \quad
Y = \begin{bmatrix}
Y_1 \\
\vdots \\
Y_M
\end{bmatrix} \in \mathcal{Y}, \quad
S = \begin{bmatrix}
S_1 \\
\vdots \\
S_H
\end{bmatrix} \in \mathcal{S},
\]

\[
X_I \in \{0,1,\ldots,l_{X_I}\}, \quad U_I \in \{0,1,\ldots,l_{U_I}\}, \quad Y_I \in \{0,1,\ldots,l_{Y_I}\}, \quad S_I \in \{0,1,\ldots,l_{S_I}\},
\]

\[
x = \begin{bmatrix}
x_1 \\
\vdots \\
x_n
\end{bmatrix} \in \mathbb{R}^n, \quad u = \begin{bmatrix}
u_1 \\
\vdots \\
u_r
\end{bmatrix} \in \mathbb{R}^r, \quad y = \begin{bmatrix}
y_1 \\
\vdots \\
y_m
\end{bmatrix} \in \mathbb{R}^m, \quad s = \begin{bmatrix}
s_1 \\
\vdots \\
s_h
\end{bmatrix} \in \mathbb{R}^h,
\]

\[
F: \mathcal{X} \times \mathcal{U} \times \mathbb{R}^h \to \mathcal{X}, \quad G: \mathcal{X} \times \mathcal{U} \times \mathbb{R}^h \to \mathcal{Y},
\]

\[
f: \mathbb{R}^n \times \mathbb{R}^r \times \mathcal{S} \to \mathbb{R}^n, \quad g: \mathbb{R}^n \times \mathbb{R}^r \times \mathcal{S} \to \mathbb{R}^m,
\]
\[ q : \mathbb{R}^n \times \mathbb{R}^r \times S \to \mathbb{R}^h, \quad Q : \mathcal{X} \times \mathcal{U} \times \mathbb{R}^h \to S. \]

\(X(0)\) and \(x(0)\) are initial conditions. The block diagram of a hybrid system is shown in Figure 4.

![Figure 4: Hybrid system block diagram.](image)

As we see from the model, the information inputs (outputs) \(S\) and \(s\) are used between continuous and discrete state systems. Continuous inputs to discrete state system are sampled with the sampling time \(T\), and, discrete inputs to continuous state system are held for \(T\) seconds.

First, we observe that \(s(t)\) does not need to have information about \(S(k)\) because \(S(k)\) is obtained from \(X(k)\) and \(U(k)\) which are known in the discrete state block. Thus \(q\) can only depend on \(x(t)\) and \(u(t)\). The same logic applies for \(S(k)\), in this case \(Q\) will only depend on \(X(k)\) and \(U(k)\). So we have

\[
q(t) = q(x(t), u(t)), \tag{3.7}
\]

\[
S(k) = Q(X(k), U(k)). \tag{3.8}
\]
Furthermore if the information between discrete and continuous blocks is not being affected by the hybrid system inputs directly then $S$ and $s$ only depend on the discrete and continuous states respectively. For this case we have a hybrid system model as

\begin{align}
X(k+1) &= F(X(k), x(kT), U(k)), \\
\frac{dx(t)}{dt} &= f(X(k), x(t), u(t)), \\
Y(k) &= G(X(k), x(kT), U(k)), \\
y(t) &= g(X(k), x(t), u(t)).
\end{align}

The configuration of a hybrid system for this case is shown in Figure 5.

![Figure 5: Hybrid system configuration.](image)

We can also model the continuous state system as a discrete time system. In this
The system equations are

\begin{align}
X(k+1) &= F(X(k), s(k), U(k)), \\
x(k+1) &= f(x(k), S(k), u(k)), \\
Y(k) &= G(X(k), s(k), U(k)), \\
y(k) &= g(x(k), S(k), u(k)), \\
s(k) &= q(x(k), u(k)), \\
S(k) &= Q(X(k), U(k)), \quad k = 0, 1, 2, \ldots.
\end{align}

### 3.1.2 Continuous Time Model

The second model is obtained by letting $T \to 0$ and simplifying the equations as shown below,

\begin{align}
X(t) &= F(X(t^-), x(t^-), U(t^-)), \\
\frac{d}{dt} x(t) &= f(X(t), x(t), u(t)), \\
Y(t) &= G(X(t^-), x(t^-), U(t^-)), \\
y(t) &= g(X(t), x(t), u(t)).
\end{align}

Then the whole state space of a hybrid system is

\[ \mathcal{X}_H = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_N \times \mathbb{R}^n = \mathcal{X} \times \mathbb{R}^n. \]
We assume that the continuous state dynamic function, \( f \), is piecewise continuous, and \( x(t) \) is absolutely continuous. Here we use Filippov's construction to find the equivalent dynamics for the discontinuity surfaces in the continuous state space [28, 69].

### 3.2 Some Properties of Hybrid Systems

In this section we define reachability, controllability and stabilizability for hybrid systems. These definitions are similar to the ones in classical system theory and automata theory.

**Definition 3.1** The set
\[
\begin{bmatrix}
\mathcal{X}^b \\
\mathcal{Y}^b
\end{bmatrix} \subseteq \begin{bmatrix}
\mathcal{X} \\
\mathcal{Y}
\end{bmatrix} \text{ is reachable from the set } \begin{bmatrix}
\mathcal{X}^a \\
\mathcal{Y}^a
\end{bmatrix} \subseteq \begin{bmatrix}
\mathcal{X} \\
\mathcal{Y}
\end{bmatrix}
\]

\[
\begin{bmatrix}
\mathcal{X} \\
\mathcal{Y}
\end{bmatrix}
\]
if \( \forall X(0) \in \mathcal{X}^a, x(0) \in \mathcal{Y}^a, \exists U(k), k = 0, 1, \ldots, K; u(t), t \in [0, KT]; \) such that \( X(K) \in \mathcal{X}^b \) and \( x(KT) \in \mathcal{Y}^b \).

**Definition 3.2** The set
\[
\begin{bmatrix}
\mathcal{X}^a \\
\mathcal{Y}^a
\end{bmatrix} \subseteq \begin{bmatrix}
\mathcal{X} \\
\mathcal{Y}
\end{bmatrix}
\]
is 1-step returnable if when \( X(0) \in \mathcal{X}^a \) and \( x(0) \in \mathcal{Y}^a, \exists U(0) \in \mathcal{U} \) and \( u(t), t \in [0, T] \); such that \( X(1) \in \mathcal{X}^a \) and \( x(t) \in \mathcal{Y}^a, t \in [0, T] \).
Definition 3.3 The set \[ \begin{bmatrix} \mathcal{X}^a \\ \Omega^a \end{bmatrix} \subseteq \begin{bmatrix} \mathcal{X} \\ \mathbb{R}^n \end{bmatrix} \] is stabilizable if it is reachable from \[ \begin{bmatrix} \mathcal{X}^0 \\ \Omega^0 \end{bmatrix} \] and 1-step returnable. The sets \( \mathcal{X}^0 \) and \( \Omega^0 \) consist of the possible initial conditions.
CHAPTER IV

Discretization of Continuous State Systems

4.1 Discretization and Nondeterministic Discrete State Systems

Consider a continuous state system (plant) and an interface as shown in Figure 6.

![Figure 6: Discretization of a Continuous State System.](image)

The plant is represented by the equations

\[ \dot{x} = f(x, u) \]
\[ y = g(x, u) \]  (4.1)

where \( x = [x_1 \ldots x_n]' \in \mathbb{R}^n \), \( u = [u_1 \ldots u_r]' \in \mathbb{R}^r \), \( y = [y_1 \ldots y_m]' \in \mathbb{R}^m \).
The interface is a model of the discrete sensor which provides the information on the system state. It is assumed that the state space of the continuous system is partitioned into regions, usually forming a grid which is not necessarily uniform. The interface generates the signals uniquely associated with each of these regions.

If each continuous state variable is discretized (or quantized) individually (or separately), we will get a partition in the form of a grid. Such a partition of the state space is shown, for the two dimensional case, in Figure 7.

More formally, for each continuous state variable $x_i$, $i \in \{1, \ldots, n\}$ let there be a finite set of points $l_i(j)$, $j = j_{\text{min}}; j_{\text{min}} + 1, \ldots, j_{\text{max}}$, such that $l_i(j) < l_i(j + 1)$. 

![Figure 7: The Partition of the State Space.](image)
We denote

\[ X_i = j \text{ iff } x_i \in [l_i(j), l_i(j + 1)), \quad i \in \{1, \ldots, n\}, \ j \in \{j_{\text{min}}^i, \ldots, j_{\text{max}}^i - 1\}, \]

and,

\[ X_i = j_{\text{min}}^i - 1 \text{ iff } x_i < l_i(j_{\text{min}}^i), \]

\[ X_i = j_{\text{max}}^i \text{ iff } x_i \geq l_i(j_{\text{max}}^i). \]

The variable \( X = [X_1 \ldots X_n]' \) takes values in a finite set \( \mathcal{X} \), which can be encoded by any finite set with the same cardinality.

By assumption there is only a finite number of values of the control \( u \), these can also be encoded similarly by a variable \( U \) taking values from a finite set \( \mathcal{U} \).

The plant+interface model form a nondeterministic automaton called \textit{Nondeterministic Discrete State System} (NDSS) given by

\[ X(t_{k+1}) \in F(X(t_k), U(t_k)) \tag{4.2} \]

where \( F(X(t_k), U(t_k)) \) is a set of possible states that can be reached if the input \( U(t_k) \) is applied on the time interval \( t \in [t_k, t_{k+1}) \) when the plant state is in the region corresponding to \( X(t_k) \). The time \( t_k \) is defined as the first instant the state \( x \) enters the region corresponding to \( X(t_k) \).

Relation 4.2 represents a general type of a nondeterministic discrete state system. Let \( \omega(t_k) \) parametrically represent the uncertainties in the system due to the nondeterministic behavior of the corresponding discrete state system. Then 4.2 can be
considered in the form

\[ X(t_{k+1}) = F(X(t_k), U(t_k), \omega(t_k)), \]

(4.3)

\( \omega(t_k) \in \Omega(t_k) \), where \( \Omega(t_k) \) is a set of possible values of the parameter. In fact, the system behavior within each region of the partition is defined by the location of the entering point, therefore, \( \Omega(t_k) \) can be understood as a boundary set of the region.

For an arbitrary partitioning of the state space above (for a design of the interface) we obtain a nondeterministic discrete state system as in 4.3. To obtain a deterministic automata out of a continuous state system we need to choose the partitioning carefully, see [75, 47] for details on this subject.

4.2 Neighbor-Transition Dynamic Systems with Finite State Space

The discrete state systems which are obtained by discretization of continuous state systems have a strong continuity property. That is, the next discrete states are only allowed to be “neighboring” states. This forms a special class of discrete state systems (or automata) which we will call neighbor-transition discrete state systems.

A Neighbor-Transition Discrete State System (NTDSS) is defined as

\[ X(k + 1) = X(k) + F(X(k), U(k)), \]

\[ Y(k) = G(X(k), U(k)), \quad k = 0, 1, 2, \ldots \]

(4.4)
where \( F_i(\cdot, \cdot) \in \{-1, 0, 1\} \), and,

\[
X(k) = \begin{bmatrix}
X_1(k) \\
\vdots \\
X_n(k)
\end{bmatrix} \in \mathcal{X}, \quad U(k) = \begin{bmatrix}
U_1(k) \\
\vdots \\
U_r(k)
\end{bmatrix} \in \mathcal{U}, \quad Y(k) = \begin{bmatrix}
Y_1(k) \\
\vdots \\
Y_m(k)
\end{bmatrix} \in \mathcal{Y}, \quad (4.5)
\]

where

\[
X_i(\cdot) \in \mathcal{X}_i \triangleq \{-l_{X_i}, \ldots, 0, \ldots, l_{X_i}\}, \quad (4.6)
\]

\[
U_i(\cdot) \in \mathcal{U}_i \triangleq \{-l_{U_i}, \ldots, 0, \ldots, l_{U_i}\}, \quad (4.7)
\]

\[
Y_i(\cdot) \in \mathcal{Y}_i \triangleq \{-l_{Y_i}, \ldots, 0, \ldots, l_{Y_i}\}. \quad (4.8)
\]

As we see the next state in the discrete state representation above can only be in
the neighboring states as in continuous state systems. An example of a neighbor-
transition discrete state system is shown in Figure 8. This example is obtained from
quantizing a two dimensional linear stable continuous state system.

As discussed in Chapter II, every trajectory is continuous in the standard sense
in discrete state systems. We may need a stronger continuity property for finite
state systems. For example finite state systems where the state space is encoded
by discretizing the state space of a continuous system as above, preserve a stronger
continuity property. A special class of dynamic systems with finite state space can
also be defined by using a metric. In this case the next states are allowed only to the
nearest states and the concept of distance is provided by the metric considered.
**Definition 4.1** A Neighbor-Transition Dynamic System with Finite State Space is defined as a two parametric set of transformations \( \{D(k, k_0, \cdot) : k_0 \leq k \} \) (2.1), of the finite metric space \( \{X, \rho \} \), satisfying 2.2 and 2.3 such that, if \( x = D(k + 1, k, x_0) \), then the following two conditions hold:

A. \( \rho(y, x_0) \geq \rho(x, x_0), \forall y \in X - \{x_0\} \).

B. \( \rho(y, x) \geq \rho(x, x_0), \forall y \in X - \{x\} \).

Condition A implies that transitions occurring in unit time, from any point, are allowed only to the nearest points. It is natural to call such systems continuous from the right. In this case we can guarantee that the distances between points on the trajectory are nonincreasing.

Condition B, on the other hand, implies that the transitions, resulting at any point could also have come only from the nearest points. Such systems can be called continuous from the left.

When both A and B are satisfied, the distances between the sequential points on any trajectory \( T = \{x(0), x(1), \ldots\} \), where

\[
x(k + 1) = D(k + 1, k, x(k))
\]

are less than or equal to \( \rho_{\text{min}} \) and there are no other points of \( X \) closer than this distance to \( T \).

The definition introduced not only provides a formal basis for the intuitive notion
of continuity in the finite state case, but also permits introduction of a nontrivial concept of stability.

4.3 Multiresolutional Discrete State Systems

Suppose that we obtain a discrete state system as in 2.9 by discretizing a continuous state system as above. Then the state resolution of the discrete state system can be defined as the number of states in \( \mathcal{X} \). One can also define the input and output resolution of a system. In this discussion we shall only consider state resolution and refer to it simply as resolution.

To increase the resolution the standard approach is to return to the continuous state system and choose a finer partitioning. Resolution can also be increased directly, resulting in a nonunique representation. Such an exercise may be useful for fault tolerant system design. To reduce the resolution, however, we may use the high resolution discrete state system. In this section we provide some mechanism to achieve this goal.

Consider a high resolution discrete state system

\[
X(k + 1) = F(X(k), U(k)),
\]

where \( X = [X_1 \cdots X_n]' \), \( X_i \in \mathcal{X}_i \), \( X \in \mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \), and, \( U \in \mathcal{U} \).

Let \( \tilde{\mathcal{X}} = \tilde{\mathcal{X}}_1 \times \cdots \times \tilde{\mathcal{X}}_n \) denote the new low resolution discrete state space. Let \( \mathcal{R}_i(m) \) represent the set of the corresponding old discrete state values of \( X_i \) for the
new discrete state value \( m \) of \( \tilde{X}_i \). For example, consider the discrete state partition in Figure 7. We may want to reduce the resolution for the state variable \( X_i \). For \( X_i = -1 \) and \(-2\) we may use \( \tilde{X}_i = -1 \) and for \( X_i = 1 \) and \( 2 \) we may use \( \tilde{X}_i = 1 \). Then we have \( \tilde{X}_i = \{-1, 0, 1\}, \mathcal{R}_i(-1) = \{-2, -1\}, \mathcal{R}_i(0) = \{0\}, \) and, \( \mathcal{R}_i(1) = \{1, 2\} \).

The new discrete state system will be a nondeterministic discrete state system as

\[
\tilde{X}(k+1) \in \tilde{F}(\tilde{X}(k), U(k)),
\]

where \( \tilde{F} \) can be obtained as

\[
\tilde{F}(\begin{bmatrix} \tilde{X}_1 \\ \vdots \\ \tilde{X}_n \end{bmatrix}, U) = \{ \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} \mid F(\begin{bmatrix} \tilde{X}_1 \\ \vdots \\ \tilde{X}_n \end{bmatrix}, U) \in \begin{bmatrix} \mathcal{R}_1(\tilde{X}_1) \\ \vdots \\ \mathcal{R}_n(\tilde{X}_n) \end{bmatrix}, \forall X_i \in \mathcal{R}_i(\tilde{X}_i) \}. \quad (4.12)
\]

We call the resolution deterministic if the reduced function \( \tilde{F} \) above has a single element, that is we obtain a deterministic low resolution discrete state system. Define the inverse function of \( \mathcal{R}_i(\cdot) \) as

\[
\mathcal{R}_i^{-1}(X_i) = \{ \tilde{X}_i \mid X_i \in \mathcal{R}_i(\tilde{X}_i) \}. \quad (4.13)
\]

We provide a condition below to obtain a deterministic resolution:

\[
\mathcal{R}_i^{-1}(F_i(X, U)) \cap \mathcal{R}_i^{-1}(F_i(Z, U)) \neq \emptyset \quad (4.14)
\]

\[
\forall i \in [1 : n], \quad \forall U \in \mathcal{U}, \quad \forall X, Z \in \begin{bmatrix} \mathcal{R}_i(q_i) \\ \vdots \\ \mathcal{R}_n(q_n) \end{bmatrix}, \text{ for any given } q_j \in \tilde{X}_j, j \in [1 : n].
\]
CHAPTER V

Stability of Discrete State Systems and Automata

5.1 Reachability and Stabilizability in Automata

In this section the definitions of reachability and stabilizability in automata are introduced and some theorems related to the definitions are considered. More details can be found in [16, 17]. In this section we use Boolean addition and multiplication operators.

Definition 5.1 A state $x^b \in \mathcal{X}$ is reachable from $x^a \in \mathcal{X}$ if for $x_0 = x^a$, there is an input string $\{u_0^b, u_1^b, \ldots, u_{k-1}^b\}$, where $k$ is finite, so that $x_k = x^b$.

Definition 5.2 A state $x^b \in \mathcal{X}$ is reachable if $x^b$ is reachable from any state $x^a \in \mathcal{X}$.

Definition 5.3 For any $\mathcal{X}^A \subseteq \mathcal{X}$ let $\overline{\mathcal{X}^A} \triangleq \mathcal{X} - \mathcal{X}^A$.

Definition 5.4 A set of states $\mathcal{X}^B \subseteq \mathcal{X}$ is reachable if for any state $x^a \in \overline{\mathcal{X}^B}$, there is an $x^b \in \mathcal{X}^B$ such that $x^b$ is reachable from $x^a$. 
**Definition 5.5** A set of states $\mathcal{X}^A \subseteq \mathcal{X}$ is 1-step returnable if for any state $x_0 = x^a \in \mathcal{X}^A$ and for any given input there exists an input $u_0^b$ such that $x_1 = x^b \in \mathcal{X}^A$.

**Definition 5.6** A set of states $\mathcal{X}^A \subseteq \mathcal{X}$ is stabilizable if $\mathcal{X}^A$ is reachable and 1-step returnable.

The above definitions are similar to those given in the literature; for example, see [56]. Let $n$ denote the dimension of the adjacency matrix $A(\cdot)$, and $\oplus$ denote the Boolean addition operation. We can define the reachability matrix $P$ as is done in graph theory (see [12, 14]),

$$P = A \oplus A^2 \oplus \cdots \oplus A^n,$$

(5.1)

where $A \overset{\Delta}{=} \sum_{i=1}^n A(u^i)$. The reachability matrix has a property that $x^i$ is reachable from $x^j$ iff $P(i,j) = 1$.

Denote $i[\cdot]$ as the $i$th row of $[\cdot]$, and $[\cdot]_j$ as the $j$th column of $[\cdot]$. We call $i[\cdot]$ ($[\cdot]_j$) "full" if all the entries of $i[\cdot]$ ($[\cdot]_j$) are 1. Then it can be shown that $x^i \in \mathcal{X}$ is reachable if and only if $i[P]$ is full. The similar results are available from graph theory, for example see [12, 14].

**Lemma 5.1** $\mathcal{X}^A \subseteq \mathcal{X}$ is reachable iff $\sum_{x^i \in \mathcal{X}^A} i[P]$ is full for the entries corresponding to the elements in $\overline{\mathcal{X}^A}$. This condition can also be stated as

$$\prod_{x^i \in \overline{\mathcal{X}^A}} \left[ \sum_{x^i \in \mathcal{X}^A} i[P] \right]_j = 1 \quad \text{if } \mathcal{X}^A \text{ is not empty}.$$
Proof: If $\mathcal{X}^A$ is reachable then from any state $x^i \in \overline{\mathcal{X}}^A$ at least one of the states in $\mathcal{X}^A$ is reachable. Hence $P(i,j) = 1$ for $x^i \in \mathcal{X}^A$. It may be concluded that $\sum_{x^i \in \mathcal{X}^A} i[P]$ is full for the entries corresponding to the elements in $\overline{\mathcal{X}}^A$.

If $\sum_{x^i \in \mathcal{X}^A} i[P]$ is full for the entries corresponding to the elements in $\overline{\mathcal{X}}^A$ then at least one of the states in $\mathcal{X}^A$ is reachable from any state $x^i \in \overline{\mathcal{X}}^A$, so $\mathcal{X}^A$ is reachable. If $\overline{\mathcal{X}}^A$ is empty then the proof is done by definition, that is, $\mathcal{X}$ is always reachable by definition.

□

Definition 5.7 $x^s \in \mathcal{X}$ is a terminal state if no input can be applied at this state.

Definition 5.8 The live part of a set $\mathcal{X}^A$ is the set $\mathcal{X}^{A+}$ such that

$$\mathcal{X}^{A+} = \mathcal{X}^A - \mathcal{X}^A^*$$

(5.2)

where $\mathcal{X}^{A^*}$ is the set of all terminal states in $\mathcal{X}^A$.

The above definitions of a “terminal state” (or “dead state”) and the “live part” of a set can be found in the literature in detail (see [56, 63]).

Lemma 5.2 $\mathcal{X}^A$ is 1-step returnable iff $\sum_{x^i \in \mathcal{X}^{A+}} i[A]$ is full for the entries corresponding to the elements in $\mathcal{X}^{A+}$. An equivalent condition is

$$\prod_{x^j \in \mathcal{X}^{A+}} \left[ \sum_{x^j \in \mathcal{X}^{A+}} i[A] \right]_j = 1$$

if $\mathcal{X}^{A+}$ is not empty.
Proof: The $i$th row of $A$ shows the states from which we can reach $x^i$ by a one-length input. If $\mathcal{X}^A$ is 1-step returnable, the entries corresponding the elements in $\mathcal{X}^{A+}$ of $\sum_{x^i \in \mathcal{X}^{A+}} i[\mathcal{I}]$ must be 1s so that for all elements in $\mathcal{X}^{A+}$ there is a state in $\mathcal{X}^A$ reachable by a one-length input. If $\mathcal{X}^{A+}$ is empty then the proof is done by definition. The contrary part of the proof is also clear. \qed

Theorem 5.1 $\mathcal{X}^A$ is stabilizable iff $\sum_{x^i \in \mathcal{X}^A} i[P]$ is full for the entries corresponding to the elements in $\mathcal{X}^A$ and $\sum_{x^i \in \mathcal{X}^{A+}} i[A]$ is full for the entries corresponding to the elements in $\mathcal{X}^{A+}$. Or we can state this condition as

$$\left( \prod_{x^i \in \mathcal{X}^A} \left[ \sum_{x^j \in \mathcal{X}^A} i[P] \right] \right) \left( \prod_{x^i \in \mathcal{X}^{A+}} \left[ \sum_{x^j \in \mathcal{X}^{A+}} i[A] \right] \right) = 1.$$

Proof: Since $\mathcal{X}^A$ is stabilizable iff $\mathcal{X}^A$ is reachable and 1-step returnable, both the conditions for reachability and 1-step returnability are used. \qed

5.2 Stability of Discrete State Systems

Let $x(k)$ denote the state of the system at time step $i$ and a trajectory be a sequence of states $T = \{x(0), x(1), \ldots\}$. Following classical Lyapunov stability theory, we introduce the notion of stability for a trajectory and an invariant set of a dynamic system with finite state space. For the discrete event system case on this subject see [57, 59].
Define the distance from a state to a set of states as

$$\rho(x, \mathcal{M}) \triangleq \min_{y \in \mathcal{M}} \rho(x, y).$$  \hspace{1cm} (5.3)

We define the $\delta$-vicinity of a set $\mathcal{M} \subset \mathcal{X}$ (or of a state $\tilde{x} \in \mathcal{X}$) as the set of states satisfying $\rho(x, \mathcal{M}) \leq \delta$ (or $\rho(x, \tilde{x}) \leq \delta$). Below we give the classical definitions of Lyapunov stability for a trajectory and an invariant set.

**Definition 5.9** The trajectory $\tilde{T} = \{\tilde{x}(0), \tilde{x}(1), \ldots\}$ of $\mathcal{D}$, is called stable if for any $\varepsilon > 0$ there exists $\delta > 0$, such that if any other trajectory $T = \{x(0), x(1), \ldots\}$ starting in its $\delta$-vicinity ($\rho(x(0), \tilde{x}(0)) \leq \delta$) remains in the $\varepsilon$-vicinity for all $k \geq 0$, i.e. $\rho(x(k), \tilde{x}(k)) \leq \varepsilon$. Furthermore if we have $\rho(x(k), \tilde{x}(k)) \to 0$ as $k \to \infty$ then $\tilde{T}$ is called asymptotically stable.

**Definition 5.10** An invariant set $\mathcal{M} \subset \mathcal{X}$ is stable if for any $\varepsilon > 0$ there exists $\delta > 0$, such that if $\rho(x(0), \mathcal{M}) < \delta$ then $\rho(x(k), \mathcal{M}) < \varepsilon$ for all $k \geq 0$. If additionally we have $\rho(x(k), \mathcal{M}) \to 0$ as $k \to \infty$ then $\mathcal{M}$ is called asymptotically stable.

**Definition 5.11** The domain of attraction of an asymptotically stable set $\mathcal{M} \subset \mathcal{X}$ is given as

$$\Omega_*(\mathcal{M}) \triangleq \{x(0) \in \mathcal{X} \mid \rho(x(k), \mathcal{M}) \to 0 \text{ as } k \to \infty\}. \hspace{1cm} (5.4)$$

Since we are dealing with finite state space systems, $\Omega_*(\mathcal{M})$ only consists of the states having a trajectory to $\mathcal{M}$.
Definition 5.12 An invariant set \( M \subset X \) is globally asymptotically stable (or asymptotically stable in the large) if \( M \) is stable and \( \Omega_*(M) = X \).

The following can be observed easily for dynamic systems with finite state space.

Lemma 5.3 Any invariant set \( M \subset X \) or any trajectory \( \tilde{T} = \{ \tilde{x}(0), \tilde{x}(1), \ldots \} \) is stable and also asymptotically stable in a dynamic system with finite state space.

Proof: As mentioned earlier, for every \( \epsilon > 0 \) we can pick \( 0 < \delta < \rho_{\text{min}} \) such that when \( \rho(x(0), M) < \delta \) (or \( \rho(x(0), \tilde{x}(0)) < \delta \)), \( x(0) \) can only be chosen from the set \( M \) (or \( \tilde{x}(0) \)) since there cannot be any other state with a distance to \( M \) (or \( \tilde{x}(0) \)) smaller than \( \rho_{\text{min}} \). Then \( \rho(x(k), M) \) becomes 0 at all times since \( M \) is an invariant set. This proves the stability and the asymptotic stability of \( M \) (or \( \tilde{T} \)).

The above lemma states that the stability and the asymptotic stability of a trajectory or an invariant set in a dynamic system with finite state space are trivially satisfied. Global asymptotic stability on the other hand is not trivial for which in the literature some theorems based on Lyapunov theory are provided, see [57, 59]. Here we give a modified Lyapunov theorem below.

Theorem 5.2 An invariant set \( M \subset X \) is globally asymptotically stable (or asymptotically stable in the large) iff there exists a scalar (or vector) function \( V(x) : X \to R \) (or \( R^n \)) which satisfies the Lyapunov conditions
(i) \( V(x) = 0 \) if \( x \in \mathcal{M} \),

(ii) \( V(x(k)) \to 0 \) as \( k \to \infty \).

**Proof:** \( (\Leftarrow) \) Let us suppose we have a function \( V(x(k)) \) which satisfies the two conditions given above. According to the Lemma 5.3 we know that \( \mathcal{M} \) is stable. Then we need to prove that from any state \( x(0) \in \mathcal{X} \) there is a trajectory to \( \mathcal{M} \). If initially \( x(0) \in \mathcal{M} \) then since \( \mathcal{M} \) is invariant we have \( x(k) \in \mathcal{M} \) or \( V(x(k)) = 0 \) for all \( k \geq 0 \). If \( x(0) \notin \mathcal{M} \) then initially \( V(x(0)) = V_0 \neq 0 \). According to condition (ii) we have \( V(x(k)) \to 0 \) as \( k \to \infty \). Thus we are sure that \( V(x(k)) \) will go to 0. Let us assume that \( V(x(k)) \to 0 \) does not mean \( x(k) \to \mathcal{M} \), that is, it takes infinite time to reach \( \mathcal{M} \). Since \( V(x(k)) \to 0 \), \( V(x(k)) \) can be arbitrarily small after a finite time. However there is a minimum value of \( \| V(x) \| \), \( V_{\min} = \min_{x \in \mathcal{X} - \mathcal{M}} \| V(x) \| \). Thus after a finite time \( k_* \) we have \( V(x(k)) = 0 \) which means \( x(k) \in \mathcal{M} \) for \( k \geq k_* \) according to the condition (i). This proves that there is a path from \( x(0) \) to \( \mathcal{M} \).

\( (\Rightarrow) \) Let us suppose the invariant set \( \mathcal{M} \subset \mathcal{X} \) is globally asymptotically stable then from any state \( x(0) \in \mathcal{X} \) there exists a trajectory to \( \mathcal{M} \). Let us choose \( V(x(k)) = \rho(x(k), \mathcal{M}) \). Then \( V(x) = 0 \) if \( x \in \mathcal{M} \) according to the definition of the metric. Also if \( x \notin \mathcal{M} \) then \( V(x) \neq 0 \). Thus the condition (i) is satisfied. Since there is a path from \( x(0) \in \mathcal{X} \) to \( \mathcal{M} \), in finite time, we obtain \( \rho(x(k_*), \mathcal{M}) = 0 \). Since \( \mathcal{M} \) is an invariant set we get \( V(x(k)) = \rho(x(k), \mathcal{M}) = 0 \) for \( k \geq k_* \). Thus the condition (ii) is also satisfied. \( \square \)
As we see in the Lyapunov global stability theorem given above we do not require $V(x(k))$ to be a non-increasing function. We only need $V(x(k)) \to 0$ as seen in the proof. That is because the stability and the asymptotic stability of an invariant set are trivially satisfied in dynamic systems with finite state space. However it may be convenient to give the theorem below (also see [57, 59]).

**Theorem 5.3** $V(x)$ satisfies the Lyapunov conditions in Theorem 5.2 if

(i) $V(x) = 0$ for $x \in \mathcal{M}$,

(ii) $V(x) > 0$ for $x \not\in \mathcal{M}$

(iii) $V(x(k+1)) - V(x(k)) < 0$ for $x(k) \not\in \mathcal{M}$.

**Proof:** Conditions (i) and (ii) requires that $V(x)$ is a positive definite function. Thus the condition (i) of Theorem 5.2 is satisfied. Condition (iii) requires that $V(x(k))$ is a decreasing function of $k$. Thus $V(x(k))$ must go to a limit $l \geq 0$. Let us assume $l \neq 0$. Then $x(k) \not\in \mathcal{M}$ and $V(x(k)) \geq l$. Thus there is a positive number $\Delta_l$ such that $V(x(k+1)) - V(x(k)) < -\Delta_l$ for all $x(k)$ which satisfies $V(x(k)) \geq l$ according to the condition (iii). However this means that $l$ cannot be a limit of $V(x(k))$ since $V(x(k))$ must be strictly decreasing. Thus we conclude that $l = 0$ which means $V(x(k)) \to 0$ as $k \to \infty$. This satisfies the condition (ii) of Theorem 5.2. □
To illustrate the above theorem consider the situation shown in Figure 8. As we see the Lyapunov function takes on lower values on a state trajectory of the system. Thus we conclude that the origin is globally asymptotically stable.

5.3 Refined Lagrange Stability Analysis

In classical Lagrange stability analysis boundedness issues are considered. Lagrange stability concepts can be directly applied to discrete state systems with infinite number of states. However in finite state systems Lagrange stability concepts do not
provide any particular insight since every trajectory is bounded by definition. We need to obtain information on how much a trajectory will diverge from an equilibrium point once it originates nearby. This type of stability is called *practical stability* in the literature (see [66, 50, 51, 60]). This information is more valuable than knowing if a trajectory diverges or not, which is answered by classical Lagrange stability. In this section stronger stability definitions for finite state systems are provided.

For any set $Y \subset X$ consider the number $\bar{p}(Y)$, which is defined as

$$\bar{p}(Y) = \min_{x \in X - Y, y \in Y} \rho(x, y).$$  \hspace{1cm} (5.5)

$\bar{p}(Y)$ characterizes the minimal distance of the set $Y$ to the set $X - Y$. Without loss of generality we can assume that $\bar{p}(Y) = \rho_{\text{min}}$ for all $Y \subset X$. Now let us first define the instability margin of a trajectory and an invariant set (see also [66, 50, 51, 60]).

**Definition 5.13** The instability margin of a trajectory $\bar{T} = \{\bar{x}(0), \bar{x}(1), \ldots\}$, $\psi_{\bar{T}}$, is given as

$$\psi_{\bar{T}} \triangleq \max\{\rho(\bar{x}(k), x(k)) \mid \rho(\bar{x}(0), x(0)) \leq \rho_{\text{min}}, k > 0\}. \hspace{1cm} (5.6)$$

Similarly the instability margin of a set $\mathcal{M} \subset X$, $\psi_{\mathcal{M}}$, is given as

$$\psi_{\mathcal{M}} \triangleq \max\{\rho(x(k), \mathcal{M}) \mid \rho(x(0), \mathcal{M}) \leq \rho_{\text{min}}, k > 0\}. \hspace{1cm} (5.7)$$

Below we give the refined Lagrange stability definitions which are closely related to *practical stability* in the literature [66, 50, 51, 60].
Definition 5.14 The trajectory $\tilde{T} = \{\tilde{x}(0), \tilde{x}(1), \ldots\}$ of $D$, is called $\lambda$-stable if any other trajectory $T = \{x(0), x(1), \ldots\}$ starting in the minimal vicinity of $\tilde{x}(0)$ ($\rho(x(0), \tilde{x}(0)) \leq \rho_{\text{min}}$) remains in the $\lambda$-vicinity for all $k > 0$, i.e. $\rho(x(k), \tilde{x}(k)) \leq \lambda$. Furthermore if additionally we have $\rho(x(k), \tilde{x}(k)) \rightarrow 0$ as $k \rightarrow \infty$ then $\tilde{T}$ is called asymptotically $\lambda$-stable.

Definition 5.15 A set $M \subset X$ is $\lambda$-stable if $\rho(x(0), M) \leq \rho_{\text{min}}$ then $\rho(x(k), M) \leq \lambda$ for all $k > 0$. If additionally we have $\rho(x(k), M) \rightarrow 0$ as $k \rightarrow \infty$ then $M$ is called asymptotically $\lambda$-stable.

As observed the refined Lagrange stability definitions above result in stronger requirements than classical stability definitions since we are forced to choose the initial state in not only any nonzero vicinity, but a $\rho_{\text{min}}$-vicinity (the minimal vicinity). This helps analyze the effects of a possible disturbance or noise occurring around an initial condition (or an invariant set) on the system behavior. The instability margin gives a measure on how much a trajectory (or a set of states) is unstable. For example $\psi_M = 0$ shows the strictest stability property since the next state must be in $M$ right away if $x(0)$ is in the minimal vicinity. If $\psi_M = \rho_{\text{min}}$ then we are sure that the next state will not go further away from the minimal vicinity once started there. The higher $\psi_M$ the worse the stability of $M$.

From the refined Lagrange stability definitions we see that a set $M$ (or a trajectory $\tilde{T}$) is $\lambda$-stable if and only if $\lambda \geq \psi_M$ (or $\lambda \geq \psi_T$). Thus to check the refined Lagrange
stability we need an upper bound for $\psi_M$ as provided in the theorem below.

**Theorem 5.4** Let a Lyapunov function $V(x): \mathcal{X} \rightarrow R^+ \cup \{0\}$ for a set $\mathcal{M} \subset \mathcal{X}$ satisfy the conditions

(i) $V(x) = 0$ iff $x \in \mathcal{M}$,

(ii) $V(x(k + 1)) - V(x(k)) \leq 0$ for $x(k) \notin \mathcal{M}$.

$\psi_X^Y$ is an upper bound for $\psi_M$ ($\psi_M \leq \psi_X^Y$) where

$$\psi_X^Y \triangleq \max \{ \rho(x, \mathcal{M}) \mid V(x) \leq \max \{ V(y) \mid \rho(y, \mathcal{M}) \leq \rho_{\min}, y \in \mathcal{X}\}, x \in \mathcal{X} \}, \quad (5.8)$$

and $\mathcal{M}$ is $\psi_X^Y$-stable. Furthermore if $V(x)$ satisfies

(ii*) $V(x(k + 1)) - V(x(k)) < 0$ for $\rho_{\min} \leq \rho(x(k), \mathcal{M}) \leq \psi_M$

then $\mathcal{M}$ is asymptotically $\psi_X^Y$-stable.

**Proof:** Consider the values of the function $V(x)$ in the minimal vicinity of $\mathcal{M}$. The maximum value can be calculated as $V_m = \max \{ V(y) \mid \rho(y, \mathcal{M}) \leq \rho_{\min}, y \in \mathcal{X} \}$. Because of the condition (ii) we know that $V(x)$ is non increasing. Then, since the initial condition starts in the minimal vicinity, for all the next states we have $V(x) \leq V_m$. Thus the maximum distance the state trajectory can reach will be in these states. Therefore we come up with an upper limit as given in 5.8. Since $\psi_X^Y \geq \psi_M$ we say that $\mathcal{M}$ is $\psi_X^Y$-stable. Furthermore if we have the condition (ii*) then we are sure that $V(x)$ is strictly decreasing. And as discussed in the proofs of
the classical stability theorems, after a finite \( k \) we have \( V(x(k)) = 0 \) thus \( x(k) \in \mathcal{M} \).

Therefore \( \mathcal{M} \) is asymptotically \( \psi^V_{\mathcal{M}} \)-stable.

Above we give the conditions for \( \lambda \)-stability by means of a Lyapunov function for a general \( \lambda \). Here we give a corollary for \( \rho_{\min} \)-stability.

**Corollary 5.1** A set \( \mathcal{M} \subset \mathcal{X} \) is \( \rho_{\min} \)-stable iff for all \( x \) in the minimal vicinity of \( \mathcal{M} \), \( (\rho(x, \mathcal{M}) \leq \rho_{\min}) \), we have \( \rho(\mathcal{D}(k + 1, k, x) \subset \mathcal{M}) \leq \rho_{\min} \) for all \( k \geq 0 \).

**Proof:** The condition given in the theorem equivalent to the condition that when initially we choose \( \rho(x(0), \mathcal{M}) \leq \rho_{\min} \) then we have \( \rho(x(k), \mathcal{M}) \leq \rho_{\min} \). Then we obtain the same condition of Definition 5.15. \( \square \)

Note that if we have an autonomous finite state system,

\[
x(k + 1) = f(x(k)) = \mathcal{D}(k + 1, k, x(k)),
\]

then Corollary 5.1 becomes that \( \mathcal{M} \) is \( \rho_{\min} \)-stable iff \( \rho(f(x), \mathcal{M}) \leq \rho_{\min} \) for all \( x \) in the minimal vicinity. As we see this is an easier condition to check for \( \rho_{\min} \)-stability of \( \mathcal{M} \). Below we define the global asymptotic \( \lambda \)-stability of a set.

**Definition 5.16** A set \( \mathcal{M} \subset \mathcal{X} \) is globally asymptotically \( \lambda \)-stable if it is \( \lambda \)-stable and globally asymptotically stable.

**Corollary 5.2** A set \( \mathcal{M} \subset \mathcal{X} \) is globally asymptotically \( \lambda \)-stable if there exist a Lyapunov function \( V(x) : \mathcal{X} \rightarrow \mathbb{R}^+ \cup \{0\} \) which satisfies the conditions
(i) $V(x) = 0 \quad \text{iff} \quad x \in M$,  

(ii) $V(x(k + 1)) - V(x(k)) < 0 \quad \text{for} \quad x(k) \notin M$,  

(iii) $\lambda \geq \max\{\rho(x, M) \mid V(x) \leq \max\{V(y) \mid \rho(y, M) \leq \rho_{\min}, y \in \mathcal{X}\}, x \in \mathcal{X}\}$. 

**Proof:** This is the direct result of Theorem 5.3 and Theorem 5.4. □
CHAPTER VI

Hybrid System State Space and Stability

6.1 The Hybrid System State Space

In this section we restrict ourselves to hybrid systems with no inputs and consider the continuous time model (Section 3.1.2). Note that since the number of states in the discrete state system portion is finite we can label them and provide a model with only one state variable. Consider the hybrid system

\[
X(t) = F(X(t^-), x(t^-)),
\]
(6.1)

\[
\frac{d}{dt}x(t) = f(X(t), x(t)),
\]
where \( X \in \mathcal{X} = \{0, 1, \ldots, N - 1\} \) and \( x \in \mathbb{R}^n \).

Define

\[
\Omega_{ij} \triangleq \{x \in \mathbb{R}^n \mid j = F(i, x)\}, \quad i, j \in \mathcal{X},
\]
(6.2)

to indicate the continuous subsystem states effect on discrete transitions.
Figure 9: Hybrid System discrete state transition.

Figure 9 illustrates a discrete state transition occurring due to continuous states being in a certain region of the state space. As shown in Figure 9, when the continuous state is in the region $\Omega_{ij}$ the discrete state jumps from $i$ to $j$. Note that the union of all regions $\Omega_{ij}$ for any state $i$ fills the continuous state space,

$$\bigcup_{j=0}^{N-1} \Omega_{ij} = \mathbb{R}^n \text{ for all } i \in \mathcal{X}. \quad (6.3)$$

We also define

$$\Omega_j \triangleq \bigcup_{i=0}^{N-1} \Omega_{ij}, \quad j \in \mathcal{X}, \quad (6.4)$$

which denotes the only region in which the system function $f(j, x)$ may participate in the continuous dynamics of hybrid system.

In the hybrid system representation 6.1 we may have some situations where the next states may not be determined. As shown in Figure 10 when the continuous state is in region $A$ the discrete system will switch between the discrete states $i$ and $j$. Since the derivative of the continuous state variable will not be continuous in this situation in general, the continuous dynamics is not identified in this region. However we assume that in a regional switching situation as mentioned above, the
hybrid system spends equal times in the switching discrete states.

Thus, the continuous state dynamics in a regional switching situation of $k$ states will be assumed to be

$$\frac{d}{dt} x(t) = \frac{(f(i, x(t)) + \cdots + f(l, x(t)))}{k}.$$ \hfill (6.5)

However after occurrence of a switching, the system trajectory may not be determined, as illustrated in Figure 10. In these cases we need to supply some additional information to eliminate the uncertain behavior.

Given an $N$ dimensional row vector $C = [C(0)C(1) \cdots C(N-1)] \in \mathcal{X}^N$ let us define the region

$$\Omega_C \triangleq \Omega_{0C(0)} \cap \Omega_{1C(1)} \cap \cdots \cap \Omega_{(N-1)C(N-1)}.$$ \hfill (6.6)

$\Omega_C$ denotes the region in which the discrete state $i$ will immediately go to $C(i)$ for $i \in \mathcal{X}$. Thus we partition the continuous state space into $N^N$ regions. Some of the
regions may be empty. Note that we have

$$\bigcup_{C \in \mathcal{X}^N} \Omega_C = \mathbb{R}^n. \quad (6.7)$$

We provide an example of a partition in Figure 11. As we see in Figure 11(c) the representative functions in each region are found. In $\Omega_{[00]}$ regardless of the discrete state we use the function $f(0, x)$ since the discrete state 1 becomes 0 immediately and stays as 0 in this region. In $\Omega_{[10]}$ the discrete state switches between 0 and 1 thus we use the arithmetical average of the two functions $f(0, x)$ and $f(1, x)$. In $\Omega_{[01]}$, on the other hand, the behavior of the continuous state system depends on the discrete state of the system. Thus hybrid systems differ from a general kind of nonlinear systems since they have discrete memory. We need to track the discrete state variable also to find out about the system behavior in general.

In general in each region $\Omega_C$ the representative functions can be found as given in the following algorithm.

**Algorithm 6.1** Finding actual consequent states of $i$ in region $\Omega_C$.

1) Let $C = [C(0)C(1)\cdots C(N-1)]$, $C(i) \in \mathcal{X}$, $i \in \mathcal{I}$.

2) For the discrete state $i \in \mathcal{I}$ find $C(i)$.

3) Take the resulting state as the initial state and return to the second step above.

After at most $N$ steps the resulting sequence of states will repeat themselves since there are only $N$ states in $\mathcal{X}$. 
Figure 11: Obtaining the partition of a hybrid system.
4) Take these repeating sequence as a set of states $\mathcal{X}_C(i)$ which denotes the actual
csequent states of $i$ in region $\Omega_C$.

Thus the representative function for the region $\Omega_C$ and the initial state $i$ can be
found as

$$f_C(i, x) \triangleq \frac{1}{|\mathcal{X}_C(i)|} \sum_{j \in \mathcal{X}_C(i)} f(j, x).$$

(6.8)

Therefore we obtain all the functions to represent the continuous state space. If there
is no regional switching, $\mathcal{X}_C(i)$ will have just one element and the original system
functions will be used. A condition for no regional switchings in a hybrid system can
be given as

$$\bigcup_{i,j,k,\ldots,m \in \mathcal{X}} \Omega_{ij} \cap \Omega_{jk} \cap \cdots \cap \Omega_{mi} = \emptyset. \quad (6.9)$$

Therefore we obtain a *representative hybrid system* from the original hybrid system
such that the resulting hybrid system has no regional switchings and no unnecessary
jumps as

$$X(t) = \tilde{F}(X(t^-), x(t^-)) = F(\dot{X}(t^-), x(t^-)),
\frac{dx(t)}{dt} = \tilde{f}(X(t), x(t)) = f_{C_x(t)}(X(t), x(t)),$$

(6.10)

where $\dot{X}(t^-) \in \mathcal{X}_{C_{x(t^-)}}(X(t^-))$, and, $C_x \in \mathcal{X}^N$ is defined such that $x \in \Omega_{C_x}$. If the set
$\mathcal{X}_{C_x}(X)$ above has more than one element this means that the original hybrid system
has an uncertainty so that choosing a specific $\dot{X}$ for the representative hybrid system
eliminates the uncertainty. Thus for the representative hybrid system we choose a
specific path after occurrence of regional switching.
6.2 Stability of a Continuous State in Hybrid Systems

In this section we consider the classical stability concept [32, 28] of a continuous state of hybrid system 6.1.

Definition 6.1 $\bar{x} \in \mathbb{R}^n$ is an equilibrium state of a hybrid system if $f_C(i, \bar{x}) = 0$ for all $i \in \mathcal{X}$.

Definition 6.2 The equilibrium state $x = 0$ is stable if for any $\varepsilon > 0$ there exists a $\delta > 0$ such that for any $\|x(0)\| < \delta$ and for any $X(0) \in \mathcal{X}$ we have $\|x(t)\| < \varepsilon$ for all $t \geq 0$. Otherwise the equilibrium state is unstable.

Definition 6.3 The equilibrium state $x = 0$ is globally asymptotically stable if it is stable and for any initial conditions $x(0) \in \mathbb{R}^n$ and $X(0) \in \mathcal{X}$ we have $x(t) \to 0$ as $t \to \infty$.

From Figure 11(c), let us obtain two nonlinear systems by choosing $f(0, X)$ and $f(1, X)$ for the region $\Omega_{[01]}$. One may wonder if it is sufficient or necessary for the global asymptotic stability of a hybrid system to have both nonlinear systems to be globally asymptotically stable. However as shown in Figure 12(a) although both nonlinear systems are stable, the hybrid system is not. Also as in Figure 12(b) although both nonlinear systems are unstable, the hybrid system is stable. Hence we cannot analyze hybrid systems by just splitting a hybrid system into a collection of nonlinear systems.
Here we give sufficient conditions for the stability of a hybrid system following classical Lyapunov stability theory [32, 28, 69].

**Theorem 6.1** If there exists a continuous scalar function $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ of the continuous state $x$, with continuous first order partial derivatives, such that

(i) $V(x)$ is positive definite,

(ii) $\dot{V}(x) = \nabla V \cdot f_C(i, x) \leq -w(x)$, where $w(x)$ is a positive definite function, for all $i \in \mathcal{X}$, in all the regions $x \in \Omega_C, C \in \mathcal{X}^N$,

(iii) $V(x) \rightarrow \infty$ as $\|x\| \rightarrow \infty$,

then the equilibrium point at the origin of the continuous state space of the hybrid system is globally asymptotically stable.
**Proof:** The theorem follows from Theorem 1, §15 in (Filippov 1988). Here for all the dynamics in all the regions we require the condition (ii) so that the Lyapunov function will decrease along any hybrid system trajectory. □

Note that the above theorem gives a sufficient condition. Furthermore we cannot always find a Lyapunov function even if the hybrid system is globally asymptotically stable (consider Figure 12(b)). Note also that we do not need to consider discrete state dynamics since the necessary information for hybrid system stability is contained in $f_C(X,x)$.

If there are no regional switchings in the hybrid system, the above theorem can lead to simpler conditions for the hybrid system Lyapunov function.

**Theorem 6.2** If there exists a continuous scalar function $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ of the continuous state $x$, with continuous first order partial derivatives, such that

(i) $V(x)$ is positive definite,

(ii) $\dot{V}(x,i) = \nabla V \cdot f(i,x) \leq -w(x)$, where $w(x)$ is a positive definite function, in all the regions $x \in \Omega_i$, for $i \in \mathcal{X}$,

(iii) $V(x) \rightarrow \infty$ as $\|x\| \rightarrow \infty$,

then the equilibrium point at the origin of the continuous state space of the hybrid system with no regional switchings is globally asymptotically stable.
Proof: If there is no regional switchings in the hybrid system the functions $f_C(i, x)$ in Theorem 6.1 need not be considered. Instead we can consider the original functions $f(i, x)$. Since there is no regional switchings, the function $f(i, x)$ is in effect only in $\Omega_{ii}$. Thus the theorem follows from Theorem 6.1.

Define the region

$$\tilde{\Omega}_j \triangleq \{ x \in \mathbb{R}^n \mid x \in \Omega_C, C(p) = j, j \in \mathcal{X}_C(j), p \in \mathcal{X}, C \in \mathcal{X}_N \}, \quad j \in \mathcal{X}, \quad (6.11)$$

which denotes the only region in which the system function $f(j, x)$ actually participates in the continuous dynamics of hybrid system. Note that $\tilde{\Omega}_j \subset \Omega_j$. Following Theorem 6.1, the corollary below can be provided:

**Corollary 6.1** If there exists a continuous scalar function $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ of the continuous state $x$, with continuous first order partial derivatives, such that

(i) $V(x)$ is positive definite,

(ii) $\dot{V}(x, i) = \nabla V \cdot f(i, x) \leq -w(x)$, where $w(x)$ is a positive definite function, in all the regions $x \in \tilde{\Omega}_i$, for $i \in \mathcal{X}$,

(iii) $V(x) \rightarrow \infty$ as $\|x\| \rightarrow \infty$,

then the equilibrium point at the origin of the continuous state space of the hybrid system is globally asymptotically stable.
Proof: Here if the condition (ii) is valid, that is if the Lyapunov function decreases for all the participating dynamic functions in all the regions, then it must also decrease along the switching trajectory. This gives the condition (ii) of Theorem 6.1. □

The difference between the above corollary and Theorem 6.2 is that Corollary 6.1 considers the switching regions also. Here we use the fact that if the Lyapunov function decreases for all the dynamic functions participating in the switching, then it must decrease along the switching trajectory also.

As mentioned, using a single Lyapunov function may not be enough to prove stability due to the discrete states of a hybrid system. To overcome this difficulty one possibility is to use multiple Lyapunov functions as given in the theorem below.

**Theorem 6.3** If there exist continuous scalar functions $V_1(x), V_2(x), \cdots, V_k(x) : \mathbb{R}^n \to \mathbb{R}$ of the continuous state $x$, with continuous first order derivatives in each region $\Omega_C, C \in \mathcal{X}^N$, and $V_1(x) = V_2(x) = \cdots = V_k(x)$ for all $x \in \partial \Omega_C$ (boundary of $\Omega_C$), $C \in \mathcal{X}^N$ such that

(i) $V_j(x)$ is positive definite for all $j \in [1, k]$,

(ii) For all regions $\Omega_C, C \in \mathcal{X}^N$ and for all discrete states $i \in \mathcal{X}$, there exists a $j \in [1, k]$ such that $\dot{V}_j(x, i) = \nabla V_j \cdot f_C(i, x) \leq -w(x)$ where $w(x)$ is a positive definite function.

(iii) $V_j(x) \to \infty$ as $\|x\| \to \infty$ for all $j \in [1, k]$,
then the equilibrium point at the origin of the continuous state space of the hybrid 

system is globally asymptotically stable.

Proof: In the above theorem we require that the Lyapunov functions have the 
same value on the boundary of each region \( \Omega_C \), but possibly different values inside. 
Thus if the conditions of the above theorem are satisfied we guarantee that the value 
of the Lyapunov function, in effect, decreases in all the regions. Thus the trajectory 
cannot be trapped inside any region except at the origin. When the trajectory reaches 
the boundary of a region, since all the values of the Lyapunov functions are the same 
in the boundary, the Lyapunov function must have strictly decreased, that is there 
cannot be any loops in the trajectory. Then according to Theorem 6.1 the hybrid 
system is globally asymptotically stable. \( \square \)

We now assert that using at most \( N \) Lyapunov functions is enough to prove the 
global asymptotic stability of a globally asymptotically stable hybrid system. This 
assertion may be easily verified. Since we have \( N \) different discrete state values in the 
hybrid system there can be at most \( N \) different dynamics \( f_C(i, x) \) in a region. Thus 
we may need at most \( N \) different Lyapunov functions to prove the global asymptotic 
stability.
6.3 Hybrid State Stability

The state of a hybrid system can be thought as an expanded state vector \( \begin{bmatrix} X \\ x \end{bmatrix} \) with full state space \( \mathcal{X} \times \mathbb{R}^n \). This state space has the combination of discrete and continuous states and it is quite different than ordinary continuous or discrete state spaces. Both the discrete and continuous dynamics are to be considered at the same time. In this section the stability of the origin \((X = 0, x = 0)\) of the hybrid system 6.1 is considered. The definitions can be extended for the stability of hybrid invariant sets.

First, the origin is to be invariant, that is the origin is to be an equilibrium point.

**Definition 6.4** \( \begin{bmatrix} \dot{X} \\ \dot{x} \end{bmatrix} \in \mathcal{X} \times \mathbb{R}^n \) is an equilibrium state of hybrid system 6.1 if \( F(\dot{X}, \dot{x}) = \dot{X} \) and \( f(\dot{X}, \dot{x}) = 0 \).

If the classical Lyapunov stability theory is applied directly, the following definition of stability is obtained.

**Definition 6.5** The equilibrium state \( \begin{bmatrix} 0 \\ 0 \end{bmatrix} \) of hybrid system 6.1 is stable if for any \( \varepsilon > 0 \) there exists a \( \delta > 0 \) such that for any \( \| \begin{bmatrix} X(0) \\ x(0) \end{bmatrix} \| < \delta \) we have \( \| \begin{bmatrix} X(t) \\ x(t) \end{bmatrix} \| < \varepsilon \) for all \( t \geq 0 \). Otherwise the equilibrium state is unstable.
The definition of hybrid state stability above requires that in a sufficiently small compact neighborhood of continuous state $x = 0$ the discrete state cannot change, that is $X$ has to be zero for any continuous state in that neighborhood, once the initial discrete state is zero. This is because when $\varepsilon$ is given sufficiently small, to obtain $\| \begin{bmatrix} X(t) \\ x(t) \end{bmatrix} \| < \varepsilon$, $X(t)$ has to be chosen as 0 for $t > 0$ once $X(0) = 0$ since $X$ can assume only finite number of values (or discrete values). Otherwise, that is if the discrete state jumps to other values other than 0, we may no longer satisfy the norm condition above. Thus the condition

$$\exists \delta > 0 \text{ such that } F(0, x(0)) = 0, \quad \forall \|x(0)\| < \delta$$

must be satisfied, otherwise the origin will not be stable. If condition 6.12 is satisfied, on the other hand, the only condition for hybrid state stability is the stability of the origin of the system

$$\frac{d}{dt} x(t) = f(0, x(t)).$$

As a result, the origin of the hybrid system $(X = 0, x = 0)$ is stable if and only if the origin $(x = 0)$ of system 6.13 is stable and the condition 6.12 is satisfied.

Asymptotic stability can be defined as follows.

**Definition 6.6** The equilibrium state $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ of the hybrid system 6.1 is globally asymptotically stable if it is stable and for any initial conditions $x(0) \in \mathbb{R}^n$ and $X(0) \in \mathcal{X}$ we have $x(t) \to 0$ and $X(t) \to 0$ as $t \to \infty$. 
From the definition above we observe that the discrete state of the hybrid system must go to 0 in finite time. This is because $X(t) \to 0$ means there exists a $\tilde{t}$ such that $X(t) = 0$ for $t \geq \tilde{t}$ since $X(t)$ can assume only discrete values. That is actually a desirable behavior. Otherwise the discrete state may go to 0 in infinite time or may never go to 0 in which case to talk about global asymptotic stability of the origin will not make sense.

Some other definitions of stability for hybrid systems can also be given. For example only the continuous state behavior of a hybrid system can be investigated. In this case the conditions of hybrid state stability given above are not required but the continuous state is required to be stable for any discrete state dynamics (see Section 6.2 above).

In the definition of the hybrid state stability above, it is required that condition 6.12 is satisfied. This requirement may be too strong for some hybrid systems. The following definition can be provided for hybrid system 3.19-3.20.

**Definition 6.7** The state $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ of the hybrid system 3.19-3.20 is globally asymptotically reachable if for any initial conditions $x(0) \in \mathbb{R}^n$ and $X(0) \in \mathcal{X}$ there exist inputs $U(t) \in \mathcal{U}$ and $u(t) \in \mathbb{R}^r$ for $t \geq 0$ such that $x(t) \to 0$ and $X(t) \to 0$ as $t \to \infty$.

The above definition can, of course, be used for system 6.1 too, in which case there is no input to the hybrid system (or the inputs to the hybrid system have no effect).
In the above definition the stability of the origin is not required. In this case again the discrete state must reach 0 in finite time but the continuous state may reach 0 in infinite time. This type of property may be desirable for many hybrid systems. The following definition can also be given.

**Definition 6.8** The state \( \begin{bmatrix} 0 \\ 0 \end{bmatrix} \) of the hybrid system 3.19-3.20 is globally reachable if for any initial conditions \( x(0) \in \mathbb{R}^n \) and \( X(0) \in \mathcal{X} \) there exist inputs \( U(t) \in \mathcal{U} \) and \( u(t) \in \mathbb{R}^r \) for \( t \in [0,\bar{t}] \) such that \( x(\bar{t}) = 0 \) and \( X(\bar{t}) = 0 \).

For global reachability it is required that the hybrid state go to the origin in finite time, that is, both the discrete and the continuous states must be zero in finite time. If there is no input and the origin is an equilibrium state, the state of the hybrid system will not leave the origin once it reaches to the origin.

Lyapunov theory may be applied to hybrid state stability problems as shown in the following theorem.

**Theorem 6.4** Consider the representative hybrid system 6.10 and a Lyapunov function \( V(X,x) : \mathcal{X} \times \mathbb{R}^n \to \mathbb{R} \) of the hybrid state \( \begin{bmatrix} X \\ x \end{bmatrix} \), with continuous first order partial derivatives in \( x \), such that for any given \( X \in \mathcal{X} \) and \( x \in \mathbb{R}^n \), \( \dot{X} \triangleq F(X,x) \)

(i) \( V(X,x) \) is positive definite, that is, \( V(0,0) = 0 \) and \( V(X,x) > 0 \) for \( X \neq 0 \) and \( x \neq 0 \),
(ii) either $V(X, x) > V(\tilde{X}, x)$,

or $V(X, x) = V(\tilde{X}, x)$ and $\dot{V}(X, x) = [\frac{\partial V(\tilde{X}, x)}{\partial x_1} \ldots \frac{\partial V(\tilde{X}, x)}{\partial x_n}] \cdot f(X, x) \leq -w(X, x)$,

where $w(X, x)$ is a positive definite function,

(iii) $V(X, x) \to \infty$ as $\|x\| \to \infty$,

then the representative hybrid system 6.10 is globally asymptotically stable.

**Proof:** In the representative hybrid system 6.10 we will not have a regional switching situation or an uncertainty as mentioned before. We follow the Lyapunov theory of dynamic systems with discontinuous right hand sides (Theorem 1, §15 in [28]). Note that the discrete state part of the initial hybrid state $X \in \mathcal{X}$ and $x \in \mathbb{R}^n$ immediately jumps to $\tilde{X}$. Thus if we have $V(X, x) > V(\tilde{X}, x)$ then $V$ decreases along the trajectory. Note that we do not need the condition $V(\tilde{X}, x) - V(X, x) < -\hat{w}(X)$ where $\hat{w}(X)$ is a positive definite function, as in discrete time systems, since $X$ can assume only finite number of states. If $V(X, x) = V(\tilde{X}, x)$ (either due to $X = \tilde{X}$ or not) the next discrete state will be $\tilde{X}$ thus we need to consider the function $V(\tilde{X}, x)$.

The direction of the continuous state trajectory on the other hand depends on the function $f(X, x)$. Thus we need to check the derivative of the Lyapunov function (with discrete state $\tilde{X}$) along the dynamics of the hybrid system. If the conditions given above are satisfied, in any trajectory of the hybrid system $V$ will strictly decrease. Since we use a positive definite function $w(X, x)$, the trajectory cannot be trapped in
any point other than the origin and the discrete state must reach to \( X = 0 \) in finite
time. Note also that since \( X \) can have only finite number of values, in a sufficiently
small neighborhood of the continuous state \( x = 0 \) the discrete state cannot change
from \( X = 0 \) once \( X = 0 \) because \( V \) has continuous first order partial derivatives for
the continuous state variable. Therefore the condition 6.12 is automatically satisfied.

\[ \square \]

6.4 Linear Hybrid Systems

A special class of hybrid systems called linear hybrid systems is now considered, such
that, the continuous state system portion is modeled as a linear discrete time system.

Hybrid system equations considered are given as follows

\[
\begin{align*}
X(k + 1) &= F(X(k), x(k), U(k)), \\
x(k + 1) &= A_{X(k)} x(k),
\end{align*}
\]

where \( x \in \mathbb{R}^n, \ X \in \mathcal{X} = \{1, 2, \cdots, N\}, \ U \in \mathcal{U} = \{1, 2, \cdots, R\}, \ A_i \in \mathbb{R}^{n \times n}. \)

As seen from the above equations the continuous state matrices \( A_i \) are chosen
according to the discrete states of the hybrid system. Since there are \( N \) discrete
states the continuous state matrices are chosen from the set \( \mathcal{A} \triangleq \{A_1, A_2, \cdots, A_N\} \).

The next discrete state of the hybrid system is chosen according to the current discrete
state, the current continuous state, and, the current discrete input. We consider the
stability of the origin of the continuous state space as in Section 6.2 with external
discrete inputs. The stability definitions for this case are given below.

**Definition 6.9** The equilibrium state \( x = 0 \) is stable if for any \( \varepsilon > 0 \) there exists a \( \delta > 0 \) such that for any \( \| x(0) \| < \delta \) and for any \( X(0) \in \mathcal{X} \) and \( U(k) \in \mathcal{U} \) we have \( \| x(k) \| < \varepsilon \) for all \( k \geq 0 \). Otherwise the equilibrium state is unstable.

**Definition 6.10** The equilibrium state \( x = 0 \) is globally asymptotically stable if it is stable and for any initial conditions \( x(0) \in \mathbb{R}^n \) and \( X(0) \in \mathcal{X} \) and \( U(k) \in \mathcal{U} \) we have \( x(t) \to 0 \) as \( t \to \infty \).

The continuous state system portion of hybrid system 6.14 can be thought of as system A.43 (See Appendix A). Hence the following result can be obtained.

**Corollary 6.2** The origin of the continuous state space of hybrid system 6.14 is globally asymptotically stable if \( \mathcal{A} \) is asymptotically stable.

The proof of stability of hybrid systems may be too difficult in many cases. Here Corollary 6.2 can be used to check the stability of a hybrid system or design controllers for hybrid systems. Corollary 6.2 gives a sufficient condition. If the matrix set \( \mathcal{A} \) of a hybrid system is asymptotically stable, a discrete state noise or perturbation to the hybrid system will not destabilize it. Actually this type of stability may be desirable in many cases for hybrid systems. Hence when we design control for hybrid systems we may want the closed loop hybrid system to have a stability property
as mentioned above, that is, the closed loop continuous state matrix set is to be asymptotically stable. Here it is also straightforward to see that the hybrid system is not asymptotically stable if $A$ is not asymptotically stabilizable.

Consider hybrid system 6.14 with discrete state dynamics as given in Figure 13. The discrete state transitions are due to the continuous states and the discrete inputs. In this case the hybrid system is asymptotically stable if the matrix set $\{A_1, A_2, A_3\}$ is asymptotically stable. When checking the asymptotic stability of $\{A_1, A_2, A_3\}$ all possible sequences from this set are considered. However as seen from the discrete state dynamics, not all the sequences actually occur. For example after the state 3 only the state 1 is allowed but not the others. We may make use of the special sequences of the discrete states according to the discrete dynamics and find a more strict sufficient condition for the asymptotic stability of the hybrid system.

In the discrete state sequence in Figure 13 either 1,2 or 1,2,3 will be chosen at the discrete state 1. Thus it can be concluded that the hybrid system will be

![Figure 13: A discrete state system with 3 states.](image)
asymptotically stable if the matrix set \( \{ A_2A_1, A_3A_2A_1 \} \) is asymptotically stable. Here another matrix set is formed according to the discrete dynamics of the hybrid system. Hence a more strict sufficient condition is obtained for the asymptotic stability.

A finite set of matrices may not be always obtained by considering the discrete dynamics. Consider a discrete dynamics as given in Figure 14. In this case the hybrid system is asymptotically stable if the set

\[ \{ A_1A_2, A_3A_2, A_3A_4A_3A_2, A_3A_4A_3A_4A_3A_2, \ldots \} \]

is asymptotically stable. The above set is obtained as considering the state 2 as the initial state and finding all the discrete state loops which begins and ends with the state 2. The problem, however, is that the resulting matrix set becomes infinite. Hence sometimes it may not be possible to obtain a finite set of matrices which exactly match the discrete state dynamics. We know that if the set \( \mathcal{A} = \{ A_1, A_2, A_3, A_4 \} \) is asymptotically stable the hybrid system will be asymptotically stable. Also if the set \( \mathcal{A}^2 \) is asymptotically stable the hybrid system will be asymptotically stable. In set \( \mathcal{A}^2 \) the products of two matrices from \( \mathcal{A} \) are used. However here we can also make use of the discrete dynamics. It is known, for example, that after the state 1, the
next state cannot be 3. Thus we should not include the matrix $A_3A_1$ in $A^2$ when considering the asymptotic stability of the hybrid system. Here a set is obtained as given below

$$A^2_F = \{A_2A_1, A_1A_2, A_3A_2, A_2A_3, A_4A_3, A_3A_4\}.$$  

Now it can be concluded that the hybrid system is asymptotically stable if $A^2_F$ is asymptotically stable. In this way a more strict sufficient condition is obtained as compared to checking the asymptotic stability of $A$ since $A^2_F$ is a subset of $A^2$.

In general let us define the set

$$A^m_F \triangleq \{A_{X(m)}A_{X(m-1)} \cdots A_{X(1)} \mid A_{X(j)} \in A, \ X(i+1) = F(X(i), x, u), \quad (6.15)\}$$

$$X(j) \in \mathcal{X}, \ j \in [1 : m], \ i \in [1 : m - 1], \ x \in \mathbb{R}^n, \ u \in \mathcal{U} \}.$$  

Then the following result is obtained.

**Corollary 6.3** The origin of the continuous state space of hybrid system 6.14 is globally asymptotically stable if $A^m_F$ is asymptotically stable for any $m \geq 1$. 
CHAPTER VII

Control Design

7.1 Sliding Mode Control Design in Discrete State Systems

We use the definition of the sliding mode given in [25], [79]. In our notations it can be formulated as follows:

**Definition 7.1** The point \( x \in X \) is a sliding mode point of the system \( D \) at instant \( k \) if a solution \( x_0 \) of the equation

\[
x = D(k, k - 1, x_0)
\]

(7.1)

exists and is not unique.

**Definition 7.2** The trajectory \( T = \{x(k_0), x(k_0 + 1), \ldots\} \) of \( D \) is called a sliding mode trajectory if it consists of the sliding mode points.

In these definitions we specified the time instants in order to cover the nonstationary case, but from now on we will consider only stationary sets of sliding mode points, which means that the above definitions will be used without the time dependence.
Definition 7.3 An integral manifold of $\mathcal{D}$ (i.e. an invariant set consisting of system trajectories) $\mathcal{M}$, is called a sliding manifold\(^1\) if it consists of sliding mode trajectories.

For example the manifold $\mathcal{M} = \{X \in \mathcal{X} | X_2 = 0\}$ in Figure 8 is a sliding manifold.

Consider the neighbor-transition dynamic systems with finite state space given in Section 4.2. The distance between any of two neighboring points on the same trajectory is assumed to be equal to $\rho_{\text{min}}$ (which is the minimal distance). For the systems where the inverse of this statement is also valid, the sliding mode manifolds and trajectories are asymptotically $\rho_{\text{min}}$-stable. We formulate this as a theorem.

Theorem 7.1 Let $\mathcal{D}$ satisfy

$$x, y \in \mathcal{X}, \rho(x, y) = \rho_{\text{min}} \iff \begin{cases} x = \mathcal{D}(k, k - 1, y) \quad \text{or} \\ y = \mathcal{D}(k, k - 1, x) \end{cases}$$

for all $k = 0, 1, 2, \ldots$. Then any sliding manifold $\mathcal{M}$ is asymptotically $\rho_{\text{min}}$-stable, moreover, for any $x(k)$ from the $\rho_{\text{min}}$-vicinity of $\mathcal{M}$ we have

$$x(k + 1) \in \mathcal{M},$$

where $x(k + 1) = \mathcal{D}(k + 1, k, x(k))$.

Theorem 7.1 implies that if $\mathcal{M}$ is a sliding manifold, then every trajectory entering its neighborhood is absorbed by $\mathcal{M}$ in one step.

\(^1\)Although "invariant set" is more natural here we keep "manifold" to be consistent with sliding mode literature.
Now let us consider the Discrete State System 2.9. If the control law is fixed, i.e. the input $u$ is a function of the state $x$:

$$u(k) = g(x(k)), \quad (7.4)$$

then to the automaton

$$x(k + 1) = F(x(k), u(k)), \quad (7.5)$$

there corresponds a stationary dynamic system defined by the equality

$$D(k, k_0, x_0) = \underbrace{\tilde{F}(\ldots \tilde{F}(x_0) \ldots)}_{k-k_0}, \quad (7.6)$$

where $\tilde{F}(x) = F(x, g(x))$.

Our purpose is to find a control $u(k)$, such that the closed loop system has stable sliding manifolds in the sense of the definitions given.

The usual approach to Sliding Mode Control (SMC) design follows the two steps:

1. Define the manifold(s) such that the system constrained on the manifold (intersection of manifolds) is stable.

2. Obtain the control to force the system to the manifold.

In our examples we use the hierarchical approach for reaching the desired sliding set.
When the state reaches the first manifold it slides over the manifold to reach the second manifold and so on as shown in Figure 15. Eventually the state reaches the desired state or tracks the desired trajectory.

For example, consider the objective of stabilizing the system at the state $x_0$. To accomplish this goal by using the SMC idea we define the manifolds $M_i$ for $i \in \{1, 2, \cdots, q\}$ such that

$$X \supset M_1 \supset M_2 \supset \cdots \supset M_q = \{x_0\}. \quad (7.7)$$

where

$$n = \dim(X) \geq \dim(M_1) \geq \dim(M_2) \geq \cdots \geq \dim(M_q) = 0. \quad (7.8)$$

The figure is for explanation purposes, and should not be taken too literally.
Define $S_i \in B^{1 \times N}$, where $N$ is the number of states in the automaton, such that $S_i$ has 0 at the $j$th entry iff $x^j \in M_i$. Then we can write

$$M_i = \{ x \mid S_i x = 0 \} \quad i \in \{1, 2, \ldots, q\}$$

(7.9)

The task of the controller is to make the state reach the manifold $M_i$ if the state is in $M_{i-1}$ currently. We choose $M_0 = \mathcal{X}$ and $M_q = \{x_0\}$. To accomplish the task, the controller must use a control sequence such that the next state is in $M_i$ if possible. That is, $x(k+1) \in M_i$. From 2.7 and 7.9 it follows that

$$S_i x(k+1) = 0$$  

(7.10)

$$S_i A(u(k)) x(k) = 0$$  

(7.11)

Hence at the time instant $k$ we can use any control input $u(k) \in U_k$ where

$$U_k = \{ u \in \mathcal{U} \mid F(x(k), u) \in M_i \},$$  

(7.12)

or

$$U_k = \{ u \in \mathcal{U} \mid S_i A(u)x(k) = 0 \},$$  

(7.13)

if $U_k$ is not empty. If $U_k$ is empty, the next state may not be in $M_i$.

In the case that it takes multiple time steps to reach $M_i$, we may need to refer to an auxiliary function (or a guide) to choose a suitable control. Here we use Lyapunov functions as a guide. Let a Lyapunov function $V_i(x)$ satisfy the conditions in Theorem 5.3 for $M_i$. Then the input at the time instant $k$ can be chosen from the set

$$U_k = \{ u \in \mathcal{U} \mid V_i(F(x(k), u)) < V_i(x(k)) \}.$$  

(7.14)
Therefore, according to Theorem 5.2, the state trajectory reaches $\mathcal{M}_i$. We may use the same approach to reach the other inner manifolds. One example for such a Lyapunov function is $V_i(x) = d_{\text{min}}(x, \mathcal{M}_i)$ where $d_{\text{min}}$ gives the minimum time steps in which $\mathcal{M}_i$ can be reached from the state $x$. However it may be difficult to find the function $d_{\text{min}}$ generally. Thus we may need to use a simpler function for $V_i(x)$. We could relax some conditions on $V_i(x)$ such as instead of a strictly decreasing function we can choose a non-increasing function. Then $u(k)$ is to be chosen from

$$U_k = \{ u \in \mathcal{U} \mid V_i(F(x(k), u)) \leq V_i(x(k)) \}. \quad (7.15)$$

Since $V_i(x(k))$ may not be strictly decreasing at each time steps, this kind of control may take longer to reach $\mathcal{M}_i$. In particular, if we do not know the system exactly or if there is a disturbance there may be not much we can do. In such cases one can use the control

$$u(k) = \arg \min_{u \in \mathcal{U}} V_i(F(x(k), u)). \quad (7.16)$$

Note that we can use the control in 7.16 in the general case once we decide how to choose $V_i(x)$. The sliding mode control method described above can be used to design a suitable control for discrete state systems.

In DSS representation 2.9, on the other hand, let us assume we choose the manifold as
\[ \mathcal{M} = \{ x \mid Sx = 0 \} \]  

(7.17)

where \( S \in \mathbb{R}^{q \times n} \). If we choose a particular Lyapunov function candidate as

\[ V(x) = (Sx)^TMSx = x^TSTMSx \]  

(7.18)

where \( M \in \mathbb{R}^{q \times q} \) positive definite matrix then \( V(x) \) satisfies the first and the second conditions in Theorem 5.3. Thus we need

\[ V(x(k+1)) - V(x(k)) = F(x(k), u(k))^TSTMSF(x(k), u(k)) - x(k)^TSTMSx(k) < 0. \]  

(7.19)

If we can find the control \( u(k) \) satisfying 7.19 then \( u(k) \) leads the state trajectory to reach \( \mathcal{M} \). Again if there is disturbance or we do not know the system exactly we may choose the best control available as

\[ u(k) = \arg \min_{u \in \mathcal{U}} F(x(k), u)^TSTMSF(x(k), u). \]  

(7.20)

### 7.2 Sliding Mode Control Design in Hybrid Systems

In this section we consider a hybrid system consisting of a plant which is modeled as a time-invariant, continuous time, continuous state system (CSS), an interface, and, a discrete state system (DSS) controller as shown in Figure 16.

As seen in Chapter IV the discretization of a continuous state system can result in a nondeterministic discrete state system (NDSS) as in 4.3. In discrete state system
4.3 The transition instants $t_k$ may vary and depend on the control and "disturbance" $\omega$. Hence, after assigning control values for 4.3 i.e. given the control law $U$ as a function of the state $X$:

$$U(t_k) = G(X(t_k)),$$

(7.21)

we deal with the system

$$X(t_{k+1}) = F'(\omega(t_k), X(t_k)),$$

(7.22)

where the transitions instants $t_k$ are function of the current state and time

$$t_{k+1} = T(\omega(t_k), X(t_k)).$$

(7.23)

To this system there corresponds a nonstationary discrete time set of mappings satisfying the semigroup property 2.2

$$x_k = D(k, k_0, x_0),$$

(7.24)
where \( x_k = X(t_k) \), and \( D \) is defined by the equality

\[
D(k, k_0, x_0) = \bar{F}(k, \bar{F}(k - 1, \ldots \bar{F}(k_0, x_0) \ldots)),
\]

(7.25)

with \( \bar{F}(k, x) = F(x, G(x), \omega(t_k)) \).

If the lengths of all time intervals \([t_k, t_{k+1})\) are greater than some nonzero positive number \( \delta \)

\[
\min_k |t_{k+1} - t_k| > \delta > 0
\]

(7.26)

then the family of mappings \( D \) also satisfies 2.3 and in this case it is represents a well defined discrete time dynamic system. In case, when the transitions can occur instantly and infinitely often 2.3 does not hold and additional clarification is needed.

In the application of sliding mode control to continuous state systems, high frequency switching of the control variable is allowed so as to maintain the system state on the prescribed manifold or at least in a small boundary layer in spite of the disturbances and parameter variations. One feature of the closed loop system in this case is characterized by the finite time of convergence to the sliding manifold. The behavior of the constrained system, in practice, is defined by averaging the chattering motion around the manifold. The control can be substituted by its equivalent value resulting in this motion.

The same situation can take place in 7.22. If high frequency switchings occur we should consider the equivalent system, which is obtained by averaging.

We introduce the following definition:
Definition 7.4 Two states $X$ and $X'$ of the finite state space $\mathcal{X}$ are called $\mathcal{D}$-equivalent at time $t_k$ if

$$X = F'(\omega(t_k), X')$$

(7.27)

and

$$t_k = T(\omega(t_k), X'),$$

(7.28)

or,

$$X' = F'(\omega(t_k), X)$$

(7.29)

and

$$t_k = T(\omega(t_k), X).$$

(7.30)

Considering the set of equivalence classes

$$\bar{\mathcal{X}} = \mathcal{X}/\mathcal{D}$$

(7.31)

we can introduce a new system $\bar{F}$, induced by $\mathcal{D}$:

$$\bar{X}(t_{k+1}) = \bar{F}(t_k, \bar{X}(t_k)) \Leftrightarrow X(t_{k+1}) = F'(\omega(t_k), X(t_k)), X(t_k) \in \bar{X}(t_k).$$

(7.32)

If a particular class contains more than one element (i.e. instant switchings occur between its states) it can be associated with the equivalent value. Such a construction leads to the description of sliding motion in a hybrid system, which is a natural generalization of the corresponding concept in ordinary differential equations and satisfies the definitions introduced above for discrete state systems. The corresponding
discrete state configuration can be obtained for continuous state plant and interface as described above. Then sliding mode control for hybrid systems can be designed as introduced in Section 7.1.

If high frequency switchings are obtained in the discrete state controller of the hybrid system we can substitute the control for the continuous part by its equivalent value and the function $F'$ of the corresponding discrete state system by $\bar{F}$, which is an equivalent system. In $\bar{F}$ there are no instant transitions, which means that the corresponding discrete time automaton is well defined.

7.3 Interconnections of Hybrid Systems

In this section we study serial and feedback connections of hybrid systems and show that interconnected hybrid systems can also be modeled as hybrid systems. Here we consider discrete time hybrid systems. The results are also applicable to continuous time hybrid systems. The sample and hold devices will not be shown in the figures.

7.3.1 Serial Connection

Let us consider two hybrid systems connected serially as shown in Figure 17. We can redraw Figure 17 so that we have a hybrid system representation as shown in Figure 18.
Figure 17: A serial connection of two hybrid systems.

In the resulting system we have

$$X(k) = \begin{bmatrix} X_a(k) \\ X_b(k) \end{bmatrix}, \quad x(t) = \begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix},$$

$$X(k+1) = F(X(k),x(kT),U(k)) = \begin{bmatrix} F_a(X_a(k),x_a(kT),U(k)) \\ F_b(X_b(k),x_b(kT),G_a(X_a(k),x_a(kT),U(k))) \end{bmatrix},$$

$$\frac{d}{dt} x(t) = f(X(k),x(t),u(t)) = \begin{bmatrix} f_a(X_a(k),x_a(t),u(t)) \\ f_b(X_b(k),x_b(t),g_a(X_a(k),x_a(t),u(t))) \end{bmatrix},$$

$$Y(k) = G(X(k),x(kT),u(k)) = G_b(X_b(k),x_b(kT),G_a(X_a(k),x_a(kT),U(k))),$$

$$y(t) = g(X(k),x(t),u(t)) = g_b(X_b(k),x_b(t),g_a(X_a(k),x_a(t),u(t))).$$

Parallel connection of hybrid systems can also be shown to be modeled as a hybrid system.
7.3.2 Feedback

Consider the hybrid system with feedback shown in Figure 19. As seen from the figure feedforward path are not used in the hybrid system to avoid an algebraic loop which may occur due to the feedback. That is, the function $G_a$ and $g_a$ do not depend on the inputs of the hybrid system. Actually we can think the hybrid system in the figure as a serial connection of a controller hybrid system and a plant hybrid system.

The whole feedback system then can be represented as a hybrid system as shown in Figure 20. In this case we let...
Figure 19: A feedback hybrid system.

\[ X(k) = X_a(k), \quad x(t) = x_a(t), \]

\[ X(k+1) = F(X(k), x(kT), U(k)) = F_a(X(k), x(kT), P(G_a(X(k), x(kT)), U(k))), \]

\[ \frac{d}{dt} x(t) = f(X(k), x(t), u(t)) = f_a(X(k), x(t), p(g_a(X(k), x(t)), u(t))), \]

\[ Y(k) = G(X(k), x(kT), u(k)) = G_a(X(k), x(kT)), \]

\[ y(t) = g(X(k), x(t), u(t)) = g_a(X(k), x(t)). \]

Hence as we conclude, interconnections of hybrid systems can be represented in hybrid system framework.
7.4 Lyapunov Based Control for Hybrid Systems

In this section we consider the hybrid system continuous time model in Section 3.1.2. Consider the closed loop feedback hybrid system in Figure 21 where $P$ and $p$ represent the feedback functions from the outputs of the plant to the inputs of the controller. As shown in Section 7.3 this closed loop system can also be considered as a hybrid system. That is, the analysis methods for hybrid systems can be directly applied for the closed loop feedback hybrid systems. The resulting discrete state of the closed loop hybrid system can be denoted as $X = \begin{bmatrix} X_p \\ X_c \end{bmatrix}$ and the resulting continuous state
Figure 21: A closed loop feedback hybrid system.

as \( x = \begin{bmatrix} x_p \\ x_c \end{bmatrix} \). That is, we consider the combination of the plant and controller states as the resulting state of the closed loop hybrid system.

Consider the closed loop hybrid system given in Figure 22. The open loop hybrid system is represented by the equations

\[
\begin{align*}
X(t) &= F(X(t^-), x(t^-), U(t^-)), \\
\frac{d}{dt} x(t) &= f(X(t), x(t), u(t)),
\end{align*}
\]  

(7.33)

where \( X \in \mathcal{X}, \ x \in \mathbb{R}^n, \ U \in \mathcal{U}, \) and, \( u \in \mathbb{R}^r.\)

First assume that the feedback \( U = P(X) \) is designed according to the specifications in the discrete state space. The feedback \( u = p(x) \) is to be designed such that the origin of the continuous state space \( (x = 0) \) is globally asymptotically stable as
defined in Section 6.2.

We design the required feedback by the help of Corollary 6.1. Assume that we have a Lyapunov function $V(x) : \mathbb{R}^n \to \mathbb{R}$ of the continuous state $x$, with continuous first order partial derivatives, such that

(i) $V(x)$ is positive definite,

(ii) $V(x) \to \infty$ as $\|x\| \to \infty$.

Then the only condition left is the condition (ii) of Corollary 6.1 for global asymptotic stability. To achieve this goal we can use the input $u$. We need

$$
\dot{V}(x, X) = \nabla V \cdot f(X, x, u) \leq -w(x), \quad \forall X \in \mathcal{X}, \forall x \in \mathbb{R}^n,
$$

(7.34)
where \( w(x) \) is a positive definite function. Then if a \( u = p(x) \) can be found such that condition 7.34 is satisfied then the closed loop hybrid system will be globally asymptotically stabilized. Alternatively we can design the best \( p(x) \) in terms of the Lyapunov function given, as

\[
U = p(X, x) = \arg \min_{u \in \mathbb{R}} \left( \max_{x \in X} \nabla V \cdot f(X, x, u) \right).
\]  

(7.35)

Figure 23: A hybrid feedback closed loop hybrid system.

Consider the closed loop hybrid system given in Figure 23. The resulting feedback system can also be shown to be represented as a hybrid system. Here both the feedback functions \( U = P(X, x) \) and \( u = p(X, x) \) will be designed. Again we consider the global asymptotic stability of the origin of the continuous state space. Assume we have a Lyapunov function \( V(x) : \mathbb{R}^n \to \mathbb{R} \) of the continuous state \( x \), with continuous
first order partial derivatives such that the conditions (i) and (ii) above are satisfied. Define

\[
\begin{bmatrix}
M(x) \\
m(x)
\end{bmatrix} = \arg \min_{X \in \mathbb{R}, u \in \mathbb{R}^r} \nabla V \cdot f(X, x, u). 
\] (7.36)

\(M(x)\) and \(m(x)\) give the best discrete state and the best continuous input for a given continuous state \(x\) in terms of the Lyapunov function given. Also define

\[
p(X, x) = \arg \min_{u \in \mathbb{R}^r} \nabla V \cdot f(X, x, u),
\] (7.37)

which gives the best continuous feedback for given discrete and continuous states. \(P(X, x)\) can be designed such that the best discrete state will be chosen immediately from initial discrete and continuous states and we will obtain \(u = p(X, x) = m(x)\). Thus we need

\[
P(X, x) \in \{ U \mid M(x) = F(X, x, U) \}. 
\] (7.38)

Therefore the feedback functions \(P(x, x)\) and \(p(X, x)\) can be chosen as above to achieve the best control in terms of the Lyapunov function given. If we have

\[
\dot{V}(x, X) = \nabla V \cdot f(X, x, p(X, x)) \leq -w(x), \quad \forall X \in \mathcal{X}, \forall x \in \mathbb{R}^n, 
\] (7.39)

where \(w(x)\) is a positive definite function then the closed loop hybrid system will be globally asymptotically stabilized.

Now consider the hybrid state stability that is the global asymptotic stability of the origin of the hybrid state space \((X = 0, x = 0)\). Consider the closed loop hybrid
system in Figure 23 and a Lyapunov function \( V(X, x) : \mathcal{X} \times \mathbb{R}^n \to \mathbb{R} \) of the hybrid state \(
abla \begin{bmatrix} X \\ x \end{bmatrix} \), with continuous first order partial derivatives in \( x \), such that for any given \( X \in \mathcal{X} \) we have

(i) \( V(X, x) \) is positive definite, that is, \( V(0, 0) = 0 \) and \( V(X, x) > 0 \) for \( X \neq 0 \) and \( x \neq 0 \),

(ii) \( V(X, x) \to \infty \) as \( \|x\| \to \infty \).

Then if additionally the condition (ii) of Theorem 6.4 is satisfied we can conclude that the closed loop hybrid system is globally asymptotically stable. For this purpose the feedback functions need to be chosen as

\[
P(X, x) \in \{ U \in \mathcal{U} \mid V(X, x) \geq V(F(X, x, U), x) \},
\]

\[
p(X, x) \in \{ u \in \mathbb{R}^n \mid \frac{\partial V(F(X, x, P(X, x)), x)}{\partial x_1} \cdots \frac{\partial V(F(X, x, P(X, x)), x)}{\partial x_n} \cdot f(X, x, u) \leq -w(X, x) \},
\]

where \( w(X, x) \) is a positive definite function. The best performance can be achieved in terms of the Lyapunov function given by choosing

\[
P(X, x) = \arg \min_{U \in \mathcal{U}} V(F(X, x, U), x) - V(X, x),
\]

\[
p(X, x) = \arg \min_{u \in \mathbb{R}^n} \left[ \frac{\partial V(F(X, x, P(X, x)), x)}{\partial x_1} \cdots \frac{\partial V(F(X, x, P(X, x)), x)}{\partial x_n} \right] \cdot f(X, x, u).
\]

Therefore the feedback functions \( P(X, x) \) and \( p(X, x) \) can be designed as desired.
CHAPTER VIII

Applications and Examples

8.1 A Robot in a Two Dimensional Work Space

Consider a gantry robot in a two dimensional work space as shown in Figure 24.

![Diagram of a robot in a two-dimensional workspace]

Figure 24: A Robot in A Two Dimensional Work Space.

The robot may only stay in the white boxes shown in the figure. The system is
represented as a DSS as

\[ x(k + 1) = x(k) \oplus u(k) \]  

(8.1)

where

\[
\begin{bmatrix}
  x_1(k) \\
  x_2(k)
\end{bmatrix},
\begin{bmatrix}
  u_1(k) \\
  u_2(k)
\end{bmatrix},
\]

\[ x_i \in \{1, 2, \ldots, 8\}, \; u_i \in \{-1, 0, 1\}, \; i \in \{1, 2\}. \]

\[ x_1(k) \] and \[ x_2(k) \] are the East-West (E-W) and North-South (N-S) positions of the robot at time instant \( k \). \( u_1(k) \) and \( u_2(k) \) are the inputs to the robot such that

\[
\begin{align*}
  u_1(k) &= \begin{cases} 
  -1 & \text{means go W} \\
  0 & \text{means do not move E or W} \\
  1 & \text{means go E}
\end{cases} \\
  u_2(k) &= \begin{cases} 
  -1 & \text{means go S} \\
  0 & \text{means do not move N or S} \\
  1 & \text{means go N}
\end{cases}
\]

The operator \( \oplus \) is defined as

\[
x(k) \oplus u(k) = \begin{cases} 
  x(k) + u(k) & \text{if } x(k) + u(k) \text{ is valid} \\
  x(k) & \text{otherwise}
\end{cases}
\]

Let us choose a metric \( \rho(x, \tilde{x}) \) as

\[
\rho(\begin{bmatrix}
  x_1 \\
  x_2
\end{bmatrix}, \begin{bmatrix}
  \tilde{x}_1 \\
  \tilde{x}_2
\end{bmatrix}) = |x_1 - \tilde{x}_1| + |x_2 - \tilde{x}_2|. \]  

(8.2)
The task of the robot is to reach the box $x_0 = [4, 8]'$. Initially the robot is in any valid box.

To obtain a control for reaching a desired state $\ddot{x} = [\ddot{x}_1, \ddot{x}_2]'$ we simply define a Lyapunov function candidate as

$$V(x) = \rho(x, \ddot{x}). \tag{8.3}$$

Assuming there is no invalid box blocking the way between the desired state and the current state, we find the desired control by using the third Lyapunov condition as shown in 7.16

$$u(k) = \arg \min_{u \in \mathcal{U}} V(x(k + 1)) \tag{8.4}$$

$$= \arg \min_{u \in \mathcal{U}} V(x(k) + u(k)) \tag{8.5}$$

$$= \arg \min_{u \in \mathcal{U}} |x_1(k) + u_1(k) - \ddot{x}_1| + |x_2(k) + u_2(k) - \ddot{x}_2| \tag{8.6}$$

$$= -\text{sign}(x(k) - \ddot{x}). \tag{8.7}$$

Hence we obtain the best control to reach $\ddot{x}$ from an initial state.

Now let us choose the manifolds. In Figure 25, Region(1) through Region(6) are defined. We choose the manifolds as

$$\mathcal{M}_i = \bigcup_{k=i+1}^6 \text{Region}(k), \quad i \in \{0, 1, \cdots, 5\} \tag{8.8}$$

Thus if initially the state is in Region(1) (in $\mathcal{M}_0$), we need to reach $\mathcal{M}_1$ (Region(2)) first. After that the state eventually reaches $\mathcal{M}_2$, $\mathcal{M}_3$, $\mathcal{M}_4$ and $\mathcal{M}_5$. In $\mathcal{M}_5$
Figure 25: The Regions in The Two Dimensional Work Space.

(Region(6)) the state reaches $x_0$. We define the desired state for each region as a function $f(x(k)) = \begin{bmatrix} f_1(x_1(k), x_2(k)) \\ f_2(x_1(k), x_2(k)) \end{bmatrix}$ where

$$f(x) = \begin{cases} [3 \ 2]' & \text{if } x \text{ is in Region(1)} \\ [3 \ 4]' & \text{if } x \text{ is in Region(2)} \\ [6 \ 5]' & \text{if } x \text{ is in Region(3)} \\ [6 \ 7]' & \text{if } x \text{ is in Region(4)} \\ [4 \ 8]' & \text{if } x \text{ is in Region(5)} \\ [4 \ 8]' & \text{if } x \text{ is in Region(6)} \end{cases}$$

Thus if initially the state is in Region(1) we need to reach Region(2) first. After that the state eventually reaches other regions in the order. In Region(6) the state
needs to reach $x_0$. Then the control to perform the task is obtained using 8.7 as

$$u(k) = -\text{sign}(x(k) - f(x(k))).$$

(8.10)

Hence for each region the robot will go in the direction of the box specified by $f(x)$.

The system with this controller is shown in Figure 26.

Let us, as an example, choose the initial condition as $x(0) = [5 \ 2]'$. The state trajectory is shown in Figure 27. As we see the robot performs its task properly.

We see that for the closed loop system, the instability margin for the $\{x_0\}$ is zero and the system is asymptotically globally stable. It can furthermore be shown that the set of states of any trajectory of the closed loop system is 0-stable, which is a very strong type of stability property.
Now we consider what happens if the system has a disturbance $d(k) = [d_1(k) \ d_2(k)]'$ as

$$x(k + 1) = (x(k) \oplus u(k)) \oplus d(k)$$

(8.11)

where $d_1(k), d_2(k) \in \{-1, 0, 1\}$ is noise. If $d(k) = 0$ then the robot performs its task as before but if $d(k) \neq 0$ then the next state may not be as before. Let us consider a $d(k)$ as shown in Figure 28. The state trajectory of the disturbed system is shown in Figure 29. As we see it takes more steps to reach $x_0$ but the robot performs its task properly.
Figure 28: The noise $d(k)$.

Figure 29: The State Trajectory of the Disturbed Robot.
8.2 A Double Integrator Example

Consider a continuous state system as a double integrator given by

\[
\begin{align*}
\dot{x}_1 &= x_2, \\
\dot{x}_2 &= u,
\end{align*}
\tag{8.12}
\]

\[
\begin{align*}
\dot{x}_1 &= x_2, \\
\dot{x}_2 &= u,
\end{align*}
\tag{8.13}
\]

where \( u \) can take values from the set \( U = \{-1, 0, +1\} \).

The above continuous state system is to be controlled by a discrete state system through an interface as in Section 7.2. The state partition prescribed by the interface is shown in Figure 30.

![Figure 30: A State Partition for the Double Integrator.](image)

The corresponding nondeterministic discrete state system is

\[
X(k + 1) \in F(X(k), U(k))
\tag{8.14}
\]

where \( X(k) = [X_1(k) \ X_2(k)]' \) and \( U(k) \in U \).
As an example, we observe that

\[
 F(\begin{bmatrix} 2 \\ 2 \end{bmatrix}, -1) = \{ \begin{bmatrix} 2 \\ 1 \end{bmatrix} \},
\]

(8.15)

\[
 F(\begin{bmatrix} 2 \\ 2 \end{bmatrix}, 0) = \{ \begin{bmatrix} 2 \\ 2 \end{bmatrix} \},
\]

(8.16)

\[
 F(\begin{bmatrix} 2 \\ 2 \end{bmatrix}, +1) = \{ \begin{bmatrix} 2 \\ 2 \end{bmatrix} \}.
\]

(8.17)

As we see for \( X(k) = [2 \ 2]' \) the next state is deterministic (that is \( F \) has a single element). For the inputs 0 and +1 the state trajectory stays in the same discrete state all the time. If \( X(k) = [2 \ 2]' \) is not a desired state, a good control must use \( U(k) = -1 \) for this case to reach another discrete state.

Also we see that

\[
 F(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, -1) = \{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 0 \end{bmatrix} \},
\]

(8.18)

\[
 F(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, 0) = \{ \begin{bmatrix} 2 \\ 1 \end{bmatrix} \},
\]

(8.19)

\[
 F(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, +1) = \{ \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \end{bmatrix} \}.
\]

(8.20)

that is, for \( X(k) = [1 \ 1]' \) and the inputs -1 and +1 the next state is not deterministic.

Discrete state diagram for this system is shown in Figure 31.
Consider a controller of the form:

\[ U(k) = -\text{sign}(X_1(k) + X_2(k)). \] (8.21)

That is if \( X_1 + X_2 > 0 \) we apply \( u(t) = U(k) = -1 \) and if \( X_1 + X_2 < 0 \) we apply \( u(t) = U(k) = +1 \), when \( X_1 + X_2 = 0 \) we apply \( u(t) = U(k) = 0 \).

Now let us choose the initial condition of the continuous state system as \( x(0) = [1.5 \ 1.5]' \). The continuous state trajectory of the system is shown in Figure 32, and the corresponding discrete state trajectory is shown by the wider arrows in Figure 31.

Although no high frequency switchings occur in the discrete state system, the
sliding mode is in effect by definition along the lines shown in Figure 32 since many other trajectories end up reaching the same manifold.

If we change the controller as

\[
U(k) = \begin{cases} 
-1 & \text{for } X_1(k) + X_2(k) > 0 \\
1 & \text{for } X_1(k) + X_2(k) \leq 0 
\end{cases}
\]  

(8.22)

the continuous state trajectory becomes as shown in Figure 33. In this case we have high frequency switchings along the horizontal lines.

The corresponding discrete state trajectory is shown in Figure 34 and the high frequency switching states are indicated in shaded regions. In the corresponding equivalent system the shaded states with common sliding boundary are considered as one state. The resulting discrete state system is well defined and has the sliding manifold in the sense of the definitions introduced in Section 7.1.
Figure 33: The Continuous State Trajectory of the Hybrid System for a Different Control.

Note that the state trajectory reaches to state $X = [0 0]'$. Since we use the partitioning above, for a while the trajectory leaves the state $[0 0]'$, but the control
makes the trajectory reach it again. If we use a finer partitioning of the state space we can control the state trajectory more precisely.

8.3 A Robot with a Gripper

Consider the robot with the gripper shown in Figure 35.

![Figure 35: A robot with a gripper.](image)

The robot can go to positions 0, 1, 2, 3. It has two input variables. One of the input variables commands the robot to go to the left or right. The other one commands the gripper to hold or to release the object. The duty of the robot is to take the objects from position 0 to position 3 and to continue to do this job.

A discrete time hybrid system model for this robot can be constructed as follows. The hybrid system equations are

\[
X_1(k + 1) = U_1(k),
\]

\[
X_2(k + 1) = F_2(X_2(k), U_2(k), s(kT)),
\]
\[
\frac{d}{dt}x(t) = -ax(t) + aS(k),
\]

\[Y_1(k) = X_1(k),\]

\[Y_2(k) = X_2(k),\]

\[y(t) = x(t),\]

\[s(t) = x(t),\]

\[S(k) = Q(X_2(k), U_2(k)),\]

where

\[X_1(k) = \begin{cases}
    0 & \text{means the gripper holds,} \\
    1 & \text{the gripper does not hold,}
\end{cases}\]

\[U_1(k) = \begin{cases}
    0 & \text{means to hold,} \\
    1 & \text{means to release,}
\end{cases}\]

\[X_2(k) \text{ is the sampled position,} \]

\[U_2(k) = \begin{cases}
    0 & \text{means to go to the right,} \\
    1 & \text{means to go to the left,}
\end{cases}\]

\[x(t) \text{ is the real position,} \]

\[S(k) \text{ is the desired position.} \]

\[Q \text{ is given in Table 1, } a \text{ is a positive real constant, and,}\]

\[F_2(X_2, U_2, s) = \begin{cases}
    X_2 & \text{If } |Q(X_2, U_2) - s| > 0.05, \\
    Q(X_2, U_2) & \text{otherwise.}
\end{cases}\]

\[Q(X_2(k), U_2(k)) \text{ is the desired position of the robot while } s(kT) \text{ represents the real position. When } |Q(X_2(k), U_2(k)) - s(kT)| > 0.05 \text{ the robot is not called to be at the desired position, hence, } X_2 \text{ does not change. As soon as } |Q(X_2(k), U_2(k)) - s(kT)| \leq 0.05, \text{ we let } X_2(k + 1) = Q(X_2(k), U_2(k)).\]
The reachability and stabilizability of some sets for this hybrid system can be checked as follows. It is trivial to see that the set

\[ \left[ \begin{array}{c} \chi^b \\ \Omega^b \end{array} \right]_j \triangleq \left\{ \left[ \begin{array}{c} i \\ j \\ l \end{array} \right] ; i \in \{0,1\}, l \in [j-0.05, j+0.05] \right\} \]  \quad (8.23)

for any \( j \in \{0,1,2,3\} \) is reachable from the set.
And note that the sets \( \mathcal{X}^b \) and \( \Omega^b \) are stabilizable but \( \mathcal{X}^b \) and \( \Omega^b \) are not.

The set

\[
\begin{bmatrix}
\mathcal{X}^c \\
\Omega^c
\end{bmatrix} \triangleq \begin{bmatrix}
i \\
1 \\
l
\end{bmatrix} ; i \in \{0, 1\}, l \in [2.95, 3.05]
\]  \hspace{1cm} (8.25)

is not reachable from \( \begin{bmatrix}
\mathcal{X}^a \\
\Omega^a
\end{bmatrix} \) because if you choose \( X(0) = 0 \) and \( x(0) = 0 \), \( x(t) \) follows \( X_2(k) \) and always we get \( |x(t) - X_2(k)| \leq 1 \). So for \( x(t) \in [2.95, 3.05] \) we get \( 2 \leq X_2(k) \leq 3 \) but not \( X_2(k) = 1 \).

For these sets, it is not difficult to check the reachability and stabilizability conditions. This is not always the case. For example, checking if the set

\[
\begin{bmatrix}
\mathcal{X}^d \\
\Omega^d
\end{bmatrix} \triangleq \begin{bmatrix}
1 \\
2 \\
l
\end{bmatrix} ; l \in [2.37, 2.38]
\]  \hspace{1cm} (8.26)
is reachable from \[ \begin{bmatrix} X^a \\ \Omega^a \end{bmatrix} \] is not trivial and may require too many numerical calculations since the inputs to give to the system to reach the set cannot be simply obtained.

The implication is that sometimes it may be too difficult to check the reachability for hybrid systems.

The duty of the robot is to take the objects from position 0 to position 3. One possible controller having two states is

\[
\begin{align*}
X_3(k+1) &= F_3(X_3(k), U_3(k)), \\
Y_3(k) &= X_3(k), \\
Y_4(k) &= F_3(X_3(k), U_3(k)),
\end{align*}
\]

where

\[
X_3(k) = \begin{cases} 
0 & \text{robot goes to the right,} \\
1 & \text{robot goes to the left,}
\end{cases}
\]

and \( F_3 \) is given in Table 2.

The connection equations are

\[
\begin{align*}
U_3(k) &= Y_2(k), \\
U_2(k) &= Y_3(k), \\
U_1(k) &= Y_4(k).
\end{align*}
\]
The whole system is shown in Figure 36 and the closed loop hybrid system equations are given below.

\begin{align*}
X_1(k+1) &= F_3(X_3(k), X_2(k)), \\ 
X_2(k+1) &= F_2(X_2(k), x_3(k), s(kT)), \\ 
X_3(k+1) &= F_3(X_3(k), X_2(k)), \\ 
\frac{dx(t)}{dt} &= -ax(t) + aS(k), \\ 
Y_1(k) &= X_1(k),
\end{align*}

Table 2: $F_3(X_3, U_3)$

<table>
<thead>
<tr>
<th>$U_3$</th>
<th>$X_3$</th>
<th>$F_3(X_3, U_3)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Figure 36: Hybrid system model of the controlled robot with the gripper.

\[ Y_2(k) = X_2(k), \quad (8.38) \]
\[ y(t) = x(t), \quad (8.39) \]
\[ s(t) = x(t), \quad (8.40) \]
\[ S(k) = Q(X_2(k), X_3(k)). \quad (8.41) \]

Let us choose \( a = 0.4 \) and \( T = 1 \) with zero initial conditions. The output of the system is shown in Figure 37. As we see the robot performs as desired.
Figure 37: The outputs of the hybrid system when \( a = 0.4 \) with zero initial conditions.

8.4 Hybrid System Numerical Example

Consider the hybrid system given by equations

\[
X(t) = F(X(t^-), x(t^-)),
\]

\[
\frac{dx}{dt}(t) = f(X(t), x(t)),
\]

where \( x \in \mathbb{R}^2 \), \( X \in \{0, 1\} \), and,

\[
F(0, x) = \begin{cases} 
0 & x_2 \geq 0, \\
1 & x_2 < 0,
\end{cases} \quad F(1, x) = \begin{cases} 
0 & x_1 < 0, \\
1 & x_1 \geq 0,
\end{cases}
\]
The eigenvalues of $A_0$ and $A_1$ are given as

$$\sigma_{A_0} = \{-0.55 \pm j0.545\}, \quad \sigma_{A_1} = \{-0.025 \pm j1.358\}.$$  

The regions defined in Section 6.1 for this example are shown in Figure 38. The
hybrid system was simulated choosing a sampling time $T = 0.01$. The continuous state trajectories for different initial conditions are shown in Figure 39.

Discrete state switchings can be observed in the third quadrant of the continuous state space ($\Omega_{[10]}$). Actually, as claimed before, the dynamics in this region is obtained as the arithmetical average of the participating dynamics. Thus we have

$$\tilde{A} = (A_0 + A_1)/2 = \begin{bmatrix} 0.025 & -0.125 \\ 0.5 & -0.6 \end{bmatrix}. \quad (8.46)$$
The eigenvalues of the matrix \( \tilde{A} \) are \{\(-0.1, -0.475\)\}, so the behavior in the region \( \Omega_{\Omega_0} \) is as expected.

In the first quadrant of the continuous state space (\( \Omega_{\Omega_1} \)) the dynamics is chosen according to the discrete states. Due to the loops of the trajectory in this region, we cannot find any Lyapunov function to prove the stability of hybrid system. However, by the help of multiple Lyapunov functions (Theorem 6.3), we could prove that the hybrid system is actually stable. First Lyapunov functions are chosen as

\[
V_1(x) = 0.5x_1^2 + 0.425x_2^2, \\
V_2(x) = 0.5(x_1 - 0.12x_2)^2 + 0.4178x_2^2,
\]

which are both positive definite. Note also that for \( x_1 = 0 \) or \( x_2 = 0 \) (for the boundaries of the regions) we have \( V_1(x) = V_2(x) \).

The derivative of \( V_1 \) along the first dynamics \( (X = 0) \) and the derivative of \( V_2 \) along the second dynamics \( (X = 1) \) are

\[
\dot{V}_1(x) = -0.1(x_1 - 2.875x_2)^2 - 0.0234x_2^2, \\
\dot{V}_2(x) = -0.03(x_1 - 0.5167x_2)^2 - 0.012x_2^2,
\]

which are both negative definite. Thus it can be concluded that the hybrid system in this example is stable.
8.5 Modeling of a Car as a Hybrid System

A simple hybrid system model for a car as shown in Figure 40 can be given as

\[
X(t) = X(t^-) \oplus U(t^-) \triangleq \begin{cases} 
X(t^-) + U(t^-) & \text{if } X(t^-) + U(t^-) \in \{0,1,2\}, \\
X(t^-) & \text{otherwise}, 
\end{cases}
\]

(8.51)

\[
\frac{dx(t)}{dt} = \begin{cases} 
0 & \text{if } X(t) \in \{0,1\}, \\
u(t) & \text{if } X(t) = 2, 
\end{cases}
\]

(8.52)

where \( X \in \{0,1,2\}, x \in \mathbb{R}, U \in \{-1,0,1\}, \) and, \( u \in \mathbb{R}. \) The car hybrid system model has a discrete \( (X) \) and a continuous \( (x) \) state variable. \( X = 0 \) means the car stops, \( X = 1 \) means the car is running in idle condition, \( X = 2 \) means the car is ready to move. \( x \) is the position of the car on the road. In “Move” mode the car speed is controlled by the continuous input \( u. \) In “Stop” and “Idle” modes the car cannot move. The mode of the car (the discrete state \( X) \) can be changed using the discrete input \( U. \)

The objective is that for any initial position \( (x(0)) \) and any initial mode \( (X(0)) \) of the car, the car is to go to the position 0 \( (x = 0) \) and stop \( (X = 0). \) One possible
controller to achieve the goal is to choose

\[ U(t) = P(x(t)) = \begin{cases} 
1 & \text{if } x(t) \neq 0, \\
-1 & \text{if } x(t) = 0,
\end{cases} \quad (8.53) \]

\[ u(t) = p(x(t)) = -\text{sign}(x(t)). \quad (8.54) \]

![Figure 41: Closed Loop Hybrid System Model for the Car.](image)

The closed loop hybrid system with this controller is shown in Figure 41 and can be described by the equations

\[ X(t) = F(X(t^-), x(t^-)) = \begin{cases} 
X(t^-) \oplus 1 & \text{if } x(t^-) \neq 0, \\
X(t^-) \oplus (-1) & \text{if } x(t^-) = 0,
\end{cases} \quad (8.55) \]

\[ \frac{d}{dt} x(t) = f(X(t), x(t)) = \begin{cases} 
0 & \text{if } X(t) \in \{0, 1\}, \\
1 & \text{if } X(t) = 2. \end{cases} \quad (8.56) \]
As we see the closed loop system can also be represented as a hybrid system. The representative hybrid system for the above original closed loop hybrid system can be obtained as

\[
X(t) = \begin{cases} 
2 & \text{if } x(t^-) \neq 0, \\
0 & \text{if } x(t^-) = 0,
\end{cases} \quad (8.57)
\]

\[
\frac{d}{dt} x(t) = \begin{cases} 
-\text{sign}(x) & \text{if } x(t) \neq 0, \\
0 & \text{if } x(t) = 0.
\end{cases} \quad (8.58)
\]

The behavior of the closed loop hybrid system can be easily obtained by the help of the representative hybrid system above. If \(x(0) \neq 0\), that is the position of the car is not at zero, first \(X\) will become 2, that is the car will start to move. Then \(x\) will reach to 0 in finite time by the equation \(\dot{x}(t) = -\text{sign}(x(t))\). Finally at the position 0, \(X\) will become 0, that is the car will stop at the position 0. Thus the car with the given controller will perform as desired.

The closed loop hybrid system for this example is not actually stable since condition 6.12 is not satisfied because in any small compact neighborhood of the continuous state \(x = 0\) the discrete state jumps to \(X = 2\) and does not stay at \(X = 0\). Therefore a Lyapunov function as in Theorem 6.4 cannot be found. Note that if we choose \(V(0,0) = 0\), since \(V(X,x)\) should be continuous for \(x\), we have \(V(0,h) \approx 0\) for \(h \approx 0\). However since for \(X = 0\) and \(x = h\) the discrete state jumps to \(X = 2\) \(V\) need to decrease for a specific amount. This can not happen because we can choose \(h\) as
small as desired. Thus a Lyapunov function cannot actually be found.

On the other hand the origin is globally reachable since any initial hybrid state reach to the origin in finite time. Note that if we used the dynamic \( \dot{x}(t) = -x(t) \) for “Move” mode of the car (that is the feedback \( p(x(t)) = -x(t) \)) the continuous state would go to 0 but never reach 0 thus the discrete state would never be 0. Thus the objective of the controller would not be achieved for this case. The hybrid state \( \begin{bmatrix} 2 \\ 0 \end{bmatrix} \) can also be chosen to be the final state to be achieved, that is, the car may be desired to go to position 0 with “Move” mode is still active. For this purpose the feedback function \( P(x) \) can be modified. In this case the equilibrium state \( (x = 2, x = 0) \) will be globally asymptotically stable.
CHAPTER IX

Conclusion

In this thesis we provide models for discrete state and hybrid systems. We introduce the classical and refined analysis methods for both discrete and hybrid state space. We consider some control strategies to obtain a desirable behavior. Furthermore we provide some examples of discrete state and hybrid systems.

In Chapter II, dynamic systems with finite state space are introduced. First we provide a model for automata using the well studied adjacency matrix and output matrix. Then we introduce discrete state systems (DSS) which are automata in the vector form. DSS can be considered as a network of automata and like that of discrete time systems in $\mathbb{R}^n$. A discrete state system can also be seen as a single automaton. A one-to-one mapping is provided from a discrete state system to a single automaton.

In Chapter III, the models of hybrid systems are introduced. Hybrid systems are modeled as a combination of discrete and continuous state systems. A discrete time model is obtained by a discrete time discrete state system and either a discrete or
continuous time continuous state system. In this case the discrete state can change only at certain points on the time axis. In continuous time models, however, the discrete states can change any time according to discrete inputs or continuous states. Some properties like reachability and stabilizability of a set of hybrid states are also defined in this chapter.

In Chapter IV, we consider the discretization of continuous state systems. It is shown that partitioning the continuous state space produces a nondeterministic discrete state system, which is represented in the form of a deterministic system with unknown disturbances. The discrete state systems which are obtained from continuous state systems by discretization (or quantization) have a special property that the next discretized states may only be in the "neighboring" states. This type of discrete state systems are called neighbor-transition discrete state systems. The discretization is made by the help of an interface (or an A/D converter) which discretizes and encodes the continuous states to the discrete states. The issue of resolution is discussed for the discretized continuous state systems. Some mechanism is provided to obtain a lower resolution discrete state system from a higher one.

In Chapter V, we consider stability issues for discrete state systems and automata. The reachability and stabilizability of a set of states are defined for automata in the usual manner. The theorems to check these properties are provided using the reachability matrix which has been widely studied in the graph theory. We define classical
stability for discrete state systems. The main difficulty in introducing classical stability concepts for systems where the state space is a finite set is that the standard definitions of continuity and stability are either not clear or trivial. As demonstrated, the standard definition implies that all possible trajectories are stable. The global asymptotic stability, on the other hand, is not trivial for which some theorems based on Lyapunov theory are provided as in the literature \[57, 59\]. The only way to resolve the above contradiction of applying the classical stability theory to discrete state systems is in considering a class of systems with a more restrictive condition of trajectory continuity. For this purpose a refined Lagrange stability concept similar to \textit{practical stability} in the literature (see \[66, 50, 51, 60\]) is provided to fill the gap for dynamic systems with discrete state space. We provide the theorems to check refined Lagrange stability based on Lyapunov theory.

In Chapter VI, the hybrid system state space and various types of stability for hybrid systems are considered. The state space of the continuous state system portion is divided into \(N^N\) regions so that in every region depending on the discrete state of hybrid system the continuous state system dynamics to be considered are established. In hybrid systems infinite frequency discrete state switching in a region can occur. In this case we assume the function of the continuous dynamics is obtained as the arithmetical average of the original system functions participating in the switching. We use Filippov's construction [28] and apply and modify it to hybrid systems to find
out the state trajectory. We investigate the uncertain behavior occurring after the
infinite frequency discrete state switchings. We show that hybrid systems in general
are quite different than classical nonlinear systems so that special analysis methods
must be developed for hybrid systems. We provide a method for obtaining a repre­
sentative hybrid system from an original hybrid system by using the representative
functions in each region.

We define a classical stability concept of a continuous state in hybrid systems.
This approach may be useful especially if our goal is to reach an equilibrium point in
continuous state space and stabilize the system at that point regardless the discrete
states. Then following Lyapunov theory we provide some results on stability. As dis­
cussed, using a single Lyapunov function may not be sufficient to prove the stability
of a hybrid system. We investigate using multiple Lyapunov functions to prove the
stability. A hybrid state stability concept which considers the stabilization of both
the discrete and continuous states is also defined and studied. Classical Lyapunov
stability concept is applied to hybrid state stability. Conditions are found for hybrid
state stability of the origin. In global asymptotic stability both the discrete and con­
tinuous states is to go to the origin but the discrete state must reach 0 in finite time.
A Lyapunov theorem for global asymptotic stability is provided. Global reachability
of the origin is also defined. A special class of hybrid systems, such that the contin­
uous state system portion is modeled as a linear discrete time system, called linear
hybrid systems, is considered for stability. Some theorems for the global asymptotic stability of the origin of the continuous state space are provided by the help of a recent developed theory of stability of matrix sets.

In Chapter VII, we consider control design issues for discrete state and hybrid systems. The sliding mode control approach is developed for discrete state systems in a metric space. A concept of trajectory continuity as introduced by neighbor-transition dynamic systems with finite state space allows the use of sliding mode design techniques similar to that in continuous state systems. Discrete state system configuration permits us to consider integral manifolds analogous to the lower dimensional manifolds in continuous state space. This approach is then applied to a class of hybrid systems, which consist of a continuous plant, an interface and a discrete state system controller. The use of a sliding mode control strategy in the discrete state system controller allows us to obtain a robust closed loop system and therefore to eliminate the problems of a disturbance or a nondeterministic behavior. We show that interconnections of hybrid systems can be modeled as hybrid systems in our framework. This allows us to analyze the closed loop hybrid systems in the hybrid system framework considered. Control strategies are developed for hybrid systems based on Lyapunov functions. The Lyapunov theory developed for hybrid systems is applied to design a feedback control for a desired type of stability.

In Chapter VIII, we provide some examples of discrete state and hybrid systems.
First a robot in a two dimensional work space is modeled as a discrete state system. The sliding mode control found, similar to continuous case, can provide a closed loop system insensitive to strong disturbances. Second, we consider a double integrator and model it by a nondeterministic discrete state system by discretization. Then using a discrete state sliding mode controller we stabilize the system as desired. Then a robot with a gripper is modeled as a hybrid system and some hybrid system properties are investigated. Furthermore a discrete state controller is designed for the robot to obtain a desirable behavior. We provide a continuous time hybrid system numerical example and study hybrid system state space and stability concepts. Finally we model a car as a hybrid system and design a feedback to achieve the desired behavior. We investigate the closed loop hybrid system for stability analysis.

In Appendix A, we define and investigate stability of sets of matrices, and, apply the results to some control systems. Two types of stability are defined for sets, asymptotic stability and stabilizability. Asymptotic stability requires all possible products from the matrix set considered to be asymptotically stable, whereas in asymptotic stabilizability one particular sequence product of matrices is enough to be asymptotically stable. The upper and lower spectral radius of matrix sets are defined, which are the two possible generalizations of regular spectral radius. The upper and lower spectral radius give the highest and the lowest possible (normalized) spectral radius which can be obtained from the products of the matrices considered respectively.
Some necessary and sufficient conditions are provided for asymptotic stability and stabilizability using Lyapunov theory and special norms. Some properties of USR and LSR are provided. Some constructive methods are given to prove asymptotic stability using special Lyapunov functions of norms. It is also shown how to check asymptotic stability of sets using quadratic Lyapunov functions and the linear matrix inequalities. Some control problems are considered which can be solved by the theory of stability of matrix sets. It is shown that the theory of stability of matrix sets is helpful in the stability analysis and control design.

In future work, many other analysis and control design techniques developed in control and system science can be applied to discrete state and hybrid systems. The current models of discrete state and hybrid systems can be applied to many systems in real world and the methods developed can help solve the problems of stability analysis and control design. A more sophisticated modeling of real systems and control can be achieved by the help of the theory of hybrid systems.
Appendix A

Stability of a Set of Matrices

A.1 Introduction

In this chapter we consider a time varying discrete time system with state matrices which are chosen from a given set. This type of systems arise in many applications in control, for example, in linear feedback systems with communications delays [45], in hybrid systems [21, 64], in jump systems [49], in gain scheduling [68], and, in discrete time systems with time varying parametric uncertainty [43, 36]. Consider a system, for example, switching between two configurations represented by the trajectories given in Figure 42 (a) and (b), both trajectories are stable. Different trajectories are obtained by proper switching between those systems as shown in Figure 42 (c) and (d). The system in (c) becomes unstable although both the systems indicated by (a) and (b) are stable. Furthermore, the trajectory (d) obtained from (a) and (b) is much "more stable" than (a) and (b). In analyzing switching systems as above, we will encounter products of transition matrices. This provides the motivation for this
Figure 42: Two stable trajectories (a and b), and, an unstable (c) and a stable (d) trajectories obtained by switching between the systems a and b.

Although there are some recent results on the boundedness and convergence of matrix products from finite sets [15, 5, 46], the problem of deciding stability for some discrete event systems is shown to be unsolvable [77]. This result suggests that a simple algorithm is not likely to be obtained for set stability. Some conservative approaches need to be considered to decide stability. In the literature [65, 15] the
joint spectral radius (JSR) (which is a generalization of spectral radius in the usual sense) is defined so that asymptotic stability of given matrices depends on whether JSR is less than one. In this chapter, an upper (USR) and lower (LSR) spectral radius are defined for a given set of matrices. The USR is the same variable called JSR in the literature. USR gives the largest (normalized) spectral radius obtained from a set of matrices. One may also be interested in the lowest (normalized) spectral radius obtained from a set of matrices (LSR).

Returning to the example in Figure 42, the “rate of stability” of the system in (c) and (d) refer to USR and LSR respectively. That is, USR quantifies how unstable a system switching between a set of dynamics can be made, while LSR quantifies how stable the system can be made.

As a physical illustration, the configurations in Figure 42 (a) and (b) can also be thought as the economic recovery plans of two parties in a country. Effect of election of the parties on the resulting economy is also answered by USR and LSR. As seen from Figure 42 (c), although both parties may have good economic recovery plans, switching between the parties may result in worse and worse situations in that country.
A.2 The Problem Definition

Consider a time varying discrete time system

\[ x(k + 1) = A(k)x(k), \quad k = 0, 1, 2, \ldots \]  \hspace{1cm} (A.1)

where \( A(k) \in \mathcal{A} = \{A_1, A_2, \ldots, A_N\}, \ A_i \in \mathbb{R}^{n \times n} \). Assume that in the discrete time system above, state matrices are chosen from a certain set, \( \mathcal{A} \). The stability definitions are given below.

**Definition A.1** A set of matrices, \( \mathcal{A} \), is called asymptotically stable if discrete time system (A.1) is asymptotically stable for all possible sequences of matrices in \( \mathcal{A} \).

**Definition A.2** A set of matrices, \( \mathcal{A} \), is called asymptotically stabilizable if discrete time system (A.1) is asymptotically stable for a sequence of matrices in \( \mathcal{A} \).

A matrix \( A \) is called asymptotically stable if all the eigenvalues of \( A \) are inside the unit disc. For asymptotic stability all the products from a set of matrices are required to be asymptotically stable. However, for asymptotic stabilizability, one particular sequence of products is enough to be asymptotically stable. If \( \mathcal{A} \) is asymptotically stable, then system (A.1) is asymptotically stable for all possible combination of matrices. This property may be desirable for many systems. Asymptotic stabilizability of \( \mathcal{A} \) means that, by using a proper sequence of matrices from \( \mathcal{A} \), system (A.1) can be made asymptotically stable. This property may be useful for designing a control for system
A.1. If $\mathcal{A}$ is not asymptotically stabilizable, then system A.1 is not asymptotically stable for any sequence of matrices from $\mathcal{A}$.

The regular spectral radius $\rho(\cdot)$ of a matrix $A$ is defined as ([37]):

$$\rho(A) = \max \{ |\lambda| \mid \lambda \text{ is an eigenvalue of } A \} , \quad (A.2)$$

where $|\lambda|$ indicates the absolute value of the complex number $\lambda$.

The spectral radius has the following properties ([37]):

$$\rho(A) = \lim_{k \to \infty} \|A^k\|^{1/k} , \quad (A.3)$$
$$\rho(A) = \inf \{ \|A\| \mid \| \cdot \| \text{ is a matrix norm} \} , \quad (A.4)$$

where $\| \cdot \|$ is any matrix norm. If a matrix $A$ is asymptotically stable we have $\rho(A) < 1$.

The definitions for the upper (USR) and lower (LSR) spectral radius of a set of matrices are given below. Let $\mathcal{A}/\rho$ denote the set $\{ \frac{1}{\rho} A_1, \frac{1}{\rho} A_2, \cdots, \frac{1}{\rho} A_N \}$.

**Definition A.3** The upper spectral radius (USR)\(^1\) of a set $\mathcal{A}$, $\overline{\rho}(\mathcal{A})$, is defined as

$$\overline{\rho}(\mathcal{A}) \triangleq \inf \{ \rho > 0 \mid \mathcal{A}/\rho \text{ is asymptotically stable} \} . \quad (A.5)$$

\(^1\)USR is known as joint spectral radius (JSR) in the literature [65, 15, 33]. Here this term is used to make it match with "lower spectral radius."
Definition A.4 The lower spectral radius (LSR) of a set \( \mathcal{A} \), \( \rho(\mathcal{A}) \), is defined as

\[
\rho(\mathcal{A}) \triangleq \inf \{ \rho > 0 \mid \mathcal{A}/\rho \text{ is asymptotically stabilizable.} \} \tag{A.6}
\]

In the definitions above, USR and LSR give the highest and the lowest possible (normalized) spectral radii which can be obtained from the products of the matrices from the set \( \mathcal{A} \) respectively. Here it is easy to see that \( \mathcal{A} \) is asymptotically stable if \( \overline{\rho}(\mathcal{A}) < 1 \) and \( \mathcal{A} \) is asymptotically stabilizable if \( \underline{\rho}(\mathcal{A}) < 1 \).

The joint spectral radius (or the upper spectral radius in our terms) can also be defined using any matrix norm \( \| \cdot \| \) ([65, 15, 34]) as

\[
\rho(\mathcal{A}) = \limsup_{k \to \infty} (\hat{\rho}_k(\mathcal{A}, \| \cdot \|))^{1/k}, \tag{A.7}
\]

where

\[
\hat{\rho}_k(\mathcal{A}, \| \cdot \|) \triangleq \sup \{ \| \prod_{i=1}^k M_i \| \mid M_i \in \mathcal{A} \}. \tag{A.8}
\]

\( \hat{\rho}_k \) gives the maximum norm of the all possible matrix products with length \( k \) from the set \( \mathcal{A} \). Another definition for the upper spectral radius using the spectral radius of matrix products is given ([15, 5]) as

\[
\rho(\mathcal{A}) = \limsup_{k \to \infty} (\overline{\rho}_k(\mathcal{A}))^{1/k}, \tag{A.9}
\]

where

\[
\overline{\rho}_k(\mathcal{A}) \triangleq \sup \{ \rho(\prod_{i=1}^k M_i) \mid M_i \in \mathcal{A} \}. \tag{A.10}
\]
\( \bar{\rho}_k \) is the maximum spectral radius of the all possible matrix products with length \( k \) from the set \( \mathcal{A} \). Since the matrix products with length \( k \) are used, the \( k \)th root of the resulting number is taken and the upper spectral radius is obtained by searching for larger matrix products as given in A.9.

In [5], it is shown that definitions given in A.7 and A.9 are indeed the same for any bounded set of matrices. The equations in A.7 and A.9 are not necessarily equal, however, for some unbounded infinite sets of matrices as shown in [15]. Here only bounded sets of matrices are considered.

Similarly for lower spectral radius, the lowest possible spectral radius needs to be obtained from the set \( \mathcal{A} \) as

\[
\rho(\mathcal{A}) = \lim_{k \to \infty} \inf (\bar{\rho}_k(\mathcal{A}))^{1/k},
\]  

(A.11)

where

\[
\bar{\rho}_k(\mathcal{A}) \triangleq \inf \{ \rho(\prod_{i=1}^{k} M_i) \mid M_i \in \mathcal{A} \}.
\]  

(A.12)

\( \rho_k \) is the minimum spectral radius of the all possible matrix products with length \( k \) from the set \( \mathcal{A} \).

### A.3 Some Results for Stability of Sets

Equations A.9 and A.11 provide a natural way to compute the upper and lower spectral radius. For this, all possible products of matrices from \( \mathcal{A} \) with a certain
length are considered. Then the maximum (or minimum) spectral radius of the matrices from these products is obtained, \( \bar{\rho}_k(A) \) (\( \lambda_k(A) \)). Since \( k \) products are used in this case, the \( k \)th root of the resulting number is taken. This gives a lower (or upper) bound for USR (or LSR). The larger the number of matrices in the products, the more accurate the desired number is approximated. This actually is a good algorithm for calculating USR or LSR. The problem, however, is that the maximum (or minimum) may be achieved in arbitrarily large number of products\(^2\). Thus, we may never be certain that by considering a finite length of products the actual USR (or LSR) is achieved.

Following Lyapunov theory (for example see [54]) the two theorems below can be given.

**Theorem A.1** \( A \) is asymptotically stable iff there exists a continuous scalar function \( V(x) : \mathbb{R}^n \rightarrow \mathbb{R} \) of the state \( x \) of system \( A.1 \), with continuous first order partial derivatives, such that

(i) \( V(x) \) is positive definite,

(ii) \( V(x) > V(A_i x) \), \( \forall x \neq 0, \forall i \in [1 : N], \)

\(^2\)Recently, it has been shown in [77] that some stability problems for discrete event systems are algorithmically unsolvable. Based on this result, it is rather unlikely to find a simple algorithm for the above set stability problem in the general case. Therefore more conservative approaches need to be considered to decide on asymptotic stability.
(iii) $V(x) \to \infty$ as $\|x\| \to \infty$.

**Proof:** Sufficiency is obvious from Lyapunov theory of time variant discrete time systems. For necessity consider a Lyapunov function candidate as

$$V(x) = \sup \{\|x\| + \sum_{i=0}^{\infty} \|A(i)A(i-1)\cdots A(0)x\| \mid A(\cdot) \in \mathcal{A}\}.$$  \hfill (A.13)

We first need to show that $V(x)$ given above is well defined. Note that since $\mathcal{A}$ is asymptotically stable, given $x$, it is possible to find a $K > 0$ and $0 < \rho < 1$ such that

$$\|A(i)A(i-1)\cdots A(0)x\| < K\rho^i, \forall A(\cdot) \in \mathcal{A}.$$  \hfill (A.14)

Then we obtain

$$V(x) < \|x\| + K\sum_{i=0}^{\infty} \rho^i = \|x\| + \frac{K}{1-\rho}.$$  \hfill (A.15)

Since the last quantity in the above inequality is finite, $V(x)$ is well defined. $V(x)$ already satisfies conditions (i) and (iii). From A.13 we obtain

$$V(x) = \max \{V(A_1x), V(A_2x), \cdots, V(A_Nx)\} + \|x\|,$$  \hfill (A.16)

which shows that the condition (ii) is also satisfied. \hfill \□

**Theorem A.2** $\mathcal{A}$ is asymptotically stabilizable iff there exists a continuous scalar function $V(x): \mathbb{R}^n \to \mathbb{R}$ of the state $x$ of system A.1, with continuous first order partial derivatives, such that

(i) $V(x)$ is positive definite,
(ii) $V(x) > V(A_i x)$, $\forall x \neq 0 \exists i \in [1 : N]$,

(iii) $V(x) \to \infty$ as $\|x\| \to \infty$.

**Proof:** If $\mathcal{A}$ is asymptotically stabilizable then there exists an asymptotically stable discrete time system as in A.42 from Observation A.1. Thus there must be a Lyapunov function as given above. If there is such a Lyapunov function, which is guaranteed to be strictly decreasing in time for at least one sequence, then $\mathcal{A}$ is asymptotically stabilizable. \(\square\)

In the literature [65, 5], the norm condition stated below is given for stability.

**Theorem A.3** $\mathcal{A}$ is asymptotically stable iff there exist a norm $\| \cdot \|_\xi$ such that

$$
\|A_i\|_\xi < 1, \forall A_i \in \mathcal{A}.
$$

(A.17)

**Proof:** If $\mathcal{A}$ is asymptotically stable then according to Theorem A.1 there exists a Lyapunov function as given in A.13. It is easy to see that $V(x)$ satisfies norm conditions. Thus we can choose $\|x\|_\xi = V(x)$ as defined in A.13. Since $V(x) > V(A_i x)$ for all $i \in [1 : N],$

$$
\|A_i\|_\xi = \max_x \frac{\|A_i x\|_\xi}{\|x\|_\xi} = \max_x \frac{V(A_i x)}{V(x)} < 1, \forall i \in [1 : N],
$$

(A.18)

as stated in A.17.
If there is such a norm then we choose $V(x) = \|x\|$; then, the conditions (i) and (iii) of Theorem A.1 are already satisfied. Note that

$$\max_{x} \frac{V(A_ix)}{V(x)} = \max_{x} \frac{\|A_ix\|}{\|x\|} = \|A_i\| < 1, \ \forall i \in [1:N].$$

(A.19)

From the above equation we obtain

$$V(x) > V(A_ix), \ \forall x \neq 0, \ \forall i \in [1:N],$$

(A.20)

which is the condition (ii) of Theorem A.1. Thus $A$ is asymptotically stable. □

Denote the norm of a set of matrices $\|A\|$ as $\sup \{\|A\| \mid A \in \mathcal{A}\}$. Theorem A.3 leads to (also see [65, 5]) :

Corollary A.1 The upper spectral radius has the following property

$$\bar{\rho}(\mathcal{A}) = \inf \{\|A\| \mid \| \cdot \| \text{ is a matrix norm}\}.$$

Proof: According to Definition A.3, $A/\rho$ is just marginally stable when $\rho = \bar{\rho}(\mathcal{A})$.

Then, according to Theorem A.3, we obtain

$$\inf \{\max\{\frac{1}{\rho}\|A_1\|, \frac{1}{\rho}\|A_2\|, \ldots, \frac{1}{\rho}\|A_N\|\} \mid \| \cdot \| \text{ is a matrix norm}\} = 1.$$  

(A.21)

Due to the linear multiplicativity of norms we conclude with the above corollary. □

A.4 Some Properties of USR and LSR

It is straightforward that for diagonal or upper-triangular matrices $\bar{\rho}(\mathcal{A}) = \max_i \rho(A_i)$. In [33] the following result is also reported.
Lemma A.1 [33] If $A_1$ and $A_2$ are simultaneously upper-triangularized or simultaneously Hermitianized then $\rho(\{A_1, A_2\}) = \max\{\rho(A_1), \rho(A_2)\}$.

Denote the following as

$$
A^k \triangleq \{ \prod_{i=1}^{k} M_i \mid M_i \in \mathcal{A} \}, \tag{A.22}
$$

$$
T_1 A T_2 \triangleq \{ T_1 A T_2 \mid A \in \mathcal{A} \}, \quad T_1 \in \mathbb{C}^{m \times n}, \quad T_2 \in \mathbb{C}^{n \times p}, \tag{A.23}
$$

$$
\langle \mathcal{A} \rangle \triangleq \{ \sum_{i=1}^{N} p_i A_i \mid \sum_{i=1}^{N} |p_i| \leq 1 \}. \tag{A.24}
$$

Some of the properties of the upper and lower spectral radius are given as follows.

Lemma A.2 Consider sets of $(n \times n)$ matrices $\mathcal{A}$ and $\mathcal{B}$. The following properties are valid.

1. $\bar{\rho}(\{A\}) = \underline{\rho}(\{A\}) = \rho(A)$,

2. $\bar{\rho}(\mathcal{A}) \geq \rho(\mathcal{A})$,

3. $\bar{\rho}(\mathcal{A}) \geq \max_i \rho(A_i)$, $\underline{\rho}(\mathcal{A}) \leq \min_i \rho(A_i)$,

4. $\bar{\rho}(a\mathcal{A}) = |a|\bar{\rho}(\mathcal{A})$, $\underline{\rho}(a\mathcal{A}) = |a|\underline{\rho}(\mathcal{A})$,

5. $\bar{\rho}(T\mathcal{A}T^{-1}) = \bar{\rho}(\mathcal{A})$, $\underline{\rho}(T\mathcal{A}T^{-1}) = \underline{\rho}(\mathcal{A})$, where $T \in \mathbb{C}^{n \times n}$,

6. $\bar{\rho}(\mathcal{A}^k) = \bar{\rho}(\mathcal{A})^k$, $\underline{\rho}(\mathcal{A}^k) = \underline{\rho}(\mathcal{A})^k$,

7. $\bar{\rho}(\mathcal{A}^k \cup \mathcal{A}^l) = \bar{\rho}(\mathcal{A}^k)$, $\underline{\rho}(\mathcal{A}^k \cup \mathcal{A}^l) = \underline{\rho}(\mathcal{A}^k)$, where $k \geq l$. 

8. \( p(A) \geq p(B) \), \( p(A) \leq p(B) \), where \( A \supset B \),

9. \( p(A \cup B) \geq \max\{p(A), p(B)\} \), \( p(A \cup B) \leq \min\{p(A), p(B)\} \),

10. \( p(A \cap B) \leq \min\{p(A), p(B)\} \), \( p(A \cap B) \geq \max\{p(A), p(B)\} \),

11. \( p(A \cup B) = p(A) \), where \( B \subset \langle A \rangle \),

12. \( p(\langle A \rangle) = p(A) \),

13. \( p(A \cup \{I\}) = \max\{1, p(A)\} \), \( p(A \cup \{I\}) = \min\{1, p(A)\} \).

**Proof:** Many of the above properties are obvious. We provide some of the proofs below.

6. Suppose a periodic sequence of matrices \( M_1M_2 \cdots M_i \) gives the highest (normalized) spectral radius as in A.9 and A.10. Then for the set \( \langle A \rangle \) the sequence \( (M_1M_2 \cdots M_i)^k \) will give the highest (normalized) spectral radius. Since we use \( k \) times the matrices as in \( A \) we take the \( k \)th root and obtain the result.

11. Consider a set \( \tilde{A} = A \cup \{\tilde{A}\} \) where \( \tilde{A} = \sum_{i=1}^{N} p_i A_i \) and \( \sum_{i=1}^{N} |p_i| \leq 1 \). When calculating the USR of \( \tilde{A} \) we consider the matrix multiplications from \( \tilde{A} \). However the addition of the matrix \( \tilde{A} \) does not increase or decrease the USR of \( A \). It is obvious that the addition of \( \tilde{A} \) will not decrease the USR (the property 9 above). It will also not increase the USR of \( A \) because when \( \tilde{A} \) is multiplied
with the other matrices as

\[ M_1 M_2 \cdots \tilde{A} \cdots M_l \]

then this will equal to

\[ \sum_{i=1}^{N} p_i M_1 M_2 \cdots A_i \cdots M_l. \]

Since \( \sum_{i=1}^{N} |p_i| \leq 1 \) the norm (or spectral radius) of the above product cannot exceed the maximum norm (or spectral radius) of the individual sequences in the above sum which are available from the set \( \mathcal{A} \). In this way we can add any matrices from \( \langle \mathcal{A} \rangle \) to \( \mathcal{A} \) and find that the USR of the resulting set and the USR of \( \mathcal{A} \) are indeed equal. Property 12 also follows in similar way. □

### A.5 Constructive Methods for Asymptotic Stability

Theorem A.1 states that there must be a Lyapunov function to prove asymptotic stability of \( \mathcal{A} \). However it may be very difficult to find an appropriate Lyapunov function or norm to prove asymptotic stability or stabilizability for some cases. Below a theorem is provided for constructing a suitable Lyapunov function using a special norm. The approach uses the concept of expansion developed by Siljak and others [71, 38].
Theorem A.4 Consider a full rank expansion matrix $\tilde{V} \in \mathbb{C}^{m \times l}$, $l \geq n$. If there exist matrices $Y_i \in \mathbb{C}^{l \times 1}, i \in [1 : N]$ such that

$$A_i \tilde{V} = \tilde{V} Y_i, \ i \in [1 : N],$$

(A.25)

with $\|Y_i\| < 1$ ($\| \cdot \|$ is any fixed norm) then $A$ is asymptotically stable.

Proof: A Lyapunov function can be constructed using $\tilde{V}$ as

$$V(x) = \min \{ \|y\| \mid \tilde{V} y = x, \ y \in \mathbb{C}^l \},$$

(A.26)

which is also a norm. A.26 satisfies the conditions (i) and (iii) of Theorem A.1. In the above Lyapunov function, when the expansion matrix is real and the 1-norm is considered, the unit value surface, that is the surface $V(x) = 1$ in state space of $x$, becomes a polygon with corners as the column vectors and the negative column vectors of $\tilde{V}$. For complex expansion matrices a larger class of Lyapunov functions can be obtained. $V(x)$ also have a multiplicative property, that is, $V(ax) = |a|V(x)$. That is why we only need to prove that the dynamics in $A$ map the unit value surface strictly inside. For this, consider a point $x$ on the unit value surface. According to A.26 we have a $\tilde{y} \in \mathbb{C}^l$ such that $\tilde{V} \tilde{y} = x$ and $\|\tilde{y}\| = 1$. Let us consider the map of $x$ on $A_i$,

$$V(A_i x) = \min \{ \|y\| \mid \tilde{V} y = A_i x, \ y \in \mathbb{C}^l \}.$$  \hspace{1cm} (A.27)

Since $x = \tilde{V} \tilde{y}$ we have $\tilde{V} y = A_i \tilde{V} \tilde{y}$, or from A.25 we obtain $\tilde{V} y = \tilde{V} Y_i \tilde{y}$. Since $\|Y_i\| < 1$ we can choose $y = Y_i \tilde{y}$ and obtain $\|y\| = \|Y_i \tilde{y}\| < 1$. This proves that
$V(A_i x) < V(x)$ which is the condition (ii) of Theorem A.1. Thus it can be concluded that $A$ is asymptotically stable.

In the above theorem a method is provided for proving asymptotic stability using $l$ vectors. In general, for an asymptotically stable set of matrices it is enough to consider a finite number of vectors. However, for some cases an arbitrarily large number of vectors may be needed to prove stability. An upper bound for USR can be given using the above theorem as follows.

**Corollary A.2** Given a full rank expansion matrix $\tilde{V} \in \mathbb{C}^{n \times l}$, $l \geq n$,\n
$$\bar{\rho}(A) \leq \max_i \min\{\|Y\| \mid A_i \tilde{V} = \tilde{V} Y, \ Y \in \mathbb{C}^{l \times l}\}. \quad (A.28)$$

**Proof:** Following Theorem A.4 we can say that if

$$\max_i \min\{\|Y\| \mid A_i \tilde{V} = \tilde{V} Y, \ Y \in \mathbb{C}^{l \times l}\} < 1, \quad (A.29)$$

then $A$ is asymptotically stable. The corollary follows due to the linear multiplicativity of the upper spectral radius. \hfill \Box

One may use a linear programming algorithm to obtain the value in A.28. Consider a quadratic type of Lyapunov function $V(x) = x^T P x$ which leads to the following.

**Corollary A.3** If there exist positive definite matrices $P, Q_1, Q_2, \ldots, Q_N \in \mathbb{R}^{n \times n}$ such that

$$A_i^T P A_i - P = -Q_i, \ \forall A_i \in A, \quad (A.30)$$

then $A$ is asymptotically stable.
Proof: Choose a quadratic Lyapunov function as \( V(x) = x^T P x \) then the corollary follows from Lyapunov theory of linear systems.

The condition of Corollary A.3 is not actually necessary to prove asymptotic stability of sets. To show this consider a set \( \{Q, W\} \) where

\[
Q = \begin{bmatrix}
0.25 & 0.75 \\
0.05 & 0.75
\end{bmatrix}, \quad W = \begin{bmatrix}
0.4 & -0.8 \\
1.3 & -0.8
\end{bmatrix}.
\]  \hspace{1cm} (A.31)

There is no \( P \) which satisfies condition A.30 for \( \{Q, W\} \) although it is asymptotically stable. Actually, if the second power of the set \( \{Q, W\} \) \( (\{Q, W\}^2 = \{QQ, QW, WQ, WW\}) \) is considered, choosing

\[
P = \begin{bmatrix}
0.5 & -0.3 \\
-0.3 & 0.5
\end{bmatrix},
\]

condition A.30 is satisfied for \( \{Q, W\}^2 \), thus \( \{Q, W\}^2 \) is asymptotically stable. Since \( \overline{\rho}(\{Q, W\}) = \overline{\rho}(\{Q, W\}^2)^{1/2} < 1 \) (see Property 6 of Lemma A.2), \( \{Q, W\} \) is indeed asymptotically stable although there is no quadratic Lyapunov function to prove it.

Another method for checking asymptotic stability of \( \mathcal{A} \) is given below.

**Corollary A.4** \( \mathcal{A} \) is asymptotically stable iff there exist a positive definite matrix \( P \in \mathbb{R}^{n \times n} \) such that

\[
A^T PA - P < 0, \quad \forall A \in \mathcal{A}^k, \quad \text{for some} \quad k \geq 1,
\]  \hspace{1cm} (A.32)

where the inequality is in the sense of negative definiteness.
**Proof:** If for some $k$ the above inequality holds then $A^k$ is asymptotically stable, or, $\bar{\rho}(A^k) < 1$. Since $\bar{\rho}(A) = \bar{\rho}(A^k)^{1/k}$ we conclude that $\bar{\rho}(A) < 1$, or, $A$ is asymptotically stable. If $A$ is asymptotically stable, 2 norm of the products of matrices in $A$ must decrease after a certain finite length of products. Thus a finite $k$ can be found to obtain a $P$ which satisfy A.32.

As an example there is no quadratic function to prove asymptotic stability of the set $\{1.2Q, W\}^k$ for $k = 1, 2, 3, \text{ and, } 4$. However for $\{1.2Q, W\}^5$,

$$P = \begin{bmatrix} 0.525 & -0.3 \\ -0.3 & 0.475 \end{bmatrix}$$

satisfy A.32, thus it can be concluded that $\{1.2Q, W\}$ is asymptotically stable. The set $\{1.21Q, W\}$ on the other hand is not asymptotically stable because we have $\rho((1.21Q)(1.21Q)WW) = 1.0029 > 1$ although both $1.21Q$ and $W$ are asymptotically stable.

Given $k$, the condition A.32 can be checked for some $P$ by using linear matrix inequalities (LMI) (see [7, 52]). Consider a set $A$ (or $A^k$). The condition A.30 is to be checked for a $P$. First note that if $A_i \in A$ is not asymptotically stable it can be directly concluded that $A$ is not asymptotically stable. Thus the assumption can be made that all the matrices in $A$ are asymptotically stable. Then from Lyapunov theory of linear systems the following is obtained,

$$P = \sum_{k=0}^{\infty} A_i^{kT}Q_i A_i^k, \quad \forall i \in [1 : N].$$ (A.33)
It is a fact that any positive definite matrix \((n \times n)\) can be written as an affine combination of \(\tilde{n} = n(n + 1)/2\) linearly independent positive (semi) definite matrices. Choose any \(\tilde{n}\) linearly independent positive (semi) definite \((n \times n)\) matrices \(Q_{1,1}, Q_{1,2}, \ldots, Q_{1,\tilde{n}}\) for \(Q_1\). Now \(Q_1\) can be represented as

\[
Q_1 = \sum_{j=1}^{\tilde{n}} w_j Q_{1,j} \tag{A.34}
\]

by using some positive or negative (or zero) \(w_j\)'s.

Obtain \(Q_{i,j}, \ i \in [2 : N], \ j \in [1 : \tilde{n}]\) by solving the Lyapunov equation

\[
A_i^T P_j A_i - P_j = -Q_{1,j} \tag{A.35}
\]

for \(P_j\) (since \(A_1\) is asymptotically stable this is possible), and, choosing

\[
Q_{i,j} = A_i^T P_j A_i - P_j. \tag{A.36}
\]

Then choose \(Q_i, \ i \in [2 : N]\) as

\[
Q_i = \sum_{j=1}^{\tilde{n}} w_j Q_{i,j}. \tag{A.37}
\]

Thus the equation A.33 is automatically satisfied for all \(i \in [1 : N]\) as

\[
\sum_{k=0}^{\infty} A_i^{kT} Q_i A_i^k = \sum_{j=1}^{\tilde{n}} w_j \sum_{k=0}^{\infty} A_i^{kT} Q_{i,j} A_i^k = P, \quad \forall i \in [1 : N]. \tag{A.38}
\]

Now the only condition needed is the positive definiteness of \(Q_i\)'s, that is,

\[
\sum_{j=1}^{\tilde{n}} w_j Q_{i,j} > 0, \quad \forall i \in [1 : N]. \tag{A.39}
\]
Then, since all $A_i$'s are asymptotically stable matrices, $P$ must be a positive definite matrix from Lyapunov theory. The condition A.39 can also be written as

$$\sum_{j=1}^{\bar{n}} w_j \tilde{Q}_j > 0,$$  (A.40)

where

$$\tilde{Q}_j \triangleq \begin{bmatrix} Q_{1,j} & 0 \\ 0 & Q_{2,j} \\ \vdots \\ 0 & Q_{N,j} \end{bmatrix}.$$  (A.41)

If there are (positive or negative) numbers $w_j$, $j \in [1 : \bar{n}]$ such that A.40 is satisfied then it can be concluded that $A$ is asymptotically stable. Condition A.40 can be easily checked by using linear matrix inequalities (see [52]). This gives another method for checking asymptotic stability of sets of matrices using Corollary A.4 and the LMI approach.

### A.6 Applications to Control

In system A.1 the state matrix switches to different values in set $A$ in no particular way. These kinds of systems are guaranteed to be asymptotically stable only if $A$ is asymptotically stable as defined. On the other hand, if the jump sequence can be chosen, then we can have an asymptotically stable system only if $A$ is asymptotically stabilizable as defined.
Consider another type of system

\[ x(k + 1) = A(x(k))x(k), \quad k = 0, 1, 2, \ldots \]  

(A.42)

where \( A(x(k)) \in \mathcal{A} = \{A_1, A_2, \ldots, A_N\}, \ A_i \in \mathbb{R}^{n \times n} \). In this case, the matrices are not chosen according to time but according to the state of the system. Asymptotic stability and stabilizability can also be defined for the above system. In this case the stability is considered not only for all sequences of matrices but for all designs of function \( A(x) : \mathbb{R}^n \rightarrow \mathcal{A} \). However, even for this case, the following is observed.

Observation A.1 Asymptotic stability (and stabilizability) of systems A.1 and A.42 with matrix set \( \mathcal{A} \) are equivalent.

\textbf{Proof:} If system A.1 is not asymptotically stable then there is a sequence of matrices in \( \mathcal{A} \) which is not asymptotically stable. That is, there is an initial state and a trajectory which is not asymptotically stable. Thus, using this trajectory we can design a system as in A.42 such that system A.42 is not asymptotically stable.

If system A.1 is asymptotically stable then for all sequence of matrices in \( \mathcal{A} \) all the trajectories are asymptotically stable. Thus system A.42 must be asymptotically stable. Similar conclusions can be made for asymptotic stabilizability. \( \Box \)

Hence asymptotic stability of \( \mathcal{A} \) needs to be checked to decide on asymptotic stability of system A.42. Also the answer of whether there is a design (or feedback) \( A(x) \) such that system A.42 could be asymptotically stabilizable depends directly
on asymptotic stabilizability of $\mathcal{A}$. It can also be observed that the upper (and lower) spectral radius of systems A.1 and A.42 are the same because of the linear multiplicativity.

Consider a system with discrete inputs

$$x(k + 1) = A_{u(k)}x(k), \quad k = 0, 1, 2, \cdots$$

(A.43)

where $A_u \in \mathcal{A} = \{A_1, A_2, \ldots, A_N\}$, $A_i \in \mathbb{R}^{n \times n}$, $u \in [1 : N]$. When $u(k)$ is chosen independently not in a particular way A.43 represents system A.1. System A.42, however, can be thought as one with feedback, $g$,

$$u(k) = g(x(k)),$$

(A.44)

without a particular way. Here we can say that system A.43 is asymptotically stable if and only if $\mathcal{A}$ is asymptotically stable. The best performance (in terms of stability) of feedback system A.43 and A.44 is given by the lower spectral radius of $\mathcal{A}$.

In A.43 the system switches between dynamics according to discrete input $u(k)$. In sliding mode (or variable structure) control [79, 25] systems with discontinuous inputs are considered. Consider a special case of a sliding mode control system

$$x(k + 1) = Ax(k) + Bu(k)$$

(A.45)

where

$$u(k) = \begin{cases} 
    G_1x(k), & \text{if } s(k, x(k)) > 0, \\
    G_2x(k), & \text{if } s(k, x(k)) < 0.
\end{cases}$$

(A.46)
The surface \( \{x(k) \mid s(k, x(k)) = 0\} \) is called the discontinuity surface. The feedback dynamics is chosen according to the sign of \( s(k, x(k)) \). In the above case the closed loop system equations are

\[
x(k+1) = \begin{cases} 
(A + BG_1) x(k), & \text{if } s(k, x(k)) > 0, \\
(A + BG_2) x(k), & \text{if } s(k, x(k)) < 0.
\end{cases} 
\]  

(A.47)

That is, either \( (A + BG_1) \) or \( (A + BG_2) \) is chosen as the state matrix according to the switching function \( s(k, x(k)) \). It may be difficult to decide on asymptotic stability of the system in this case due to the discontinuity. Especially if \( s(k, x(k)) \) is nonlinear or time dependent the analysis of the closed loop system may be too difficult. However, some useful results can be obtained from the stability of sets. For example, if \( \{A + BG_1, A + BG_2\} \) is asymptotically stable then the closed loop system will be asymptotically stable. If \( \{A + BG_1, A + BG_2\} \) is not stabilizable the closed loop system will not be asymptotically stable. Some results may also be adapted from sliding mode (or variable structure) control theory to obtain some analysis methods for stability of sets.

Now consider a system

\[
x(k+1) = A(k)x(k), \quad k = 0, 1, 2, \cdots 
\]  

(A.48)

where \( A(k) \in \langle A \rangle \triangleq \{ \sum_{i=1}^{N} p_i A_i \mid \sum_{i=1}^{N} |p_i| \leq 1 \} \), \( A_i \in \mathbb{R}^{n \times n} \). In this case \( A(k) \) can assume values from infinite continuous sets of matrices. The next matrix can jump any point in \( \langle A \rangle \). It can be observed that the asymptotic stability of systems A.48
and A.1 are equivalent (see the property 12 in Lemma A.2), which is a very useful result. Assume that system A.48 represents a system with time varying parametric uncertainty. As an example to such systems, consider a real system, modeled as

$$x(k+1) = A(k)x(k), \quad k = 0, 1, 2, \cdots.$$  \hspace{1cm} (A.49)

Assume that in the system above the matrix $A(k)$ is not known exactly. It is just known that $A(k)$ changes affinely between $A_1$ and $A_2$ arbitrarily fast. Here we have that $A(k) \in \{p_1 A_1 + p_2 A_2 \mid p_1 + p_2 = 1, \quad p_1 \geq 0, \quad p_2 \geq 0\}$. Hence system A.49 is guaranteed to be asymptotically stable if and only if $\{A_1, A_2\}$ is asymptotically stable.

In [10, 45], linear feedback systems with communication delays are considered. In this case for each particular feedback delay combinations a different state matrix is obtained. This type of systems can be modeled as in A.1 and analyzed using the stability theory of matrix sets. In [10, 45], a stochastic choice of state matrices as Markov chains is also studied.
BIBLIOGRAPHY


