SIMULTANEOUS CONTROL OF COUPLED ACTUATORS USING SINGULAR VALUE DECOMPOSITION AND SEMI-NONNEGATIVE MATRIX FACTORIZATION

A Dissertation
Presented to
The Academic Faculty

by

Ryder Christian Winck

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Mechanical Engineering

Georgia Institute of Technology
December 2012

SIMULTANEOUS CONTROL OF COUPLED ACTUATORS USING SINGULAR VALUE DECOMPOSITION AND SEMI-NONNEGATIVE MATRIX FACTORIZATION

Approved by:

Dr. Wayne J. Book, Advisor
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Nader Sadegh
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Jun Ueda
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Eric Feron
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Haesun Park
School of Computational Science and
Engineering
Georgia Institute of Technology

Date Approved: November 5, 2012

To my wife, Emily Winck.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This thesis considers the application of singular value decomposition (SVD) and semi-nonnegative matrix factorization (SNMF) within feedback control systems, called the SVD System and SNMF System, to control numerous subsystems with a reduced number of control inputs. The subsystems are coupled using a row-column structure that constrains the inputs to be rank-one when properly ordered in a matrix form. Past techniques for controlling systems in this row-column structure have focused on scheduling procedures that offer limited performance. The SVD and SNMF Systems permit simultaneous control of every subsystem, which increases the convergence rate by an order of magnitude compared with previous methods. In addition to closed loop control, open loop procedures using the SVD and SNMF are compared with previous scheduling procedures, demonstrating significant performance improvements.

This thesis presents theoretical results for the controllability of systems using the row-column structure and for the stability and performance of the SVD and SNMF Systems. In addition to theoretical analysis, practical challenges to the implementation of the SVD and SNMF Systems are examined, such as the need for physical multiplication of the row and column inputs and the need to compute the SVD and SNMF online, in real-time. Numerous simulation examples are provided that demonstrate the theoretical concepts, compare the performance of the various techniques, and raise new questions. In particular, a dynamic simulation of a pin array device, called Digital Clay, and two physical demonstrations are used to assess the feasibility of the SVD and SNMF Systems for specific applications.

CHAPTER 1

INTRODUCTION

This thesis presents new techniques to provide simultaneous control of a large set of subsystems using a limited number of control inputs. The subsystems are coupled using a row-column structure that allows mn subsystems to be controlled using $m + n$ inputs, resulting in a dimensionality constraint on the control input. Feedback control of these subsystems is accomplished in a simultaneous manner using singular value decomposition (SVD) or semi-nonnegative matrix factorization (SNMF) to reduce the dimension of the control signal to match the rank constraint of the row-column structure. Theoretical guarantees of controllability, stability and performance for these systems are presented along with methods for their physical implementation.

There are many examples of systems comprised of numerous individual subsystems in various fields of engineering practice and research, such as control of deformable mirrors for optics, microcantilever arrays, power networks, arrays for manufacturing systems with large numbers of actuators, swarm robots, and distributed manipulation [1]-[6]. Feedback control in the context of such systems is made difficult by the many control inputs required. Those inputs can be either communication signals or power signals that actuate the subsystems. Control inputs consume resources, including physical space, signal capacity, time, and communication bandwidth, and/or add complexity. For example, in an electrical system, the inputs may be voltage signals from a D/A board. In that case, the cost of an input includes the need for more cabling, D/A hardware, and power resources [7]. In the case of a microelectromechanical systems

(MEMS) device the primary concern may be the space required for wiring all of the subsystems [8]. In a fluid system, the inputs may be fluid flow through a channel, where more inputs require more control valves and fluid channels [9]. In a networked control system, additional system inputs require greater communication bandwidth and computational abilities [10], [11]. Therefore, systems comprised of many subsystems can benefit from reducing the number of inputs required for control. The challenge then is to maintain control performance while reducing system requirements. Distributed computation and decentralized control provide one way to reduce the number of inputs required to control a system [8]. However, in some situations, centralized control may be needed. For example, if each subsystem does not have the ability to perform computation, all of the computation must be performed centrally. There have been few attempts to reduce the number of inputs in large-scale systems using centralized computation. This thesis addresses that challenge by coupling the subsystems by their inputs using a row-column structure.

The row-column structure as described in this thesis reduces the number of inputs needed to control a large set of subsystems. The idea is to couple the subsystems by a grid of m rows and n columns. Every subsystem in a given column shares the control input for that column and likewise for each row. Therefore, rather than using a separate input for every subsystem, one input is used for every column of subsystems and one input is used for every row of subsystems. Thus mn subsystems can be controlled using only $m + n$ inputs.

The basic concept of the row-column structure has been used in many applications to reduce the number of required inputs. One variation of this method is

2

commonly used to reduce inputs for LCD screens [12]. The method allows for high resolution screens with fewer control signals. The subsystems are the pixels of the screen, which are arranged in a grid as shown in Figure 1. Each pixel consists of a capacitor and liquid crystal in parallel. The row inputs control the transistor at each of the pixels in their row, and the column inputs regulate the voltage to each of the pixels in their column. To generate an image, the transistors in one row are switched to permit current flow and the column inputs charge the capacitors to a set value. When the capacitors in that row are charged to the appropriate voltage across the liquid crystal, the transistors in the row are switched off. Then the transistors in the next row are switched on and the capacitors in that row are charged. This continues until the process loops back to the first row. The capacitors hold the charge across the liquid crystal until the next cycle. One of the challenges is to cycle through all of the rows fast enough such that the human eye cannot detect the cycling. The higher the resolution of the screen, the faster the procedure must cycle. This pattern is referred to as a raster scan, or line scanning.
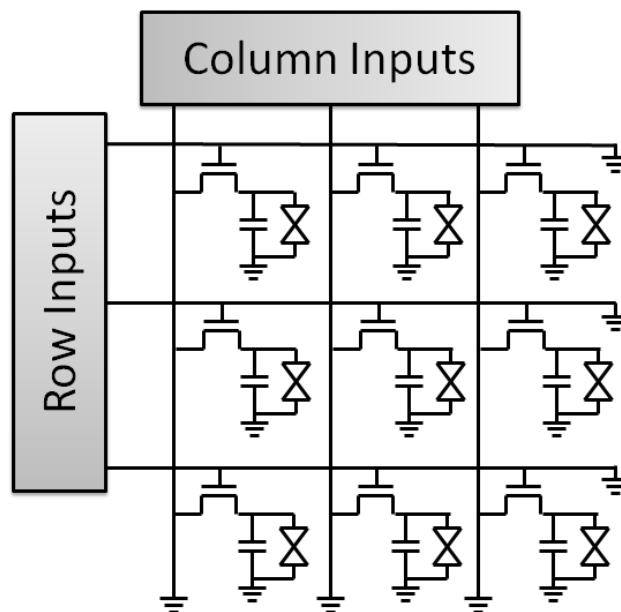
Figure 1 The electric circuit of an LCD screen is an example of the row-column structure.

In the example of the LCD screen, the subsystems are physically arranged in a grid, but this is in no way necessary for the application of the row-column structure. Consider a serial robotic manipulator with mn joints and a motor used to control each joint. Although the motors are distributed down the length of the arm, they can be coupled electronically by creating an electrical row-column structure.

Although the row-column structure significantly reduces the number of inputs required to control a large set of subsystems, it couples the subsystems by their inputs, making control of the subsystems challenging. In order to control every subsystem, a line scanning procedure is often used. As will be discussed, the line scanning procedure, while effective, is often too slow for many applications, particularly those involving motion systems. Various techniques will be discussed in the literature review concerning attempts to improve upon the line scanning procedure. However, the proposed methods so far have provided only incremental improvements. The control concepts presented in this thesis offer an entirely different approach to the problem that significantly improves the performance of systems characterized by the row-column structure and raises a number of interesting questions, both theoretical and practical, concerning dimension reduction within a feedback loop. In addition to the presentation of these new techniques, this thesis will address a number of those questions. The new techniques make use of the SVD and SNMF to reduce the dimension of the control signal, allowing for control of every subsystem in the row-column structure simultaneously, which significantly improves the speed of response of subsystems in the row-column structure.

To begin, some background on related work will be presented in Chapter 2. In particular, a pin array device, called Digital Clay, will be described, which provides a

motivating example that will be discussed throughout this thesis. Chapter 3 will explain

in detail the concept of input dimension reduction using the row-column structure and

show that the resulting system is completely controllable if the subsystems themselves

are completely controllable. Chapter 4 will discuss the application of the SVD and SNMF

to perform feedback control. Chapter 5 is devoted to the primary theoretical contributions

of this thesis, examining the stability and performance of these feedback systems.

Chapter 6 presents open loop and command generation techniques that are similar in

concept to the feedback techniques of Chapter 4. Simulation examples are provided in

Chapter 7 to demonstrate the theoretical results and prompt discussion of important

characteristics of the response of systems with dimension reduction. The physical

implementation of these concepts is discussed in Chapter 8. Finally, in Chapter 9, the

concepts are physically demonstrated using both a grid of DC motors and a grid of RC

circuits.

CHAPTER 2

BACKGROUND AND MOTIVATION

A significant amount of research has examined the control of systems containing large numbers of subsystems [5], [7], [13]-[22]. Much of this work focuses on coordinating the various subsystems to accomplish a single global task using distributed computation and decentralized control [5], [13], [15], [16]. These techniques do reduce the number of inputs required by a large system, although that aspect has not generally been the motivation for the research. However, there has also been a substantial amount of research in how to deal with the large numbers of inputs required to control these systems. Most of that research focuses on the communication requirements for distributed and network systems [10], [11]. Less work has been done on control strategies to reduce the number of inputs required for non-networked systems. Nevertheless, one area that has received some significant consideration is in the use of a row-column structure [7], [9], [17], [19]. The majority of that research focuses on pin array devices. Pin arrays are 3D surface displays consisting of a grid of small pins that can move vertically to display a surface. One example of a pin array device is Digital Clay, shown in Figure 2 [20].
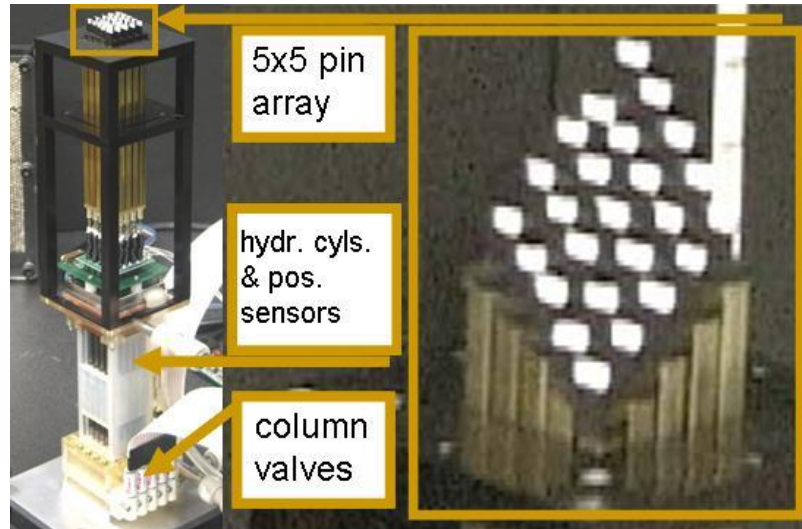
Figure 2 The current version of Digital Clay (*left*). A close-up of the array displaying a sloped surface (*right*).

Motivating Example: Digital Clay

Although not the focus of this work, Digital Clay provides a convenient motivating example for the improved control techniques presented in this thesis. It allows users to interact visually, tactilely, and haptically with virtual or remote objects. Digital Clay is unique among pin arrays in its use of hydraulic actuation of the pins, and it has patented linear position sensors to provide feedback for control and human interaction. The major design challenges for a pin array device stem from the competing design criteria of high pin resolution, small device size, fast surface generation, high force, and feasible cost. For Digital Clay, the desire for a large, high resolution grid requires many small hydraulic cylinders. To obtain a fast response and high force output, while also maintaining a compact device size, these cylinders must be controlled by small valves able to withstand high pressures and produce significant flow rates. All of these factors add to the cost of the device. In order to meet these criteria, the row-column structure has been employed in the design of Digital Clay, as it has been in other pin arrays [7], [17], [19], [21], [22].

Digital Clay uses a row-column structure that reduces the number of components and the number of control signals and thus reduces the total device size and cost. For Digital Clay, which has an $n \times n$ array of pins, the n^2 cylinders are controlled using $2n$ valves, as shown in Figure 3. As indicated in the figure, cylinder (A) is controlled using the row valve (C) and the column valve (D). The column valve uses compressed air to open and close the control adaptors (B) in each column, thereby controlling the hydraulic flow from the row valve into the cylinder. The extra valve in the bottom left of the diagram is used on the current prototype to connect the row valves to the high and low pressure sources. This implementation would likely be modified when using the techniques of this thesis, as will be discussed in Chapter 8.
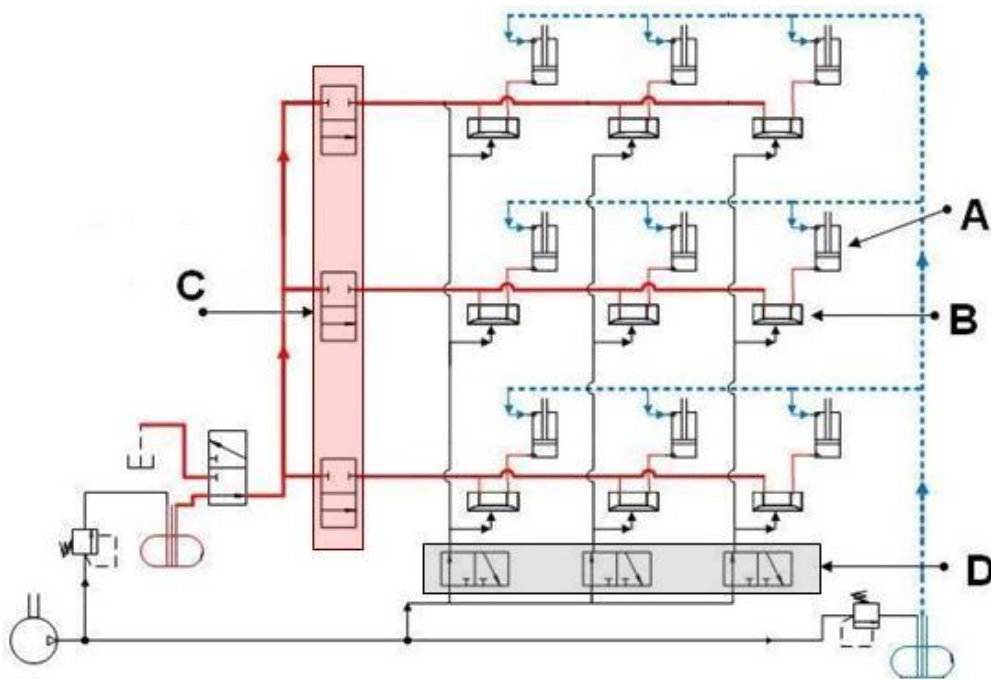
Figure 3 An abbreviated hydraulic schematic of the current Digital Clay prototype.

The most common method used to control pin arrays, including Digital Clay, is a line scanning procedure. The line scanning procedure moves the pins in the array one row at a time, as shown in Figure 4. In application to pin arrays, this procedure permits closed

8

loop control of each individual actuator while it's in motion, but in an $n \times n$ array, only n of the n^2 actuators are employed at any one time. Thus, while this is a simple and effective way to control an array, it is too slow for many applications that require real-time interaction, and it does not permit simultaneous motion of every pin, which may not be visually appealing.
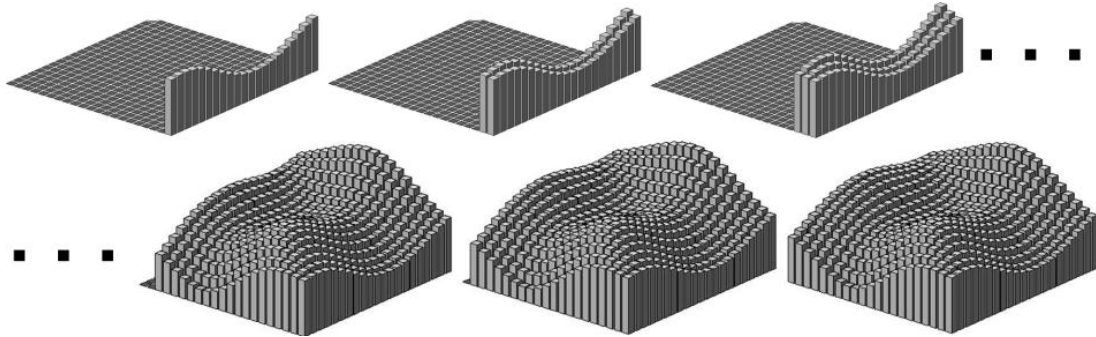
Figure 4 The line scanning technique creating a surface for a pin array display.

As an example, consider the line scanning procedure as it is used for Digital Clay. The pins have a stroke length of 50 mm and a maximum velocity of about 200 mm/s. An average surface takes approximately 3 seconds to generate for the current 5×5 grid prototype. However, if this were scaled to a 100×100 grid, for instance, it would take about 1 minute to generate a surface. If real-time interaction is desired, it would not be possible at those speeds. Without increasing the speed of the individual actuators, the speed of positioning the set of actuators that are coupled using the row-column structure needs to be improved. This could be done by independently controlling the cylinders, which would generate a surface in less than a second (assuming the necessary flow rate can be achieved), but would require 10,000 valves. A new control technique is needed to generate the surface using only the 200 valves of the row-column structure, but generating the surface at speeds on the order of seconds rather than minutes.

9

Control Improvements

There have been attempts to improve on the line scanning procedure for pin array devices [7], [9], [17], [19]. Flemming and Mascaro expanded the allowable sets of actuators at each scheduled step from actuators in one column (line scanning) to actuators in a row, a column, or a box pattern [17]. Nakatani et al. used linear programming techniques to improve on the surface generation time of line scanning for a pin array [19]. The result still involved a scheduling procedure where different rows and columns are either on or off. The linear programming technique simply provided a way to determine the optimal actuation order to minimize the surface generation time in a manner that allowed the actuators to be controlled independently. Cho et al. also controlled a pin array using a row-column system [7], [22]. Their unique contribution was the concept of iterating through every actuator in the grid, one at a time, fast enough relative to the system time constant of the actuators so that the control for each actuator worked similarly to pulse-width modulation (PWM). This is close to simultaneous control because the use of PWM moves the actuators in a continuous fashion, but it is still a scheduling procedure in the sense that a single actuator is commanded at a time. The main drawback is the time constant constraint. The speed of response is limited for each actuator and for the whole system, because it requires the iteration through all of the actuators to be done at a high rate. This constraint becomes multiplicatively greater with increasing array size.

While these methods offer improvements to line scanning, they involve a scheduling procedure that permits only a subset of actuators to receive an input at any one time so that each actuator can be controlled with an independent feedback controller.

10

Zhu suggested that to dramatically speed up the generation of a pin array the desired surface can be decomposed into intermediate surfaces, which can be achieved, one at a time, through the constant velocity motion of every pin [20]. That would require the simultaneous motion of every pin, but he provided no method to determine these intermediate surfaces or to provide feedback control for the pins during this simultaneous motion. To the author's knowledge, no attempt has been made to use a single controller to control all of the actuators simultaneously. The techniques in this thesis take advantage of the particular nature of the row-column structure to simultaneously control all of the input-coupled subsystems by reducing the dimension of the control signal. To the author's knowledge, no previous work has examined the effect of dimension reduction of a control signal within a feedback loop. In addition to pin arrays, the techniques described in this thesis are applicable to any system with large numbers of subsystems where the subsystems' inputs can be coupled by the row-column structure.

CHAPTER 3

ROW-COLUMN STRUCTURE

This chapter examines the row-column structure. A mathematical description of the row-column structure is presented that considers the combination of row and column signals to be multiplicative. This presents a rank constraint on the total set of inputs to all of the subsystems within the structure. The use of $m + n$ inputs to control mn subsystems raises a question as to the controllability of the resulting system. The controllability of the entire system is proven assuming the complete controllability of the individual subsystems.

Mathematical Description

Before providing a general mathematical model of the row-column structure for an arbitrary system, the case where each subsystem is independently controlled will first be described. This is done to outline a less conventional way of representing the system using a matrix representation that matches the physical row-column grid but with scalar operations. Consider then a set of mn decoupled, single-input, single-output, possibly nonlinear subsystems. The input-output mapping of the ij^{th} subsystem will be defined in the following manner,

$$y_{ij}(t) = H_{ij}u_{ij}(t), \tag{1}$$

where the operator, $H_{ij}: \mathbb{R} \to \mathbb{R}$, is defined by the dynamic system model,

$$\dot{x}_{ij}(t) = f_{ij}\left(t, x_{ij}(t), u_{ij}(t)\right),$$

$$y_{ij}(t) = g_{ij}\left(t, x_{ij}(t), u_{ij}(t)\right), \tag{2}$$

where $x_{ij}(t) \in \mathbb{R}^{p_{ij}}$, $u_{ij}(t) \in \mathbb{R}$, and $y_{ij}(t) \in \mathbb{R}$. Therefore, the mapping for every

subsystem can be represented in matrix form,

$$Y(t) = HU(t) = \begin{bmatrix} H_{11}u_{11}(t) & \cdots & H_{1n}u_{1n}(t) \\ \vdots & \ddots & \vdots \\ H_{m1}u_{m1}(t) & \cdots & H_{mn}u_{mn}(t) \end{bmatrix}, \tag{3}$$

where $U(t)$ and $Y(t) \in \mathbb{R}^{m \times n}$ and $H: \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$.

By placing a feedback loop around each subsystem, these independent loops can

be represented using a single feedback loop as shown in Figure 5. The double-lined

arrows will be used throughout this thesis to represent signals within a loop that are in

matrix form as opposed to vector form.
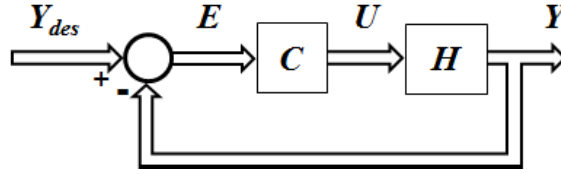
Figure 5 Matrix feedback control loop for mn subsystems.

As with $U(t)$ and $Y(t)$, $E(t)$ and $Y_{des}(t) \in \mathbb{R}^{m \times n}$. If the controller, $C: \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$,

represents a linear mapping, it can be represented in the frequency domain as

$$U_{ij}(s) = C_{ij}(s)E_{ij}(s). \forall i, j. \tag{4}$$

At first glance, this seems like an overly complicated way to represent many independent

feedback loops, but this representation will become important when the number of inputs

is reduced and each subsystem cannot be independently controlled.

To reduce the number of inputs, a row-column structure is used to couple the

subsystems in such a way that the input to each subsystem is the product of what will be

called a row input and a column input. An example for a 3×3 grid is shown in Figure 6.

In this example, 9 subsystems are controlled using 6 inputs. The multiplication of the row

and column inputs is done physically, not within the controller. For example, as will be

discussed in Chapter 8, the row and column inputs for Digital Clay can be defined

respectively as pressure and the inverse of resistance. These physically multiply to

determine the fluid flow to each cylinder.
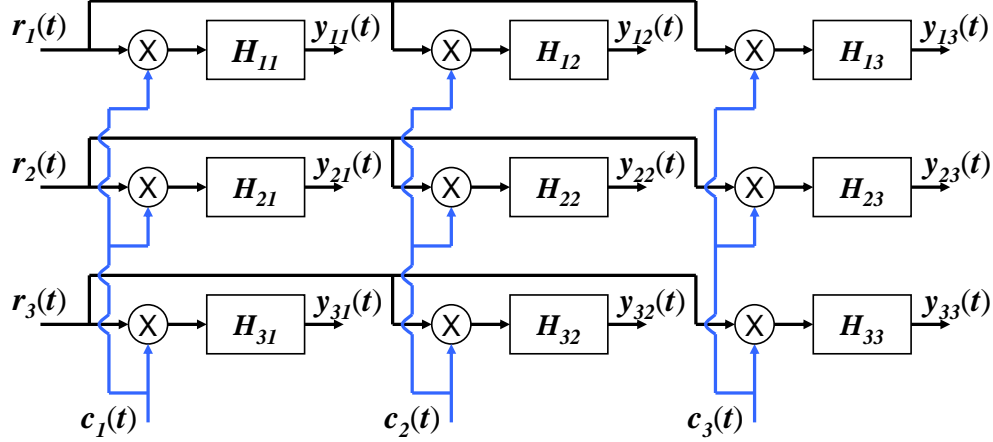
Figure 6 A 3 × 3 example of the row-column structure.

Now the subsystems' input-output mapping is expressed as

$$y_{ij} = H_{ij}\big(r_i(t), c_j(t)\big). \tag{5}$$

By coupling the subsystems in this manner, the number of inputs is reduced from mn to

$m + n$. However, in doing so, the control input matrix, $U(t)$ in (3), cannot be arbitrarily

defined. Instead, the inputs to each subsystem are defined by a new control signal matrix,

$$\widehat{U}(t) = \begin{bmatrix} r_1(t)c_1(t) & \cdots & r_1(t)c_n(t) \\ \vdots & \ddots & \vdots \\ r_m(t)c_1(t) & \cdots & r_m(t)c_n(t) \end{bmatrix}. \tag{6}$$

$\widehat{U}(t)$ can be rewritten as an outer product of the vector of the row inputs and the vector of

the column inputs,

$$\widehat{U}(t) = r(t)c(t)^T, \tag{7}$$

where $r(t) \in \mathbb{R}^m$ and $c(t) \in \mathbb{R}^n$. The effect of using this row-column input structure is

to reduce the dimension of the column space of the control input matrix, $U(t)$, from

$\min\{m, n\} \to 1$. Therefore, the new feedback loop will be the same as in Figure 5 with

14

the exception of the input being $\widehat{U}(t)$ instead of $U(t)$. The physical coupling of the subsystems is expressed by this rank-one constraint. Because the system is now underactuated, the first question that must be answered is whether the system remains completely controllable after the inputs are reduced and coupled, assuming the subsystems themselves are completely controllable when controlled independently.

Controllability

This section will demonstrate that the row-column coupling of the inputs does not reduce the theoretical controllability of the overall system. If each of the subsystems is completely controllable by itself, then the entire system shown in Figure 6 will also be completely controllable. The proof of the controllability of systems with this rank-one input constraint is simple for a set of linear subsystems if the composite systems made up of all of the subsystems in each row or each column are completely controllable using only the input from their respective row or column. In that case, simply define either the row or column inputs as constants, and the resulting linear, time-invariant (LTI) system will be completely controllable. However, if that is not the case, for example if every subsystem is exactly the same, then the controllability question is more challenging. To analyze this problem, a system model is needed. Given a set of LTI subsystems coupled using the row-column structure, a state-space representation can be expressed as

$$\dot{x}_{11}(t) = A_{11}x_{11}(t) + B_{11}r_1(t)c_1(t),$$
$$\dot{x}_{12}(t) = A_{12}x_{12}(t) + B_{12}r_1(t)c_2(t),$$
$$\dot{x}_{21}(t) = A_{21}x_{21}(t) + B_{21}r_2(t)c_1(t) \dots$$
$$\dot{x}_{mn}(t) = A_{mn}x_{mn}(t) + B_{mn}r_m(t)c_n(t).$$

(8)

For a given subsystem, $x_{ij}(t) \in \mathbb{R}^{p_{ij}}$, $A_{ij} \in \mathbb{R}^{p_{ij} \times p_{ij}}$, $B_{ij} \in \mathbb{R}^{p_{ij} \times 1}$, and $r_i(t)$ and $c_j(t)$ are scalars. Notice that the system is nonlinear in its inputs. Using this nonlinear model, controllability can be proven for the entire system on the basis that the individual subsystems, (A_{ij}, B_{ij}), are completely controllable. The proof will make use of the controllability gramian for an LTI system,

$$W_c(0, t) = \int_0^t \Phi(t_0, t) B B^T \, \Phi(t_0, t)^T dt, \tag{9}$$

and the controllability gramian for a linear, time-varying (LTV) system with $B(t)$ being time varying,

$$W_c(t_0, t_f) = \int_{t_0}^{t_f} \Phi(t_0, t) B(t) B(t)^T \, \Phi(t_0, t)^T dt. \tag{10}$$

Using these, the following theorem concerning the controllability of the system in (8) can be proven.

Theorem 1: Given a set of mn subsystems coupled by the row-column structure as specified by (8), the entire system will be completely controllable on any arbitrary interval, $[t_0, t_f]$, $t_0 < t_f$, if and only if all of the subsystems are completely controllable given an independent input.

Proof: The proof of the necessary condition is immediate. If any subsystem is not completely controllable then there is no way for the entire system to be completely controllable.

For sufficiency, it must be shown that a set of inputs exists to drive the system from any initial state to any desired state on any arbitrary interval $[t_0, t_f]$. The approach taken will involve three primary steps. Notice that (8) is not a linear system as it is nonlinear in its inputs. Therefore, the first step will be to define a set of column inputs,

$c_j(t)$, and treat the system as an LTV system. By showing that this system is completely controllable using the row inputs, $r_i(t)$, the original system in (8) is also shown to be completely controllable. The specific choice of column input is similar to the line scanning procedure, although there exist an infinite number of choices. The second step will show that the controllability gramian for the LTV system reduces to a block diagonal form due to the choice of the column inputs in the first step. The final step is to show that the individual blocks of this controllability gramian are the controllability gramians for each subsystem. Therefore, the controllability of the subsystems implies the controllability of the entire system.

For the first step, the jth column input is defined as

$$c_j(t) \triangleq h(t - T_1) - h(t - T_2), \tag{11}$$

where h is the step function defined by

$$h(t - T) = \begin{cases} 0 \text{ if } t < T \\ 1 \text{ if } t \geq T \end{cases} \tag{12}$$

and

$$T_1 = \left(t_0 + \frac{(j-1)}{n}(t_f - t_0) + \varepsilon \right)$$

$$\tag{13}$$

$$T_2 = \left(t_0 + \frac{j}{n}(t_f - t_0) - \varepsilon \right),$$

where n is the number of columns and $0 < \varepsilon \ll 1$. An example where $n = 3$, $t_0 = 1$, and $t_f = 10$ is shown in Figure 7.
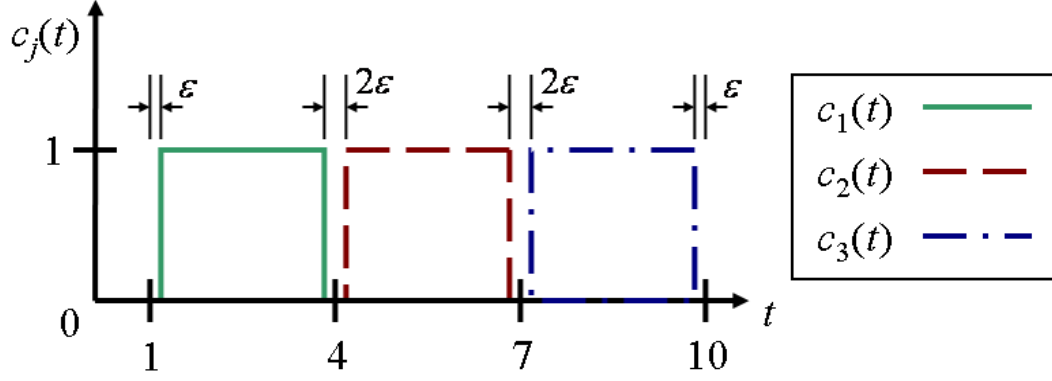
Figure 7 The column inputs for a system where $n = 3$, $t_0 = 1$, and $t_f = 10$.

This input, $c_j(t)$, has the properties

$$c_\ell(t) c_k(t) = 0, \forall \ell \neq k, \text{ and} \tag{14}$$

$$c_\ell^2(t) = \begin{cases} 1 \text{ if } T_1 < t < T_2 \\ 0 \text{ otherwise} \end{cases}. \tag{15}$$

With the column inputs thus defined, the system in (8) can be rewritten as an LTV

system,

$$\dot{x}(t) = Ax(t) + B(t)r(t), \tag{16}$$

with:

$$x = [x_{11}(t) \quad \cdots \quad x_{1n}(t) \quad x_{21}(t) \quad \cdots \quad x_{2n}(t) \quad \cdots \quad x_{m1}(t) \quad \cdots \quad x_{mn}(t)]^T; \tag{17}$$

$$r(t) = [r_1(t) \quad r_2(t) \quad \cdots \quad r_m(t)]^T; \tag{18}$$

$$A = \begin{bmatrix} A_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & A_{1n} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & A_{21} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & A_{1n} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & A_{m1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & A_{mn} \end{bmatrix}; \text{ and} \tag{19}$$

18

$$
B = \begin{bmatrix}
c_1(t)B_{11} & 0 & 0 & 0 \\
\vdots & 0 & 0 & 0 \\
c_n(t)B_{1n} & 0 & 0 & 0 \\
0 & c_1(t)B_{21} & 0 & 0 \\
0 & \vdots & 0 & 0 \\
0 & c_n(t)B_{2n} & 0 & 0 \\
0 & 0 & \ddots & 0 \\
0 & 0 & 0 & c_1(t)B_{m1} \\
0 & 0 & 0 & \vdots \\
0 & 0 & 0 & c_n(t)B_{mn}
\end{bmatrix}. \tag{20}
$$

The systems corresponding to different row inputs are decoupled. Therefore, without loss of generality, the entire system in (8) can be shown to be completely controllable by showing that the LTV system corresponding to the ith row input,

$$
\dot{x}_{\text{row}i}(t) = \begin{bmatrix} \dot{x}_{i1}(t) \\ \vdots \\ \dot{x}_{in}(t) \end{bmatrix} = \begin{bmatrix} A_{i1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & A_{in} \end{bmatrix} \begin{bmatrix} x_{i1}(t) \\ \vdots \\ x_{in}(t) \end{bmatrix} + \begin{bmatrix} c_1(t)B_{i1} \\ \vdots \\ c_n(t)B_{in} \end{bmatrix} r_i(t), \tag{21}
$$

is completely controllable for any arbitrary row, i.

For the second step, it will be shown that the controllability gramian of the system in (21) is block diagonal because of the choice of $c_j(t)$. Because every subsystem is completely controllable, it can be transformed into controllable canonical form,

$$
A_{ij} = \begin{bmatrix}
0 & 1 & \cdots & 0 \\
0 & 0 & \ddots & \vdots \\
0 & 0 & \cdots & 1 \\
a_{ij0} & a_{ij1} & \cdots & a_{ij(p_{ij}-1)}
\end{bmatrix} \text{ and} \tag{22}
$$

$$
c_j(t)B_{ij}(t) = [0 \quad \cdots \quad 0 \quad c_j(t)]^T. \tag{23}
$$

The state transition matrix for the LTV system in (21) is

$$
\Phi_i(t_0, t) = e^{A_i(t_0 - t)} = \begin{bmatrix}
e^{A_{i1}(t_0 - t)} & 0 & 0 \\
0 & \ddots & 0 \\
0 & 0 & e^{A_{in}(t_0 - t)}
\end{bmatrix}, \tag{24}
$$

and the controllability gramian for the system in (21) is

19

$$W_{i_c}(t_0, t_f) = \int_{t_0}^{t_f} \Phi_i(t_0, t) B_i(t) B_i(t)^T \, \Phi_i(t_0, t)^T dt. \tag{25}$$

Because of the properties of the column inputs given in (14) and (15), the product, $B_i(t)B_i(t)^T$, can be represented block diagonally,

$B_i(t)B_i^T(t)$

$$= \begin{bmatrix}
0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 & c_1^2(t) & 0 & \cdots & 0 & c_1(t)c_2(t) & 0 & \cdots & 0 & c_1(t)c_n(t) \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 & c_1(t)c_2(t) & 0 & \cdots & 0 & c_2^2(t) & 0 & \cdots & 0 & c_2(t)c_n(t) \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 & c_1(t)c_n(t) & 0 & \cdots & 0 & c_2(t)c_n(t) & 0 & \cdots & 0 & c_2^2(t)
\end{bmatrix}$$

$$= \begin{bmatrix}
0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 & c_1^2(t) & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & \cdots & 0 & c_2^2(t) & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & c_2^2(t)
\end{bmatrix} \tag{26}$$

$$= \begin{bmatrix}
B_{i1}(t)B_{i1}(t)^T & 0 & 0 \\
0 & \ddots & 0 \\
0 & 0 & B_{i1}(t)B_{in}(t)^T
\end{bmatrix}, \text{ where}$$

$$B_{ij}(t)B_{ij}(t)^T = \begin{bmatrix}
0 & 0 & 0 \\
0 & \ddots & 0 \\
0 & 0 & c_j^2(t)
\end{bmatrix}.$$

$B_i(t)B_i^T(t)$ and the state transition matrix are block diagonal with the ij^{th} subsystems.

Notice that the off-diagonal terms $B_{i\ell}(t)B_{ik}(t)^T = 0, \forall \ell \neq k$ because of the

20

orthogonality of the column inputs. Since $B_i(t)B_i^T(t)$ and $\Phi_i(t_0, t)$ are block diagonal, the controllability gramian in (25) will be positive definite if and only if

$$W_{ij_c}(t_0, t_f) = \int_{t_0}^{t_f} e^{A_{ij}(t_0-t)} B_{ij}(t)B_{ij}(t)^T e^{A_{ij}(t_0-t)^T} dt \tag{27}$$

is positive definite $\forall j \subset [1, n]$. Using the mean value theorem for integrals, (27) becomes

$$W_{ij_c}(T_1, T_2) = \int_{T_1}^{T_2} e^{A_{ij}(T_1-t)} \begin{bmatrix} 0 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix} e^{A_{ij}(T_1-t)^T} dt =$$

$$\int_{T_1}^{T_2} e^{A_{ij}(T_1-t)} B_{ij}B_{ij}{}^T e^{A_{ij}(T_1-t)^T} dt, \tag{28}$$

where (28) is also the controllability gramian for the ij^{th} subsystem given an independent input on the interval $[T_1, T_2]$. Since the assumption is that every subsystem is completely controllable, (28) must be positive definite $\forall i, j, t_0$, and t_f. Therefore, (25) is positive definite and, as a result, the entire system in (8) is completely controllable on any arbitrary interval $[t_0, t_f]$. $\qquad\square$

Theoretically, the row-column system's controllability is determined solely by the controllability of the individual subsystems. In reality, as with any system, the true ability of the system to obtain any arbitrary state also depends on the system bandwidth, bounds on control inputs, and other physical constraints. By defining the column inputs as above, the bandwidth limitation is increased relative to independently controlling every subsystem. For example, in a motion control scenario, it would not be possible to achieve a maximum velocity of each subsystem simultaneously using the column inputs defined in (11) because in the time required to switch from the first column input to the next, the subsystems in the first column would slow down due to friction and inertia.

Therefore, although theoretical controllability is maintained in spite of coupling

the subsystems in this row-column structure, the space of practically obtainable states is

reduced when also considering constraints on the control inputs. The selection of the

column inputs in (11) is very similar to the line scanning procedure. Each column of

subsystems is moved independently. The controllability proof hinges on the orthogonality

of the column inputs, and more generally is due to the row and column inputs forming

bases of the spaces $\mathbb{R}^m$ and $\mathbb{R}^n$ respectively. The theoretical controllability is maintained

because arbitrary rank matrices can be generated by sums of rank-one matrices formed

from the outer product of these basis vectors. This is the reason the line scanning

technique, discussed in Chapter 1 and Chapter 2, works. The generation of each line is a

rank-one input. When all of these rank-one commands are combined, the result is a set of

arbitrary-rank state matrices. The challenge then is to reduce the dimension of the inputs

in such a way that the stability of the system is maintained and the negative effects on

performance are minimized, such as a reduction of the practically reachable state space.

CHAPTER 4

FEEDBACK CONTROL USING THE SVD AND THE SNMF

This chapter examines the use of the SVD and the SNMF to reduce the dimension of the control signal such that it can be used within the row-column structure. Returning to the feedback loop in Figure 5 on page 13, if this feedback loop were applied to a system using the row-column structure, the control input would not be physically achievable. As described in Chapter 3, the input must be constrained to rank-one matrices. Therefore, rather than scheduling various rank-one inputs such that only a subset of the subsystems is controlled at any one instant, the goal is to control every subsystem simultaneously while providing a way to ensure that the rank-one constraint is met and also guaranteeing that the subsystems converge to their desired values. The SVD provides an elegantly simple way to accomplish these goals with little added complexity to the control design. Some systems, however, such as Digital Clay, have an added constraint of nonnegativity for either the row or column inputs. For those systems, the SVD is not practical, but the SNMF can provide a way to meet all of the constraints. The following sections provide a description of how the SVD and the SNMF can be used to permit simultaneous feedback control of subsystems coupled by the row-column structure. The SVD will be discussed first, beginning with background information and then describing how it is used within a feedback loop. Then, a similar discussion will be made using the SNMF. Finally, a simple simulation example will be provided to demonstrate the effect of the dimension reduction via the SVD or the SNMF on the overall system response.

Background on Singular Value Decomposition

The SVD of a matrix is generally represented as

$$A = Z\Sigma V^T, \tag{29}$$

where the matrix, $A \in \mathbb{R}^{m \times n}$, is rank-$r$, where $r \leq \min\{m, n\}$. The left and right singular vectors of A are the columns of $Z \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$, which are orthogonal matrices. $\Sigma \in \mathbb{R}^{m \times n}$ is a matrix of the singular values of A ordered on the diagonal such that

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min\{m,n\}} \geq 0. \tag{30}$$

The SVD can also be written as a sum of rank-one matrices,

$$A = \sum_{i=1}^{r} \sigma_i z_i v_i^T, \tag{31}$$

where σ_i is the ith singular value, z_i is the ith left singular vector, and v_i is the ith right singular vector. The maximum singular value of a matrix is equivalent to the 2-norm of that matrix,

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \sigma_1. \tag{32}$$

In addition to (32), the following theorem and corollary present some properties of the SVD that are useful to understanding its application in this paper. For a proof of this theorem and more on the SVD see Watkins [23].

Theorem 2: Given a rank-r matrix, $A \in \mathbb{R}^{m \times n}$, with the SVD given by (31), an arbitrary matrix, $B \in \mathbb{R}^{m \times n}$, and a matrix, $A_k \in \mathbb{R}^{m \times n}$, where

$$A_k = \sum_{i=1}^{k} \sigma_i z_i v_i^T \tag{33}$$

with rank $k < r$, then

$$\min\{\|A - B\|_2 \mid rank(B) \leq k\} = \|A - A_k\|_2 = \sigma_{k+1}. \qquad (34) \ \square$$

In other words, the truncated SVD of A provides the best low-rank approximation (LRA)

of A with respect to the matrix 2-norm. The norm of the error between A and the

truncated SVD of A is equivalent to the maximum of those singular values excluded in

the truncation. In addition to the matrix 2-norm, it can also be shown that the SVD

provides the best LRA with respect to the Frobenius norm given by [24],

$$\|A\|_F = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}|^T \right)^{1/2} = \left(\sum_{i=1}^{r} \sigma_i^2 \right)^{1/2}. \qquad (35)$$

Corollary: If the best rank-one approximation of A is given by $z_1 \sigma_1 v_1^T$, then the greatest

singular value of the error, $E_1 = A - z_1 \sigma_1 v_1^T$, is σ_2. Therefore, the best rank-one

approximation of E_1 is $z_2 \sigma_2 v_2^T$, and the greatest singular value of the error, $E_2 = E_1 -$

$z_2 \sigma_2 v_2^T$, is σ_3. This pattern continues r times until $E_r = E_{r-1} - z_r \sigma_r v_r^T = 0$. $\qquad \square$

By taking successive rank-one approximations of a rank-r matrix and the

subsequent rank-$(r - i)$ error matrices and adding the solutions at each step, the result

will converge monotonically to the exact matrix in r steps, giving the entire SVD of A.

Previous Use of the SVD in Feedback Control

The SVD possesses unique properties that have been used in many different

control applications, including analysis of controllability and observability, model

reduction, multi-input multi-output (MIMO) frequency analysis, robust control, and

analysis of sensitivity to plant variations [25]-[29]. The SVD of a plant transfer matrix

has also been used within a feedback loop as part of a decoupling feedback control law

for MIMO systems [29]-[31]. That technique, called the SVD Controller, allows a MIMO

design to be decomposed into a set of single-input single-output (SISO) designs. Lau and

Jensen used this technique for interactions at a given frequency [30]. Hovd, et al. applied the idea to systems where U and V^T were not frequency dependent, proving optimality for their case [31]. Brambilla and D'Elia extended the work of Lau to a controller that allowed the designer to trade off removing directionality with robustness by adjusting a weighting term between an inverse-based control and the SVD Controller [32]. Anthonis and Ramon extended the work of Hovd to mechanical systems where U and V^T were not constrained to be unitary or constant [33].

In this previous work, the SVD of the plant transfer matrix was used to design a decoupling MIMO controller. In contrast, in this thesis, the SVD of the control signal matrix is determined, instead of the plant transfer matrix. As a result, any type of control law (PID, state feedback, etc.) may be used in this control structure, and the SVD must be calculated at each iteration of the control loop. A description of the SVD System in that context is given in the following section.

The SVD for Feedback Control: The SVD System

As demonstrated in (7) on page 14, it is clear that the input for the row-column structure must be rank-one. Theorem 2 states that the SVD provides the best rank-one approximation of that input with respect to either the 2-norm or Frobenius norm.

Therefore, the new control input is defined as

$$\widehat{U} = rc^T = \sigma_1 z_1 v_1^T, \tag{36}$$

where z_1 and v_1 are the left and right singular vectors corresponding to σ_1, the maximum singular value of the control input U. The new feedback loop will be called the SVD System and is shown in Figure 8.
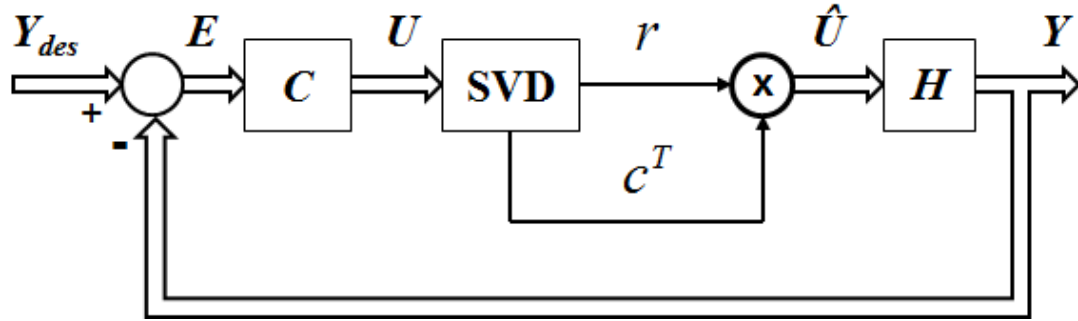
Figure 8 The SVD System.

In Figure 8, $r(t) \in \mathbb{R}^m$ and $c(t) \in \mathbb{R}^n$ are the row and column inputs that multiply using the row-column structure to produce the input to each subsystem. As with the system in Figure 5, $Y_{des}(t), E(t), U(t), \hat{U}(t)$, and $Y(t) \in \mathbb{R}^{m \times n}$, and $H: \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$ and $C: \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$. Recall that although the signals are matrices, the operations, H and C are scalar operations. It is similar to the feedback control loop in Figure 5, except that the dimension of the control signal is reduced using the SVD. The SVD decomposes a matrix into a set of basis vectors for $\mathbb{R}^m$ and $\mathbb{R}^n$ and selects the best rank-one direction in which to apply a control input every time through the loop. This dimensionality reduction has the effect of coupling the subsystems in a nonlinear manner by their inputs.

One important advantage of this control structure is that it does not dictate the choice of control law. Any type of classical, modern, robust, nonlinear, centralized or decentralized, or other control technique could be used, because the only change is the dimension reduction of the control input coming out of the controller. For instance, if every subsystem were a single-input single-output system (SISO), then a separate controller could be designed for each SISO subsystem. The row-column coupling would be accounted for by the SVD rank-one approximation.

Reducing the dimension in this way is advantageous compared with previous attempts at control within the context of the row-column structure. Whereas previous

attempts required only a subset of the subsystems to be active at any instant in time, essentially creating a scheduling procedure to iterate through the subsystems, the SVD System allows the control of all the subsystems to be carried out simultaneously by accounting for the dependence among the subsystems in the feedback controller. Additionally, there is no need to predefine a trajectory, so the system is able to respond quickly and correct for disturbances.

To gain a thorough understanding of the concept, it is helpful to step through the feedback loop in Figure 8. First, the difference between the current and desired position of *each subsystem* is found, $e_{ij}(t) = y_{desij}(t) - y_{ij}(t)$. Then the controller for *each subsystem* acts on the error of *each subsystem*, $u_{ij}(t) = C_{ij}e_{ij}(t)$. The matrix, $U(t)$, will be full-rank in general. Therefore, the largest singular value of $U(t)$, $\sigma_1(t)$, and its corresponding singular vectors, $z_1(t)$ and $v_1(t)$, are found, and the rank-one approximation, $\widehat{U}(t)$, is obtained through the system's row-column multiplication, with $r(t) = z_1\sqrt{\sigma_1}$ and $c(t) = v_1\sqrt{\sigma_1}$. This multiplication is not done centrally in the controller, but rather is part of the physical system architecture. Finally, *each entry* in $\widehat{U}$ is an input to *each subsystem*, $y_{ij}(t) = H_{ij}\widehat{u}_{ij}(t)$.

The effect of input coupling on the system is easily seen in the case of full state feedback regulation of a set of linear subsystems. For example, consider a set of mn LTI, single input, single output, decoupled subsystems of arbitrary order, p. This system can be represented in state space form as

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{mn} \end{bmatrix} = \begin{bmatrix} A_1 & 0 & 0 & 0 \\ 0 & A_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & A_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{mn} \end{bmatrix} + \begin{bmatrix} B_1 & 0 & 0 & 0 \\ 0 & B_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & B_{mn} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{mn} \end{bmatrix} \qquad (37)$$
$$= Ax + Bu$$

28

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{mn} \end{bmatrix} = \begin{bmatrix} C_1 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & C_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{mn} \end{bmatrix} + \begin{bmatrix} D_1 & 0 & 0 & 0 \\ 0 & D_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & D_{mn} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{mn} \end{bmatrix} \tag{38}$$
$$= Cx + Du.$$

The overall state vector, $x, \dot{x} \in \mathbb{R}^{mnp \times 1}$, is comprised of the state vectors for each

subsystem, $x_i, \dot{x}_i \in \mathbb{R}^{p \times 1}$. The overall system input and output is $u, y \in \mathbb{R}^{mn \times 1}$.

Therefore, $A \in \mathbb{R}^{mnp \times mnp}$, $B \in \mathbb{R}^{mnp \times mn}$, $C \in \mathbb{R}^{mn \times mnp}$, and $D \in \mathbb{R}^{mn \times mn}$, and for

each subsystem, $A_i \in \mathbb{R}^{p \times p}$, $B_i \in \mathbb{R}^{p \times 1}$, $C_i \in \mathbb{R}^{1 \times p}$, and $D_i \in \mathbb{R}^{1 \times 1}$. Assuming

independent control of each subsystem were possible, (37) becomes

$$\dot{x} = Ax - BKx, \text{ where}$$

$$K = \begin{bmatrix} k_1 & 0 & 0 & 0 \\ 0 & k_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & k_{mn} \end{bmatrix}. \tag{39}$$

The overall control gain matrix is $K \in \mathbb{R}^{mn \times mnp}$, and the control gain vector for each

subsystem is $k_i \in \mathbb{R}^{1 \times p}$.

However, if the SVD System is used, the control input for each subsystem would

be

$$\hat{u} = \text{vec}(\hat{U}) = \text{vec}(\sigma_1(K_x)u_1(K_x)v_1^T(K_x)), \tag{40}$$

where

$$K_x = \begin{bmatrix} k_1 x_1 & \cdots & k_n x_n \\ \vdots & \ddots & \vdots \\ k_{mn-n+1} x_{mn-n+1} & \cdots & k_{mn} x_{mn} \end{bmatrix}. \tag{41}$$

The operation, $\text{vec}(\cdot)$, represents the linear transformation of a matrix into a column

vector by stacking the columns of the matrix. Therefore, the input to the ith subsystem is

now nonlinearly dependent on the state of every subsystem and not just linearly

dependent on x_i.

While the SVD presents a simple method by which to reduce the dimension of the control signals, it requires four-quadrant multiplication within the row-column structure. For many systems this may not be economically feasible to build. The row-column structure for Digital Clay, for instance, has column inputs defined as the inverse of resistance, which cannot take negative values. The actual physical implementation of Digital Clay will be discussed in Chapter 8. In order to reduce the dimension of the control signal for these types of systems, semi-nonnegative matrix factorization (SNMF) is used in place of the SVD. Before discussing the SNMF's use for simultaneous control, some background information on the SNMF will be provided.

Background on Semi-Nonnegative Matrix Factorization

Semi-nonnegative matrix factorization (SNMF) is a low-rank approximation method under nonnegativity constraints on one of the low-rank factors [34]-[35]. Given a matrix $M \in \mathbb{R}^{m \times n}$, the SNMF finds two matrices $W \in \mathbb{R}^{m \times z}$ and $H \in \mathbb{R}^{n \times z}$, such that

$$M \approx WH^T \text{ subject to } H_{ij} \geq 0, \forall i, j, \tag{42}$$

where $z < \min\{m, n\}$ denotes the desired low-rank, and $H_{ij} \geq 0$ means that each element of H is nonnegative. Similarly, non-negativity may be imposed on W instead of H. In this thesis the focus is on rank-one approximations, such that $w \in \mathbb{R}^{m \times 1}$ and $h \in \mathbb{R}^{n \times 1}$. Finding appropriate vectors, w and h, can be done by solving the problem

$$\min_{u \in \mathbb{R}^m, v \in \mathbb{R}^n} \|M - wh^T\|_F^2, \text{subject to } h \geq 0. \tag{43}$$

The problem in (43) is a non-convex optimization problem. It is easy to check that its objective function is non-convex. A practical alternating minimization algorithm for (43) can be developed based on the block coordinate descent method [36]. The algorithm begins by initializing w with random real numbers and alternatively solves the following

problems until convergence. Denoting the values of w and h at the ℓth step by $w^{(\ell)}$ and $h^{(\ell)}$, for $\ell = 0, 1, 2, \cdots$, w and h are updated by

$$h^{(\ell+1)} \leftarrow \arg\min_{h \in \mathbb{R}^n} \left\| M - w^{(\ell)} h^T \right\|_F^2 \text{ s.t. } h \geq 0, \text{ and}$$

$$w^{(\ell+1)} \leftarrow \arg\min_{w \in \mathbb{R}^m} \left\| M - w \left(h^{(\ell)} \right)^T \right\|_F^2. \tag{44}$$

The solutions to these sub-problems can be written in closed forms

$$h^{(\ell+1)} \leftarrow \left[\frac{M^T w^{(\ell)}}{(w^{(\ell)})^T w^{(\ell)}} \right]_+ \text{ and } w^{(t+t)} \leftarrow \frac{M h^{(\ell+1)}}{(h^{(\ell+1)})^T h^{(\ell+1)}}, \tag{45}$$

where the $[\cdot]_+$ operator represents a projection defined element-wise as

$$([h]_+)_i = \max(h_i, 0). \tag{46}$$

The closed-form solutions in (45) can be efficiently computed because they only involve matrix-vector multiplications. The convergence of this iterative algorithm is detected by checking the Karush Kuhn Tucker (KKT) conditions of (43) [37].

Imposing non-negativity constraints in the low-rank approximation of matrices has been useful in a wide range of applications. Nonnegative matrix factorization (NMF), in which non-negativity is imposed on both of the low-rank factors, was popularized by Lee and Seung [38]. They demonstrated that the NMF is able to extract physically meaningful representations using matrices from text documents and facial images. Research on the NMF has been actively conducted both in applications such as bioinformatics and signal processing and in efficient algorithms for its computation [37], [39], [40]. Cho et al. used a variant of the NMF in controlling a robotic hand [42]. For the SNMF, in which non-negativity is imposed on only one of the low rank factors, Park and Kim, and Ding et al. studied algorithms and applications in text mining and clustering [34]-[35]. In this thesis, the SNMF is used to reduce the dimension of the control signals

31

in a feedback control loop. To the author's knowledge, the SNMF has not previously

been applied in the context of feedback control systems.

SNMF for Feedback Control: The SNMF System

The SNMF System is similar to the SVD System except that the SNMF is used in

place of the SVD to reduce the dimension of the control signals to one. Figure 9 below

shows a block diagram of the SNMF System where each signal, $Y_{des}(t)$, $E(t)$, $U(t)$ and

$Y(t) \in \mathbb{R}^{m \times n}$. As previously discussed, with the row-column structure the input to the

physical system, $\widehat{U}(t) = r(t)c(t)^T$. Here, the additional constraint, $c(t) \geq 0$, is also

considered. However, since the output of the controller in Figure 9, $U(t)$, can have

arbitrary values and be full-rank, a rank-one approximation $U(t)$ is generated by the

SNMF.
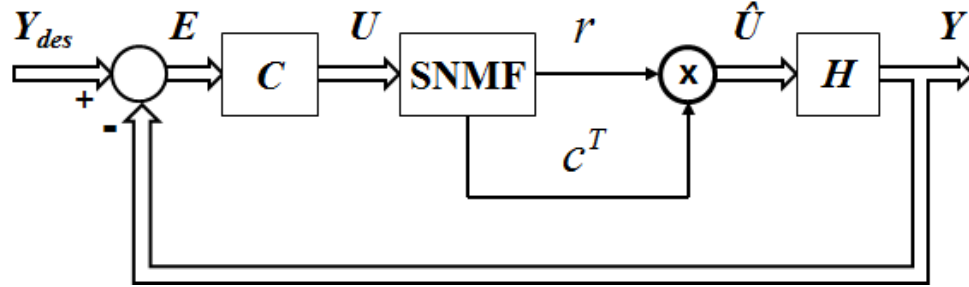
Figure 9 The SNMF System

The SNMF System functions in the same manner as the SVD System. The control

loop is similar to the SVD System control structure except that, at each iteration through

the feedback loop, the number of control signals is reduced from mn to $m + n$ using the

SNMF. Then, the rank-one approximation of $U(t)$ is found by the physical row-column

multiplication of the row and column signals: $\widehat{U}(t) = r(t)c(t)^T$. For example, given the

feedback control in (39), the approximated control input for the SNMF System would be

defined as

$$\hat{u} = \text{vec}(\hat{U}) = \text{vec}(W(K_x)H(K_x)^T), \text{subject to } H \geq 0. \tag{47}$$

The SNMF does not provide convergence guarantees like the SVD, and so is more

difficult to analyze theoretically, and the system response does not perform quite as well.

However, when the added constraint of nonnegativity is inherent to the physical system,

the SNMF System provides an important method for control. This added constraint will

be shown to be common in multiple potential applications, including Digital Clay.

Example System Response

This section describes the effect of the SVD and the SNMF dimension reduction

on the feedback loop as applied to the regulation of a set of mn moving masses. The

masses are given nonzero initial velocities, and feedback control is used to return the

masses to rest by adding damping to the system. The assumptions for this simulation are

that the multiplication of the row and column inputs is exact and immediate, every

subsystem is dynamically equivalent, and no noise is present. If the masses could be

controlled independently, the feedback loop for this system is as shown in Figure 10,

where $u_{ij}(t)$ is the force on the ij^{th} mass, $v_{ij}(t)$ is the velocity of the ij^{th} mass, and

$$G_{ij}(s) = \frac{V_{ij}(s)}{U_{ij}(s)} = \frac{1}{ms}. \tag{48}$$

If the force inputs to the masses are coupled using the row-column structure, then the

SVD System is shown in Figure 11. As in the feedback loop in Figure 8, $U(t), V(t),$

$\hat{U}(t)$, and $G(t) \in \mathbb{R}^{m \times n}$, but the operations are scalar. The SVD System response will be

compared to the independently controlled (IC) system to highlight the effect of the

dimension reduction. For both the SVD System and IC system, all of the masses are

$m = 10$ kg. A simple proportional controller is used for each subsystem, both in the IC

system and the SVD System. The control gain is $k = 10$.
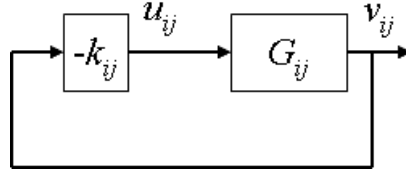
Figure 10 Regulation of mass ij's velocity using independent feedback control.
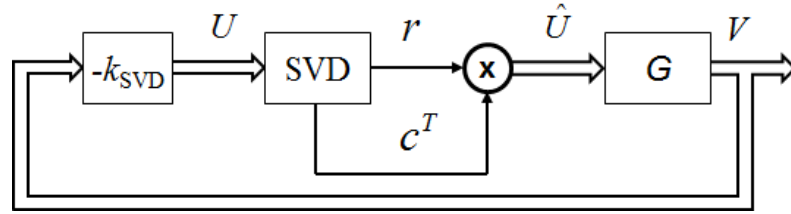
Figure 11 Regulation of a set of mn masses' velocities using the SVD System.

Given a random set of initial velocities,

$$V_{init} = \begin{bmatrix} -1.70 & 4.96 & 5.23 \\ 5.04 & 1.80 & 6.18 \\ -0.78 & 4.16 & 0.60 \end{bmatrix},$$ (49)

the IC subsystems' responses are shown in Figure 12 on page 37. As expected, the mass

velocities decay exponentially to zero with no overshoot. The SVD System response is

shown in Figure 13 on page 37. The effect of the SVD nonlinearity is clearly seen in the

response. Some of the masses overshoot the equilibrium point and others move away

from the equilibrium point before converging. Those behaviors are a direct result of the

SVD reducing the dimension of the input. In spite of the nonlinear behavior, it is possible

to predict the behavior qualitatively. The SVD System response for every mass shows a

significant change in velocity in Figure 13 at about 0.51 seconds and 2.7 seconds. The

fact that there are two significant changes is a direct result of the initial velocity matrix

being rank-3, and is caused by the change in direction of the first singular vectors of

$U(t)$. Therefore, to get a better understanding of the behavior of the SVD System, it is

34

more informative to look at the singular values and singular vectors of the input than at the system response.

Figure 14 and Figure 15 on page 38 show the singular values of the force input matrix, $U(t)$, for both the IC system and the SVD System respectively. The singular values represent the magnitude of the input to all of the subsystems and the singular vectors represent the direction of the input, or how the input is distributed through the grid of mn subsystems. Whereas with IC the singular values decay simultaneously, for the SVD System, only the largest singular value decays at any instant in time. This is the essence of the dimension reduction.

The control input is initially in the direction of the singular vector corresponding to the first singular value. After about 0.51 seconds, when the first singular value has decayed in magnitude to the value of the second singular value and then below it, the direction of the input changes to the singular vector corresponding to what was originally the second singular value. Until that point, the first singular values for the SVD System and for the IC system converge at the same rate. Thus, the convergence rate of the SVD System is a function of the relative magnitudes of the singular values of the input matrix. This means that the masses will converge to zero more quickly given an initial velocity matrix with a lower rank, or with the smaller singular values having a lower magnitude relative to the larger singular values. If the initial velocity matrix is rank-one, then the response of the SVD System will be the same as the system with IC.

The cause of the reduced convergence rate can be understood by looking at the matrix $z_1 v_1^T$. Initially, the rank-one input is only in the direction of the first singular vectors, then, at about 0.51 seconds, $z_1 v_1^T$ changes between what were originally the first

and second singular vectors. Figure 16 on page 38 shows the change at 0.51 seconds. Other than the initial switch, which is longer due to system inertia, the switching of the singular vector directions occurs every 0.001 seconds, or at every step of the numerical solver. At about 2.7 seconds, the input alternates between all three sets of singular vectors, switching at each time step of the solver. This is also evident in the way that all of the singular values decay more or less at the same, slower rate after 2.7 seconds, as seen in Figure 15.

The SNMF System can also be applied to this example, using the same control gain. The response of the system is shown in Figure 17 on page 39. Notice that the response of the SNMF System is nearly the same as the SVD System, but the convergence is slower. This is due to the nonnegativity constraint as well as the nature of the numerical solutions. Rather than looking at the singular values of the control input, Figure 18 shows the control input for mass 1,1. Notice that it initially has fewer discontinuous switches, although at particular instances it discontinuously changes value due to the numerical solution. At $t \approx 0.5$ seconds the input begins switching between a negative value and zero. This is due to the nonnegativity constraint. The rapid switching of the input can be seen more clearly in the enlarged section to the right of the plot. Clearly, the response of the SNMF System is less consistent than the SVD System, but the response still converges at nearly the same rate and nearly the same trajectory, even though the control input contains both positive and negative values.

Undoubtedly there is a relationship between the relative magnitudes of the singular values and the convergence rate. What has been taken for granted in this discussion is the fact that the SVD System converges at all. The next chapter will

examine theoretically the stability and performance of the SVD System and the effect of dimension reduction on stability by relating the stability of the SVD System to the stability of a set of independently controlled subsystems. The stability of the SNMF System will also be briefly discussed, but the results are less notable.
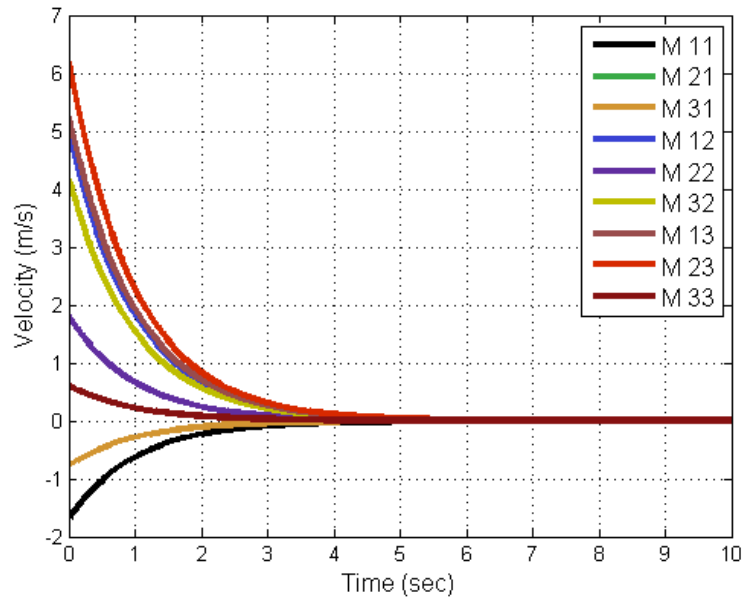
Figure 12 Velocity of 9 masses controlled independently.
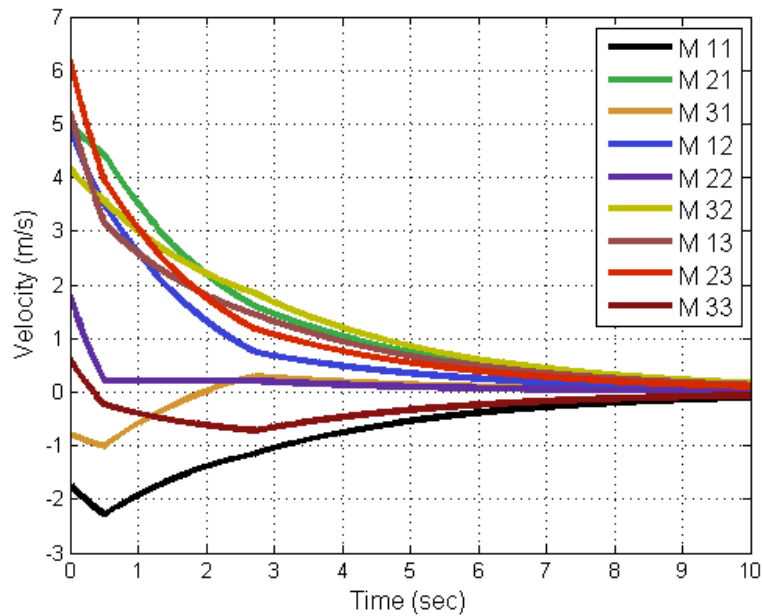
Figure 13 Velocity of 9 masses controlled using the SVD System.
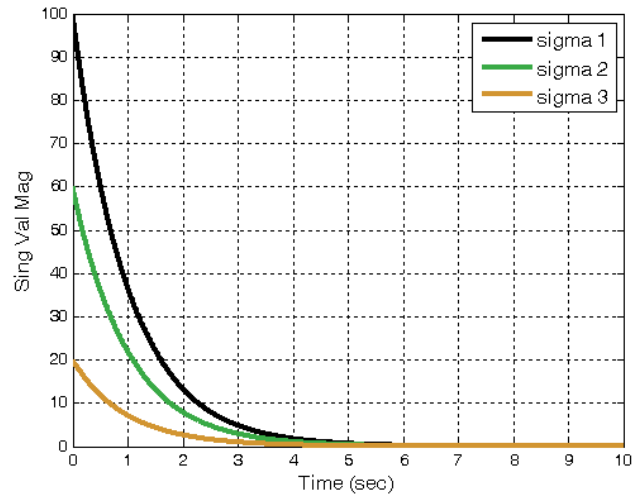
Figure 14 Singular values of $V(s)$ for the IC system.
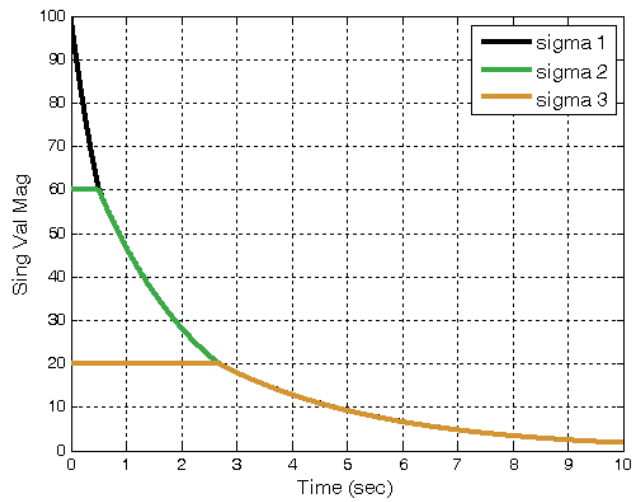
Figure 15 Singular values of $V(s)$ for the SVD System.
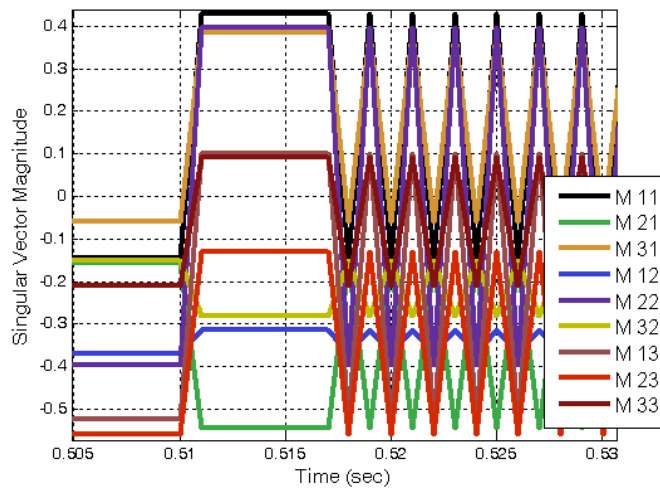
Figure 16 The rank-one matrix defined by $z_1 v_1^T$, at $t \approx 0.51$ seconds.
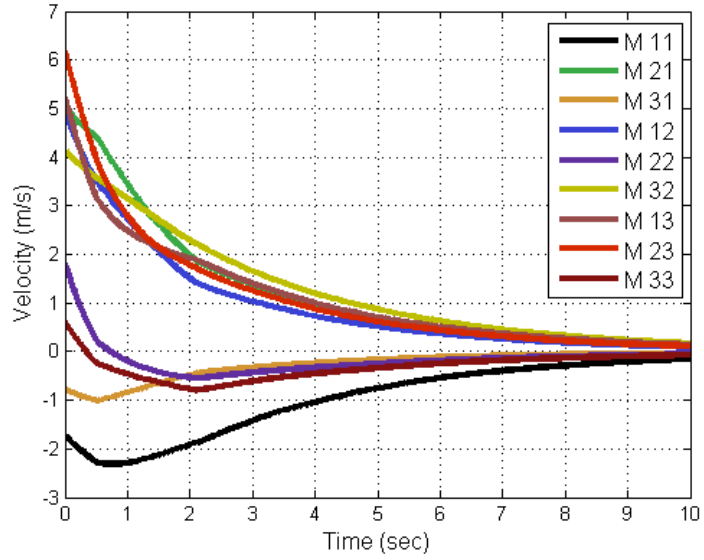
Figure 17 Velocity of 9 masses controlled using the SNMF System.
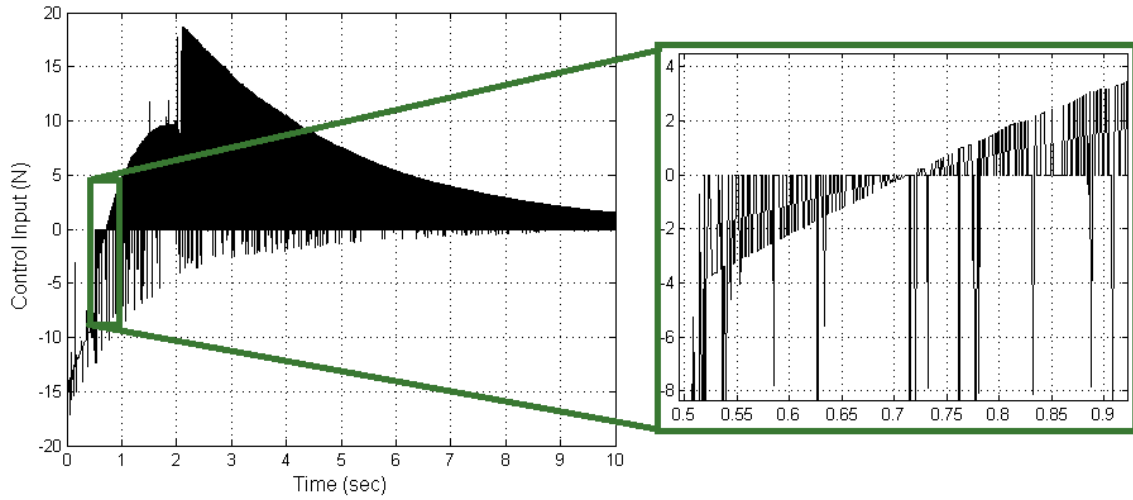
Figure 18 The control input for mass 1,1 (*left*) and an enlarged view (*right*).

CHAPTER 5

STABILITY AND PERFORMANCE

This chapter discusses the stability of the SVD System and the SNMF System in separate sections. For both the SVD and SNMF Systems, stability will be analyzed by treating the rank-one approximation as a sector-bounded nonlinearity. For the SVD System, Lyapunov analysis will be used to show exponential stability with guarantees on the rate of convergence. That performance guarantee will be related to the performance of independently controlled (IC) systems.

The SVD System

This section describes stability results for systems where the rank of a signal within the control loop is reduced using the SVD. Although the rank-one approximation is the focus of this thesis, in this section, a rank-r approximation is considered for generality. There are two primary goals of this discussion of the SVD System. The first is to provide practical measures of stability for use in designing controllers that guarantee system stability. That will be done for a number of different types of subsystems and controllers.

The second goal is to compare the stability and performance of the SVD System with the stability of a system where the subsystems can be controlled independently. If this relationship is established, then the control design can be done with little consideration of the dimension reduction, with the outcome guaranteeing both system stability and bounds on the rate of convergence. This relationship will be shown for particular classes of linear systems and for particular linear control laws. This goal is

motivated by preliminary simulations that have suggested that, given a set of decoupled, linear time-invariant (LTI) subsystems that have eigenvalues with negative real parts and a set of equivalent, decoupled, LTI feedback controllers that result in a stable closed loop system when each subsystem is independently controlled, if the same controllers are also applied to the SVD System, as shown in Figure 8 on page 27, the resulting closed loop system would also be stable. Furthermore, given a set of decoupled, LTI subsystems that have at least one eigenvalue with a positive real part and a set of equivalent, decoupled, LTI feedback controllers that results in a stable closed loop system for a certain steady-state control gain when each subsystem is independently controlled, if the same controllers were also applied to the SVD System, the steady-state control gain would need to be amplified by $\min\{m, n\}$.

This section is divided into two primary subsections. First, the SVD System stability will be analyzed for the most general case where the subsystems can be nonlinear, coupled, and time-varying. The approach taken in this subsection is to treat the SVD low rank approximation (LRA) as a sector bounded nonlinearity and to apply the small-gain theorem and passivity theories. The Circle Criterion will be applied to explore the special case of LTI subsystems while treating the SVD LRA as a bounded nonlinearity. In the next subsection, the primary results of this chapter will restrict the subsystems to LTI systems with additional restrictions on subsystem coupling and subsystem variation. Lyapunov's direct method will be used to compare the stability of a set of subsystems that are independently controlled and a set controlled using the SVD System. For higher-order subsystems, the specific choice of LQR control will be analyzed, resulting in guarantees of system stability and performance.

The SVD as a Sector-bounded Nonlinearity

In this subsection, the SVD LRA is treated as a bounded nonlinearity. To begin

with, the small-gain theorem provides an immediate, albeit conservative, stability

condition. Next, it will be shown that the SVD LRA meets a set of passivity conditions,

which can be used to directly derive conditions for stability. Finally, by limiting the

physical subsystems and controllers to LTI systems, and using sector bounds on the SVD

LRA, an absolute stability condition is derived using the Circle Criterion. The system is

assumed to be composed of mn single input, single output subsystems. In the feedback

loop in Figure 19 the signals, $E(t), U(t), Y(t),$ and $Y_{des}(t) \in \mathbb{R}^{m \times n}$, and the signal, $\widehat{U}(t)$,

is the LRA of the signal, $U(t)$. The plant and controller are, in general, operators such

that $C: \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$ and $H: \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$.
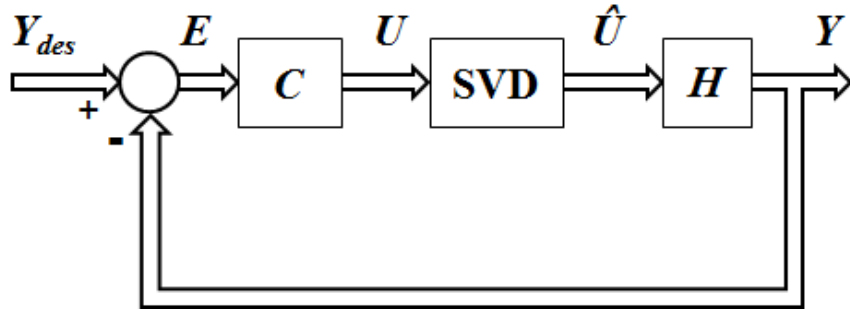
Figure 19 The SVD System with the SVD LRA of the control input signal.

The controller and the plant can, in general, be nonlinear. They can also include

coupling between the subsystems, meaning that the output of one subsystem could

depend on the input of another subsystem.

Although the matrix representation of the loop signals is convenient for

discussing the application of the SVD, to apply the small-gain theorem, the loop signals

must first be converted to $y_{des}(t), e(t), u(t), \hat{u}(t),$ and $y(t) \in \mathbb{R}^{mn}$, where $e = \text{vec}(E)$

for example. That makes the plant and controller operators such that, $C: \mathbb{R}^{mn} \to \mathbb{R}^{mn}$ and

$H: \mathbb{R}^{mn} \rightarrow \mathbb{R}^{mn}$. If the controller or plant were linear, then it could be represented as a diagonal transfer function matrix that is $mn \times mn$. Therefore, Figure 19 is converted to the loop in Figure 20. Assuming that C is linear, Figure 20 can be rewritten in the form given in Figure 21.
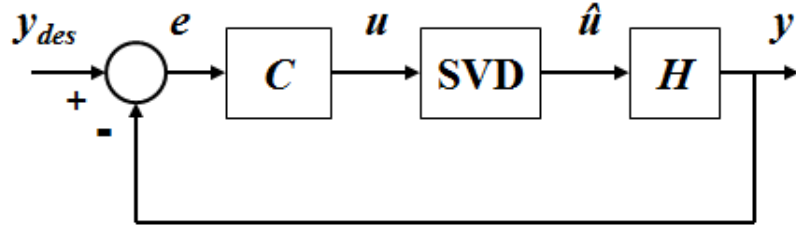
Figure 20 The vector representation of the SVD System.
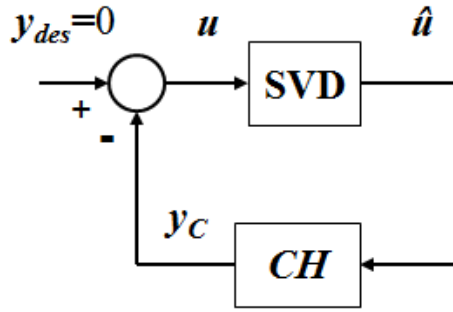
Figure 21 The vector representation of the SVD System for application of the small-gain theorem.

The system in Figure 21 can be represented by

$$\hat{u}(t) = \text{SVD}(u(t)), \text{ and}$$
$$y_c(t) = CH(\hat{u}(t)). \tag{50}$$

CH is the operator defined by the controller and plant dynamics, and SVD is the operator defined in (33) with the exception that in (50), $u(t) = \text{vec}(U)$ and $\hat{u}(t) = \text{vec}(\hat{U}(t))$. Here, the SVD LRA of U is defined as

$$\hat{U} = \text{SVD}(U) = \sum_{i=1}^{r} \sigma_i z_i v_i^T. \tag{51}$$

It is assumed throughout this chapter that CH is internally stable, i.e., if H is linear, then there are no unstable pole-zero cancellations.

43

Now that the proper block diagram form has been established, the small-gain theorem can be applied. For purposes of the below theorem, the $\mathcal{L}_2$ gain of the SVD operation in Figure 21, from $u(t)$ to $\hat{u}(t)$ is defined as

$$\|\hat{u}\|_2 \leq \gamma_1 \|u\|_2, \tag{52}$$

and the $\mathcal{L}_2$ gain of the operation, CH in Figure 21, from $\hat{u}(t)$ to $y_C(t)$ is defined as

$$\|y_c\|_2 \leq \gamma_2 \|\hat{u}\|_2. \tag{53}$$

The small-gain theorem states that this feedback loop is finite-gain $\mathcal{L}_2$ stable if $\gamma_1 \gamma_2 < 1$ [43]. Now that the system is defined, the application of the small-gain theorem will be examined.

Small-gain Theorem

Theorem 3: Given the feedback loop in Figure 21, assume that the system defined by CH in Figure 21 is a finite-gain $\mathcal{L}_2$ stable system. Then, the SVD System, given by the feedback interconnections in Figure 21, is finite-gain $\mathcal{L}_2$ stable if $\gamma_2 < 1$.

Proof: It must be shown that $\gamma_1 \gamma_2 < 1$. γ_1 in (52) can be found by noting that

$$\|\hat{u}\|_2 = \left\|\hat{U}\right\|_F = \sqrt{\sum_{i=1}^{r} \sigma_i^2}, \text{ and} \tag{54}$$

$$\|u\|_2 = \|U\|_F = \sqrt{\sum_{i=1}^{\min\{m,n\}} \sigma_i^2}, \quad \text{where } r \leq \min\{m, n\}. \tag{55}$$

Therefore,

$$\min\{\|U\|_F\} = \left\|\hat{U}\right\|_F, \text{and} \tag{56}$$

$$\gamma_1 = 1, \forall r. \tag{57}$$

Thus for stability,

44

$$\gamma_1 \gamma_2 < 1 \rightarrow \gamma_2 < 1. \qquad (58) \;\square$$

The result of Theorem 3 is conservative, and is independent of the rank of the approximation. In fact, the bound, $\gamma_1 = 1$, is obtained when $\hat{u} = u$ and the approximation is exact.

Passivity

Tighter stability bounds can be gained by showing that the SVD operation is passive, output strictly passive, and input strictly passive. This discussion of passivity proves these observations for the SVD rank-r approximation, where $r < \min\{m, n\}$. In the previous discussion, a small-gain theorem condition was stated that implies that the SVD approximation is in the sector $[-I, I]$, where $I \in \mathbb{R}^{mn \times mn}$ is the identity matrix. New sector bounds will be determined that reduce the conditions on the dynamic system for stability. The SVD LRA is treated as a memoryless bounded nonlinearity. It is memoryless in that its output depends only on the current input in time.

First, some definitions will be provided for the passivity of a memoryless nonlinearity,

$$y = h(u). \qquad (59)$$

(59) is passive if

$$u^T y \geq 0. \qquad (60)$$

It is input strictly passive if, given some function, $\varphi(u)$,

$$u^T y \geq u^T \varphi(u) > 0, \forall u \neq 0. \qquad (61)$$

It is output strictly passive if, given some function, $\rho(y)$,

$$u^T y \geq y^T \rho(y) > 0, \forall u \neq 0. \qquad (62)$$

In addition, passivity will be defined for a dynamic system,

$$\dot{x} = f(x, u), \tag{63}$$

$$y = h(x, u), \tag{64}$$

where $f: \mathbb{R}^{mnp_{ij}} \times \mathbb{R}^{mn} \to \mathbb{R}^{mnp_{ij}}$, $h: \mathbb{R}^{mnp_{ij}} \times \mathbb{R}^{mn} \to \mathbb{R}^{mn}$, $f(0,0) = 0$, and $h(0,0) =$

0. The system is passive if there exists a continuously differentiable positive semidefinite

storage function, $V(x)$, such that

$$u^T y \geq \dot{V}, \forall (x, u) \in \mathbb{R}^{mnp_{ij}} \times \mathbb{R}^{mn}. \tag{65}$$

It is input strictly passive if

$$u^T y \geq \dot{V} + u^T \varphi(u) \text{ and } u^T \varphi(u) > 0, \forall u \neq 0, \\ \forall (x, u) \in \mathbb{R}^{mnp_{ij}} \times \mathbb{R}^{mn}. \tag{66}$$

It is output strictly passive if

$$u^T y \geq \dot{V} + y^T \rho(y) \text{ and } y^T \rho(y) > 0, \forall u \neq 0, \forall (x, u) \\ \in \mathbb{R}^{mnp_{ij}} \times \mathbb{R}^{mn}. \tag{67}$$

It is strictly passive if

$$u^T y \geq \dot{V} + \psi(x), \text{ and } \psi(x) > 0, \forall (x, u) \in \mathbb{R}^{mnp_{ij}} \times \mathbb{R}^{mn}. \tag{68}$$

With these definitions, the SVD LRA will be shown to be passive, input strictly

passive, and output strictly passive, and to belong to the sectors $[0, \infty]$, $[0, I]$,

$[(r/n)I, \infty]$, and $[\alpha I, ((\alpha - 1)/((n/r)\alpha - 1))I]$, $\forall 0 \leq \alpha < r/n$. Then stability can be

determined for the feedback system in (50) using commonly known stability criterion for

feedback loops with passive elements and the definitions of passivity for dynamic

systems.

Theorem 4: The SVD rank-r approximation is passive for all r.

Proof: To prove passivity of the SVD LRA, it must be shown that

$$u^T \hat{u} \geq 0. \tag{69}$$

Defining the SVD LRA as in (51), (69) can be rewritten as

46

$$u^T \hat{u} = \left(\sigma_1 \text{vec}(z_1 v_1^T) + \sigma_2 \text{vec}(z_2 v_2^T) + \cdots \right.$$
$$+ \sigma_n \text{vec}(z_n v_n^T) \big)^T \left(\sigma_1 \text{vec}(z_1 v_1^T) + \sigma_2 \text{vec}(z_2 v_2^T) + \cdots \right.$$
$$\left. + \sigma_r \text{vec}(z_r v_r^T) \right) = \sum_{i=1}^{r} \sigma_i^2, \tag{70}$$

because the singular vectors are orthonormal. Therefore,

$$u^T \hat{u} = \sum_{i=1}^{r} \sigma_i^2 \geq 0. \tag{71} \ \square$$

Corollary 1: The SVD LRA belongs to the sector $[0, \infty]$.

Proof: This follows directly from the definition of passivity. $\square$

Theorem 5: The SVD LRA is output strictly passive for all r.

Proof: This is done by showing that, for some function, $\rho(\hat{u})$,

$$u^T \hat{u} \geq \hat{u}^T \rho(\hat{u}) > 0, \forall \hat{u} \neq 0. \tag{72}$$

Choosing $\rho(\hat{u}) = \hat{u}$ and recalling (71), the solution can be found by

$$u^T \hat{u} = \sum_{i=1}^{r} \sigma_i^2 = \left\| \hat{U} \right\|_F^2 = \hat{u}^T \hat{u} > 0, \forall \hat{u} \neq 0. \tag{73} \ \square$$

Corollary 2: The SVD rank-r approximation belongs to the sector $[0, I]$.

Proof: This is shown by proving

$$\hat{u}^T [\hat{u} - u] \leq 0 \rightarrow \hat{u}^T \hat{u} \leq u^T \hat{u}, \tag{74}$$

which has already been done in proving Theorem 5. $\square$

Theorem 6: The SVD rank-r approximation is input strictly passive for all r.

Proof: This is shown by proving that, for some function, $\varphi(u)$,

$$u^T \hat{u} \geq u^T \varphi(u) > 0, \forall u \neq 0. \tag{75}$$

Choosing $\varphi(u) = \eta u$, where η is a constant,

$$u^T \eta u = \eta \sum_{i=1}^{\min\{m,n\}} \sigma_i^2 > 0, \forall u \neq 0. \tag{76}$$

In addition,

$$\max(u^T u) = n\sigma_1^2, \text{assuming that } n < m. \tag{77}$$

When the maximum is obtained for $u^T u$,

$$u^T \hat{u} = r\sigma_1^2. \tag{78}$$

Therefore, $\eta = r/n$, such that

$$u^T \hat{u} \geq (r/n)u^T u. \tag{79} \square$$

Corollary 3: The SVD rank-r approximation belongs to the sector $[(r/n)I, \infty]$.

Proof: This is shown by proving

$$u^T [\hat{u} - (r/n)u] \geq 0 \rightarrow u^T \hat{u} \geq (r/n)u^T u, \tag{80}$$

which has already been done in proving Theorem 6. $\square$

Finally, Corollary 2 and Corollary 3 can be combined to provide a general sector bound.

Theorem 7: The SVD rank-r approximation belongs to the sector

$$\left[\alpha I, \left(\frac{\alpha - 1}{(n/r)\alpha - 1}\right)I\right], \quad \forall 0 \leq \alpha < r/n. \tag{81}$$

Proof: To prove this sector bound, begin with

$$[\hat{u} - K_1 u]^T [\hat{u} - K_2 u] \leq 0, \tag{82}$$

where $K_1 = \alpha I$ and $K_2 = \beta I$, with $\alpha > 0$ and $\beta > 0$. Substituting these relationships into (82) and simplifying, results in

$$u^T \hat{u} \geq \frac{\alpha\beta}{((\alpha + \beta) - 1)} u^T u. \tag{83}$$

Comparing (83) with (79) results in the relationship

$$\frac{\alpha\beta}{((\alpha + \beta) - 1)} = \frac{r}{n}. \tag{84}$$

48

Solving for β,

$$\beta = \left(\frac{\alpha - 1}{(n/r)\alpha - 1}\right). \tag{85}$$

Therefore, the sector bound is

$$\left[\alpha I, \left(\frac{\alpha - 1}{(n/r)\alpha - 1}\right) I\right]. \tag{86} \square$$

Notice that the sector bound in (81) captures the sector bounds in Corollary 2 and

Corollary 3 because

$$\alpha = 0 \rightarrow \left(\frac{0 - 1}{(n/r)0 - 1}\right) = 1, \text{and}$$

$$\lim_{\alpha \to (r/n)} \left(\frac{\alpha - 1}{(n/r)\alpha - 1}\right) = \infty. \tag{87}$$

Now that the passive properties of the SVD have been presented, these properties

can be used to establish stability criterion for the feedback loop in Figure 21. The

following theorems present stability criteria for various types of systems, CH, of the form

given in (63) and (64) on pages 46 and 46. Since the passivity and sector bounds of the

SVD LRA have already been proven, the proofs of the following stability theorems can

be found directly in Khalil and will not be repeated here [43].

Theorem 8: If the system defined by CH in (63) and (64) is passive, then the feedback

system in (50) on page 43 is passive, and, furthermore, if the storage function of CH is

positive definite, then the origin of the closed loop system is Lyapunov stable. $\square$

Theorem 9: If the system defined by CH in (63) and (64) is strictly passive and time-

invariant, then the origin of the closed loop system in (50) is uniformly asymptotically

stable. If the storage function for CH is radially unbounded, then the closed loop system

is globally uniformly asymptotically stable. $\square$

Theorem 10: If the system defined by CH in (63) and (64) is time-invariant, zero-state observable, and has a positive definite storage function satisfying

$$\hat{u}(t)^T y_c(t) \geq \dot{V}_{CH} + y_c(t)^T \rho(y_c(t)), \tag{88}$$

then the origin of the closed loop system is asymptotically stable if

$$\mu^T(\rho(\mu) + (r/n)\mu) > 0, \forall \mu \neq 0. \tag{89} \ \square$$

Using the sector condition in (81), another less conservative stability condition can be defined by using input feedforward and output feedback, as in Figure 22, where

$$[K_1, K_2] = \left[\alpha I, \left(\frac{\alpha - 1}{(n/r)\alpha - 1} \right) I \right], \quad \forall 0 \leq \alpha < (r/n), \tag{90}$$

and $K = K_2 - K_1$. Using these loop transformations, Theorem 9 can be applied to $\widetilde{CH}$ instead of CH because $\widetilde{SVD}$ is in the sector $[0, \infty]$ [43].

Figure 22 The SVD nonlinearity is transformed into the sector $[0, \infty]$.

The results based on passivity are less conservative than the small-gain theorem result. In the next subsection, the sector bounds of the SVD LRA will be applied to a feedback loop with LTI subsystems.

<u>Absolute Stability</u>

If the class of systems, CH, in Figure 21 is restricted to LTI systems, then the

Circle Criterion presents another stability condition. Therefore, the system, CH, can be

defined in state space form as

$$\dot{x} = Ax + B\hat{u}, \tag{91}$$

$$y_c = Cx + D\hat{u}, \tag{92}$$

where $x \in \mathbb{R}^{mnp_{ij}}$, $\hat{u}$, and $y \in \mathbb{R}^{mn}$ is the SVD LRA. The transfer function matrix for the

linear system in (91) and (92) is defined as

$$CH(s) = -C(sI - A)^{-1}B + D. \tag{93}$$

Theorem 11: The system, $CH(s)$ in Figure 21 and defined by (93), is globally uniformly

asymptotically stable if

$$[I + K_2 CH(s)][I + K_1 CH(s)]^{-1} \tag{94}$$

is strictly positive real (strictly passive) for

$$[K_1, K_2] = \left[\alpha I, \left(\frac{\alpha - 1}{(n/r)\alpha - 1}\right) I\right], \forall 0 \leq \alpha < (r/n). \tag{95}$$

Proof: It has already been demonstrated that the SVD LRA shown in (50) on page 43 is

within the sector given in (81). Therefore, the proof of Theorem 11 follows directly from

Khalil [43]. □

The Circle Criterion presents a nice visualization of the effect of changing α on

the stability bounds since the sector condition is diagonal and the subsystems are single-

input single-output. As shown in Figure 23, when $\alpha = 0$ the circle has an infinite radius

and the boundary is a line at -1. As α increases, the circle moves to the right along the

real axis, and the radius decreases, until $\alpha = r/n$. Therefore, the control designer can

select α based on this trade-off, but once α is selected, it is fixed for all frequencies. The

stability condition is that the Nyquist plot $CH(j\omega)$ cannot enter the circle and encircles it in a counterclockwise manner a number of times equal to the number of poles of $CH(s)$ with positive real parts. For the case where $\alpha = 0$, the Nyquist plot $CH(j\omega)$ must lie to the right of the vertical line at -1.
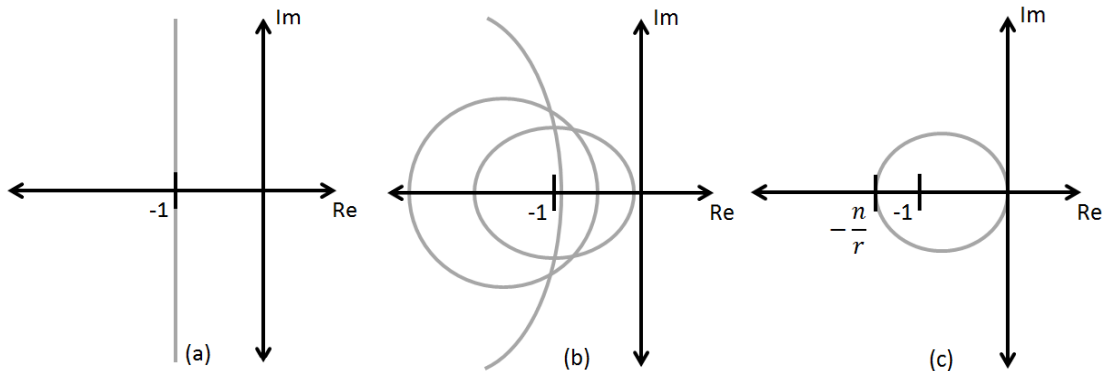
Figure 23 A graphical representation of the Circle Criterion. When $\alpha = 0$, the stability boundary is a line at -1 (a). As α increases, the radius of the circle decreases and the center of the circle moves to the right (b). When $\alpha = r/n$ the stability boundary is a circle centered at $-n/2r$ with a radius of $n/2r$ (c).

In this discussion of stability, the SVD LRA has been treated as a bounded nonlinearity. The stability conditions apply to a large class of subsystems. The subsystems can be nonlinear and/or can be arbitrarily coupled. In the next subsection, the actual SVD LRA feedback will be included in Lyapunov stability analysis to derive non-conservative stability conditions. In order to accomplish this, various constraints will be placed on the physical system and controller, restricting the class of systems to which the results can be applied. Nevertheless, the results are sufficiently general to be useful in practice. When they are too restrictive, for example, if the physical subsystems are nonlinear or coupled, then the results from the current discussion can be applied.

Linear Time-invariant Subsystems

This subsection explores the stability of systems comprised of linear, time-invariant (LTI) subsystems using Lyapunov's direct method. To begin with, the particular

52

analysis of first-order subsystems will be explored. Then, the expansion to higher-order subsystems will be discussed. For each of these, a state space representation will be used with full state feedback control being assumed, either by direct measurement or through the use of an observer. Also, in each case, a comparison will be made between the stability of an independently controlled set of subsystems and the SVD System. In addition, the convergence rates of the two systems are related, providing bounds on the reduction in performance for the SVD System relative to IC. For the higher-order subsystems, the specific LQR controller will be examined due to unique properties that aid in the stability analysis and due to its widespread application.

First-order Subsystems

For an $m \times n$ grid of possibly coupled, first-order LTI subsystems, the state-space representation is

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{mn} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1mn} \\ a_{21} & a_{22} & \cdots & a_{2mn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{mn1} & a_{mn2} & \cdots & a_{mnmn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{mn} \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{mn} \end{bmatrix} = Ax + u. \quad (96)$$

Although the entire system is higher-order, each subsystem is first-order, represented by a_{ii}. The off-diagonal terms in the A matrix represent the coupling between the subsystems. It is assumed that this coupling is such that A is symmetric, $a_{ij} = a_{ji}$. Assuming IC of each subsystem, and assuming that the controller is decoupled and that the same controller is used for each subsystem, a full state feedback controller takes the form

$$\dot{x} = Ax - kIx = (A - kI)x. \quad (97)$$

53

Using the SVD System with a rank-r approximation, $r \leq \min\{m, n\}$, for control with a

reduced number of inputs and maintaining the same assumptions of the system given

above, the feedback control is

$$u_{\text{SVD}} = \text{vec}\left(\sum_{i=1}^{r} \sigma_i(-k_{\text{SVD}}X)z_i(-k_{\text{SVD}}X)v_i(-k_{\text{SVD}}X)^T\right)$$

$$= -k_{\text{SVD}} \cdot \text{vec}\left(\sum_{i=1}^{r} \sigma_i(X)z_i(X)v_i(X)^T\right), \tag{98}$$

where $x = \text{vec}(X)$. Therefore, the closed loop system is

$$\dot{x} = Ax - k_{\text{SVD}} \cdot \text{vec}\left(\sum_{i=1}^{r} \sigma_i(X)z_i(X)v_i(X)^T\right). \tag{99}$$

Theorem 12: Given the IC system in (97) and the SVD System in (99), the SVD System

has a globally exponentially stable equilibrium point, $x = 0$, if there exists a k for the

independently controlled system such that $x = 0$ is a globally exponentially stable

equilibrium point of the IC system, and

$$k_{\text{SVD}} \geq (n/r)k, \tag{100}$$

assuming $n = \min\{m, n\}$. Additionally, if every eigenvalue in the matrix, A, in (97) and

(99), has a negative real part or zero real part that is semi-simple, then the SVD System

will have a globally exponentially stable equilibrium point, $x = 0$, if

$$k_{\text{SVD}} \geq k. \tag{101}$$

Proof: First, consider the IC system. To analyze the stability of $x = 0$ for this system, a

quadratic Lyapunov function is used,

$$V = \left(\frac{1}{2}\right)x^T x. \tag{102}$$

The time derivative of this Lyapunov function is

$$\dot{V} = \left(\frac{1}{2}\right)(\dot{x}^T x + x^T \dot{x}). \tag{103}$$

54

Substituting (97) into (103) results in

$$\dot{V} = (1/2)x^T A x + (1/2)x^T A^T x - kx^T x = x^T A x - k \cdot \text{trace}(X^T X). \quad (104)$$

Using the same Lyapunov function for the SVD System, (103) becomes

$$\dot{V}_{\text{SVD}} = x^T A x - k_{\text{SVD}} \cdot \text{vec}\left(\sum_{i=1}^{r} \sigma_i(X)z_i(X)v_i(X)^T\right)^T x$$

$$= x^T A x - k_{\text{SVD}} \cdot \text{trace}\left(\left(\sum_{i=1}^{r} \sigma_i(X)z_i(X)v_i(X)^T\right)^T X\right) \quad (105)$$

$$= x^T A x - k_{\text{SVD}}\left(\sum_{i=1}^{r} \sigma_i^2(X)\right).$$

Now, (104) and (105) will be compared. First consider the case where A has at least one eigenvalue with positive real part. Then, assuming k is positive, for the IC system in (97) to be globally exponentially stable,

$$\dot{V} = x^T A x - k \cdot \text{trace}(X^T X) = -x^T Q x \leq -\lambda_{\min}\{Q\}\|x\|_2^2, \text{where } Q \\ = Q^T > 0, \quad (106)$$

where $\lambda_{\min}\{\cdot\}$ is the minimum eigenvalue. Since

$$n\sigma_1^2(X) \geq \text{trace}(X^T X), \quad (107)$$

if the SVD System feedback gain is chosen as in (100), then

$$k_{\text{SVD}}\left(\sum_{i=1}^{r} \sigma_i^2(X)\right) \geq k \cdot \text{trace}(X^T X), \quad \forall x. \quad (108)$$

Therefore, if k is chosen to satisfy (106) and k_{SVD} to satisfy (100), then

$$\dot{V}_{\text{SVD}} = x^T A x - k_{\text{SVD}}\left(\sum_{i=1}^{r} \sigma_i^2(X)\right) \leq \dot{V} = -x^T Q x, \quad (109)$$

and

$$\dot{V}_{\text{SVD}} \leq -\lambda_{\min}\{Q\}\|x\|_2^2, \text{where } Q = Q^T > 0, \quad (110)$$

55

Thus the conditions for global exponential stability of $x = 0$ for the SVD System can be derived by the relationship in (100).

Now, consider the case where A has eigenvalues with only negative real parts or zero real parts that are semi-simple. Then for both (97) and (99), if k and k_{SVD} are nonnegative, global exponential stability is immediate. If they are negative, then k must satisfy

$$|k \cdot \text{trace}(X^T X)| < |x^T A x|, \ \forall x. \tag{111}$$

Since

$$\sigma_1^2 \leq \text{trace}(X^T X), \tag{112}$$

if the SVD System feedback gain is chosen as in (101), then if both k and k_{SVD} are chosen to be negative,

$$\left| k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right) \right| \leq |k \cdot \text{trace}(X^T X)|, \tag{113}$$

and

$$\dot{V}_{\text{SVD}} = x^T A x - k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right) \leq x^T A x - k \cdot \text{trace}(X^T X)$$
$$\leq -\lambda_{\min}\{Q\} \|x\|_2^2, \text{where } Q = Q^T > 0. \tag{114}$$

Thus the condition for global exponential stability of $x = 0$ for the SVD System can be derived by the relationship in (101). ☐

The nice general result for first-order subsystems is that using the same input as IC for the SVD System doesn't destabilize the system relative to IC if the unforced system is stable to begin with. In other words, the controller can be designed and the stability checked as if the control is independently performed. Even if the system is unstable initially, the simple relationship in (100) makes it easy to design a controller for

the independently controlled system and to adjust the gain to meet the requirements of the

SVD System. In addition to providing a condition for stability of the SVD System,

Theorem 12 also provides a way to evaluate the performance of the SVD System, as will

now be explained in Corollary 4.

Corollary 4: Assuming $k_{\text{SVD}}, k > 0$ and $m \geq n > 0$, the exponential convergence of the

SVD System in (145) is bounded from above by the convergence of the IC system in

(143) for all $x \in \mathbb{R}^{mn \times 1}$ if and only if

$$k_{\text{SVD}} \geq (n/r)k. \tag{115}$$

Proof: The proof of sufficiency follows directly from the proof of the first part of

Theorem 12, as given in (109) and (110), recognizing that (109) holds regardless of the

sign of the eigenvalues of A. The necessary portion can be proven by counterexample.

Assume that $k_{\text{SVD}} = \mu k$, where $0 < \mu < (n/r)$. If $X = I$, where I is the identity matrix,

then

$$k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right) = \mu k r < k \cdot \text{trace}(X^T X) = kn. \tag{116}$$

Therefore,

$$\dot{V}_{\text{SVD}} = x^T A x - \mu k r > \dot{V} = x^T A x - kn, \text{ for } X = I. \tag{117} \,\square$$

By Corollary 4, for the SVD System to maintain the same performance as the IC system

over the entire state space, the gain of the SVD System must be n/r times the gain for

IC. However, even with a much lower gain, the SVD System will converge at nearly the

same rate as IC for a lower rank X. For example, if $k_{\text{SVD}} = k$ and X is rank-one, then

both systems will follow exactly the same trajectory. Simulation examples discussed in

Chapter 7 will confirm this result.

One of the downsides of Theorem 12 is that it does not provide a stability condition for the SVD System alone but requires finding a stable gain for the IC system. Thus another corollary to Theorem 12 is provided that creates stability limits for the SVD System without the need to compare it with IC.

Corollary 5: For the system shown in (99), if every subsystem is the same and dynamically decoupled, such that $A = aI$ and $a \neq 0$, then the origin of the SVD System is globally exponentially stable if

$$k_{\text{SVD}} > (n/r)a. \tag{118}$$

Furthermore, if $a < 0$, then the origin of the SVD System is globally exponentially stable if

$$k_{\text{SVD}} > a. \tag{119}$$

Proof: The proof proceeds similarly to the proof of Theorem 12 above without the need for comparing the SVD System with the IC system, though the comparison is still valid. Using the Lyapunov function in (102), its derivative becomes

$$\dot{V}_{\text{SVD}} = a \cdot \text{trace}(X^T X) - k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right). \tag{120}$$

Recalling the relationship in (107) and using the relationship in (118) results in

$$k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right) > a \cdot \text{trace}(X^T X), \forall x. \tag{121}$$

Therefore, defining $\gamma = (r/n)k_{\text{SVD}} - a > 0$,

$$\dot{V}_{\text{SVD}} = a \cdot \text{trace}(X^T X) - k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right) \leq -\gamma \|x\|_2^2. \tag{122}$$

If a is restricted to be negative, then, using the relationships in (112) and (119),

$$\left| k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right) \right| < |a \cdot \text{trace}(X^T X)|, \; \forall x. \tag{123}$$

Therefore, defining $\rho = k_{\text{SVD}} - a > 0$,

$$\dot{V}_{\text{SVD}} = a \cdot \text{trace}(X^T X) - k_{\text{SVD}} \left(\sum_{i=1}^{r} \sigma_i^2(X) \right) \leq -\rho \|x\|_2^2. \tag{124} \; \square$$

Because exponential stability provides guarantees of the convergence rate, it is simple to compare the convergence SVD System and IC. Consider, for example, a set of 9 equivalent decoupled subsystems in a 3×3 grid with $a = 2$. Clearly, $k > 2$ will stabilize those systems given IC and $k_{\text{SVD}} > 6$ will stabilize the SVD System using a rank-one approximation, as predicted by Theorem 12. However, if $k \triangleq 5$, this does not mean that k_{SVD} must be 15 for stability, as might be concluded from Theorem 12. Rather, the stability condition remains the same from Corollary 5. With $k = 5$, the convergence of the IC system is given by $-3\|x\|_2^2$. For the SVD System to guarantee the same convergence for all x, then $k_{\text{SVD}} = 15$, so that $\dot{V}_{\text{SVD}} \leq -3\|x\|_2^2$. Thus for the SVD System to provide at least the same speed of response as IC, the control gain must be n/r times greater than for IC. It is important to emphasize that this is only to guarantee the speed of response for all $x \in \mathbb{R}^{mn \times 1}$. In fact, if X is rank-one, then $k_{\text{SVD}} = k$ will provide the same convergence rate.

These results for first-order subsystems are useful for predicting stability and performance for those systems. However, stability is rarely an issue for first-order systems, and the performance is simple to comprehend since it is determined by a single gain. It is tempting to directly apply the same Lyapunov approach to higher-order subsystems, but it does not work out as nicely because the control gain, k_{SVD}, cannot be

pulled out of the LRA as in (105). In the next discussion the stability and performance of

higher-order subsystems will be explored.

Higher-order Subsystems

Two approaches will be presented to analyze the stability of higher-order

subsystems. The first applies to systems with a set of LTI subsystems that are the same

and have eigenvalues with negative real parts. The second approach examines the

particular application of LQR control for systems whose subsystems have arbitrary

eigenvalues. In each case, the connection between the SVD System and the IC system

will play a key role.

First, consider the set of mn LTI single input, single output, decoupled

subsystems of arbitrary order, p, defined in (37) and (38) on pages 28 and 29. Now,

assume that all of the subsystems are the same such that $A_i = A_j$, $B_i = B_j$, $C_i = C_j$, and

$D_i = D_j$. Also, assume the use of the same full state feedback controller for each

subsystem as in (39) with $k_i = k_j$.

For this system, only the rank-one approximation will be considered. A new way

to represent the SVD rank-one approximation is also used.

Theorem 13: The SVD rank-one approximation of a matrix, A, can be obtained by

multiplication on the right or left by a symmetric rank-one projection matrix, defined

respectively by the outer product of either the first left or right singular vectors of that

matrix with themselves, $z_1 \sigma_1 v_1^T = z_1 z_1^T A = A v_1 v_1^T$.

Proof: Given a matrix $A \in \mathbb{R}^{m \times n}$, with the SVD, $A = Z\Sigma V^T$, then the SVD rank-one

approximation, A_1, of A can be expressed as:

$$A_1 = z_1 z_1^T A = z_1 z_1^T Z \Sigma V^T = z_1 [1 \quad 0 \quad \cdots \quad 0] \Sigma V^T$$
$$= z_1 [\sigma_1 \quad 0 \quad \cdots \quad 0] V^T = z_1 \sigma_1 v_1^T. \tag{125}$$

A similar procedure can be followed for multiplication on the right by $v_1 v_1^T$. The solution

will be equivalent to (125). Clearly, $z_1 z_1^T$ and $v_1 v_1^T$ are rank-one matrices because they

are defined by an outer product of two vectors. It is also easily verified that these

matrices are symmetric, $(z_1 z_1^T)^T = z_1 z_1^T$, and projections, $(z_1 z_1^T)^2 = z_1 (z_1^T z_1) z_1^T =$

$z_1 (1) z_1^T = z_1 z_1^T$. The same can be shown for $v_1 v_1^T$. □

Corollary 6: The symmetric rank-one projection matrix, $z_1 z_1^T$ or $v_1 v_1^T$, has unit 2-norm.

Proof: This follows directly from the fact that the symmetric rank-one projection matrix

is found by the outer product of a singular vector, which is a unit vector, with itself. □

The left or right singular vectors must be known in order to know the rank-one

projection matrix that can be used to obtain the SVD rank-one approximation. However,

the properties of the projection matrix are general, permitting its application here. To do

this, the SVD rank-one approximation will be represented as multiplication by a block

diagonal matrix, $M(u) \in \mathbb{R}^{mn \times mn}$,

$$\hat{u} = \begin{bmatrix} \hat{u}_1 \\ \vdots \\ \hat{u}_{mn} \end{bmatrix} = \begin{bmatrix} z_1(u)z_1(u)^T & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & z_1(u)z_1(u)^T \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_{mn} \end{bmatrix} = M(u)u. \tag{126}$$

Therefore, using this representation, the SVD System can be represented by

$$\dot{x} = Ax - B\hat{u} = Ax - BM(Kx)Kx. \tag{127}$$

The following theorem will apply to a more general class of systems, where $M(Kx)$ is

not restricted to be made up of the left singular vectors of U, but instead is made up of

any unit vector,

$$M(u) = \begin{bmatrix} \mu(u)\mu(u)^T & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mu(u)\mu(u)^T \end{bmatrix}, \text{s.t.} \|\mu(u)\|_2 = 1 \; \forall u \tag{128}$$
$$\in \mathbb{R}^{mn \times 1}.$$

The SVD System is a subset of the systems defined by (127) and (128).

Theorem 14: Given the system with minimal realization shown in (37), assume that the eigenvalues of A have negative real parts and that the eigenvalues of $A - BK$ in the system given in (39) on page 29 satisfy the same properties, the origin of the system described in (127), with $M(u)$ given in (128), is globally exponentially stable if there exists a matrix, $P_i = P_i^T > 0$, that solves the linear matrix inequality (LMI),

$$A_i^T P_i + P_i A_i < 0$$
$$(A_i - B_i K_i)^T P_i + P_i (A_i - B_i K_i) < 0. \tag{129} \;\square$$

The proof of Theorem 14 is quite complicated because of the need to analytically compute the eigenvalues of large matrices, but can be readily shown for a system of a 2×2 grid of subsystems that are second-order.

Proof: Given a system of a 2×2 grid of subsystems that are second-order, the dynamic equations can be represented by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} A_1 & 0 & 0 & 0 \\ 0 & A_2 & 0 & 0 \\ 0 & 0 & A_3 & 0 \\ 0 & 0 & 0 & A_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} B_1 & 0 & 0 & 0 \\ 0 & B_2 & 0 & 0 \\ 0 & 0 & B_3 & 0 \\ 0 & 0 & 0 & B_4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}, \text{where} \tag{130}$$

$$A_i = A_j = \begin{bmatrix} 0 & 1 \\ -a_1 & -a_2 \end{bmatrix}, \text{and } B_i = B_j = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{131}$$

To analyze the stability of the unforced $(u = 0)$ system, the following quadratic Lyapunov function is used:

$$V = x^T P x, \tag{132}$$

where $P = P^T > 0$ is a solution to the LMI in (129). Taking the time derivative of (132) results in

$$\dot{V}_{uf} = x^T(A^T P + PA)x = -x^T Qx \leq -\lambda_{\min}\{Q\}\|x\|_2^2. \tag{133}$$

Because of the block diagonal nature of A with equivalent blocks, P and Q can be represented in a similar manner as

$$P = \begin{bmatrix} P_1 & 0 & 0 & 0 \\ 0 & P_2 & 0 & 0 \\ 0 & 0 & P_3 & 0 \\ 0 & 0 & 0 & P_4 \end{bmatrix}, \text{ and } Q = \begin{bmatrix} Q_1 & 0 & 0 & 0 \\ 0 & Q_2 & 0 & 0 \\ 0 & 0 & Q_3 & 0 \\ 0 & 0 & 0 & Q_4 \end{bmatrix}, \text{ where} \tag{134}$$

$$P_i = P_j = \begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix} \text{ and } Q_i = Q_j = \begin{bmatrix} q_1 & q_2 \\ q_2 & q_3 \end{bmatrix}. \tag{135}$$

If full state feedback, as in (39), is used such that $k_i = k_j$, then (133) becomes

$$\dot{V}_{ff} = x^T((A - BK)^T P + P(A - BK))x = -x^T Rx \leq -\lambda_{\min}\{R\}\|x\|_2^2, \tag{136}$$

where R has the same block diagonal structure as Q. If instead of independent control (IC), (127) is used, then (133) becomes

$$\dot{V}_{LR} = x^T((A - BM(x)K)^T P + P(A - BM(x)K))x = -x^T W(x)x \\ \leq -\lambda_{\min}\{W(x)\}\|x\|_2^2, \forall x. \tag{137}$$

Although $W(x)$ is a function of x, it will be shown that $\lambda_i\{W(x)\}$ does not depend on x. Whereas P and Q are block diagonal with 4 blocks, where $P_i, Q_i \in \mathbb{R}^{2 \times 2}$, $W(x)$ is block diagonal with only 2 blocks such that $W_1(x) = W_2(x) \in \mathbb{R}^{4 \times 4}$. Assuming that P solves the LMI in (129), then $Q = Q^T > 0$ and $R = R^T > 0$. The next step then is to show that W_1 is similar to the matrix

$$QR = \begin{bmatrix} Q_1 & 0 \\ 0 & R_1 \end{bmatrix}. \tag{138}$$

Since the system is a 2×2 grid of identical second-order subsystems, the eigenvalues of QR are simple to compute. They are

$$\lambda\{Q_1\} = (1 + a_1)p_2 + a_2p_3$$
$$\pm (p_1^2 + (a_1^2 - 2a_1 + a_2^2 + 1)p_2^2 + (a_1^2 + a_2^2)p_3^2 \quad (139)$$
$$+ 2a_2p_1p_2 + 2a_1p_1p_3 + 2a_2p_2p_3)^{1/2}, \text{ and}$$

$$\lambda\{R_1\} = (1 + a_1 - k_1)p_2 + (a_2 - k_2)p_3$$
$$\pm (p_1^2 + (a_1^2 - 2a_1k_1 - 2a_1 + a_2^2 - 2a_2k_2 + k_1^2 + 2k_1 + k_2^2 + 1)p_2^2$$
$$+ (a_1^2 - 2a_1k_1 + a_2^2 - 2a_2k_2 + k_1^2 + k_2^2 + a_2^2)p_3^2 + (2a_2p_1 - 2k_2p_1)p_2$$
$$+ (2a_1p_1 + 2a_2p_2 - 2k_1p_1 - 2k_2p_2)p_3)^{1/2}.$$

To find the eigenvalues of $W_1(x)$, first $\mu(x)$ is defined using arbitrary functions, $\alpha(x)$

and $\beta(x)$, as

$$\mu(x)\mu^T(x) = \begin{bmatrix} \dfrac{\alpha}{\sqrt{\alpha^2 + \beta^2}} \\ \dfrac{\beta}{\sqrt{\alpha^2 + \beta^2}} \end{bmatrix} \begin{bmatrix} \dfrac{\alpha}{\sqrt{\alpha^2 + \beta^2}} & \dfrac{\beta}{\sqrt{\alpha^2 + \beta^2}} \end{bmatrix}$$

$$= \frac{1}{\alpha^2 + \beta^2} \begin{bmatrix} \alpha^2 & \alpha\beta \\ \alpha\beta & \beta^2 \end{bmatrix}. \quad (140)$$

In the interest of space, the dependence of $\alpha(x)$ and $\beta(x)$ on x is omitted. Therefore,

substituting (140) into (128), and (128) into (137) yields

$$W_1(x) = \quad (141)$$

$$\begin{bmatrix} 2p_2\phi_\alpha & p_1 + p_3\phi_\alpha + p_2\psi_\alpha & -\dfrac{2\alpha\beta k_1 p_2}{\alpha^2 + \beta^2} & \dfrac{-\alpha\beta k_1 p_3 - \alpha\beta k_2 p_2}{\alpha^2 + \beta^2} \\ p_1 + p_3\phi_\alpha + p_2\psi_\alpha & 2p_2 + 2p_3\psi_\alpha & \dfrac{-\alpha\beta k_1 p_3 - \alpha\beta k_2 p_2}{\alpha^2 + \beta^2} & \dfrac{2\alpha\beta k_2 p_3}{\alpha^2 + \beta^2} \\ -\dfrac{2\alpha\beta k_1 p_2}{\alpha^2 + \beta^2} & \dfrac{-\alpha\beta k_1 p_3 - \alpha\beta k_2 p_2}{\alpha^2 + \beta^2} & \dfrac{-\alpha\beta k_1 p_3 - \alpha\beta k_2 p_2}{\alpha^2 + \beta^2} & -\dfrac{2\alpha\beta k_2 p_3}{\alpha^2 + \beta^2} \\ \dfrac{-\alpha\beta k_1 p_3 - \alpha\beta k_2 p_2}{\alpha^2 + \beta^2} & -\dfrac{2\alpha\beta k_2 p_3}{\alpha^2 + \beta^2} & 2p_2\phi_\beta & p_1 + p_3\phi_\beta + p_2\psi_\beta \\ & & p_1 + p_3\phi_\beta + p_2\psi_\beta & 2p_2 + 2p_3\psi_\beta \end{bmatrix}$$

where

$$\phi_{\alpha=}\left(a_1 - \frac{\alpha^2 k_1}{\alpha^2 + \beta^2}\right), \psi_\alpha = \left(a_2 - \frac{\alpha^2 k_2}{\alpha^2 + \beta^2}\right), \quad (142)$$

$$\phi_\beta = \left(a_1 - \frac{\beta^2 k_1}{\alpha^2 + \beta^2}\right), \text{ and } \psi_\beta = \left(a_2 - \frac{\beta^2 k_2}{\alpha^2 + \beta^2}\right).$$

The eigenvalues of W_1 are

64

$$\lambda_{1,2}\{W_1\} = (1 + a_1)p_2 + a_2 p_3$$
$$\pm \left(p_1^2 + (a_1^2 - 2a_1 + a_2^2 + 1)p_2^2 + (a_1^2 + a_2^2)p_3^2 \right. \tag{143}$$
$$\left. + 2a_2 p_1 p_2 + 2a_1 p_1 p_3 + 2a_2 p_2 p_3\right)^{1/2}, \text{ and}$$

$$\lambda_{3,4}\{W_1\} = (1 + a_1 - k_1)p_2 + (a_2 - k_2)p_3$$
$$\pm \left(p_1^2 + (a_1^2 - 2a_1 k_1 - 2a_1 + a_2^2 - 2a_2 k_2 + k_1^2 + 2k_1 + k_2^2 + 1)p_2^2 \right.$$
$$+ (a_1^2 - 2a_1 k_1 + a_2^2 - 2a_2 k_2 + k_1^2 + k_2^2 + a_2^2)p_3^2 + (2a_2 p_1 - 2k_2 p_1)p_2$$
$$\left. + (2a_1 p_1 + 2a_2 p_2 - 2k_1 p_1 - 2k_2 p_2)p_3\right)^{1/2},$$

which are not functions of x and are the eigenvalues of QR shown in (139). Therefore,

QR and W_1 are similar matrices $\forall\, x$. Furthermore, since Q and R are positive definite, W

is also positive definite for all x, and the equilibrium point, $x = 0$, for the system in (127)

is globally exponentially stable. □

For systems of higher-order or larger grid size, it becomes difficult to directly

compute the eigenvalues in order to establish the similarity of QR and W_1, although it has

been done for third and fourth-order systems and 3×3 and 4×4 size grids. Assuming

that the LMI has a solution, the provided stability condition matches that of first-order

subsystems when A has negative eigenvalues. However, the bound on convergence is

based on the convergence of the open loop system, assuming that the choice of K

improves the speed of response of the system. No further condition for the performance

of the SVD System can be derived as in the first-order case.

Next, a second stability condition will be given for higher-order LTI subsystems,

as in (37) on page 28. From this condition, it will also be possible to derive guarantees of

performance similar to those of the first-order subsystems. For the following theorem, the

subsystems do not have to be identical. The conditions on the eigenvalues of A are also

removed, but a condition is added that the controller is an LQR controller. That is to say,

the controller defined in (39) is selected to minimize the cost function,

$$J(K, x(0)) = \int_0^\infty x^T Q x + u^T R u \, dt, \tag{144}$$

where $Q = Q^T \geq 0 \in \mathbb{R}^{mnp \times mnp}$ and $R > 0 \in \mathbb{R}^{mn \times mn}$ are block diagonal such that

$$Q = \begin{bmatrix} Q_1 & 0 & 0 & 0 \\ 0 & Q_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & Q_{mn} \end{bmatrix}, \text{and } R = \begin{bmatrix} R_i & 0 & 0 & 0 \\ 0 & R_i & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & R_i \end{bmatrix}. \tag{145}$$

$R_i \in \mathbb{R}$ is chosen to be the same for each subsystem, but $Q_i \in \mathbb{R}^{p \times p}$ is not. In general

$Q_i \neq Q_j$.

The SVD System in question will use a rank-r approximation and can be defined

as

$$\dot{x} = Ax - Bu_{\text{SVD}}, \tag{146}$$

where

$$u_{\text{SVD}} = \text{vec}\left(\sum_{i=1}^r \sigma_i(U) z_i(U) v_i(U)^T\right), \text{and } U = -\sum_{i=1}^p k_{\text{SVD}i} X_i \tag{147}$$

where each entry in X_i represents the ith state for that subsystem.

Theorem 15: The SVD System in (146) has a globally exponentially stable equilibrium

point, $x = 0$, if there exists a controller, $u = -Kx$, that solves the LQR problem in (144),

and the control gain for the SVD System is defined as

$$K_{\text{SVD}} \geq (n/2r)K, \tag{148}$$

assuming $n = \min\{m, n\}$.

Proof: The proof will proceed similarly to the proof of Theorem 12 in that the stability of

the IC system will be analyzed using Lyapunov stability theory and the result will be

used to analyze the SVD System. Therefore, to begin, the standard quadratic Lyapunov

function is used for the IC system in (37),

$$V = x^T P x. \tag{149}$$

Its derivative is

$$\dot{V} = x^T (A^T P + PA)x + u^T B^T P x + x^T P B u. \tag{150}$$

Assuming that (37) is a minimal realization, then it can be expressed by

$$A_i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_1 & -a_2 & -a_3 & \cdots & -a_p \end{bmatrix}, \text{ and } B_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}. \tag{151}$$

The matrix, $P = P^T > 0$, is block diagonal and can be represented by

$$P = \begin{bmatrix} P_1 & 0 & 0 & 0 \\ 0 & P_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & P_{mn} \end{bmatrix}, \text{ where } P_i = \begin{bmatrix} p_{11} & p_{21} & \cdots & p_{p1} \\ p_{21} & p_{22} & \cdots & p_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ p_{p1} & p_{p2} & \cdots & p_{pp} \end{bmatrix}. \tag{152}$$

Using these representations, (150) can be expressed as

$$\dot{V} = x^T (A^T P + PA)x + 2 \sum_{i=1}^{p} \left(u_i \sum_{j=1}^{mn} (p_{pj} x_{ij}) \right)$$

$$= x^T (A^T P + PA)x + 2 \cdot \text{trace} \left(U^T \left(\sum_{j=1}^{mn} p_{pj} X_j \right) \right). \tag{153}$$

Now, the feedback is defined based on LQR control. Therefore, the feedback control gain can be defined by

$$K = R^{-1} B^T P = R_i^{-1} B^T P. \tag{154}$$

The input can be redefined in matrix form as

$$U = -K_x = -R_i^{-1} \tilde{U}, \text{ where } \tilde{U} = \left(\sum_{j=1}^{mn} p_{pj} X_j \right), \tag{155}$$

where K_x is defined as in (41) on page 29, and

$$\text{vec}(U) = u = -Kx = R_i^{-1} B^T P x. \tag{156}$$

67

Substituting these relationships into (150) results in

$$
\begin{aligned}
\dot{V}_{\text{LQR}} &= x^T(A^TP + PA)x - x^TK^TB^TPx - x^TPBKx \\
&= x^T(A^TP + PA)x - 2R_i^{-1}x^TPBB^TPx \\
&= -x^T(Q + R_iK^TK)x \le -\lambda_{\min}\{Q + R_iK^TK\}\|x\|_2^2.
\end{aligned}
\tag{157}
$$

For the SVD System,

$$
\begin{aligned}
U_{\text{SVD}} &= \sum_{i=1}^{r} \sigma_i(K_{\text{SVD}}x)z_i(K_{\text{SVD}}x)v_i(K_{\text{SVD}}x)^T \\
&= -R_{\text{SVD}}^{-1}\sum_{i=1}^{r} \sigma_i(\tilde{U})z_i(\tilde{U})v_i(\tilde{U})^T,
\end{aligned}
\tag{158}
$$

where $K_{\text{SVD}} = R_{\text{SVD}}^{-1}B^TP$ and $\text{vec}(K_{\text{SVD}}x) = K_{\text{SVD}}x$. Substituting (158) into (153) results in

$$
\begin{aligned}
\dot{V}_{\text{SVD}} &= x^T(A^TP + PA)x - 2R_{\text{SVD}}^{-1} \\
&\quad \cdot \text{trace}\left(\left(\sum_{i=1}^{r}\sigma_i(\tilde{U})z_i(\tilde{U})v_i(\tilde{U})^T\right)^T\left(\sum_{j=1}^{mn}p_{pj}X_j\right)\right) \\
&= x^T(A^TP + PA)x - 2R_{\text{SVD}}^{-1} \\
&\quad \cdot \text{trace}\left(\left(\sum_{i=1}^{r}\sigma_i(\tilde{U})z_i(\tilde{U})v_i(\tilde{U})^T\right)^T\tilde{U}\right) \\
&= x^T(A^TP + PA)x - 2R_{\text{SVD}}^{-1}\sum_{i=1}^{r}\sigma_i(\tilde{U})^2.
\end{aligned}
\tag{159}
$$

Using (157), the resulting relationship is

$$
(A^TP + PA) = -Q + 2K^TK.
\tag{160}
$$

Substituting (160) into (159) results in

$$\dot{V}_{\text{SVD}} = x^T(-Q + R_i K^T K)x - 2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\widetilde{U})^2$$

$$= -x^T Q x - 2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\widetilde{U})^2 + R_i x^T K^T K x \tag{161}$$

$$= -x^T Q x - 2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\widetilde{U})^2 + R_i^{-1} x^T P B B^T P x$$

$$= -x^T Q x - 2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\widetilde{U})^2 + R_i^{-1} \cdot \text{trace}(\widetilde{U}^T \widetilde{U}).$$

Since

$$(n/r) \sum_{i=1}^{r} \sigma_i(\widetilde{U})^2 \geq \text{trace}(\widetilde{U}^T \widetilde{U}), \tag{162}$$

if $k_{\text{SVD}} \geq (n/2r)k_{\text{LQR}}$, or $R_{\text{SVD}}^{-1} \geq (n/2r)R_i^{-1}$, then

$$\dot{V}_{\text{SVD}} = -x^T Q x - 2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\widetilde{U})^2 + R_i^{-1} \cdot \text{trace}\left(\widetilde{U}^T \widetilde{U}\right) \leq -x^T Q x \tag{163}$$

$$\leq -\lambda_{\min}\{Q\}\|x\|_2^2.$$

Therefore, $x = 0$ is an exponentially stable equilibrium point of the SVD System. □

Theorem 15 provides a useful check for stability given the design of an LQR controller for IC. For example, consider the expensive control case where $(Q/R) \to 0$ in (144). Selecting k_{SVD} using Theorem 15 provides a stabilizing controller using a minimum control effort, that effort being $n/2r$ times the gain required to stabilize the IC system. However, Theorem 15 is still a sufficient condition that states that the stability is conditional on the existence of a k that solves the LQR control problem. Therefore, if the controller for the SVD System is designed by solving the LQR problem and then multiplying the gain by $n/2r$, the resulting controller is likely conservative. This is due to the fact that for (161) to be quadratically negatively definite,

$$-2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\tilde{U})^2 + R_i^{-1} \cdot \text{trace}(\tilde{U}^T \tilde{U}) < x^T Q x, \forall x. \qquad (164)$$

However, it is not known how to prove this inequality for all x. Instead, the

inequality,

$$-2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\tilde{U})^2 + R_i^{-1} \cdot \text{trace}(\tilde{U}^T \tilde{U}) < 0, \forall x, \qquad (165)$$

is used to derive (163). For example, if $k = \begin{bmatrix} 3 & 2 \end{bmatrix}$ is the solution to an LQR problem for

a set of second-order subsystems in a 4×4 grid, and $k' = \begin{bmatrix} 6 & 4 \end{bmatrix}$ is also a solution to an

LQR problem for the same set of subsystems but for a different choice of Q and R_i, then

clearly $k'_{\text{SVD}} > k' = 2k$ results in asymptotic stability even though $k'_{\text{SVD}} > k'$ does not

meet the condition in Theorem 15. Therefore, to find the boundary for stability for the

SVD System, the expensive control approach is useful. On the other hand, in designing a

controller it is often not desirable to choose one that is barely stable. Rather, a controller

is usually chosen to improve the performance of a system, as well as to stabilize it. In

addition to a stability guarantee, Theorem 15 leads to a convenient way of relating the

performance of the SVD System to IC as expressed in the following corollary.

Corollary 7: The convergence of the SVD System in (146) can be bounded for all

$x \in \mathbb{R}^{mnp \times 1}$ by the convergence of the IC system using a controller, $u = -Kx$, that

solves the LQR problem in (144) if and only if the control gain for the SVD System is

defined as

$$k_{\text{SVD}i} \geq (n/r)k_i, \qquad (166)$$

assuming $n = \min\{m, n\}$.

Proof: If $k_{\text{SVD}} \geq (n/r)k_{\text{LQR}}$, or $R_{\text{SVD}}^{-1} \geq (n/r)R_i^{-1}$, then (161) becomes

$$\dot{V}_{\text{SVD}} = -x^T Q x - 2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\tilde{U})^2 + R_i^{-1} \cdot \text{trace}(\tilde{U}^T \tilde{U})$$
$$\leq -x^T Q x - R_i^{-1} \cdot \text{trace}(\tilde{U}^T \tilde{U}) \quad (167)$$
$$= -x^T Q x - R_i x^T K^T K x = -x^T (Q + R_i K^T K) x$$
$$\leq -\lambda_{\min}\{Q + R_i K^T K\} \|x\|_2^2.$$

Therefore, $\dot{V}_{\text{SVD}}$ is bounded by the same function as $\dot{V}_{\text{LQR}}$, and both converge at a rate

bounded by the exponential

$$\|x\| \leq \left(\frac{\lambda_{\max}\{P\}}{\lambda_{\min}\{P\}} \right)^{\frac{1}{2}} \|x_0\| e^{-\left(\frac{\lambda_{\min}\{Q + R_i K^T K\}}{2\lambda_{\max}\{P\}} \right)(t - t_0)}. \quad (168)$$

The necessary portion can be proven by counterexample. Assume that $k_{\text{SVD}} = \mu k_{\text{LQR}}$, or

$R_{\text{SVD}}^{-1} = \mu R_i^{-1}$, where $0 < \mu < (n/r)$. If $X = I$, where I is the identity matrix, then

$$2R_{\text{SVD}}^{-1} \sum_{i=1}^{r} \sigma_i(\tilde{U})^2 = 2\mu R_i^{-1} r \left(\sum_{j=1}^{mn} p_{pj} \right)^2 < 2R_i^{-1} \cdot \text{trace}(\tilde{U}^T \tilde{U})$$
$$= 2R_i^{-1} n \left(\sum_{j=1}^{mn} p_{pj} \right)^2. \quad (169)$$

Therefore,

$$\dot{V}_{\text{SVD}} = -x^T Q x - 2\mu R_i^{-1} r \left(\sum_{j=1}^{mn} p_{pj} \right)^2 > \dot{V}_{\text{LQR}}$$
$$\quad (170)\square$$
$$= -x^T Q x - R_i^{-1} n \left(\sum_{j=1}^{mn} p_{pj} \right)^2, \text{ for } X = I.$$

Thus to design a controller for the SVD System, given the cost function in (144),

the weights can be selected as desired and the IC gain can be obtained by solving the

LQR problem. Using Corollary 7, the controller for the SVD System can be obtained to

provide the same convergence. However, this choice of k_{SVD} will result in faster

convergence of the SVD System for many trajectories and could potentially require more control effort than available.

The results in Theorem 14 and Theorem 15 echo the results derived for first-order subsystems in Theorem 12, but add restrictions. Theorem 14 is limited by the need to solve the LMI. That additional restriction is not surprising, considering that the dimension reduction it allows is much more general than the SVD rank-one approximation. Theorem 15 only allows for LQR controllers, but the result is less restrictive than the first-order case when the physical system is open loop unstable. As a result, Theorem 15 is beneficial in the design of LQR controllers for first-order subsystems. On the downside, the result for Theorem 15 is not conditional on the stability of the subsystems as it is in the first-order case. It is likely that the conditions for stability derived in Theorem 12 therefore do apply to systems of higher-order, although no proof has yet been found demonstrating this.

The SNMF System

The analysis of the stability of the origin of the SNMF System is more difficult than the SVD because the SNMF does not have the same breadth of properties as the SVD by which to conduct the analysis. Additionally, the SNMF is based purely on a numerical approach to a non-convex optimization problem. Nevertheless, a stability result for this system can still be obtained by the small gain theorem. The system of interest, shown in Figure 24, is given in (50) on page 43, except that the SVD operator is replaced by the SNMF rank-one approximation. It can be represented by

$$\hat{u}(t) = \text{SNMF}(u(t)), \text{ and}$$
$$y_c(t) = CH(\hat{u}(t)).$$

(171)

CH is the operator defined by the controller and plant dynamics, and the SNMF is the operator defined in (42) on page 30, with the exception that in (171), $u(t) = \text{vec}(U)$ and $\hat{u}(t) = \text{vec}(\hat{U}(t))$.
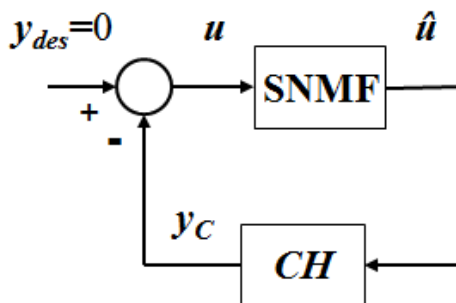
Figure 24 The SNMF System in a format for application of the small-gain theorem.

For the purposes of the theorem below, the $\mathcal{L}_2$ gain of the SNMF operator from $u(t)$ to $\hat{u}(t)$ is defined as

$$\|\hat{u}\|_2 \leq \gamma_1 \|u\|_2, \tag{172}$$

and the $\mathcal{L}_2$ gain of the operator, CH, from $\hat{u}(t)$ to $y_C(t)$ is defined as

$$\|y_c\|_2 \leq \gamma_2 \|\hat{u}\|_2. \tag{173}$$

Theorem 16: Given the feedback loop in Figure 24, assume that the system defined by CH in Figure 24 is a finite-gain $\mathcal{L}_2$ stable system. Then, the SNMF System, given by the feedback interconnections in Figure 24 is finite-gain $\mathcal{L}_2$ stable if $\gamma_2 < 1/2$.

Proof: First, (172) is rewritten as

$$\|\hat{U}\|_F \leq \gamma_1 \|U\|_F, \tag{174}$$

where $u = \text{vec}(U)$. As discussed in more detail in Chapter 4, the SNMF algorithm iteratively solves the problem

$$\min \|U - \hat{U}\|_F^2, \text{ where } \hat{U} = uv^T \text{ and } v \geq 0. \tag{175}$$

The algorithm will not return a solution where the Frobenius norm of the error is worse than its initial value. Therefore, the error given from the initial value for $\hat{U}$ forms an

73

upper bound for the error of the SNMF algorithm. If the initial guess for $\widehat{U}$ is the zero

matrix, then

$$\|U_E\|_F = \left\|U - \widehat{U}\right\|_F \le \|U - 0\|_F = \|U\|_F. \tag{176}$$

Using this inequality, a bound for $\widehat{U}$ is

$$\left\|\widehat{U}\right\|_F = \|U - U_E\|_F \le (\|U\|_F + \|-U_E\|_F) \le (\|U\|_F + \|U\|_F), \text{ or} \tag{177}$$

$$\left\|\widehat{U}\right\|_F \le 2\|U\|_F. \tag{178}$$

Therefore, $\gamma_1 = 2$ and the bound for $\mathcal{L}_2$ stability of Figure 24 is

$$\gamma_2 < 1/2. \tag{179} \ \square$$

The result in Theorem 16 is even more conservative than the small gain result for the

SVD System. However, using the relationship in (176), it is easy to see that the SNMF

rank-one approximation is passive.

Theorem 17: The SNMF rank-one approximation in (171) is passive.

Proof: Starting with (176), this relationship can be rewritten as

$$\|u - \hat{u}\|_2 \le \|u\|_2. \tag{180}$$

Therefore, $|\theta| \le \pi/2$ where

$$\cos(\theta) = \frac{\hat{u}^T u}{\|\hat{u}\|\|u\|}. \tag{181}$$

Since $\cos(|\theta|) \ge 0$, it follows that $\hat{u}^T u \ge 0$ and the SNMF low rank approximation is

passive. $\square$

Corollary 8: The SNMF rank-one approximation belongs to the sector $[0, \infty]$.

Proof: This follows directly from the definition of passivity. $\square$

As with the SVD System, the fact that the SNMF rank-one approximation is

passive leads directly to two stability theorems that are less conservative than the small

gain condition. The proofs of the following theorems can be found directly in Khalil and will not be repeated here [43].

Theorem 18: If the system defined by CH in (171) is passive, then the feedback system in (171) is passive, and, furthermore, if the storage function of CH is positive definite, then the origin of the closed loop system is Lyapunov stable.   □

Theorem 19: If the system defined by CH in (171) is strictly passive and time-invariant, then the origin of the closed loop system in (171) is uniformly asymptotically stable. If the storage function for CH is radially unbounded, then the closed loop system is globally uniformly asymptotically stable.   □

Therefore, although the stability results for the SNMF System are more conservative than those for the SVD System, they still present useful methods by which to guarantee stability, particularly the passivity condition of Theorem 17.

CHAPTER 6

OPEN LOOP AND COMMAND GENERATION TECHNIQUES

This chapter discusses open loop or command generation techniques used to define a set of intermediate reference commands that build upon one another to generate the desired reference. These techniques rely on pre-existing knowledge of the system to create the commands. In many systems, such as Digital Clay, feedback is needed due to system variations and the effect of noise and disturbances. This chapter describes the techniques in terms of surface generation of a pin array, but they apply generally to any system. One of the advantages of the line scanning technique discussed in Chapters 1 and 2 is that closed loop control can be applied independently to each cylinder when it is in motion. For simultaneous motion of the pins, the feedback must account for the row-column multiplication, as well as other system constraints. This feedback control has been described in Chapters 4 and 5. One of the challenges with these control techniques is rapid switching between various rank-one control inputs that can occur for reference commands representing desired surfaces of rank greater than one. This can be seen in the switching of the singular vectors for the SVD System in Figure 16 on page 38 and in the switching of the control input for the SNMF System in Figure 18 on page 39. This rapid switching of the control input can cause undesirable oscillations in the subsystems. However, if the difference between the reference commands and the current outputs is rank-one, then the response does not exhibit these oscillations because the constraint on the inputs has no effect. The command generation procedures presented in this chapter theoretically eliminate these oscillations by decomposing a desired surface into a series of

rank-one reference commands with respect to the current output. Also, the command generation techniques apply directly to the case of open loop control. This chapter will discuss the command generation algorithms, how they could be implemented using Digital Clay, and will present kinematic simulations comparing the techniques to line scanning.

Command Generation

The command generation procedure for a general system using the row-column structure can be expressed as a summation of intermediate surfaces or, mathematically, as a sum of rank-one matrices of the form

$$S_{des} = S_0 + \sum_{i=1}^{k} S_i = S_0 + \sum_{i=1}^{k} t_i u_i v_i^T. \tag{182}$$

S_{des} is the desired surface expressed as an $m \times n$ matrix of position values, k is the number of intermediate surfaces, S_i represents the ith rank-one intermediate surface, and t_i is the time required to generate the ith intermediate surface. This idea was first suggested by Zhu, although he made no attempt to develop the concept [20]. This representation is based on the assumption that the row and column signals multiply to form the command to each pin. Physically, this can be thought of as $u_i v_i^T$ being a velocity command, which generates an intermediate position $S_i = t_i u_i v_i^T$. S_0 represents the initial position. The intermediate surfaces build on one another to generate a desired surface. The goal is to find the sequence of S_i's such that the generation time for the final surface, S_{des}, is reduced relative to the line scanning technique and the pins move collectively and not line by line. In general, the velocity command for each cylinder is an outer product of the row and column inputs, $V = u_i v_i^T$.

In this chapter, the SVD and SNMF are used to decompose a desired surface into sums of rank-one intermediate surfaces. These decompositions will create the commands u_i and v_i to generate each intermediate surface. If feedback control is necessary, then these open loop procedures can be applied as command generators for the SVD and SNMF Systems. While the SVD is generally preferable for this task, as will be shown, when there is a nonnegativity constraint, as with Digital Clay, a modified version of the SVD or the SNMF can be used.

The SVD Procedure

One solution to the problem of generating rank-one intermediate surfaces is given by the SVD of the desired surface, as shown in Figure 25. Given the SVD defined by (31) on page 24, the matrix, $u_i v_i^T$, can be scaled so that $V = \alpha u_i v_i^T$. V is the velocity matrix and α is a scaling factor so that the absolute value of the maximum element in V represents the maximum velocity of the cylinders. By also dividing the singular values by α, the intermediate surfaces can be generated by the SVD,

$$S_{des} - S_0 = \sum_{i=1}^{r} \left(\frac{\sigma_i}{\alpha}\right) \alpha u_i v_i^T = \sum_{i=1}^{r} t_i V_i, \tag{183}$$

so that the intermediate surfaces are $\sigma_i u_i v_i^T$, and the total time to reach the final surface is

$$t_{tot} = \sum_{i=1}^{r} \left(\frac{\sigma_i}{\alpha}\right). \tag{184}$$
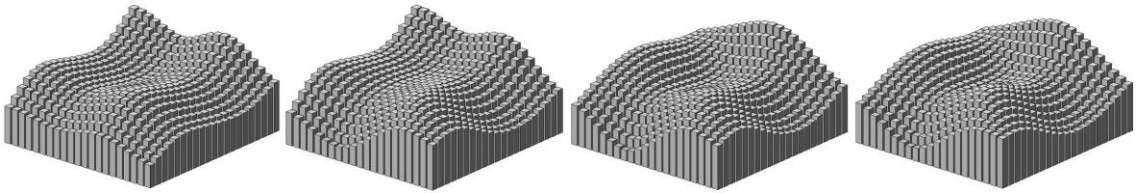
Figure 25 A rank-4 surface as produced by the SVD procedure from the first intermediate surface (*left*) to the final surface (*right*).

The above SVD procedure works for any system with four-quadrant row-column multiplication. However, for the specific application to Digital Clay, as previously discussed, it is not economically feasible to manufacture four-quadrant row-column multiplication. However, unlike with feedback control, a modification to the SVD can constrain the intermediate surfaces to meet the nonnegativity condition.

The Modified SVD Procedure

The modified SVD (MSVD) procedure works as follows: each intermediate surface with a negative v_i value is separated into two sub-intermediate surfaces, one using the positive resistance values and another using the negative resistance values. For the sub-intermediate surface created using the negative resistance values, the row and column commands are multiplied by -1 so that the resistance values become positive but the cylinders still move in the correct direction. For example, if an intermediate surface is defined by

$$u_i^T = [0.3 \quad 0.5 \quad -0.8] \text{ and } v_i^T = [0.3 \quad -0.5 \quad 0.8], \tag{185}$$

the intermediate surface is split into two sub-intermediate surfaces, one using the positive resistance values,

$$u_i^T = [0.3 \quad 0.5 \quad -0.8] \text{ and } v_i^T = [0.3 \quad 0 \quad 0.8]. \tag{186}$$

The other sub-intermediate surface uses the negative resistance values,

$$u_i^T = [-0.3 \quad -0.5 \quad 0.8] \text{ and } v_i^T = [0 \quad 0.5 \quad 0]. \tag{187}$$

This MSVD procedure can be applied directly to Digital Clay. However, as described below, dividing intermediate surfaces that have positive and negative column values requires more time to generate the desired surface as more steps are added.

Therefore, another way to account for the nonnegative constraints of Digital Clay is to use the SNMF in place of the MSVD.

The SNMF Procedure

Unlike the SVD, the SNMF naturally incorporates the nonnegativity constraints of the column commands. Suppose the SNMF of S_{des} is written as $S_{des} = WH^T$ such that $W \in \mathbb{R}^{n \times k}$, $H \in \mathbb{R}^{m \times k}$, and $H \geq 0$. Then intermediate surfaces can be generated as

$$S_{des} - S_0 = \sum_{i=1}^{k} S_i = \sum_{i=1}^{k} w_i h_i^T, \tag{188}$$

where w_i and h_i represent the ith columns of W and H, respectively. Note that column commands, h_i, are constructed to be nonnegative.

One challenge is that using successive rank-one approximations and computing all the factors simultaneously does not yield the same results when using the SNMF, as when using the SVD. A comparison of these two methods has been made with regards to surface generation, using as an example a 50×50 matrix whose rank is 40. The error norm, defined as the norm of the difference between the accumulation of the intermediate surfaces and the desired surface, is plotted in Figure 26 for an increasing number of intermediate surfaces. Although the simultaneous approximation method achieved a smaller error after utilizing all 40 surfaces, the intermediate surfaces generated by that method produced large errors. In contrast, the intermediate surfaces generated by the successive rank-one approximations minimized the error norm more quickly and monotonically.
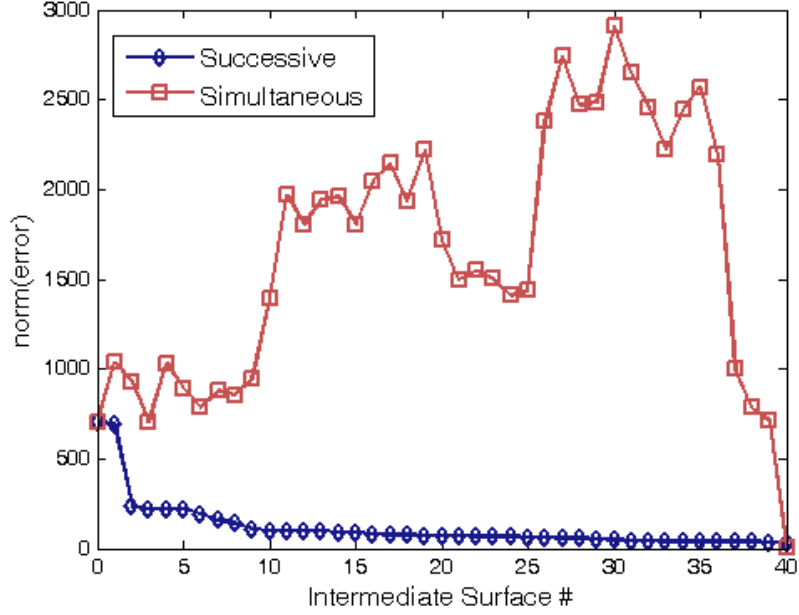
Figure 26 Error convergence for k successive rank-1 approximations and one simultaneous rank-k approximation.

Therefore, the SNMF surface generation procedure is obtained by computing successive rank-one approximations as follows: at each stage, if the error between the current accumulation of intermediate surfaces and the desired surface is given as

$$E_i = S_{des} - \left(S_0 + \sum_{j=1}^{i-1} S_j \right),$$
(189)

find a rank-one approximation of E_i subject to nonnegativity constraints on the column inputs. This amounts to solving (43) on page 30 with E_i in place of M.

One remaining concern of the SNMF procedure is that it is unknown how many rank-one approximations are needed to exactly produce the desired surface. In the example in Figure 26, after using the summation of 40 successive rank-one approximations to a matrix whose rank is 40, there remains a nonzero error. Therefore, either a supplement to the rank-one SNMF approach that enables exact surface generation is needed, or one must accept a slight amount of error. The additional step is simple.

81

Once the norm of the error matrix becomes smaller than a predefined tolerance, the rank-one SNMF approximation is stopped, and line scanning is implemented to reduce the remaining error matrix to zero in as few steps as possible. This strategy integrates the advantages of both the rank-one SNMF approach and the line scanning method. In general, the line scanning method is too slow to be used from the beginning; on the other hand, successive rank-one SNMF approximations are very efficient, but the number of rank-one steps required to achieve an exact surface can potentially be too large. Hence, the SNMF procedure uses successive rank-one SNMF approximations for the most expensive steps, and then line scanning is applied only to an error matrix of a small norm without sacrificing time efficiency. One could also augment the SNMF procedure with the MSVD to generate an exact solution in less time. However, the added time of augmenting with line scanning is very small, generally less than 1% of the total time. The goal primarily is to reduce the number of intermediate surfaces. Since the MSVD can generate up to twice as many intermediate surfaces as line scanning, line scanning is chosen to augment the SNMF in the following simulations. Were the MSVD used instead, the difference in general would be a small percent decrease in time and an increase in the number of intermediate surfaces.

Simulation Examples

A kinematic simulation was used to compare the line scanning, SVD, MSVD, and SNMF procedures. The SNMF procedure incorporated line scanning as previously discussed. The switch from the SNMF to line scanning was set to occur when the maximum error of any pin was within 1% of the total pin stroke. The simulation was purely kinematic, assuming that the pins reached maximum velocity instantaneously.

This permitted a comparison of the procedures regardless of a particular physical system so that the results could be easily extended to other applications. In general, the slower the response of the pins, the greater the time needed per each intermediate surface because the pins would need to accelerate and decelerate at each intermediate surface. Therefore, the goal was to minimize the amount of time and the number of intermediate surfaces.

The results in Table 1 show the name of the surface, its size, and its rank. Also shown is the number of intermediate surfaces, the time for each of the four procedures, and the percent improvement of the new techniques compared to line scanning. The names of the surfaces are descriptive and the surfaces are shown in Appendix A. For example, "Face" is a human face, and "Grid of Sqrs" is a near checkerboard pattern with isolated squares raised to the stroke limit. In the non-rotated version, the squares are aligned with the grid so that the surface is rank-one. The MATLAB "peaks," MATLAB "peaks NZ", and "World Map" surfaces were created using the built-in MATLAB commands, peaks and load topo. For every test, each pin's initial position was zero, except in the case of MATLAB "peaks" NZ, which began with each pin at the midpoint of its stroke. This was done to demonstrate that the techniques work for arbitrary initial positions. For every surface, at least one pin was set to the stroke limit so that the minimum possible time to generate the surface would be 0.5 seconds, the exception being MATLAB "peaks" NZ, for which the minimum time is 0.25 seconds.

The SVD procedure without modification was shown to generate surfaces up to 20 times faster than the line scanning method, and would be even faster for larger arrays and lower rank surfaces. In general, the performances of the SVD, MSVD, and SNMF

are rank dependent, whereas line scanning is size dependent. Using the MSVD instead of the SVD increased the amount of time and the number of intermediate surfaces. However, with the MSVD, the number of intermediate surfaces is guaranteed to remain below 2 times the number generated by the SVD, and the time likewise would not be more than twice as long. As shown below, the surface generation times for the MSVD were still much faster than line scanning.

The SNMF method was faster in some cases and slower in others compared with the MSVD technique. In the identity case, the SNMF method was faster than the SVD. That shows that no method offers a minimum-time solution to the general problem, but clearly all of the new procedures are preferable to line scanning. The number of intermediate surfaces used by the SNMF method was very large in some cases. This is partly due to augmenting the SNMF with line scanning. There is a design trade-off between time, accuracy and number of surfaces associated with augmenting the SNMF with line scanning. The designer must decide whether a small amount of error is tolerable or whether to supplement the SNMF with line scanning. Then, he or she must decide when to switch from the SNMF to line scanning, balancing the time and number of intermediate surfaces. Switching to line scanning increased the time by less than 1%. Another important observation is that the performance of the SNMF and the MSVD did not significantly decrease relative to the SVD for the surface MATLAB "peaks" NZ, where the pins had to move in both positive and negative directions.

Table 1 Comparison of line scanning, SNMF, SVD and MSVD.

Name	Size	Rank	Line scanning proc. # intermed. surfaces	time (sec)	SNMF procedure # intermed. surfaces	time (sec)	% Improvmt. on Line scan
Identity x50mm	3x3	3	3	1.5	4	1.2	17.2
Gradient	4x4	4	4	1.4	10	0.7	50.9
Parabola	50x50	1	50	6.1	2	0.5	91.9
Parabola w/ noise	50x50	19	50	6.3	84	1.1	83.1
Grid of Sqrs	50x50	1	50	12.0	51	0.5	95.8
Grid of Sqrs Rotated	50x50	40	50	24.5	173	8.6	64.9
Matlab "peaks"	49x49	4	49	15.4	60	0.9	94.0
Matlab "peaks" NZ	49x49	4	49	5.0	59	0.7	85.4
Face	110x82	82	82	35.0	221	3.7	89.4
World Map	180x360	180	180	68.0	545	6.6	90.3

Name	Size	Rank	SVD procedure # intermed. surfaces	time (sec)	% Improvmt. on Line scan	Modified SVD procedure # intermed. surfaces	time (sec)	% Improvmt. on Line scan
Identity x50mm	3x3	3	3	1.5	0.0	3	1.5	0.0
Gradient	4x4	4	3	0.6	55.9	5	0.7	51.3
Parabola	50x50	1	1	0.5	91.9	1	0.5	91.9
Parabola w/ noise	50x50	19	19	0.8	87.5	37	1.0	83.8
Grid of Sqrs	50x50	1	1	0.5	95.8	1	0.5	95.8
Grid of Sqrs Rotated	50x50	40	40	5.9	75.8	78	9.6	60.9
Matlab "peaks"	49x49	4	4	0.7	95.5	7	0.9	94.2
Matlab "peaks" NZ	49x49	4	4	0.5	90.6	8	0.6	88.0
Face	110x82	82	82	2.2	93.8	163	3.6	89.8
World Map	180x360	180	180	3.7	94.6	359	6.3	90.7

Command Generation

At first, the SVD and SNMF open loop procedures may not appear to be directly applicable to Digital Clay or similar systems where feedback control is necessary. However, they provide potentially useful command generators for those systems. As mentioned, the SVD and SNMF Systems can excite undesirable oscillations in systems with flexibility. For example, consider a response to a random set of inputs shown in Figure 27. Looking closely at the detailed plot on the right, it can be seen that the system is tracking minor oscillations caused by the changing values of the singular vectors for the SVD System and the input in the low-rank factors for the SNMF System. In systems with flexibility, such periodic excitations may be undesirable. In using the SVD or

SNMF procedures described in this chapter to generate commands, the input would be a rank-one or nearly rank-one matrix, resulting in smoother subsystem response. The command generation procedure can be expressed in a block diagram form as in Figure 28. The block diagram for the SNMF System and command generation takes the same form. The error feedback is used to determine when one intermediate surface has been reached and, accordingly, when to switch the command to the next intermediate surface.
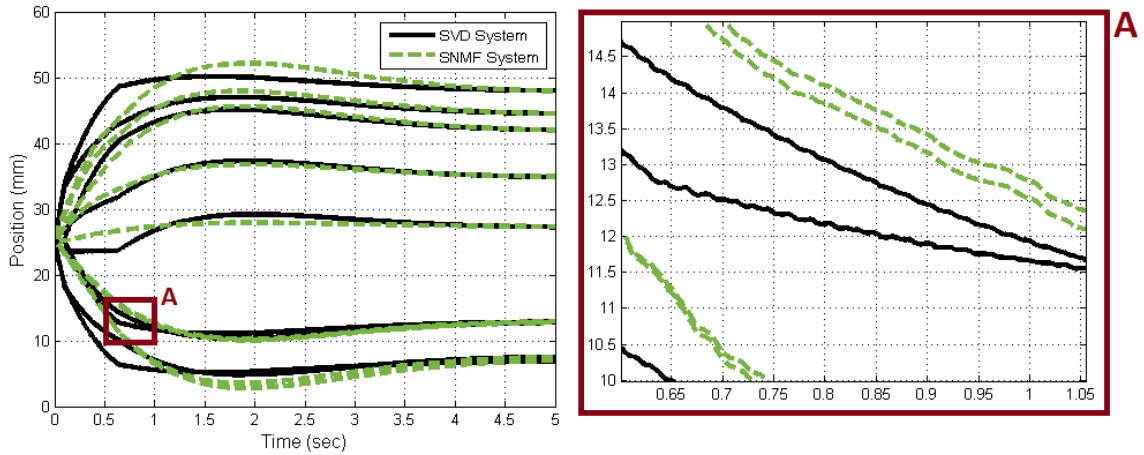
Figure 27 Response of the SVD and SNMF Systems to a set of random inputs (*left*) and a detailed plot of the oscillations (*right*) caused by the rank-one approximation.
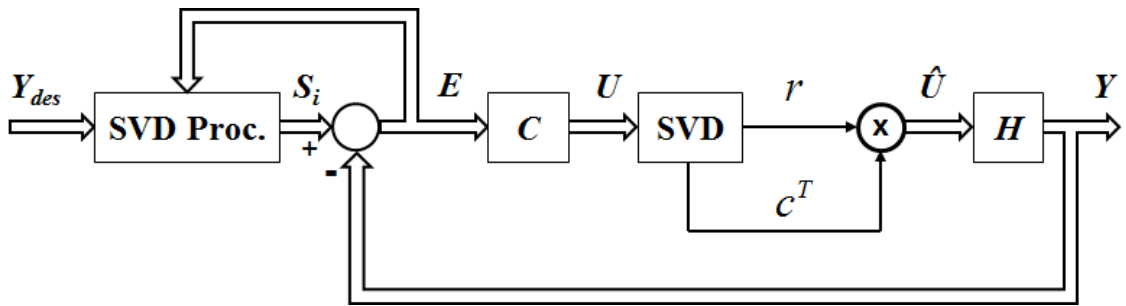
Figure 28 Command generation for the SVD System using the SVD procedure.

As a demonstration of the command generation procedures, consider a 3×3 grid with LTI subsystems described by

$$G(s) = \frac{1}{s(s+2)}. \tag{190}$$

Using a proportional controller with $k = 1$, the response of the SVD System to a step command to

$$Y_{des} = \begin{bmatrix} 40.74 & 45.67 & 12.92 \\ 45.29 & 31.62 & 27.34 \\ 6.35 & 4.88 & 47.88 \end{bmatrix} \quad (191)$$

is shown in the left plot in Figure 29. The SVD procedure creates three, rank-one intermediate surfaces resulting in the response shown in the right plot in Figure 29. Notice that the oscillations present in the response without command generation are removed by using the command generation procedure, and that this change is accomplished without significantly increasing the time required to converge. More details will be given on the convergence rate of these procedures in the next chapter focusing on the system response.

Figure 29 The response of the SVD System (*left*) and the SVD System with command generation (*right*).

CHAPTER 7

SYSTEM RESPONSE

This chapter examines the effect of dimension reduction on the response of a system. The focus is on systems comprised of a set of dynamically equivalent linear subsystems of various orders. The chapter is divided into three main sections. The first focuses on the SVD System, the second on 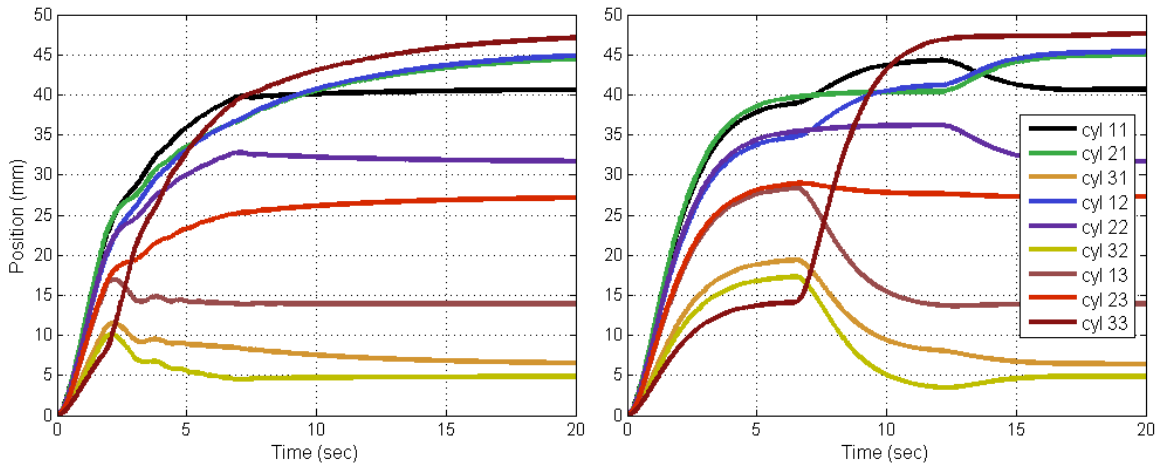the SNMF System, and the third considers the open loop techniques discussed in Chapter 5. The stability theorems proposed in Chapter 5 also will be evaluated and other effects of the dimension reduction will be examined, such as its effect on steady-state error.

The SVD System

First-Order Subsystems

This section verifies the stability and convergence conditions of the SVD System for first-order subsystems. Further, the stability and convergence rate of the SVD System is explained by analyzing the response of the singular values of the states, and the convergence rate determined by Corollary 4 is confirmed. Finally, the issue of steady-state error is examined for the tracking problem.

To begin with, consider a 3×3 grid of subsystems, each defined by the stable first-order system,

$$\dot{x}_{ij}(t) = -2x_{ij}(t) + u_{ij}(t)$$
$$y_{ij}(t) = x_{ij}(t). \tag{192}$$

The SVD System will be compared with independent control (IC). Consider first the regulation problem, where the goal is to return to the origin. State feedback is used, as

seen in   Figure 10 and Figure 11 on page 34, where $G(s)$ is the transfer function

form of the system in (192). Using state feedback for control, stability of this system can

be analyzed using Theorem 12 on page 54. For both the SVD System and for IC, it is

easily verified that the system is asymptotically stable for $k > -2$, and this is also seen in

the response of the simulated system. For $k < -2$, both systems are unstable.

The convergence of the SVD System is also compared with IC. Choosing $k = 5$

as the gain for IC, then by Corollary 4 on page 57, $k_{\text{SVD}} \geq nk$ must be chosen for the

convergence of the SVD System to be bounded by that of IC. This is an upper bound on

the convergence that is met when all of the singular values have the same magnitude. The

convergence rate of the SVD System relative to IC can be more generally understood by

comparing the two using initial conditions with singular values of different relative

magnitudes.

The response of $\|x\|_2^2$ for the SVD System and IC is shown in Figure 30 for three

types of initial conditions. The first initial condition is a rank-one matrix, the second

initial condition is a set of random values, which is full-rank and has unequal singular

values, and the third initial condition is $X = \alpha I$, where I is the identity matrix and α is a

constant. If $k_{\text{SVD}} = k$, shown in the left figures, then it can be seen that the convergence

of $\|x\|_2^2$ is the same for both the SVD System and IC for the rank-one initial condition.

The convergence for the SVD System becomes slightly slower for the random initial

condition and is slowest for the identity matrix initial condition. If $k_{\text{SVD}} = 3k$, thus

meeting the condition in Corollary 4, then the SVD System converges faster than IC for

the rank-one and random initial conditions, and both systems converge at the same rate

for the identity matrix initial condition.
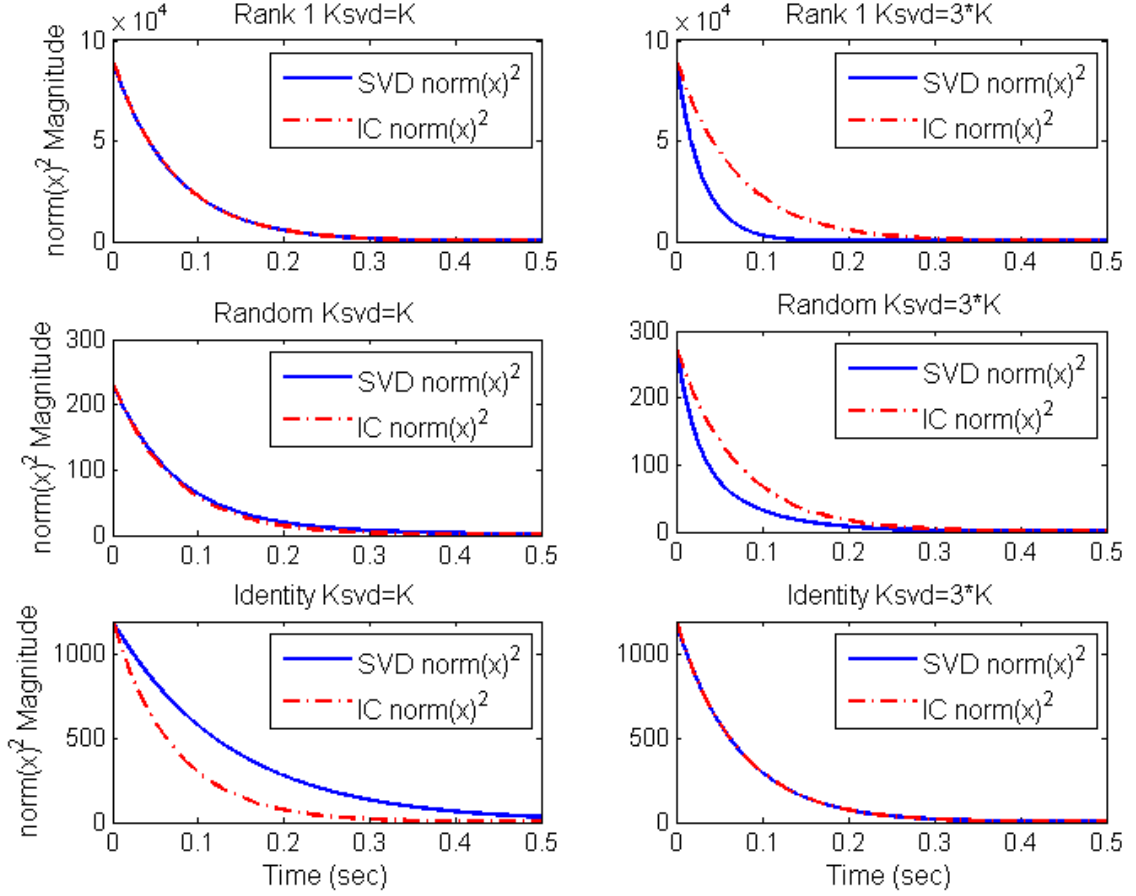
Figure 30 The response of $\|x\|_2^2$ for the SVD System and IC with $a = -2$, $k = 5$ and for $k_{SVD} = k$ (*left*), and $k_{SVD} = 3k$ (*right*).

Second, consider a 3×3 grid of unstable subsystems defined by

$$\dot{x}_{ij}(t) = 2x_{ij}(t) + u_{ij}(t)$$

$$y_{ij}(t) = x_{ij}(t). \tag{193}$$

This system can be stabilized using state feedback with $k > 2$ for the IC system. As predicted by Corollary 5, the SVD System requires a gain of $k_{SVD} > 6$ to be asymptotically stable.

The stability and convergence of the SVD System for both (192) and (193) can be further understood by observing the behavior of the singular values of the matrix of the subsystems' errors, which, in the case of regulation of first-order subsystems, is equivalent to the states. The ith singular value of the error will be referred to as σ_i^e. The

90

convergence of the singular values of the error of the SVD System can be understood as a combination of the convergence of the singular values of the error of the forced and free responses of the subsystems using IC. The relationship of the convergence rate of the SVD System and the free response of the subsystems is most easily observed when $k_{\text{SVD}} = k = 0$. Clearly, for the IC system, the subsystems remain stationary. Using the SVD System, however, the system does respond by the free response of σ_2^e and σ_3^e. While σ_1^e is stationary, σ_2^e and σ_3^e converge to zero for the system in (192), and they diverge for the system in (193). That is because the control gain does not act on σ_2^e and σ_3^e, so they follow the subsystem's free response trajectory.

When $k_{\text{SVD}} = k \neq 0$, it will be observed in the subsequent examples that σ_1^e of the SVD System follows the trajectory of σ_1^e of the forced response of the IC system, and the rest of the singular values of the SVD System follow the trajectory of the singular values of the free or unforced response of the subsystems. This explains why the stability bound for the SVD System is the same as IC when the subsystems are stable. It also helps to explain the relative convergence rates of the SVD System and IC.

If the subsystems themselves are stable, as in (192), then the convergence of the smaller singular values by the free response allows the SVD System to converge more quickly. Therefore, if the control gain is sufficiently small, the convergence rate of the SVD System is nearly identical to IC. Consider the case where $k_{\text{SVD}} = k = 0.2$. Given a set of random initial conditions, the singular values of the error for both systems are shown in Figure 31. In that case, σ_1^e for both systems converges at the same rate since, for the SVD System, σ_1^e does not converge to the same magnitude as σ_2^e during the time shown in Figure 31. Furthermore, σ_2^e and σ_3^e for the IC system converge more quickly

than for the SVD System, but the overall convergence of both systems is dominated by σ_1^e so that they have nearly the same settling time. Using a more aggressive gain, $k_{\text{SVD}} = k = 3$, causes the singular vectors of input to the SVD System to change direction when σ_1^e converges to the same magnitude as σ_2^e, slowing the convergence rate as shown in Figure 32. The SVD System's response is dependent on the relative convergence of the forced and unforced responses of the subsystems in addition to the relative magnitudes of the singular values.
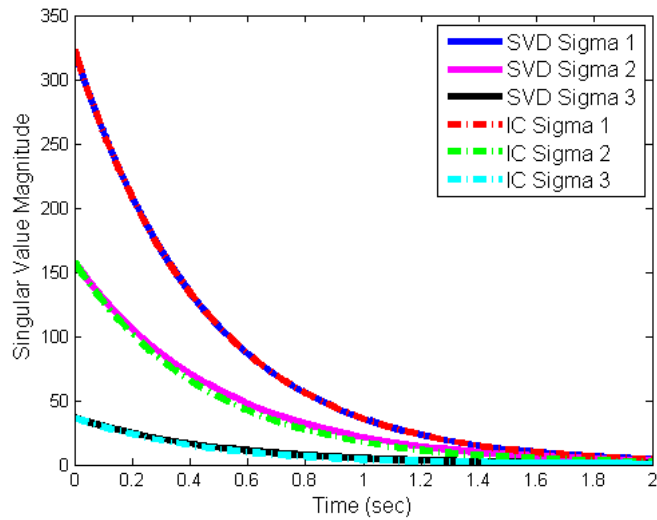

Figure 31 Singular values of the error of the IC system and the SVD System with $a = -2$ and $k_{\text{SVD}} = k = 0.2$.
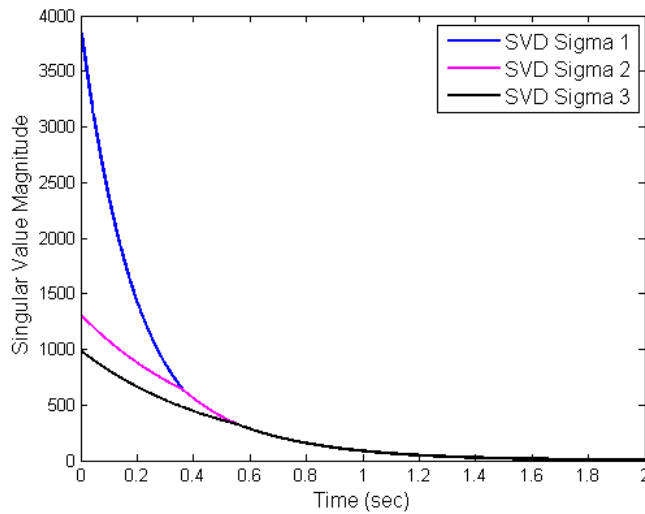

Figure 32 Singular values of the error of the SVD System for regulation with $a = -2$ and $k_{\text{SVD}} = 3$.

The relationship between the SVD System and the free and forced responses of the IC system can also be used to explain the response of the SVD System for unstable subsystems. If the subsystems themselves are unstable, as in (192), then the smaller singular values of the SVD System diverge initially. When these divergent singular values reach the same magnitude as σ_1^e, the SVD System must be able to force each singular value to converge by switching between them, and, therefore, it must have a gain required to stabilize each singular value times the number of singular values. This explains why $k_{\text{SVD}} \geq nk$ is the condition for stability in Theorem 12 for the SVD System when the subsystems are unstable. Hence, for stability of the SVD System for the subsystems in (192), the control gain is $k_{\text{SVD}} > 3a = 6$. Furthermore, just as using a higher-rank approximation reduces the gain in the stability condition, $k_{\text{SVD}} > (n/r)a$, the magnitude of the control gain of the SVD System that results in instability can be used to predict when the system will diverge. For example, if the gain is $4 < k_{\text{SVD}} < 6$, then σ_1^e converges up to the point where all three singular values have the same magnitude, and then it diverges. However, if the gain is $2 < k_{\text{SVD}} < 4$, then σ_1^e converges to the point where the first and second singular values are equivalent and then they both diverge. This is shown in Figure 33. Clearly, if the initial conditions are rank-one or rank-two, then the stability condition can be relaxed to $k_{\text{SVD}} > -2$ or $k_{\text{SVD}} > -4$, respectively, but this is unlikely to hold in a physical system, because even if the system begins at a low-rank initial condition, differences between the subsystems, disturbances, and noise, among other factors, will result in a higher rank error that will necessitate the stability condition given in Theorem 12.
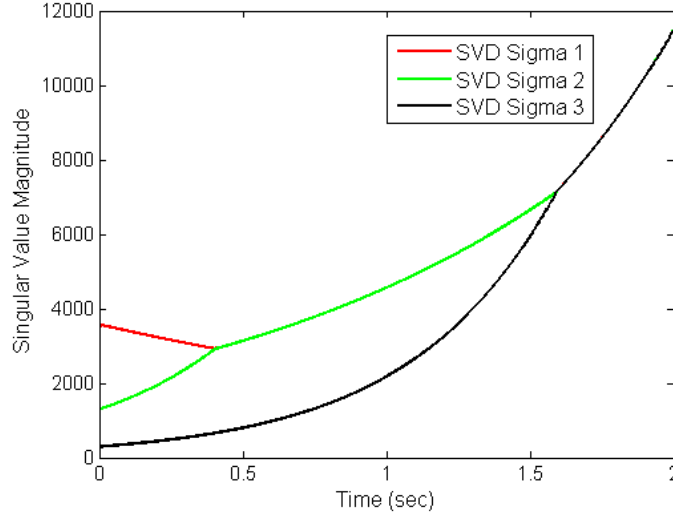
Figure 33 Singular values of the error of the SVD System for regulation with $a = 2$ and $k_{\text{SVD}} = 2.5$.

Now consider the problem of tracking a reference command. The SVD System can be represented by the block diagram in Figure 34. For the tracking problem, the error is not the same as the state and, instead, is defined by $Y_{des} - Y$. The reference command considered is a step response to a set of random values. Assuming the subsystem dynamics in (192), and with $k_{\text{SVD}} = k = 3$ and $k_o = 1$, both the SVD System and the IC system have some nonzero steady-state error. However, the SVD System has significantly more steady-state error because only σ_1^e converges, whereas all the singular values converge somewhat for the IC system. The lower singular values of the error for the SVD System do not converge because the control input, U, only acts in the direction of the first singular value. The singular values of the control input will be referred to as σ_i^{ci}. Figure 35 shows the singular values of the control input. Notice that σ_1^{ci} never converges to be the same magnitude as σ_2^{ci}. Therefore, the control input never changes directions and only acts in the direction of σ_1^{ci}. Setting $k_o = 5$ removes the steady-state error for the IC system and in σ_1^e for the SVD System. However, the SVD System retains some steady-state error because increasing k_o increases the magnitude of every σ_i^{ci}

94

proportionally. Thus, σ_1^{ci} will never converge to the same magnitude as σ_2^{ci}, and the error

of the SVD System will never converge in the directions of the lower singular values

regardless of the size of k_o. This demonstrates the need for a free integrator in the

forward path transfer function of the subsystems for the SVD System when using state

feedback. By adding integral control, as in Figure 36, the steady-state error in both

systems is removed.
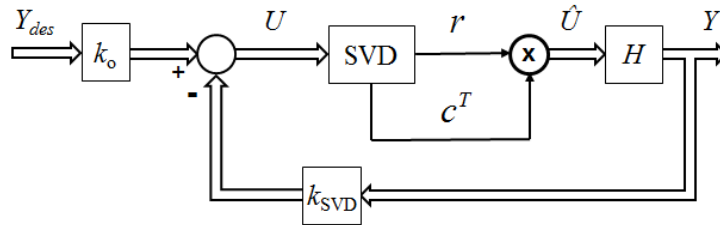

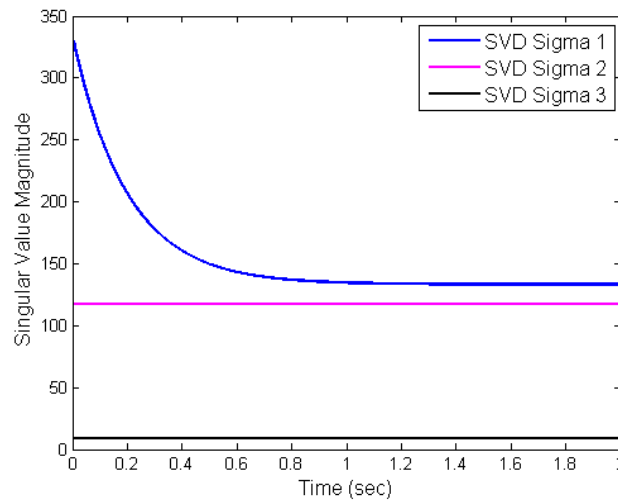**Figure 34** SVD System for reference tracking with state feedback.

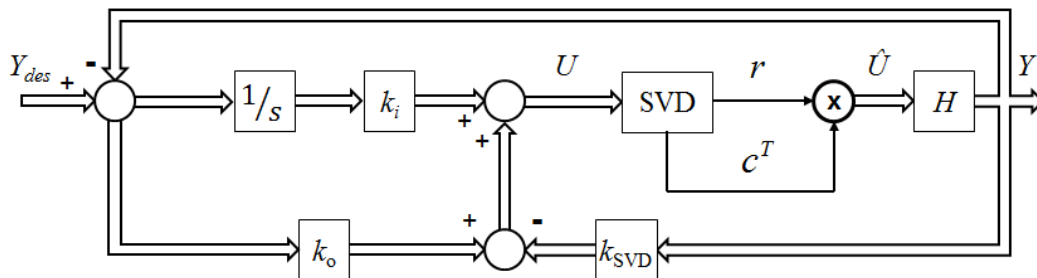**Figure 35** Singular values of the control input of the SVD System for reference tracking.

Figure 36 SVD System for reference tracking with integral control.

Higher-Order Subsystems

This section verifies the stability and convergence conditions for the SVD System for higher-order subsystems. As with first-order subsystems, the stability and convergence rate of the SVD System can be understood by analyzing the response of the singular values.

For higher-order subsystems, the stability analysis of Theorem 14 will first be examined. Consider a set of subsystems defined by

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -3 & -5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x. \tag{194}$$

If state feedback is used, $u = -kx$, then, using Theorem 14, stability for the SVD System is guaranteed by finding $P = P^T > 0$ to solve the LMI in (129) on page 62. This can be done using MATLAB software packages. For example, if k is chosen such that the IC subsystems have closed loop poles of -4 and -10, then the subsystems' feedback gains are $k = \begin{bmatrix} 37 & 9 \end{bmatrix}$. Solving the LMI problem returns a positive definite P such that the eigenvalues for the various Lyapunov functions are

$$\lambda\{A^T P + PA\} = -0.68, -0.25,$$
$$\lambda\{(A - Bk)^T P + P(A - Bk)\} = -7.5, -0.28, \text{ and} \tag{195}$$
$$\lambda\{(A - BMk)^T P + P(A - BMk)\} = -0.68, -0.25, -7.5, -0.28, \forall x,$$

with various multiplicities. The grid size only changes the multiplicities of the eigenvalues. However, as long as the subsystems are diagonalizable, since the entire state equation is block diagonal, it will also be diagonalizable. In other words, the geometric multiplicity of the eigenvalues will equal the algebraic multiplicity. For example, for a 4×4 grid, the four eigenvalues of the Lyapunov functions of the two linear systems in

96

(195) will have multiplicities of 16. The eigenvalues of the Lyapunov function for the SVD System in (195) will have multiplicities of 12, 12, 4, and 4 respectively. Generally, an $n \times n$ grid of pth-order subsystems will have p eigenvalues with multiplicities of $n^2 - n$, and p eigenvalues with multiplicities of n.

The simulated response of the IC system, the unforced system, and the SVD System verifies the exponential stability of each. As with the first-order subsystems, the convergence of the SVD System is related to the convergence of the unforced and IC systems. To show this, the regulation problem from a set of random initial conditions is studied. Figure 37 and Figure 38 show the singular values of the matrix of the errors of the output of each subsystem, $0 - Y$, for the SVD System compared with the IC system and the unforced system. These singular values are used in conjunction with the singular values of the control input matrix, U, to analyze the effect of the SVD rank-one approximation. The singular values of the control input matrix for the SVD System are shown in Figure 39. The singular values of the error will be referred to as σ_i^e, and the singular values of the control input will be referred to as σ_i^{ci}.
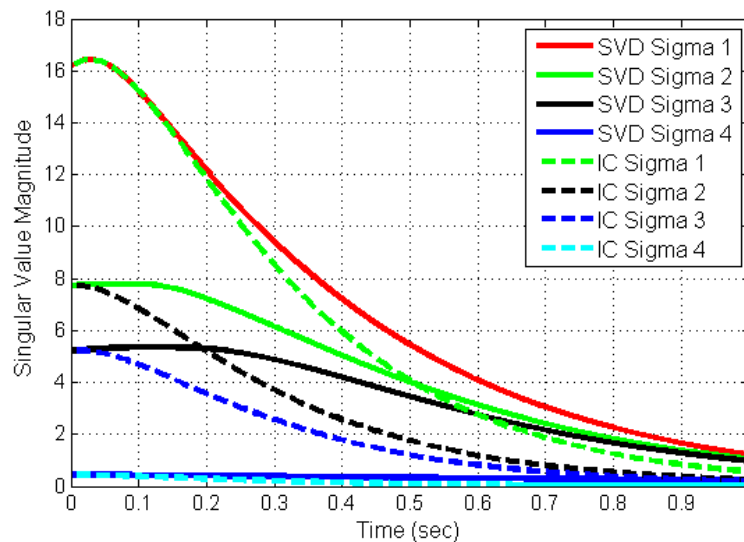
Figure 37 Singular values of the error for the SVD System (solid) and IC system (dashed).
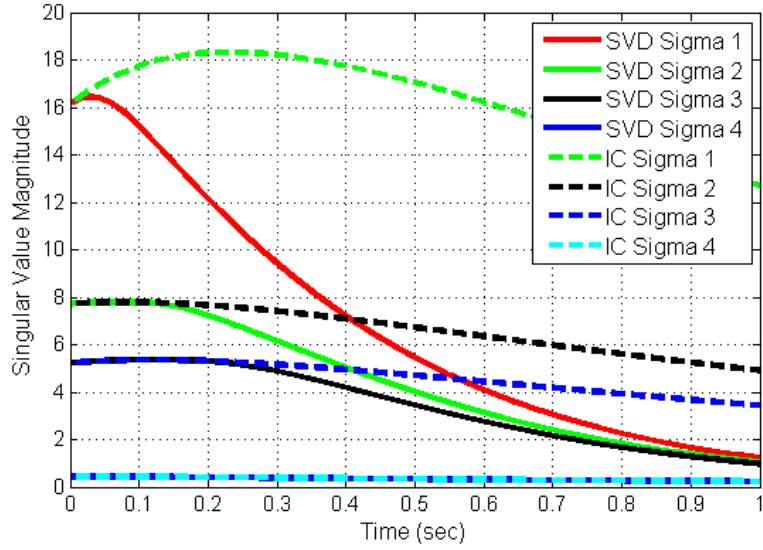
Figure 38 Singular values of the error of the SVD System (solid) and free response (dashed).
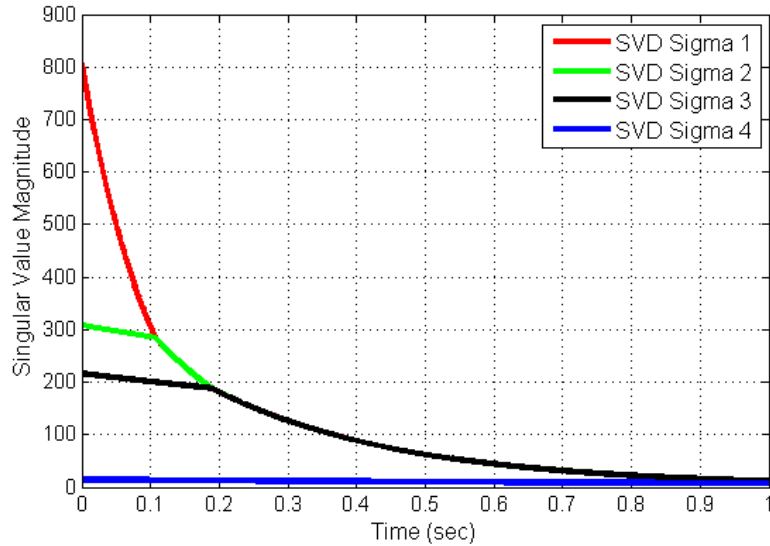
Figure 39 Singular values of the control input of the SVD System.

The important thing to note is the convergence of the SVD System compared with both the IC system's forced response and the subsystems' free response. In Figure 37, σ_1^e of both the SVD System and the IC system follow the same trajectory until $t \approx 0.1$ seconds. In Figure 39, this is the point at which the first and second singular values of the control input matrix, σ_1^{ci} and σ_2^{ci}, of the SVD System are equal in magnitude, or when the

SVD System begins forcing the system in the direction of both σ_1^{ci} and σ_2^{ci}. After that time, the convergence of σ_1^e of the SVD System is slower than that of the IC system.

In Figure 38, it can be seen that σ_2^e of the SVD System follows the trajectory of σ_2^e of the free response until $t \approx 0.1$ seconds. Additionally, σ_3^e of the SVD System follows the trajectory of σ_3^e of the free response until $t \approx 0.18$. Each of these times represents the time at which the σ_1^{ci} of the control input matrix of the SVD System converges to the same magnitude as σ_2^{ci} and σ_3^{ci}. In contrast, σ_4^e of the SVD System follows the trajectory of σ_4^e of the free response throughout the entire shown trajectory because σ_1^{ci} has not yet converged to the same magnitude as σ_4^{ci}. Therefore, the rate of convergence of the SVD System is a combination of the rates of convergence of the free response system and the independently controlled systems, as suggested by the eigenvalues in (195). The same result has also been confirmed for third-order subsystems and for grid sizes of 3×3 and 8×8, but the results are no different and therefore are not included in the interest of brevity.

The downside of Theorem 14 is that a solution to the LMI may not always be found. Then Theorem 14 gives no guarantee of stability even in cases where the SVD System demonstrates a stable response. For example, if the feedback gain is $k = [297 \quad 30]$, then the eigenvalues for all the Lyapunov functions are nearly zero ($\sim 10^{-7}$), and some are positive. However, the SVD System is shown in simulation to converge for a variety of initial conditions. Additionally, the stability analysis of Theorem 14 applies only to systems where A has negative eigenvalues.

For subsystems with positive eigenvalues, it is useful to examine the result of Theorem 15 on page 66. The response of higher-order subsystems with the LQR design is

similar in nature to the response of first-order subsystems in the way that the SVD

System response relates to IC. Consider a 4×4 grid of linear subsystems defined by

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 3 & 5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{196}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x.$$

Designing an LQR controller according to (144) on page 66 with

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } R_i = 1, \tag{197}$$

results in $k = \begin{bmatrix} 6.2 & 11.2 \end{bmatrix}$. If the gain for the SVD System is chosen as $k_{\text{SVD}} \geq 2k$, then

the system converges as predicted by Theorem 15. If the gain for the SVD System is

$k_{\text{SVD}} = 1.9k$, then the SVD System is not stable, as seen in the response of the singular

values of the control input matrix in Figure 40. Initially, σ_1^{ci} converges at the rate of σ_1^{ci}

of the forced IC system. In contrast, σ_2^{ci} and σ_3^{ci} initially diverge as in free response until

they are the same magnitude as σ_1^{ci}. These singular values then converge at a slower rate

until σ_4^{ci} diverges to approximately the same magnitude, at which point they all diverge.

If the gain is chosen such that $1.5k < k_{\text{SVD}} < 2k$, then all of the singular values will

diverge after σ_1^{ci} reaches the magnitude of σ_4^{ci}, as in Figure 40. If it is chosen so that

$1k < k_{\text{SVD}} < 1.5k$, then they will diverge after σ_1^{ci} reaches the magnitude of σ_3^{ci}. This is

shown in Figure 41. For gains of $0.5k < k_{\text{SVD}} < 1k$ or $k_{\text{SVD}} < 0.5k$, they will diverge

after σ_1^{ci} reaches σ_2^{ci} or instantly. This is the same effect as using an approximation

higher than rank-one, as shown in (148) on page 66. This is a similar result as shown for

first-order subsystems, and it demonstrates the similarity between Theorem 15 and
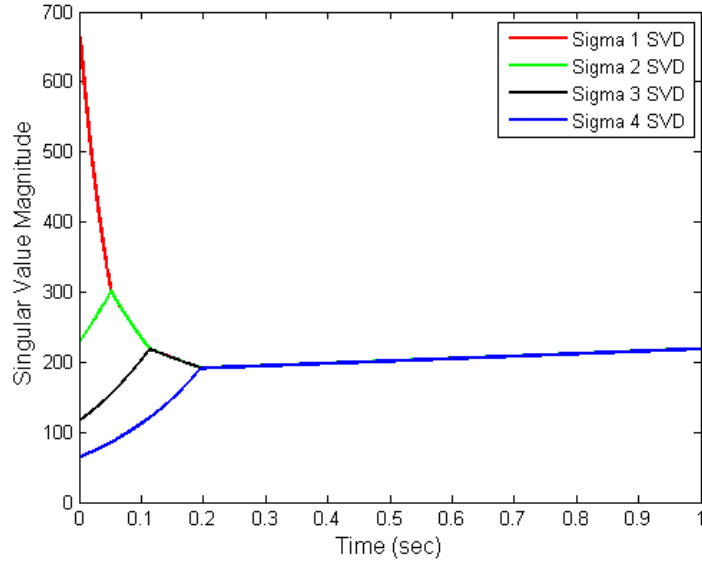
Theorem 12.

Figure 40 Singular values of the control input matrix of the SVD System for $k_{\mathrm{SVD}} = 1.9k$.
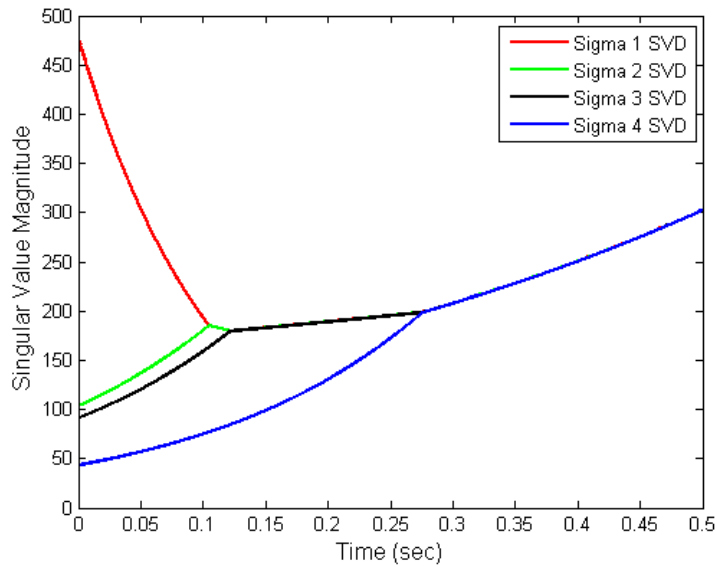
Figure 41 Singular values of the control input matrix of the SVD System for $k_{\mathrm{SVD}} = 1.3k$.

Although Theorem 15 presents a useful criterion for stabilizing unstable subsystems, it is only a sufficient condition. As mentioned in Chapter 5, the condition is not directly dependent on the choice of Q, which has a significant effect on the Lyapunov function. For example, consider an LQR controller for the system in (196) with

$$Q = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix} \text{ and } R_i = 1. \tag{198}$$

101

This choice results in $k = [13.4 \quad 17.3]$. Choosing $k_{\text{SVD}} = 1.5k$ results in an

asymptotically stable response for the SVD System, as shown in Figure 42, in spite of the

fact that it does not meet the condition of Theorem 15. It does meet the condition in (164)

on page 70, but only for the values of x that were tested in the simulation studies. There

is no guarantee of stability for the entire state space or even for a particular domain

within the state space. However, the result does suggest that there is likely a more general

stability condition that can be gained from (164). Not having that condition, the bound for

stability is best determined using the expensive control case of LQR, and the design of

the controller should be done based on Corollary 7 for the performance of the SVD
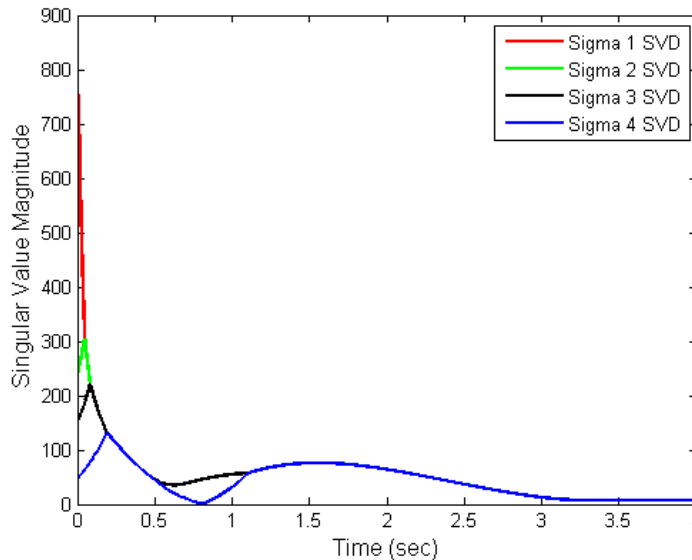
System.

Figure 42 Singular values of the control input matrix of the SVD System for $k_{\text{SVD}} = 1.5k$.

The SNMF System

The SNMF System shares many characteristics in terms of its system response

with the SVD System, such as steady-state error characteristics. However, the guarantees

for stability and convergence are not shared, and, as discussed in Chapter 4, the control

input does not track the singular directions due to the nonnegativity constraint. This

section will provide an example of the use of passivity in designing a stable controller and also will explain how this condition is a conservative one. The techniques used also apply to the SVD System. Decoupled linear subsystems will be used for simplicity; however, nonlinear, coupled subsystems and controllers can also be used. Other examples of the SNMF System will be given in Chapter 8, specifically focusing on its application to Digital Clay.

Consider the second-order LTI subsystem,

$$G(s) = \frac{1}{s(s+3)},$$ (199)

in a 3×3 grid. Using a PD controller with $k_p = 10$ and $k_d = 4$, the transfer function, $G(s)C(s)$, is positive real, or passive. Thus the controller and plant satisfy Theorem 18, and the SNMF System is Lyapunov stable. The response of the system confirms this result, as shown in Figure 43. However, the choice of $k_p = 10$ and $k_d = 2$ does not satisfy the positive real condition, and yet the system response, shown in Figure 44, remains stable. This shows that the stability condition based on passivity is conservative.
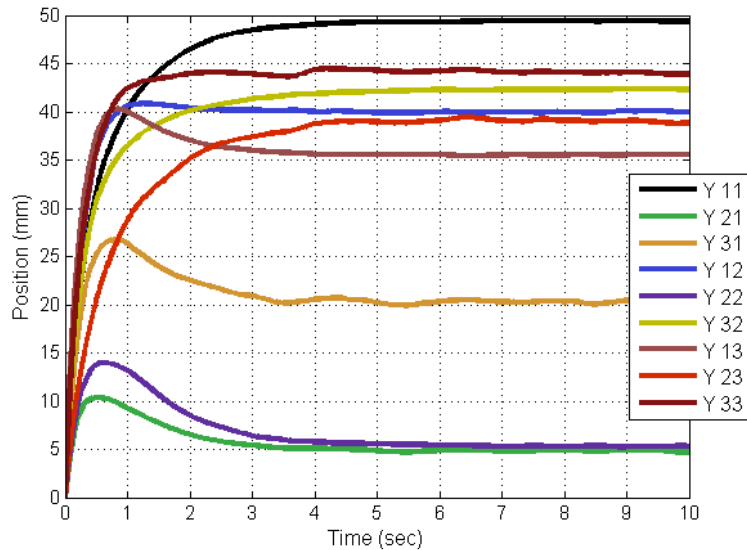

Figure 43 SNMF System response to a step command with $k_p = 10$ and $k_d = 4$.
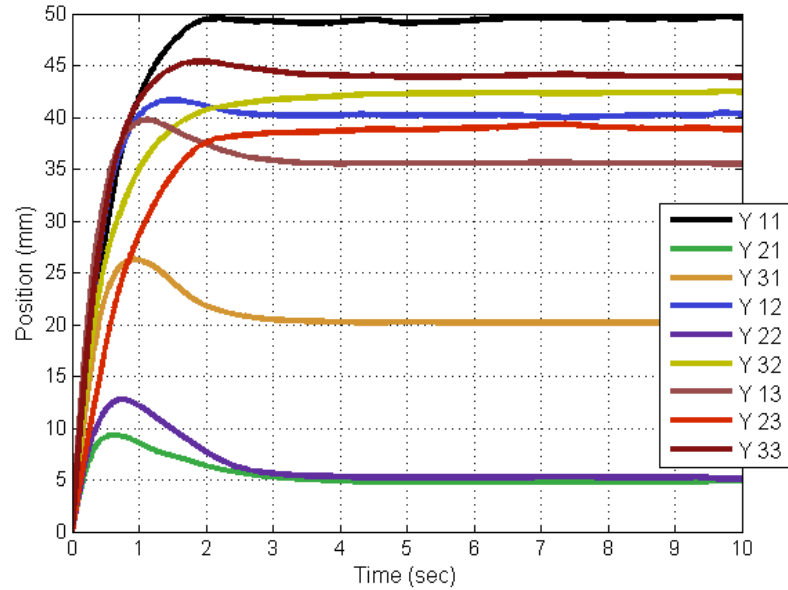
Figure 44 SNMF System response to a step command with $k_p = 10$ and $k_d = 2$.

Open Loop Techniques

Simulations were carried out to compare the various command generation techniques discussed in Chapter 6. A 36×36 grid was used with 72 inputs controlling 1,296 subsystems. The goal was to compare the command generation procedures with one another, with line scanning, and also with simply using the SVD and SNMF Systems or IC without any command generation. Second-order models with one free integrator were used. One dynamic model was controlled to have a faster response with no overshoot and another model controlled to have a slower response with some overshoot. One hypothesis being tested is that the slower the dynamics of the system, the greater the penalty for having more intermediate surfaces.

Four different reference commands were used. Three of these were based on the surfaces used in the kinematic simulations in Chapter 6. The surfaces used were the MATLAB "peaks" and "World Map" surfaces. They were cropped to match the size of the grid. In particular, the "World Map" surface was cropped into two separate test

104

reference commands called "Topo 1" and "Topo 2". A fourth reference command was selected as a random set of values between 0 and 50. The reference commands are shown using surface images in Appendix B. Rather than show the response for every condition and reference command, examples are given to aid the discussion. The focus for comparison will be the settling time and the convergence of the singular values of the error.

The 2% settling time of the maximum singular value of the error was used to compare IC, the SVD System, the SNMF System, the SVD System with SVD command generation (SVD CG), the SNMF System with SNMF command generation (SNMF CG), the SNMF System with modified SVD command generation (MSVD CG), and line scanning. The reason that the SNMF System is used in conjunction with the MSVD rather than the SVD System is that it is impossible to guarantee that the control input will share the same singular vectors as those computed by the MSVD due to model inaccuracies, noise, and other factors. Therefore, if a nonnegativity constraint exists, and the MSVD is designed to be used for that situation, the SNMF System must be used and not the SVD System. The command generators were added to the feedback systems as shown in Figure 28 on page 86. The 2% settling time of the maximum singular value of the error was chosen as the metric for comparison because it captures the convergence of the entire grid and not just individual subsystems. The 2% was with respect to the initial value of σ_1^e of the error.

The first dynamic subsystem used for testing is defined by

$$G(s) = \frac{1}{s(0.01s + 1)}.$$ (200)

To control these subsystems, a proportional controller was used with $k_p = 10$. Figure 45

shows the 2% settling time for all the tests using each of the control strategies. For all of

the reference commands, the line scanning procedure was the slowest and IC was the

fastest. Also, the SVD and SNMF Systems without the command generation procedures

were faster for all tests than those using the command generation. Between the two

feedback control strategies, the SVD System was always faster and was about twice as

fast as the SNMF System, except for the response to the "Peaks" command. The SVD

CG was the fastest command generation procedure, and the SNMF CG was faster than

the MSVD CG. This contrasts with the kinematic studies, which found the MSVD CG to

generally be faster than the SNMF CG. The difference can be accounted for in the use of

the SNMF System to find a rank-one approximation of the input within the feedback

loop. The SNMF is not guaranteed to find the minimum solution, and, therefore, it is not

guaranteed to actuate the system in the direction computed by the MSVD CG. Using the

new control strategies, the vast performance difference between the "Random" command

reference and the other test commands has to do with the rank of the commands, the

relative magnitudes of the singular values, and the magnitude and variation of the

subsystems' reference commands. The "Peaks" command was only rank-four. The other

commands were full-rank, but the singular value magnitudes of "Topo 1" and "Topo 2"

dropped off more quickly than the singular value magnitudes of "Random." This is

different than the line scanning procedure, which performed roughly the same for each

command. Its performance is relative to the size of the grid and the maximum magnitude

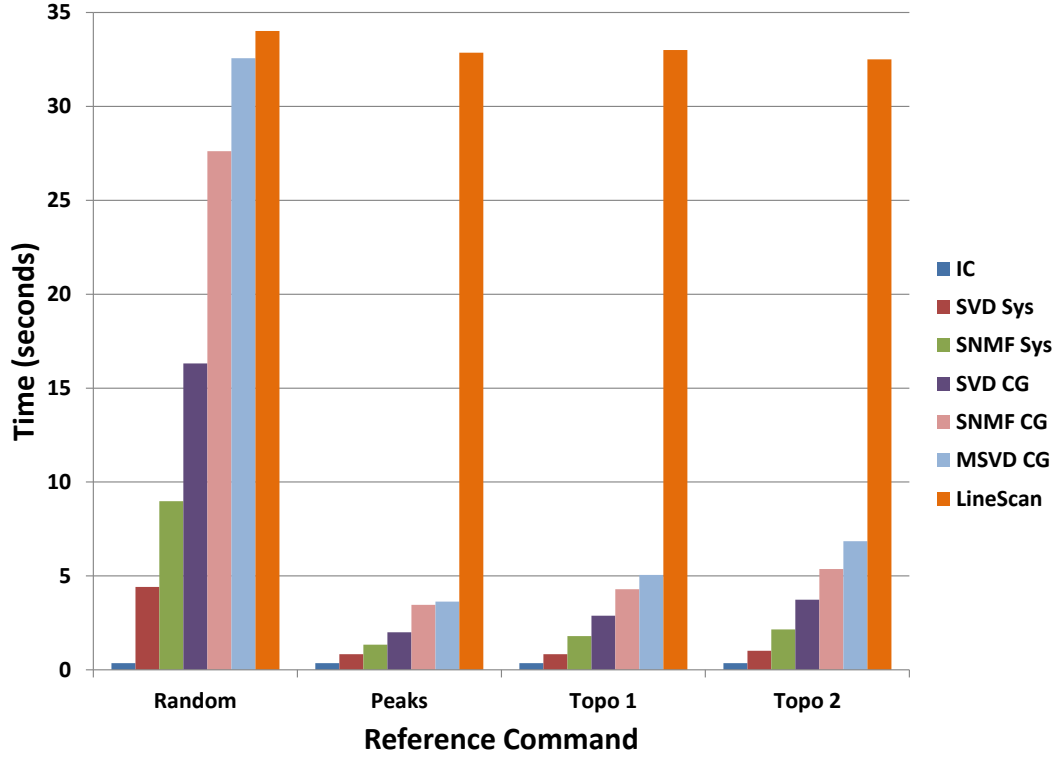of the reference commands in each line.

Figure 45 2% settling time for different reference commands using the various control techniques to control a 36×36 grid of subsystems given in (200).

A second dynamic system was tested with the model

$$G(s) = \frac{1}{s(s+1)}. \tag{201}$$

The controller was a PD controller with $K_p = 15$ and $K_d = 3$. This resulted in an underdamped response for the IC system. Figure 46 shows the 2% settling time of σ_1^e of the position error for all the tests using each of the control strategies. The bars that fade at the top of the chart represent tests that did not converge to within 10% in 30 seconds. For those tests, the magnitude of σ_1^e at 30 seconds is shown in Figure 47. The results for this system are similar to the results for (200) with some slight differences. For the "Peaks" test, the SNMF System is faster than the SVD System. In general, in these tests, the SVD and SNMF System are closer to IC because the switching of the inputs actually reduces the oscillations in the subsystems, allowing them to converge more quickly.

107

As before, the SVD and SNMF Systems without command generation were much faster than with command generation. Moreover, using the command generation techniques for (201) was even slower relative to not using them than they were for (200). That is due to the oscillatory dynamics. Because the command generation techniques do not switch from one intermediate surface to the next until the subsystems have converged, using these techniques requires the system to wait for each intermediate surface to converge before moving to the next. This additional starting and stopping is more time consuming for subsystems with slower dynamics. Additionally, for oscillatory systems, the command generator may switch to the next intermediate surface before the first one has converged. Figure 48 shows the singular values of the error for the Peaks command using the SVD CG. The first 4 seconds are shown in the plot. Notice that the SVD CG does not switch to the second intermediate surface at about 0.4 seconds due to the oscillations preventing the convergence criterion from being met. However, it does switch at approximately 1.2 seconds because the convergence criterion is met in spite of the remaining oscillations. Again, the SVD CG does not switch to the third intermediate surface until about 3.5 seconds due to the oscillations. The same behavior is seen for the other command generators. This presents a challenge in designing these command generators. Switching too fast will negate the use of the generator, and switching too slowly increases the overall convergence time of the system, as seen in these tests.
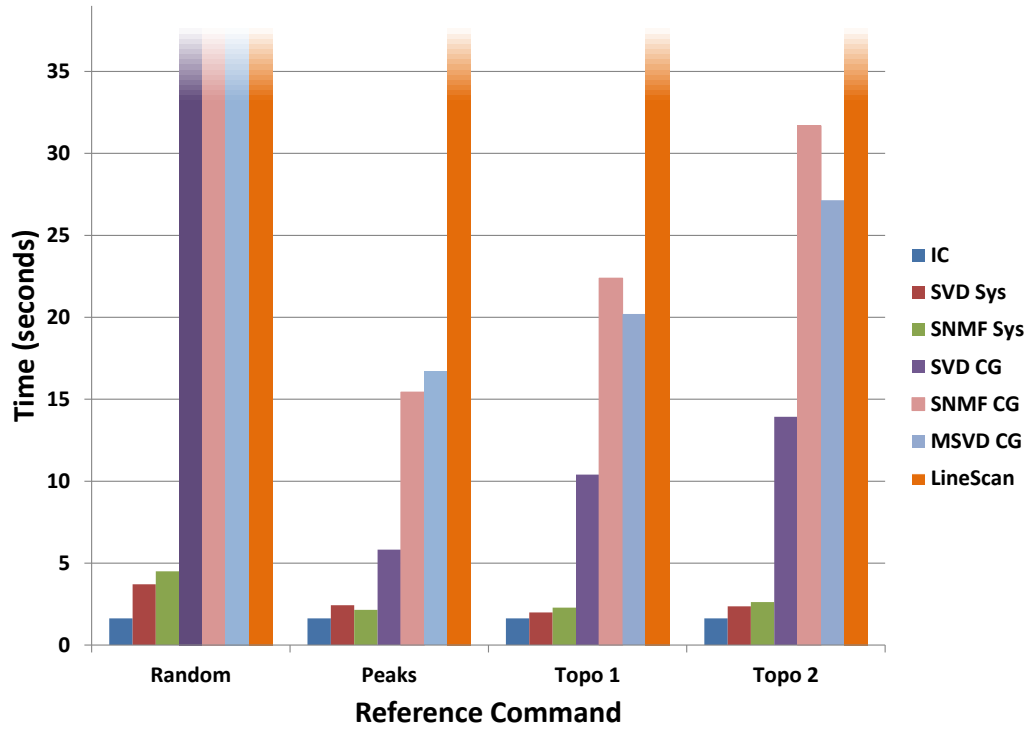
Figure 46 The 2% settling time for different reference commands using the various control techniques to control a 36×36 grid of subsystems given in (201). The bars that fade at the top represent tests that did not converge within 30 seconds.
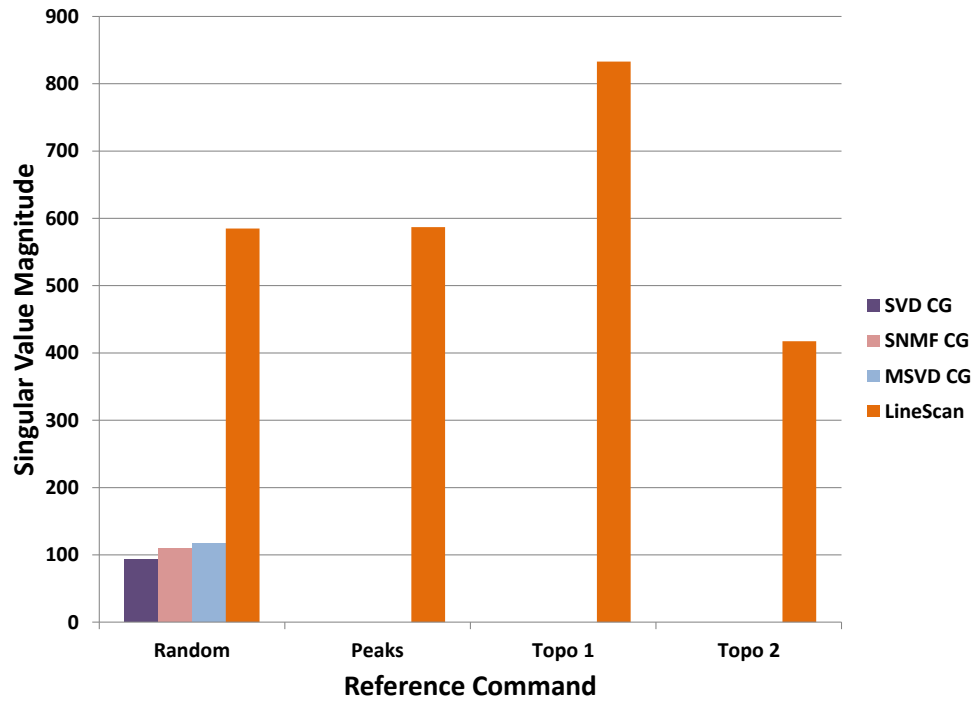

Figure 47 The magnitude of σ_1^e of the error at 30 seconds for the test that did not converge within that time.
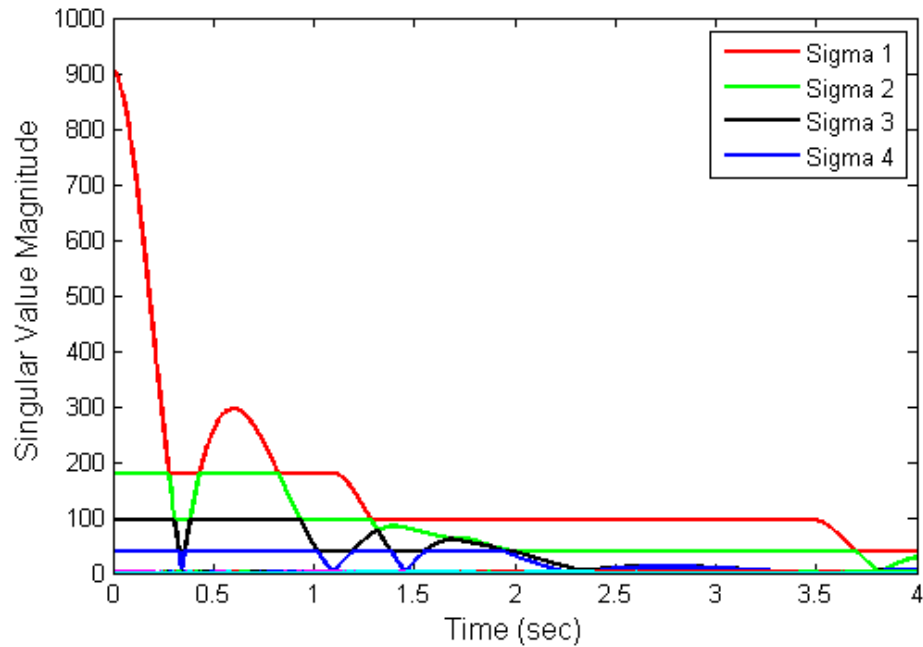
Figure 48 The singular values of the error for the Peaks command using the SVD CG for control.

To summarize, the command generation techniques add significantly more time to convergence than just using the SVD and SNMF Systems, especially for underdamped systems. However, the performance using all of the new techniques was better than using line scanning. When evaluating the dynamics of the system, the designer has additional considerations when choosing whether to switch from one intermediate surface to the next. Considering the evidence, it appears that the SVD and SNMF Systems without command generation would be preferable except in the extreme case where the feedback control techniques by themselves induce undesirable oscillations due to the switching of the command.

110

CHAPTER 8

IMPLEMENTATION

This chapter discusses some issues involved in implementing the SVD and SNMF Systems using hardware. The first section examines the physical implementation of the row-column structure with two- and four-quadrant multiplication for various applications. In the second section, the application of Digital Clay is discussed. Digital Clay provides a vehicle to demonstrate two general concerns for implementation of the SVD and SNMF Systems: dynamics within the physical multiplication of the row and column signals; and power limitations, which prevent every subsystem from simultaneously moving at a maximum rate. As seen using the example of Digital Clay, the SNMF System can effectively control the grid in spite of row and column signals that have significant dynamic characteristics themselves, assuming that the sample rate of the controller is set so that the physical row and column signals can track the desired inputs. Power limitations can have a significant effect on the performance of a controller that simultaneously actuates every subsystem. One of the advantages of the SVD and SNMF Systems is that they allow the system to make use of the available power to control every subsystem, assuming that the power is available to do so. The question then is what advantage, if any, these approaches have when the power is limited so that moving every subsystem simultaneously requires the subsystems to move at a slower rate.

The final section discusses real-time computation of the SVD within a feedback loop. Modifications to well-known computational techniques are proposed to speed up

the SVD computation for the feedback application. An extension of these concepts to computation of the SNMF will be briefly mentioned as well.

Row-Column Multiplication

This section gives examples of how physical row-column multiplication might be realized in multiple domains. In the electrical domain, the easiest solution is to use operational amplifiers. Commercial op-amp packages exist that provide two- or four-quadrant multiplication of two voltage signals, such as the Analog Devices AD633. These offer a convenient solution because the impedance of the op-amps isolates the row and column signals from the dynamics of the subsystems.

A cheaper method to obtain multiplication in the electrical domain is simply to use a variable resistor, such as a digital potentiometer, and to consider resistance as the column input and either voltage or current as the row input. In that case, only two-quadrant multiplication is obtained because resistance cannot be negative. It would also be possible to use two resistors and a switch to obtain four-quadrant multiplication. Using voltage or current and resistance as the row and column signals works so long as the input (either voltage or current) is not a state of the subsystem dynamics. If either the row or column signals are also a state of the subsystems, then the state of one subsystem will have an effect on the input to another subsystem in addition to the coupling effect of the dimension reduction. To avoid the issue of the inputs also being state variables, the potentiometer could be used to control the gain of an op-amp in an inverting or non-inverting amplifier configuration, as shown in Figure 49. For the non-inverting amplifier, the row signal is $r = V_{in}$, the column signal is determined by $R_F = R_1 c - R_1$, where c is the desired column input, and the input to the subsystem is

112

$$V_{out} = V_{in}\left(\frac{R_1 + R_F}{R_1}\right) = r\left(\frac{R_1 + R_1c - R_1}{R_1}\right) = rc. \tag{202}$$
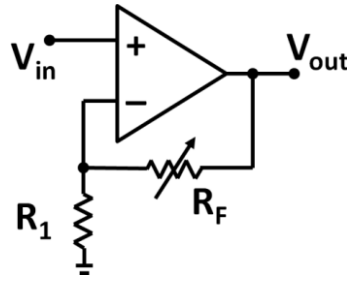

Figure 49 A non-inverting amplifier with a variable resistor.

In the fluid domain, a similar relationship between pressure and flow through a valve can be used as with voltage and current through a resistor. The row input is the supply pressure in the row and the column input is a valve orifice area. This technique will be discussed in more detail in the next section focusing on Digital Clay. Another method to obtain multiplication in the fluid domain would be to have a valve orifice that can be opened in two directions, as described in the diagram in Figure 50. The yellow and blue boxes represent separate orifices. The yellow box can slide up and down as defined by the column input, C, and the blue box can move right and left as defined by the row input, R. The input to the subsystem is then defined by the total orifice area, $A = RC$, shown in green. This could be made into a four-quadrant multiplier by using a multi-port valve as shown in Figure 51. The concept is the same as the valve in Figure 50 except that the low and high pressure ports are organized in such a way that if only R or C is negative, then the valve will open to low pressure, as shown in the current valve position. However, if both R and C are positive or negative, then the valve will be open to high pressure. Therefore, the flow through the valve can be either positive or negative and is based on the relationship

$$Q = c_d A \cdot \text{sgn}(\Delta P) \sqrt{\frac{2}{\rho} \Delta P} = c_d R C \cdot \text{sgn}(\Delta P) \sqrt{\frac{2}{\rho} \Delta P}, \tag{203}$$

where ρ is the fluid density, ΔP is the pressure difference, c_d is the coefficient of

discharge, and $\text{sgn}(\cdot)$ is the signum function.
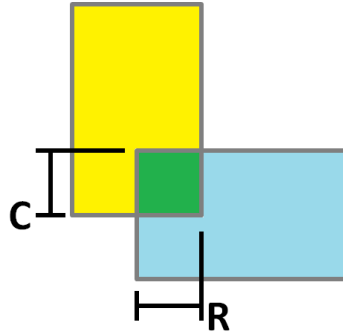

Figure 50 The valve opening can be changed in two directions.
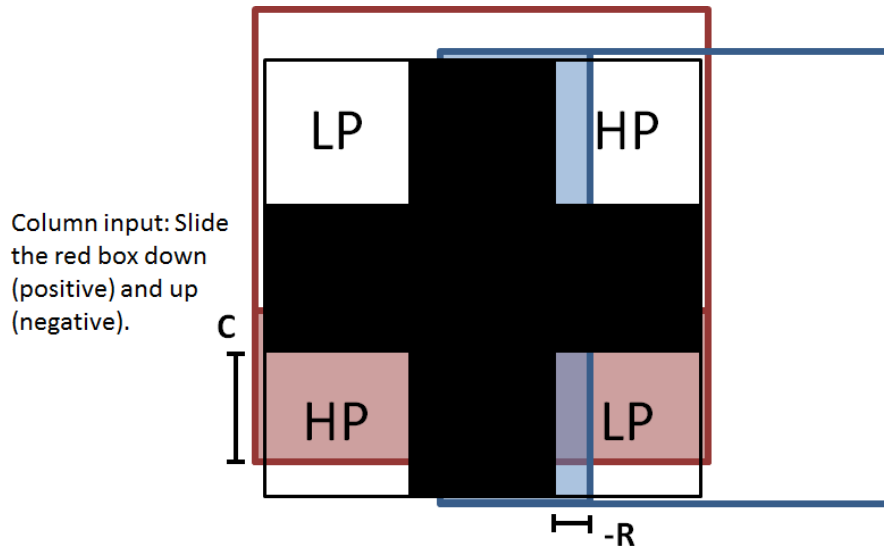

Figure 51 The valve opening in two directions can be to either high (HP) or low (LP) pressure.

Other forms of physical multiplication could be used in various domains. The

ones given here merely provide examples to inspire thought as to how multiplication

might be done in any number of domains. Multiplication is common in many physical

relationships in science and engineering, as well as other fields of study. Some of the key

aspects to keep in mind are the constraints on the row and column signals and whether or

not these signals are also states of the subsystem dynamics. The following section

provides an example of how the SNMF System can be implemented using fluid power

multiplication for Digital Clay. A simulation study confirms the feasibility of that idea. In

Chapter 9, electronic multiplication is done using the AD633 multiplier to control a set of

DC motors and also to control a grid of RC circuits.

Fluid Power Systems: Digital Clay

This section will discuss the implementation of the row-column structure for use

of the SNMF System in fluid power systems. Digital Clay is used as a case study for

these types of systems and a simulation of Digital Clay highlights some of the aspects of

the row-column structure for fluid power and similar applications. These aspects are

system power limitations, dynamics of the row and column signals, saturation of the

subsystem inputs and outputs, and nonnegativity of the column signals.

Digital Clay Model

This section describes the model used to simulate the Digital Clay cylinders and

the row-column structure. The model is based on the current prototype of Digital Clay

described in theses by Zhu and Ngoo [20], [44]. For this simulation study, the row and

column inputs, $r(t)$ and $c(t)$, for Digital Clay are considered to be the pressure in each

row and the resistance in each column. The column resistance can be achieved using the

column adaptor design of the current Digital Clay prototype shown in Figure 3 on page 8,

assuming that the column adaptors can be turned on and off fast enough for pulse width

modulation (PWM). PWM feasibility has been confirmed for the valves controlling the

column adaptors, but not for the column adaptors themselves [44]. The resistance could

115

also be created using a proportional valve with ports for every subsystem in a column. The row pressure control can be achieved by using the current row valves in conjunction with pressure sensor feedback or by using proportional pressure control valves.

The following first-order model represents the cylinder dynamics of Digital Clay:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{c_d g}{A} \sqrt{2/\rho} \end{bmatrix} c(x_v) r(p_s), \text{ where}$$

(204)

$$r(p_s) = sgn(p_s(t) - p_1)\sqrt{|p_s(t) - p_1|} \text{ and } c(x_v) = x_v(t).$$

Here, c_d is the coefficient of discharge for the column adaptor valve, and g is the gradient associated with its opening. The nonlinearity of $r(p_s(t))$ is considered to be well known and therefore can be linearized by defining

$$p_{sdes}(w(t)) = p_1 + w(t)|w(t)|,$$

(205)

where $w(t)$ is one of the SNMF low-rank factors. Therefore, the row input is

$$r(w) = sgn(p_1 + w(t)|w(t)| - p_1)\sqrt{|p_1 + w(t)|w(t)| - p_1|}$$
$$= w(t).$$

(206)

A is the cross-sectional area of the cylinder, ρ is the fluid density, p_s is the supply pressure that is controlled for each row, which creates the row signal, p_1 is the pressure in the cylinder, and x_v is the column valve position or column adaptor duty ratio for each column. This model relies on data presented by Ngoo concerning the current Digital Clay prototype [44]. Table 2 gives the values for the various model parameters.

Table 2 Parameter values for the cylinder model [44].

Parameters	Description	Values
c_d	Discharge coefficient	0.2688
ρ	Fluid density	960 kg/m^3
A_1	Piston Area	8.143 mm
p_1	Cylinder Pressure	86.2 kPa
g	Gradient Coefficient	0.927 mm

The column adaptor dynamics, as well as the row pressure dynamics, are modeled as first-order systems,

$$p_s(s) = \frac{1}{\tau_r s + 1} p_{sdes}(s), \text{ and } x_v(s) = \frac{1}{\tau_c s + 1} x_{vdes}(s), \tag{207}$$

where the time constants, τ_r and τ_c are based on the Lee Co. micro-miniature valves currently used on Digital Clay, which have a time constant of 0.005 s [44]. This value is used as a realistic reference point for the following simulation study. Depending on the specific system and choice of row and column signals this time constant can vary significantly. Therefore, the system response is examined for different values of time constants to examine the effect of the dynamics of the row and column signals on the system response. The Lee Co. valve is used to establish a reasonable range for the time constants used. In addition, saturation limits are placed on $p_s(t)$ and $x_v(t)$,

$$p_s(t) = \begin{cases} 172.4 & p_{sdes}(t) \geq 172.4 \text{ kPa} \\ p_{sdes}(t)e^{(-t/\tau_r)} & 0 < p_{sdes}(t) < 172.4 \text{ kPa, and} \\ 0 & p_{sdes}(t) \leq 0 \text{ kPa} \end{cases} \tag{208}$$

$$x_v(t) = \begin{cases} 4.88 \times 10^{-4} & x_{vdes}(t) \geq 4.88 \times 10^{-4} \text{ m} \\ x_{vdes}(t)e^{(-t/\tau_c)} & 0 < x_{vdes}(t) < 4.88 \times 10^{-4} \text{ m.} \\ 0 & x_{vdes}(t) \leq 0 \text{ m} \end{cases} \tag{209}$$

The saturation limits were determined from the current Digital Clay prototype [44]. The row signal, $r(p_s)$, can be both positive and negative, but the column signal, $c(x_v)$, is constrained to be nonnegative. This necessitates the use of the SNMF System. In addition to the saturation on the row and column signals, the output is limited by the stroke length of the cylinders, which is set at 50 mm, and a quantization nonlinearity is used to represent the A/D conversion for discrete control. A discrete PID controller is used for feedback control with a sample time, T_s. The entire system can be seen in Figure 52, where $Cyl.$ represents the cylinder dynamics in (204), P_d represents the nonlinear

117

pressure equation for the row input in (204), and P_d^{-1} represents the equation in (205). This model was validated by qualitatively comparing closed loop step responses of a single cylinder to those presented in Ngoo's thesis [44].
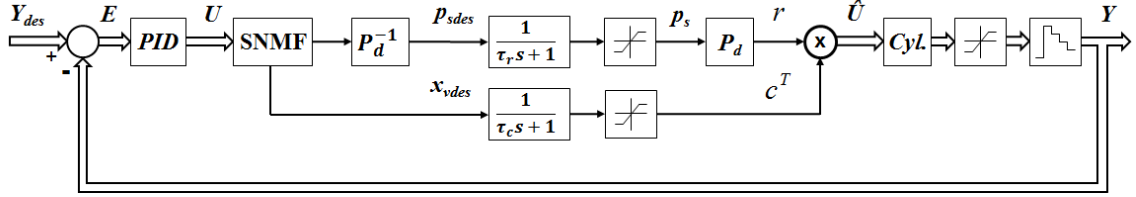

Figure 52 The SNMF System for Digital Clay.

In addition to row and column inputs with dynamics, as described below, simulations studied the effect of system-level power limitations on the performance of the SNMF System. This is done by limiting the total flow to every cylinder in the grid. If Q_i represents the flow to one subsystem, then the total flow can be represented by

$$Q_{tot} = \sum_{i=1}^{mn} Q_i. \tag{210}$$

If the total flow reaches the defined limit, Q_{max}, then the flow for each cylinder is reduced proportionally,

$$Q_{ireduced} = \begin{cases} Q_i, & Q_{tot} \leq Q_{max} \\ \left(Q_i/Q_{tot}\right)Q_{max}, & Q_{tot} > Q_{max} \end{cases}. \tag{211}$$

For the simulations in the following section, the power limit for the entire system is stated in its relation to the flow limit for each cylinder. For example, if the power limit for a 12×12 grid is set to $144Q_{max}$, then the system can supply just enough flow so that all 144 cylinders can receive their maximum flow. If the power limit is set to $12Q_{max}$, then only 12 cylinders can achieve their maximum flow, assuming none of the other cylinders

118

are moving at that time. In that case, if the flow command for the cylinders is greater than what the system can source, the flow to each cylinder is reduced according to (211).

Simulations

Experiments were conducted using the model in Figure 52. The goal was first to show that the SNMF System could be used to control Digital Clay in spite of dynamics within the row and column signals and nonlinearities, such as saturation and quantization. In addition, various degrees of power limitations were imposed to see their effect on the speed of response. The line scanning procedure and IC were used for comparison. For both of those systems, the supply pressure was set to a constant at the pressure limit and the dynamic model of the valve used at each cylinder was considered to be the same as that of the column valve in (207), but the valve was considered to be a two-position valve connecting to either high or low pressure, thus permitting negative flow. Therefore, the dynamic model for the input to the IC and line scanning systems is shared with the column inputs of the SNMF System.

The SNMF System response is demonstrated using a 12×12 grid. Figure 53 shows the response to a step input to a set of random values between 0 and 50 mm. The figure on the left shows the response of the 144 pins, and the figure on the right shows the decaying singular values of the error matrix. For this example, the power limit is $144 Q_{max}$, meaning that every cylinder can move at maximum velocity. The time constants of the row and column dynamics are set to $\tau_r = \tau_c = 0.001$ s, and the sample rate is set to $T_s = 0.005$ s, such that the physical inputs are able to track the desired inputs. In other words, each of these variables is set in such a way as to have a minimal effect on the response.

The response indicates that the SNMF System is able to control Digital Clay in spite of the nonlinear dynamics. Notice that the SNMF is able to drive the system in the direction of the maximum singular value of the error matrix, denoted as σ_1^e, until it decays to the same magnitude as σ_2^e. This makes sense because the desired surface is entirely in the positive direction, and by the Perron-Frobenius Theorem, a real square matrix with entirely positive entries will have first singular vectors with entirely positive entries [45]. Therefore, the minimum solution of the SNMF problem is $\sigma_1 z_1 v_1^T$. Once σ_1^e reaches the same magnitude of the lower singular values, the SNMF input does not match the singular vector directions exactly because they contain both positive and negative values and are thus not valid solutions to the SNMF problem.
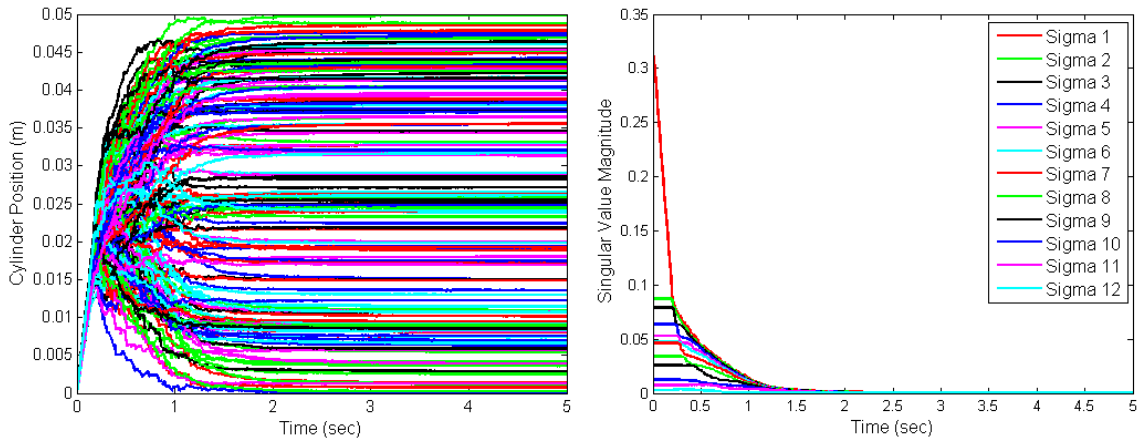

Figure 53 The response of the SNMF System to a step input to random values (*left*) and the singular values of the error matrix (*right*).

Next, the effect of the dynamics of the row and column inputs was examined, particularly in terms of the time constants of the valves and the sample time of the discrete control. If the time constant of either the row or column valve is slower than the sample time for the control, then those valves will not be able to track the command as it switches, and the response-time of the SNMF System is drastically reduced. However, the response is still well-behaved and converges, if more slowly, to the desired value.

Moreover, regardless of how slow the valves are, if the sample time of the control loop is set such that the valves can respond quickly enough to changes in the command, then the system response is the usual response for the SNMF System at that sample rate. This, of course, is assuming that the added time delay of the slower sampling rate does not destabilize the system.

Consider as an example a 3×3 grid with the time constants, $\tau_r = \tau_c = 0.001$ s, and a sample rate of $T_s = 0.01$ s. The singular values of the error converge as shown in the left plot of Figure 54. However, if the time constants are $\tau_r = \tau_c = 0.01$ s and the sample time is $T_s = 0.001$ s, then the convergence is greatly slowed, as shown in the right plot of Figure 54. The reduced convergence is due to the physical row and column signals, pressure and resistance, not being able to track to row and column commands from the discrete controller as seen in Figure 55. Therefore, for the SNMF System with significant dynamics inherent in the row and column signals, which will often be the case if these signals are power variables as with Digital Clay, it is important that the sample rate be set such that the physical row and column signals will be able to track the desired signals.

Another way to deal with row and column signal dynamics would be to apply feedback control to improve the tracking response. For example, a pressure sensor could be used to feedback the row pressure signal, improving the bandwidth of the row signal.
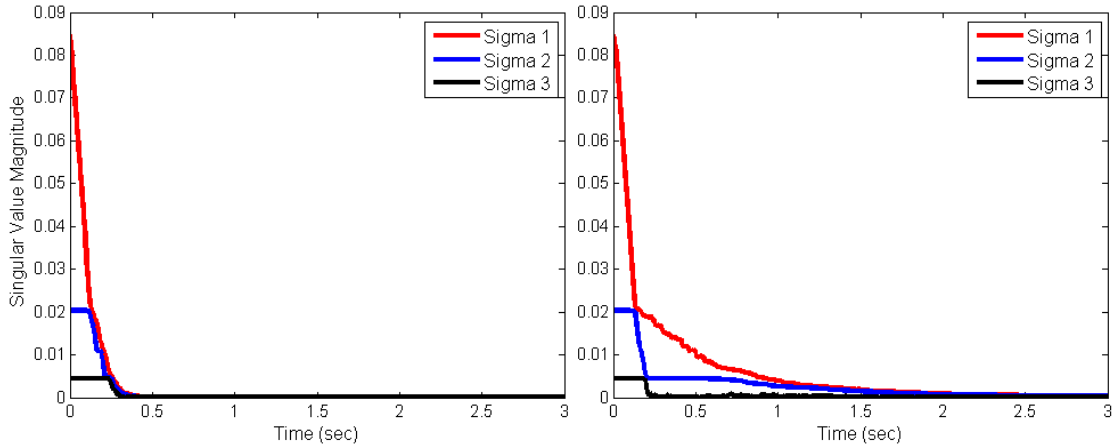
Figure 54 The singular values of the error matrix for the SNMF System with $\tau_r = \tau_c = 0.001$ s and $T_s = 0.01$ s (*left*) and with $\tau_r = \tau_c = 0.01$ s and $T_s = 0.001$ s (*right*).
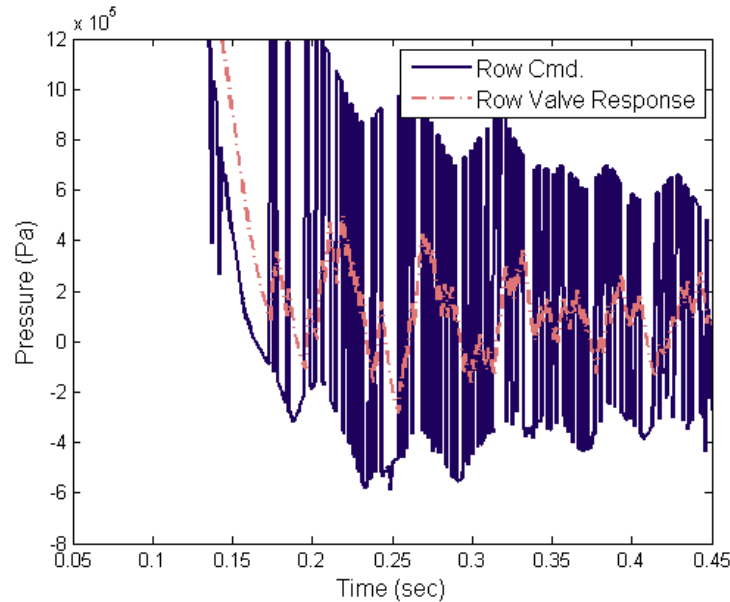

Figure 55 The response of the row signal to the row column command for the SNMF System with $\tau_r = \tau_c = 0.01$ s and $T_s = 0.001$ s.

The next tests compare the convergence of the systems at various levels of power constraints. For these tests, the time constants of the row and column valves are set to $\tau_r = \tau_c = 0.001$ s and the sample rate is set to $T_s = 0.005$ s. The SNMF System performs better than line scanning and worse than IC if the power constraint permits maximum flow of every cylinder, but the performance of the SNMF System and IC diminishes as the available system power is reduced. The performance of line scanning

122

does not diminish until the power limit is reduced below $12Q_{max}$ because only 12 cylinders are permitted to move at any one time using line scanning. If the available flow rate is only enough for 12 cylinders to obtain a maximum flow, then the SNMF System's performance is not significantly better than line scanning. In that situation, the IC system maintains a performance advantage, albeit a diminished one. All of the above statements are assuming that the surface being generated by Digital Clay is a high-rank surface. If it was low-rank and very few of the singular values had both positive and negative values, then the SNMF System would perform as well as IC regardless of the flow capacity. An example is provided below that demonstrates these characteristics using a specific, full-rank surface defined by a set of random values between 0 and 50 mm.

Figure 56 shows σ_1^e of the error matrix for the SNMF System, IC system, and line scanning for a step response to a random set of positions. The SNMF System is shown for a range of power limits from $144Qmax$, which is enough power for the entire grid to achieve maximum flow, to $9Qmax$, which is not enough power for even one line at maximum flow. The IC system is shown for comparison for a subset of these same power limits. Line scanning is shown for only two power limits because the power limit has no effect on the line scan response until it is below $12Qmax$.

For the SNMF System, σ_1^e decays initially at the same rate as it does for the IC system. It begins to decay at a lower rate at the point when σ_1^e first reaches the magnitude of σ_2^e, as seen in Figure 53. Even for this full-rank surface, the SNMF System is faster than line scanning except in the case of a severe power constraint, as shown in Figure 56. As with the other examples in this thesis, if the surface is lower-rank, then the SNMF

System's response more closely matches the response of the IC system, as shown in Figure 57.
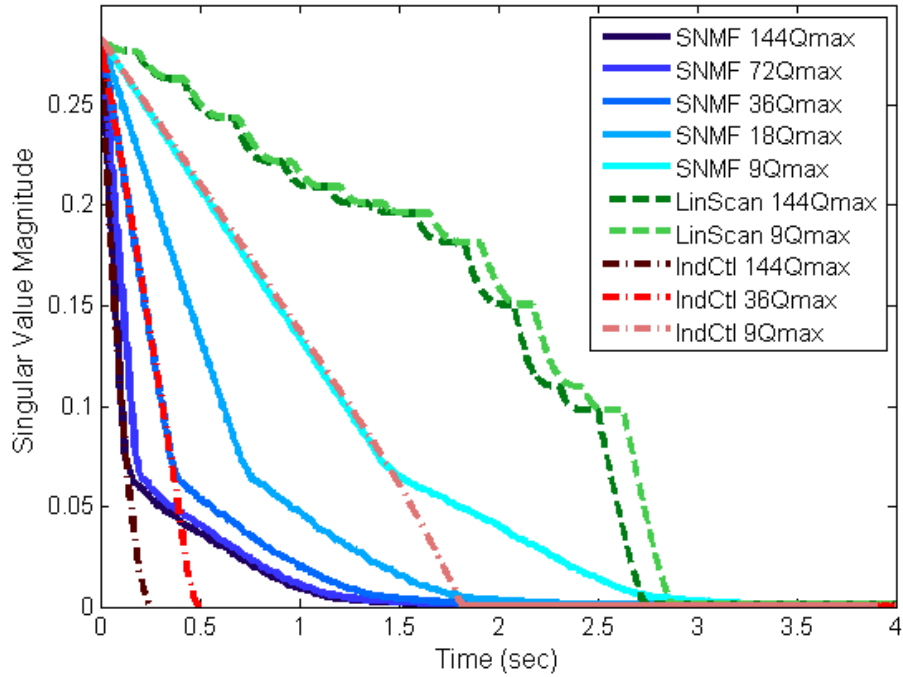

Figure 56 σ_1^e of the error matrix for a step input to a set of random values for the SNMF System, IC, and Line Scanning with varying levels of system power constraints.
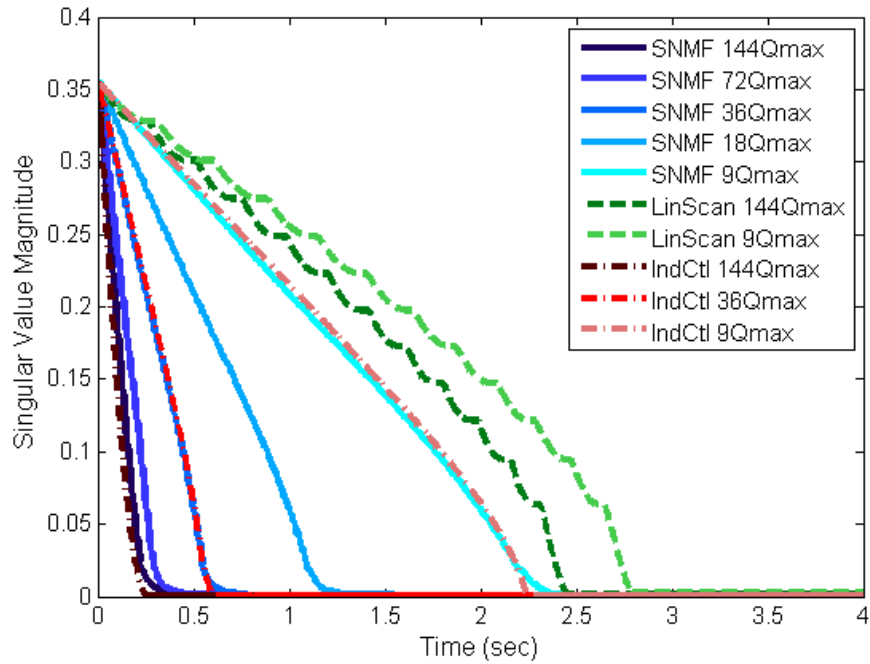

Figure 57 σ_1^e of the error matrix for a step input to a sloped surface for the SNMF System, IC, and Line Scanning with varying levels of system power constraints.

The reduction in relative performance between the SNMF System and line scanning can be explained by the fact that the SNMF permits all of the cylinders to move simultaneously. By capping the total flow available to the grid, the velocity of the cylinders will be constrained throughout the initial reduction in σ_1^e. That is revealed in Figure 56. The convergence of σ_1^e is drastically reduced by the power limitation in the portion of the trajectory where σ_1^e is initially driven towards σ_2^e. After this, the convergence is the same regardless of the power limit because the power limit is not reached. That is also seen in the performance of the SNMF System relative to IC. The SNMF System is about 1.3 seconds slower than IC regardless of the power limit. The tradeoff is that some of the cylinders must move beyond their desired position before converging. This can be seen in Figure 53 where the cylinders converging to small values initially move to about the midpoint of the cylinder stroke. This is in contrast to the IC system, which suffers less from the power constraint because the cylinders converging to smaller values converge quickly, and then require no more resources to allow an increase in the flow to the cylinders converging to larger values. This can be seen in Figure 58. Additionally, because the SNMF System is not using all of the available power resources after the point where σ_1^e initially converges to the value of σ_2^e, the control gains for the SNMF System could be raised, allowing a maximum amount of power to be used throughout the entire trajectory. That would improve the performance of the SNMF System relative to both line scanning and IC.

In summary, the SNMF System provides a practical means to control fluid power systems, such as Digital Clay, assuming that the dynamics of the row and column signals are not prohibitively slow. Additionally, the SNMF System is a significant improvement

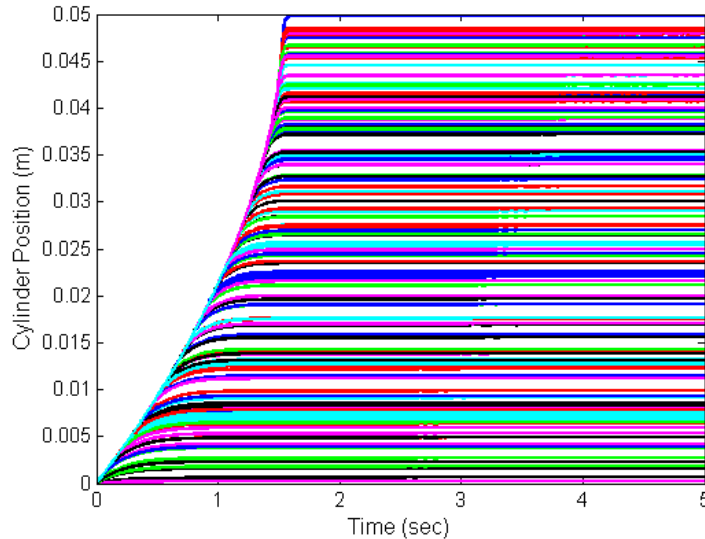on the performance of the line scanning approach except in the case of severe power limitations.

Figure 58 Response to a step to random positions for the IC system with a power constraint of $12Qmax$.

Real-Time SVD Computation

In order to implement the SVD and SNMF Systems, the rank-one approximation must be computed online during each sample time. This real-time computation often occurs at high rates on the order of 100 Hz or faster. Computational methods for both the SVD and SNMF are well known, more so for the SVD. However, there are opportunities for slight improvements on the use of these algorithms for their specific application in a feedback loop. This section examines modifications of the Power and Jacobi Methods to speed up the calculation of the SVD rank-one approximation of the control input. An extension of those concepts to the SNMF will also be mentioned.

The computation of the rank-one approximation in this context differs from its computation for an arbitrary matrix. In this application, other than during the initial sample time, information from the previous sample time can be used in the computation of the SVD during the current sample time. Thus, the SVD rank-one approximation only

126

needs to be computed from scratch one time. This section is divided into two subsections, one on the Power Method and another on the Jacobi Method. Each subsection will discuss the use of those methods for the feedback application and propose modifications to speed up their computation. The pros and cons of the modifications will also be discussed. Before describing how this can be done, there is a brief explanation of current methods used to calculate the SVD and some existing modifications for improving the computation time that are relevant here.

Background

Various algorithms have been developed to calculate the SVD of an arbitrary matrix quickly, accurately, and with numerical stability. Watkins provides a good introduction to most of these techniques [23]. The most popular algorithm is the Golub-Reinsch algorithm [46]. The algorithm is divided into two main steps. First, a series of Householder transformations reduces the matrix to a bidiagonal form. Second, the bidiagonal matrix is transformed into a purely diagonal one through an iterative procedure based on the QR algorithm with origin shifts. The Golub-Reinsch algorithm is often the most efficient and numerically stable method for finding the SVD of a generic matrix. However, two challenges arise when applying the technique to the feedback loop in Figure 8 on page 27. First, the Golub-Reinsch algorithm necessarily calculates the full SVD, where in this application only the first singular value and its corresponding singular vectors are needed. Second, the algorithm's initial step, the Householder transformations, cannot take advantage of the knowledge of the control input acquired from previous sample periods. Therefore, the Golub-Reinsch algorithm is not examined for its use in the SVD System, but it is used for comparison as it is currently the most widely used

algorithm. Instead of the Golub-Reinsch algorithm, two algorithms are examined in the following subsections for their application to the SVD System.

The first of these algorithms is commonly referred to as the Power Method [49]. It is particularly suited to this application because it can be used to calculate only a subset of the largest or smallest singular values of a matrix [23]. It is an iterative method that uses successively larger powers of a matrix to find the matrix's singular values and vectors. The Power Method is examined below to highlight the feedback problem and several modifications to the algorithm are proposed that can improve the speed and accuracy of the technique for this application.

The second algorithm that is examined is called the Jacobi Method [23]. It is based on iteratively applying Givens rotations to transform a matrix to diagonal form. A one-sided Jacobi method for computing the SVD was demonstrated by Nash [47]. It permits calculation of only the first r singular values and vectors of a matrix, where r is the rank of the matrix. This section proposes a slight modification to the algorithm that reduces the amount of computation by accurately computing only the first p singular values and vectors, where p is specified by the user. That change is most useful for larger matrices with sizes $> 100 \times 100$.

Most of the interest in the Jacobi Method is due to the ease with which the algorithm can take advantage of parallel processing [48]. While using parallel computing would speed up the calculation of the SVD, it has been the subject of much previous work and will not be considered here. However, one particular advantage of the Jacobi method for the application at hand is that it can make use of knowledge from previous iterations of the feedback loop. This idea was originally shown by Maciejewski for

finding the SVD of the Jacobian of a robotic manipulator in real-time [48]. He derived

perturbation bounds on the singular values and vectors to guarantee that they will change

in a well-defined manner. The use of this technique will be examined in this paper as it

applies to finding the rank-one approximation for the feedback loop in Figure 8.

The Power Method

The basis of the Power Method comes from Nash [49]. Defining the SVD of a

matrix as $M = U\Sigma V^T$, the algorithm, starting with an initial guess, v^0, proceeds

iteratively as follows:

$$u^{\ell+1} = \frac{Mv^\ell}{\|Mv^\ell\|} \tag{212}$$

$$v^{\ell+1} = \frac{M^T u^\ell}{\|M^T u^\ell\|} \tag{213}$$

$$\sigma^{\ell+1} = \|M^T u^\ell\|. \tag{214}$$

These iterations are continued until the convergence criteria,

$$\|u^{\ell+1} - u^\ell\| < \tau, \tag{215}$$

is met, where τ is a predefined tolerance. The approximate singular value and vectors,

$\sigma_1^{\ell+1}$, $u_1^{\ell+1}$, and $v_1^{\ell+1}$, will converge to the true largest singular value, σ_1, and its

corresponding singular vectors, u_1 and v_1, assuming that $v_1^T v^0 \neq 0$. From there, the

second singular value and vectors can be obtained by deflating the matrix,

$$M' = M - u_1^{\ell+1}\sigma_1^{\ell+1}v_1^{\ell+1}, \tag{216}$$

and iterating through (212)-(215) using M' in place of M. The solution will converge to

the second singular value and vectors. This process can be extended to compute the full

SVD. Nash provided a useful tip to accelerate the calculation of the smaller singular

values by using the last two approximations of v_1 to create the initial guess for the iterations to find v_2 [49]:

$$v_2^0 = v_1^{\ell+1} - v_1^{\ell}. \tag{217}$$

This technique speeds up the convergence to σ_i, v_i, and u_i, except when $i = 1$, which is of primary interest in this thesis. However, it will play an important role in speeding up the computation of σ_1, v_1, and u_1 in a feedback loop. An advantageous property of the Power Method is that it enables the computation of the largest singular value(s) and corresponding vectors without also requiring the computation of the rest of the singular values and vectors. However, as will be shown, gaining information of the smaller singular values and vectors can potentially aid in the computation of the largest singular value and its singular vectors within the context of the feedback loop in Figure 8.

The convergence rate of the Power Method is dependent on the relative magnitudes of adjacent singular values [23]. If adjacent singular values are of approximately the same magnitude, the convergence to the larger of the two singular values will be very slow. This presents a challenge for applying the Power Method, as described above, to the feedback control problem. Consider the example of a set of nine masses that are brought to rest from nonzero initial conditions, as described in Chapter 4. Figure 15 on page 38 shows the change of the singular values of the control input matrix $U(k)$. Notice that, initially, only the largest singular value decays as the rank-one input constraint dictates that all of the control effort be used in the direction of the maximum singular value. However, at $t \approx 0.5$ seconds, the largest singular value has decreased to the same magnitude as the second singular value. At this point, the singular values change order, and the perturbation bound on the singular vectors described by

Maciejewski becomes ill-conditioned [48]. Then, at $t \approx 2.8$ seconds, all three singular values have nearly the same magnitude. At these instances, if the Power Method is used to find the rank-one approximation, the number of iterations required for convergence in (215) greatly increases. For instance, in the example in Figure 15, the number of iterations before $t \approx 0.45$ seconds is on the order of 10^1, but as $t \rightarrow 0.5$ seconds, the number of iterations increases to on the order of 10^3. Therefore, the Power Method as presented above does not represent a viable technique for finding the rank-one approximation for the feedback system. However, as explained below, a very simple modification can make the technique quite useful in this context.

Modification 1: Capping the iterations

Instead of stopping the Power Method iterations using the convergence criterion in (215) to guarantee a level of accuracy for the largest singular value and its vectors, one modification is to stop after a certain number of iterations. This can be beneficial for real-time implementations because the number of flops for every iteration is the same, $O(mn)$, making the cost of j iterations always the same, $O(jmn)$. Therefore, by capping the number of iterations, the computation time becomes more repeatable, which allows a maximum number of iterations possible during one sample period. Another possibility is to cap the computation time itself and stop the power iterations at the end of each sampling period. The obvious effect of this modification is that the exact largest singular value and its singular vectors will not always be obtained, particularly when the largest singular value is close in magnitude to the second singular value. However, the proposed modification is still useful for the feedback control problem.

As an example, consider Figure 15 at $t \approx 0.5\ s$. Although the Power Method would require many iterations to find the exact largest singular value and its singular vectors, in very few iterations it is able to make v^ℓ and u^ℓ orthogonal to the direction of v_3 and u_3 and singular vectors corresponding to smaller singular values for larger grids, respectively. Thus,

$$u^\ell \cong \beta_1 u_1 + \beta_1 u_2,$$
$$v^\ell \cong \alpha_1 v_1 + \alpha_2 v_2,\text{ and} \tag{218}$$
$$\sigma_1 \geq \sigma^\ell \geq \sigma_2,$$

where α_1, α_2, β_1, and β_2 are constants.

Since σ_1 and σ_2 are nearly identical, it is less important which direction the control input takes between the two. Therefore, the stopping criteria can be set to $\ell = iter_{max}$ instead of (215), guaranteeing real-time implementation without significantly sacrificing performance. Returning to the example of slowing down nine masses, power iterations are used to compute the rank-one approximation, stopping after 10 iterations. The resulting approximate maximum singular value, σ^ℓ, is shown in Figure 59 to remain within the bounds given in (218). Also, up until $t \to 2.5$ seconds, v^ℓ and u^ℓ are confirmed to be orthogonal to v_3 and u_3, respectively, to less than 10^{-3}. The bounds given in (218) can also be understood in light of the stability criteria developed in Chapter 5. These stability conditions are largely dependent on the relationship

$$\left(1/n\right)\text{trace}(X^T X) \leq \sigma_1^2(X) \leq \text{trace}(X^T X). \tag{219}$$

Therefore, unless the controller is chosen such that the system is "close" to instability, the error induced by capping the iterations of the Power Method will not destabilize the system.
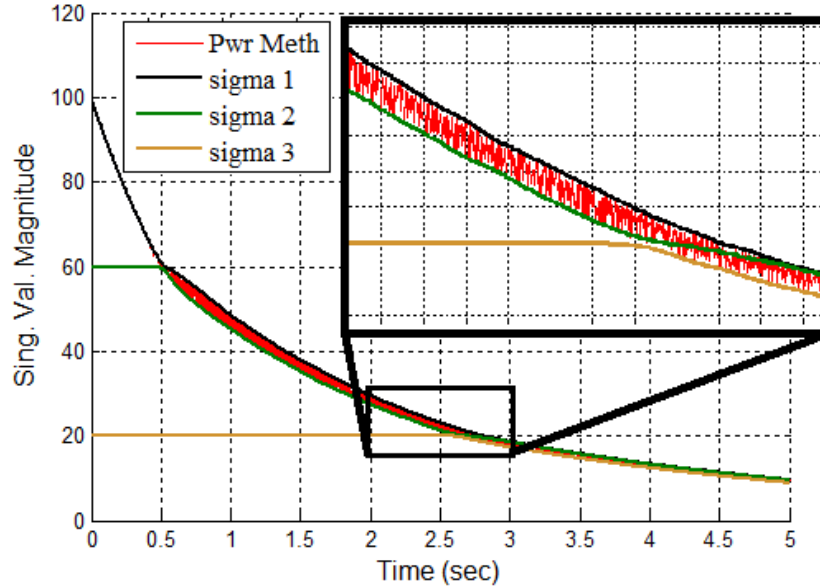
Figure 59 σ^ℓ from the Power Method and σ_1, σ_2, and σ_3 of $u(k)$ during a system repsonse to initial conditions.

Modification 2: Using previous information

Through a second modification, the Power Method can be accelerated without loss of accuracy. This is accomplished by defining $v^0(k)$ using the approximation from the previous sample period, $v^{\ell+1}(k-1)$. Only during the first sample period will the SVD approximation need to be computed from scratch. After that, the control input matrix can be represented as a perturbation of the control input from the previous sample, $U(k) = U(k-1) + \Delta U$. So in computing the approximation for $U(k)$, the initial guess, $v^0(k)$, is defined as the right singular vector of $U(k-1)$. One of the more convenient properties is that this perturbation is directly related to the sample time. The faster the sample time, the faster the rank-one approximation needs to be computed, but the smaller the perturbation, ΔU. In fact, for many linear subsystems in the feedback configuration in Figure 8, ΔU will have no effect on the singular vectors, assuming no disturbance inputs. In the example in Figure 15, the singular vectors, v_1, v_2, and v_3, remain nearly constant throughout the response. The only change is their order, as the relative magnitudes of the

133

singular values change. Therefore the exact rank-one approximation could be obtained in one power iteration because it is being initialized by the true singular vector, $v^0 = v_1$.

While this seems like a definitive solution, a serious problem emerges when the largest singular value of $U(k)$ converges to the same magnitude as the next largest singular value and then becomes smaller. When that occurs, as at $t \approx 0.5$ seconds in the example above, if $v^0(k)$ is taken as $v_1^{\ell+1}(k-1)$, then $v^0(k) = v_2(k)$, and it is orthogonal to the $v_1(k)$. Therefore, the Power Method will not converge. Even if $v^0(k)$ is close to $v_2(k)$, the solution will converge, but do so slowly. One solution to this problem involves keeping track of all of the singular vectors and not just v_1.

By keeping track of all of the right singular vectors, it is possible to detect when the singular values corresponding to these vectors change order in their relative magnitudes. This is accomplished by initializing the Power Method with each right singular vector from the previous sample period and comparing the magnitudes of the estimated singular values that result from one iteration. In that situation, the singular vector, $v_i^{\ell+1}(k-1)$, from the previous sample time that is closest to the $v_1(k)$ will result in the largest singular value after one power iteration. This approach has been applied successfully to the above example and to a separate simulation where the singular vectors gradually change over time. It has yielded exact solutions using only one or two iterations in addition to the initializing iterations. In general, it guarantees a highly accurate solution, but it requires at least $2n$ iterations of the Power Method, assuming $n \leq m$, to determine which $v_i^{\ell+1}(k-1)$ to use as $v^0(k)$. However, the process can be done partially in parallel to reduce the cost to one iteration of the Power Method using n processors.

There is another way to use previous iterations that will converge and only require about 2 iterations of the Power Method in the initialization step. This modification makes use of (217) to obtain an estimate of $v_2(k-1)$. The initialization step proceeds as follows: set $v_1^0(k) = v_1^{\ell+1}(k-1)$ and compute one iteration of the Power Method; next, set

$$v_2^0(k) = \frac{v_2^0(k-1)}{v_2^0(k-1)^T v_2^0(k-1)} \tag{220}$$

where $v_2^0(k-1)$ is given as in (217); and lastly compute one iteration of the Power Method. Compare the resulting singular value approximations and choose $v^0(k)$ as the singular vector corresponding to the larger of the two singular values. This will work only when $v^0 \neq v_1$ because, if that occurs, then $v_1^{\ell+1} - v_1^\ell = 0$ and (220) becomes ill-conditioned in the next sampling period. However, that is unlikely to occur in practice due in part to noise and unmodeled dynamics. To ensure that it never occurs, all that needs to be done is to add an infinitesimal perturbation to v^ℓ so that the Power Method must compute at least one iteration in order to satisfy (215). Using this technique for the example in Figure 15, the error between σ^ℓ and σ_1 and the required number of iterations are significantly reduced relative to the capping technique with a random v^0, as shown in Figure 59. In that case, the orthogonality between $v^{\ell+1}$ and v_3 and between $u^{\ell+1}$ and u_3 is maintained.

In summary, using (217) to estimate $v_2^0(k-1)$ and capping the number of iterations to ensure real-time operation offers a useful way determine an accurate approximation of the SVD rank-one matrix for high sample rates and large matrices. If parallel computation is possible, then tracking all of the singular vectors from one

135

iteration to the next may be possible and could provide potentially more accurate results. However, as demonstrated, computing the exact SVD by the Power Method requires many iterations, even when using data from previous sample periods. The Jacobi Method, using the modification proposed by Maciejewski, offers another way to keep track of all the right singular vectors and produce the exact SVD rank-one matrix, $\hat{U}(k)$, in a few iterations.

Jacobi Method

The Jacobi Method, as applied to the computation of the SVD, is a series of Givens rotations, called sweeps, that iteratively orthogonalize the columns of a matrix. Given a matrix, A, the goal is to find an orthogonal matrix, V, such that

$$MV = B, \tag{221}$$

where B also has orthogonal columns. Then, the SVD of M can be written as

$$M = U\Sigma V^T, \tag{222}$$

where U is obtained directly by normalizing the columns of B, and σ_i is the magnitude of the ith column of B. The matrix V is the product of a series of Givens rotation matrices that operate on the ith and j^{th} columns of A, as follows,

$$m'_i = m_i \cos(\theta) + m_j \sin(\theta), \text{ and}$$
$$m'_j = -m_i \sin(\theta) + m_j \cos(\theta). \tag{223}$$

The angle, θ, is calculated so that the columns of M' will be more orthogonal than the columns of M [49]. Convergence of this method has been proven [47]. Like the Power Method, the Jacobi Method can be initialized using V calculated from the previous sample time. This was implemented in real-time for computing the SVD of the Jacobian of a matrix by Maciejewski, who also showed perturbation bounds on both singular

values and singular vectors [48]. Applying this algorithm to the example system response, the solution for the maximum singular value is exact to within 10^{-3}. However, for large matrices of size $> 100 \times 100$, this algorithm is too slow for computation in real-time except when it is applied using a parallel architecture. To speed it up, again a modification can be made to the stopping criteria.

The stopping criteria for the algorithms used by Nash and Maciejewski is [47], [48]

$$\frac{\left(m_i^T m_j\right)^2}{\left(m_i^T m_i\right)\left(m_j^T m_j\right)} < \tau. \tag{224}$$

When this condition is met, then the particular rotation of the ith and j^{th} columns is no longer performed, thus convergence of the entire algorithm occurs when the condition is met for all i and j combinations. Since only a high level of accuracy of the first singular value and its singular vectors is needed, the iterations can be stopped only if the above condition is met where $i = 1$. Using this technique in the example in this thesis did not result in a significant increase in the error of the maximum singular value and was able to cut the number of iterations in half for most sample periods.

Discussion

No single algorithm discussed above presents a solution for every system. The closest proposed method for use in the SVD System is sampling using the Power Method with a cap on the number of iterations or computation time and using the estimate for v_2 given in (217). Although that will not yield exact solutions, it can produce useful control inputs while guaranteeing fast computation times even for matrices over $1,000 \times 1,000$, or $1,000,000$ subsystems. For example, for a $1,000 \times 1,000$ matrix, 20 power iterations

starting with a random v^0 took 0.08 seconds to compute. In contrast, the Golub-Riensch algorithm required 16.17 seconds to compute. In addition, using the approximation of the second singular vector in (217) with a cap on iterations improved the accuracy without significantly increasing computation time, and reduced it in some circumstances.

If greater accuracy is required and parallel computation is an option, the Jacobi Method can be used. However, if not computed in parallel, the Jacobi Method is too slow for large matrices, and, for small matrices, the unmodified Golub-Riensch algorithm is generally fast enough, on the order of $10^{-2} - 10^{-3}$ seconds. Note, all of the simulations were performed in MATLAB 2009b on a Dell Vostro computer running Windows 7 with an Intel Core i3 processor.

In sum, this section has presented a number of techniques that enable real-time computation down to 10^{-2} seconds, even for systems involving $1,000^2$ subsystems. Modifications to the Power and Jacobi Methods were examined to make use of information from previous sample periods and the fact that only σ_1, u_1, and v_1 need to be found.

Extension to the SNMF

Noting the similarity between (45) on page 31 and (212)-(214), the modifications made to the Power Method in the previous section can be extended to the SNMF algorithm without much difficulty. For example, the number of iterations of the SNMF algorithm can be capped to achieve similar performance as in the modified Power Method. In addition, just as v_1 from the previous sample time is used in (217) to compute an estimate of v_2 at the current sample time, which is then used to determine the initialization for the Power Method, for the SNMF algorithm, w_1 from the previous

sample of the control loop can be used to help determine the initial value for the SNMF

iterations using a relationship like

$$w_2^{(0)} = w_1^{(\ell+1)} - w_1^{(\ell)}.$$

(225)

Further study of these concepts has not been carried out and instead is left for future

work.

CHAPTER 9

PHYSICAL DEMONSTRATIONS

This chapter examines real-time implementations of the SVD and SNMF Systems using two hardware test beds. The first is a 3×2 grid of DC motors with subsystems that have a free integrator and significantly nonlinear dynamics that vary between the subsystems. The second test bed is a 4×4 grid of RC circuits with first-order linear dynamics and little variation between the subsystems. These systems do not have a free integrator, and challenges relating to these types of systems will be discussed. The main goal of this chapter is to demonstrate the feasibility and utility of the control concepts presented in this thesis.

DC Motor Control

This section provides a physical demonstration of the SVD and SNMF Systems. The goal is to present confirmation of the techniques in practice and explore some of the challenges of implementation. The techniques being explored are the SVD and SNMF Systems carried out in real-time, the modified Power Method for reducing the SVD computation time, and the use of op amp-based four-quadrant multiplication. The physical subsystems are brushless DC motors that have higher-order dynamics and vary significantly in their dynamical behavior. The SVD and SNMF Systems are compared with IC and line scanning. Before discussing the results, a brief description will be given of the system hardware.

System Description

Electro-Craft 3622-4B-N brushless DC motors were borrowed from the undergraduate labs at Georgia Tech. The motors, shown in Figure 60, have steel flywheels attached to the motor shafts to provide rotational inertia and were powered using Advanced Motion Control Brushless PWM Amplifiers. HP HEDS-5640 AO6 quadrature encoders with 500 steps per revolution were used. To create the row-column structure, both the row and the column signals were analog voltage signals. To perform the necessary multiplication, AD633 four-quadrant analog multipliers were used. The voltage output of the multipliers was obtained by the equation

$$V_{out} = \frac{V_1 V_2}{10} + Z,\tag{226}$$

where Z is an offset voltage. The offset voltage was set to 0 V. The output voltage was sent to the amplifiers to drive the motors. Unfortunately, only six motors were available, so a 3×2 grid was created. Therefore, five analog signals from the computer were used to control the six motors. The motors are identified in accordance with their placement in the grid. For example, Motor 21 is the motor in the second row and first column. MATLAB XPC Target was used for real-time control with a Quanser Q8 I/O board to perform the D/A and read the encoder signals. The sample rate was 1 kHz for all of the tests, unless otherwise noted.
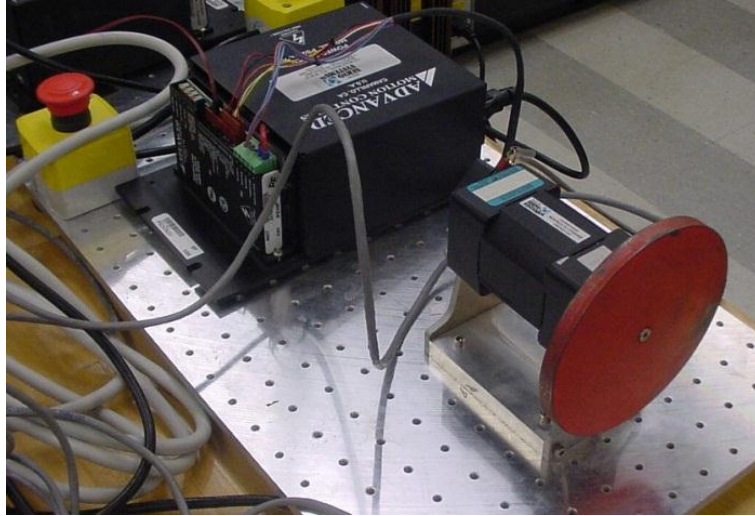
Figure 60 Brushless DC motor with flywheel (*red*) attached.
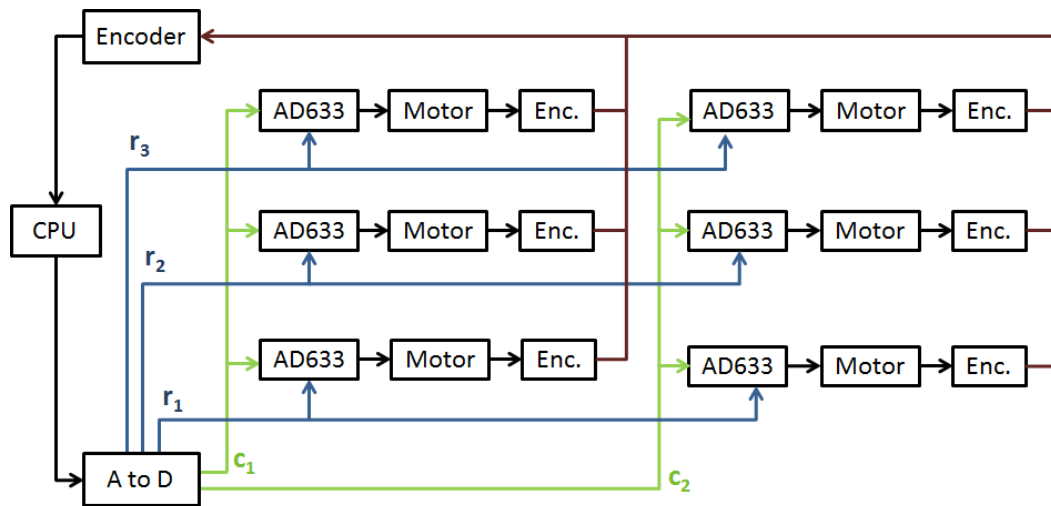
Figure 61 DC motor row-column structure with 3 row signals (*blue*), 2 column signals (*green*), and six feedback signals (*red*).

The motors have a significant nonlinear friction component that creates a deadband of $0.6 - 0.85$ V depending on the motor. This can be seen through open loop ramp responses for each motor, shown in the plot of velocity vs. input voltage in Figure 62 below. The velocity shown in Figure 62 has been filtered by an averaging filter. The variation in the motors is also clearly observed in the responses. The two most strikingly different motors are Motors 21 and 31. Motor 21 is the only motor with a primarily linear response, while Motor 31 has significant oscillations in its velocity. The oscillations in

142

Motor 31 are due to a faulty encoder, which skipped counts, causing the measured

position, and subsequently computed velocity, to drop off in magnitude at higher voltage

inputs. This fault did not have a serious negative effect on the responses shown later,

except that the actual position of the flywheel did not correspond to the measured

position.

An open loop step response also reveals the differences in the speed of response

of the motors. Figure 63 shows the unit step response for Motors 21 and 12, which were,

respectively, the fastest and slowest motors. Fitting a first-order model to approximate the

systems' responses, the time constants of the motors range from 0.83-1.905, and the open

loop gain varies from 2.96-7.93. Although the first-order models fit the data well for

those motors, it is not the case for all the motors. Furthermore, these first-order models do

not match the step responses to other magnitudes, indicating the nonlinearity of the

dynamics.
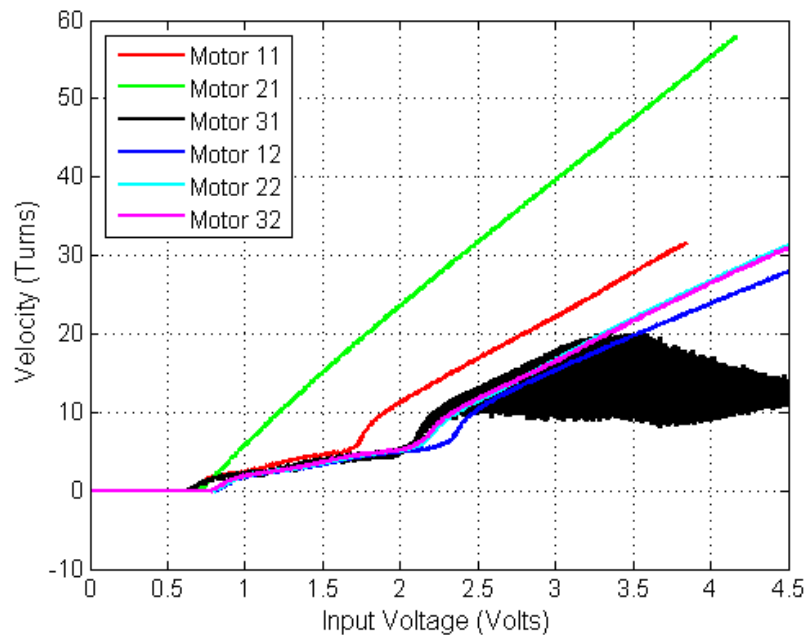
Figure 62 Open loop ramp response of each individual motor with input slope of 0.1.
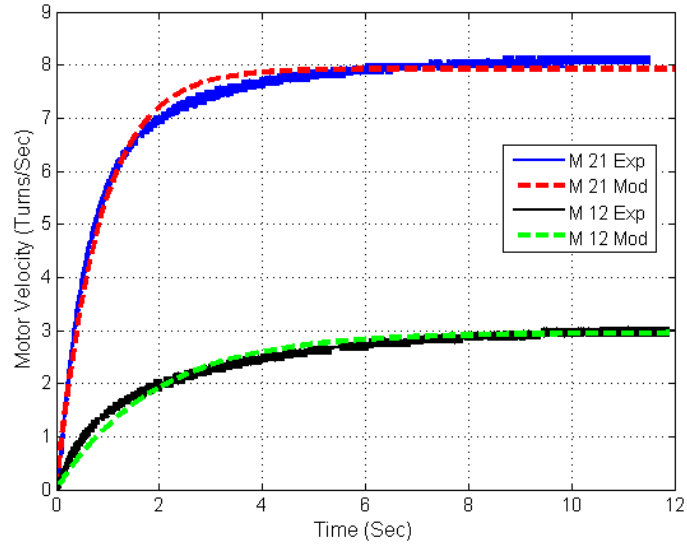
Figure 63 Motor 21 and Motor 12 open loop unit step responses.

For the following experiments, two types of controllers were used. The first is a

linear PID controller. Using that controller, Figure 64 shows an example step response

for Motor 11 to -9.4 turns, or rotations, with $k_p = 45, k_i = 15, k_d = 11$. In addition, a

nonlinear controller was used to linearize the motor deadband, as shown in Figure 65.

The nonlinear offset was equal to the deadband of the motor in the grid with the greatest

deadband, 0.85. Both types of controllers were used for all the tests with different control

gains. Only a subset of the tests will be discussed below.
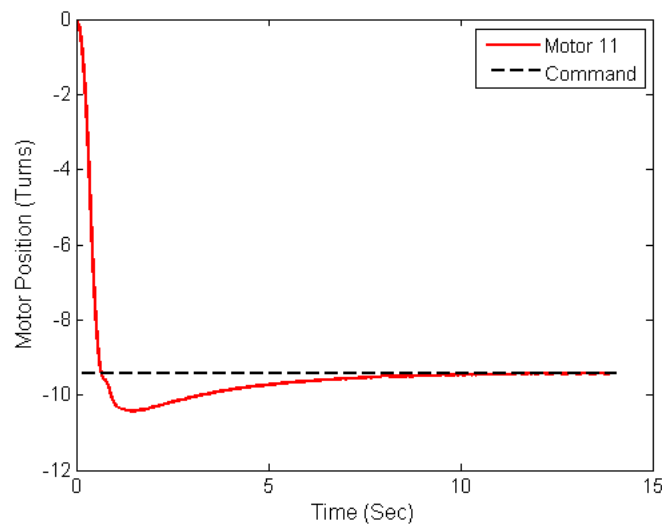
Figure 64 Step response of Motor 11 using a PID controller with $k_p = 45, k_i = 15, k_d = 11$.

144
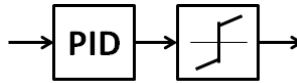
Figure 65 Nonlinear control law.

The SVD System

This section presents the SVD System, as shown in Figure 8 on page 27, for the DC motor control. The SVD System was able to run without difficulty at the 1 kHz sample rate using the Golub-Reinsch algorithm. The four-quadrant multipliers worked accurately to provide the row-column structure necessary. The entire system worked as expected in spite of challenges such as the faulty encoder and significant subsystem variability.

The stability of the SVD System and IC are compared when using identical proportional controllers for each subsystem. The overall system performance of the SVD System is examined and compared to the simulation results presented in the earlier section. This is done by examining the system step response to various reference commands, disturbance rejection, and trajectory tracking. In addition, the time required to achieve a desired position is compared with line scanning and IC, assuming the same controller for each technique.

It was shown that the stability of the SVD System using the same proportional controller for each subsystem depends on the stability of the subsystems with IC. This was done by increasing the control gain until sustained oscillations were reached such that the input oscillated between its saturation limits. For both the SVD System and the independently controlled Motor 11, this occurred at a control gain of $k_p = 40$. At that gain, Motor 11 for the SVD System and IC exhibited sustained oscillations as previously described. For lower values of k_p both systems responded without exhibiting instability.

145

To study the response of the SVD System, consider the response to a step input to the following positions:

$$Y_{des} = \begin{bmatrix} -9.4 & -0.2 \\ 8.6 & 1.6 \\ 4.6 & -5.3 \end{bmatrix} \text{ turns.} \tag{227}$$

The controller used is a PID controller with $k_p = 45, k_i = 15, k_d = 11$. The response to this command is shown in Figure 66. The system converges with varying levels of overshoot. To understand the response, the singular values of the error and the control input are examined in Figure 67 and Figure 68. Notice that σ_1^e and σ_2^e of the error matrix only change directions once at $t \approx 0.8$ sec, whereas σ_1^{ci} and σ_2^{ci} of the control input matrix change directions a number of times. The change of directions of σ_1^{ci} can be clearly seen in the change in the first singular vectors. Figure 69 shows the change in v_1^{ci} of the control input matrix.

This example also provides an opportunity to compare the performance with IC and line scanning. The 2% settling time of the SVD System shown in Figure 66 is

$$\begin{bmatrix} 6.4 & 9.3 \\ 7.9 & 10.1 \\ 7.2 & 11 \end{bmatrix} \text{ seconds.} \tag{228}$$

An interesting result is that the settling time for each motor is not dependent on the final value of the step or on the speed of response of the motor. This is partly due to settling time being a percent of the final value, but it is also due to the SVD reducing the combined error in every motor. To compare this to IC, note that the step response of the largest step, shown in Figure 64, had a settling time of 6.4 seconds. For line scanning, the largest step in each column was tested, resulting in a settling time of 12.9 seconds. Admittedly, the choice of these particular motors may not have been the best due to the varying dynamics among the motors. If the full grid could be tested using IC, the result

146

would likely be no faster than the motor selected, but it might be slower. Even still, the

SVD System is faster than line scanning, even though there are only two columns to scan

through and the reference command is full rank with a significant second singular value.

The SVD System overall is slower than the IC of Motor 11, but Motor 11 for the SVD

System actually converges at the same rate as the independently controlled motor.
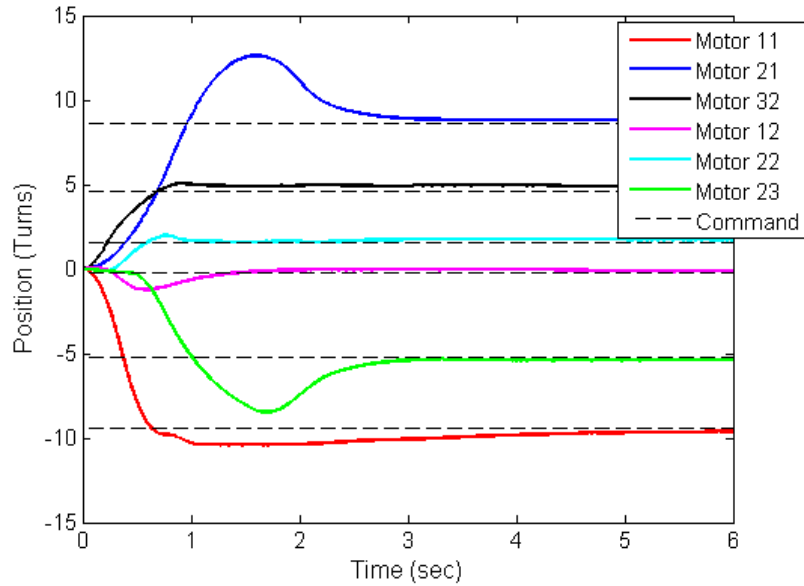
Figure 66 Response of the SVD System to a step command to (227).
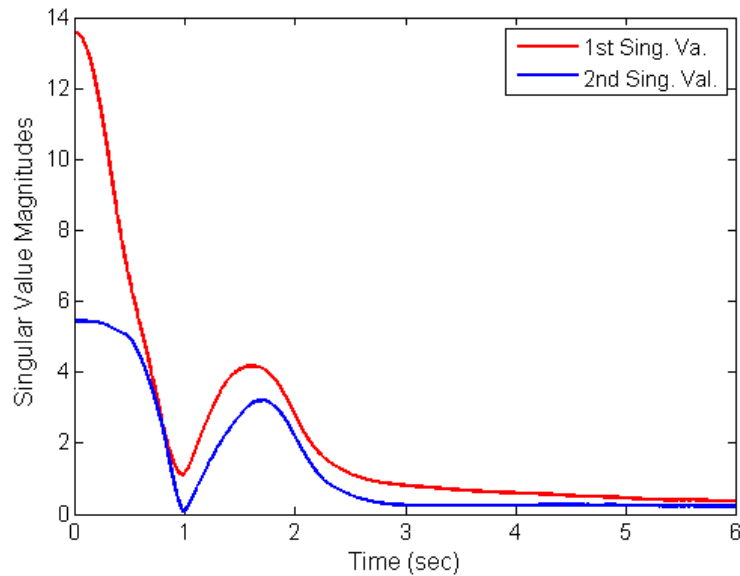
Figure 67 Singular Values of the error matrix, E.

147
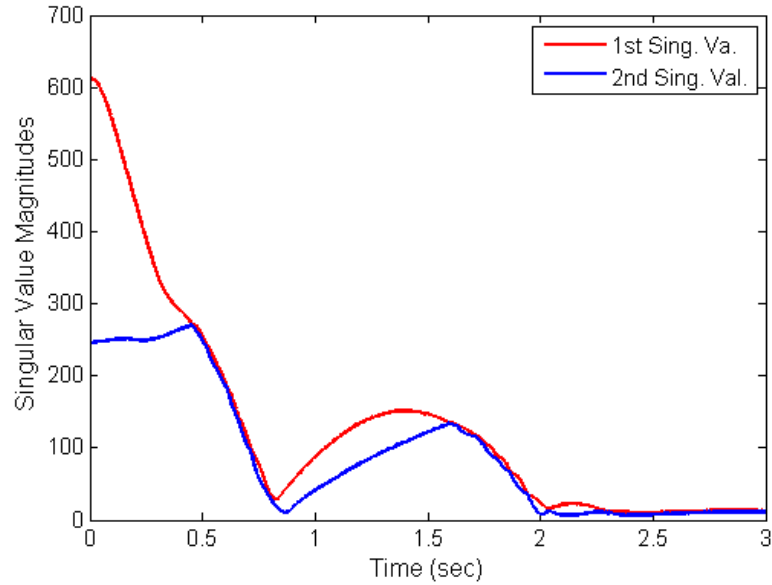
Figure 68 Singular values of the control input matrix, U.
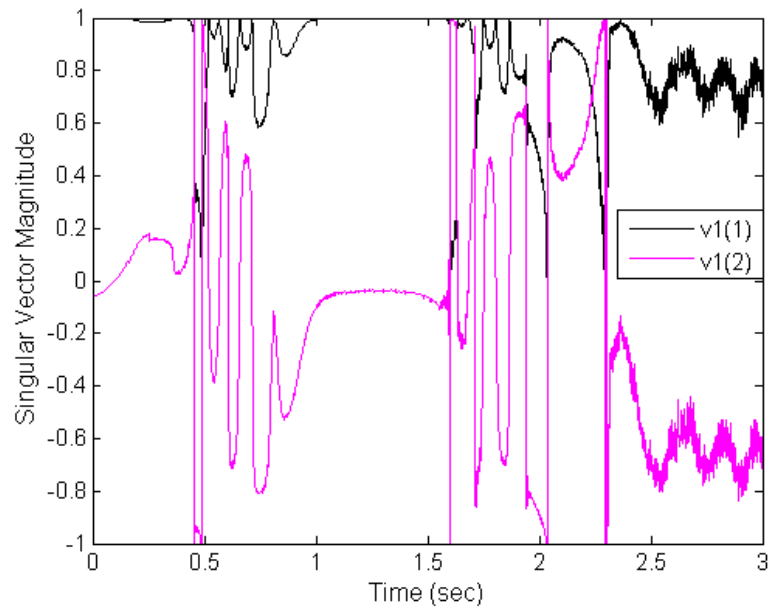
Figure 69 The first right singular vector, v_1, of the control input matrix, U.

Additionally, the disturbance rejection of the SVD System was examined by rotating the wheels from their desired positions by hand. Figure 70 shows the system response with a PID controller with gains $k_p = 37$, $k_i = 15$, and $k_d = 15$. Notice that although Motors 32 and 12 are not disturbed, they move to bring the disturbed motors quickly back to equilibrium.

148
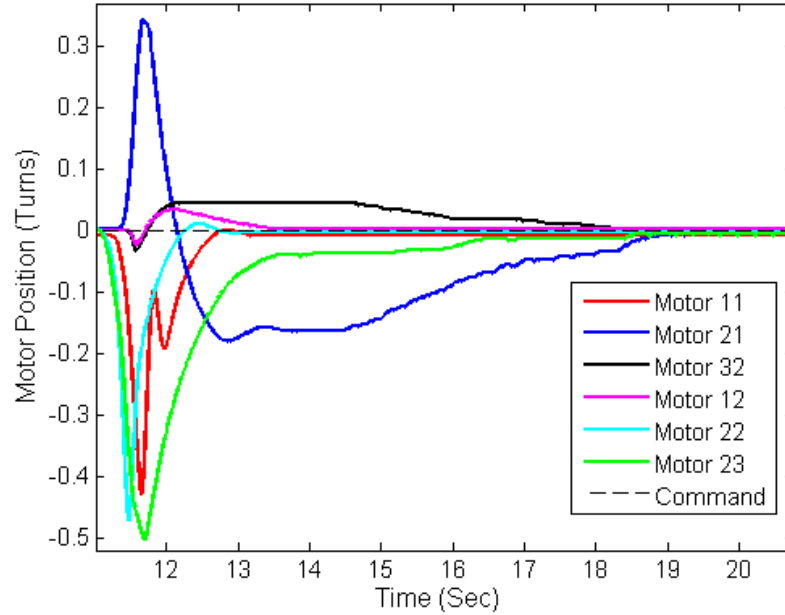
Figure 70 Disturbance rejection of the SVD System.

Trajectory tracking was tested using a number of different types of trajectories, including a sine wave, and a series of sloped surfaces. For the trajectory tracking examples, the nonlinear controller shown in Figure 65 is used with $k_p = 12$, $k_d = 5$, and the offset equal to 8.5. The response for a sine trajectory with a frequency of 1 radian per second and varying phase lags is shown in Figure 71. The same trajectory at 2 radians per second is shown in Figure 72. There is significant tracking degradation for the 2 rad/sec trajectory in spite of the bandwidth of the motors being capable of tracking a sine wave of that frequency. This demonstrates the effective reduction of the "system bandwidth" using the SVD System. However, this reduction is not as great as if line scanning were used. Also, if the phase lags were primarily aligned with either the grid's rows or columns, then the effective "system bandwidth" of the SVD System would be increased.

A separate trajectory involved a series of sloped surfaces. The interest in this response, shown in Figure 73, is not only in the tracking but also in the effect of the saturation on the SVD Response. For this surface the control gains were $k_p = 16$ and

149

$k_d = 7$. Due to the size of the motions and the magnitude of the selected gains, the control input saturates, as shown in Figure 74, causing the direction of the motion to be in neither the direction of σ_1 nor σ_2. Instead the control input is equal for every motor. This input affects both σ_1 and σ_2, as shown in the zoomed-in plot in Figure 75.
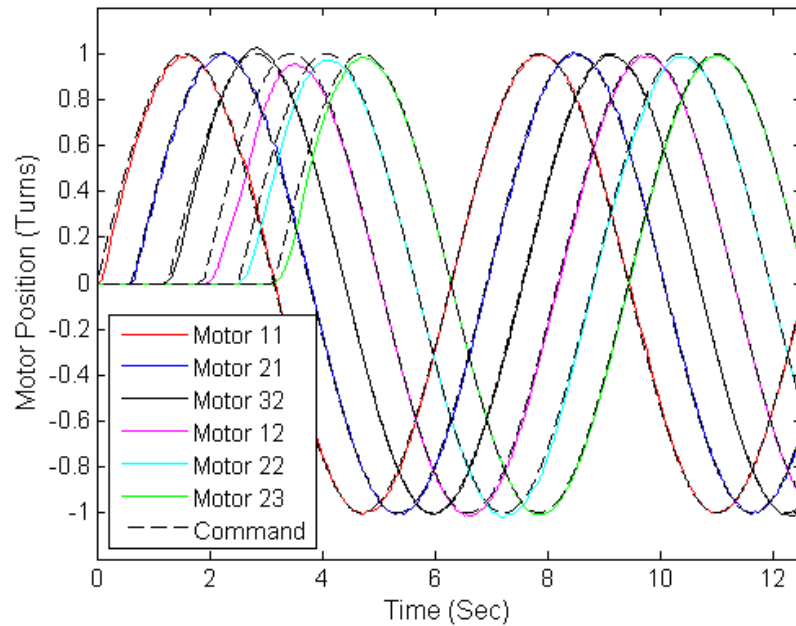
Figure 71 Tracking a sine wave with $\omega = 1$ rad/sec and different phase lags.
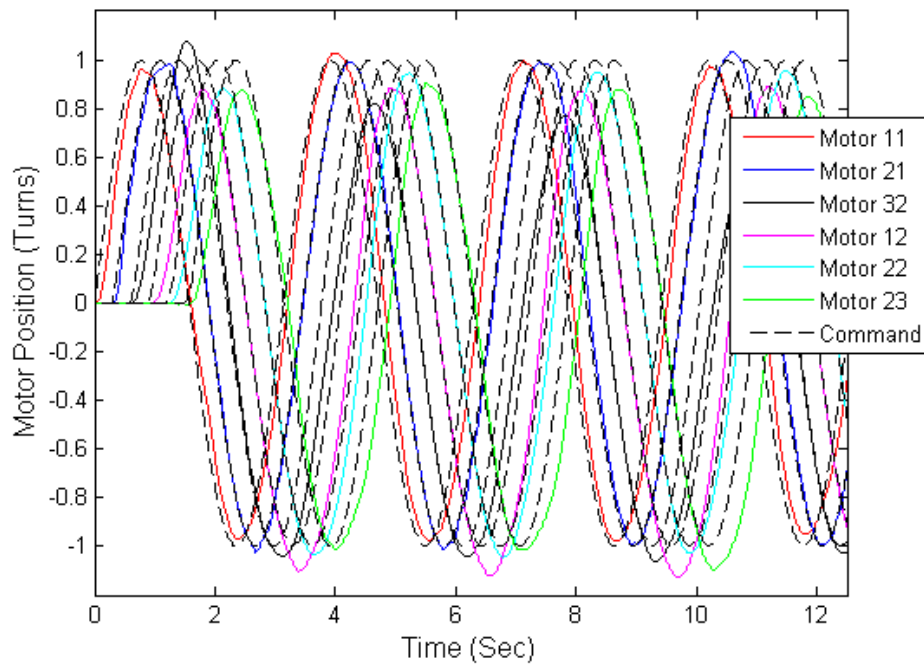
Figure 72 Tracking a sine wave with $\omega = 2$ rad/sec and different phase lags.
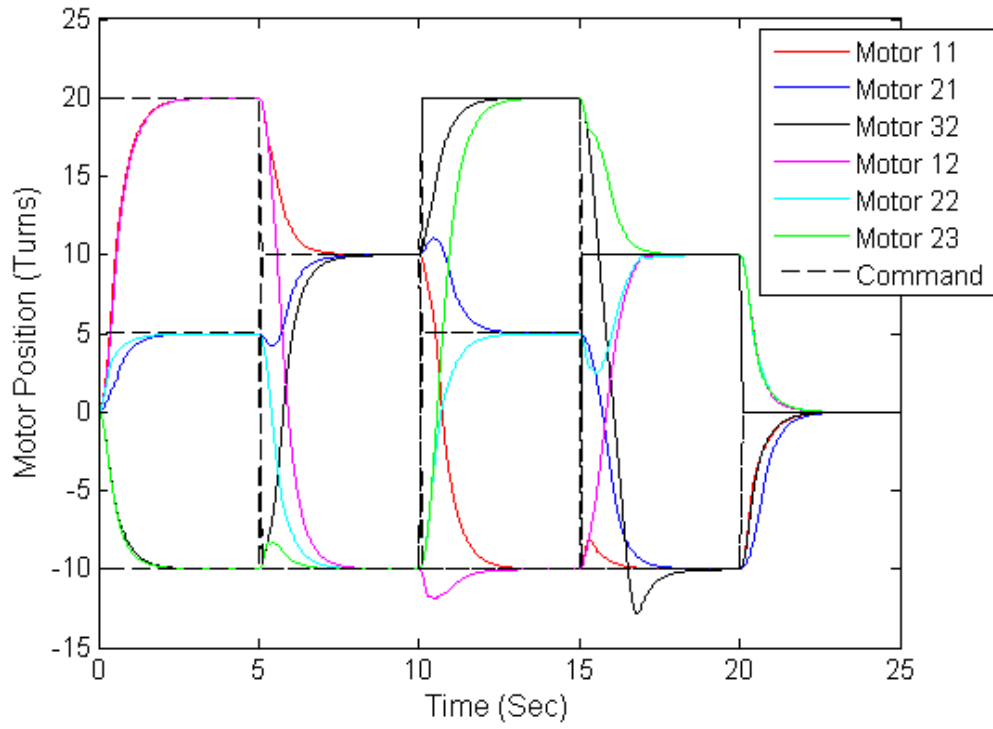
Figure 73 Tracking a series of sloped surfaces using the SVD System.
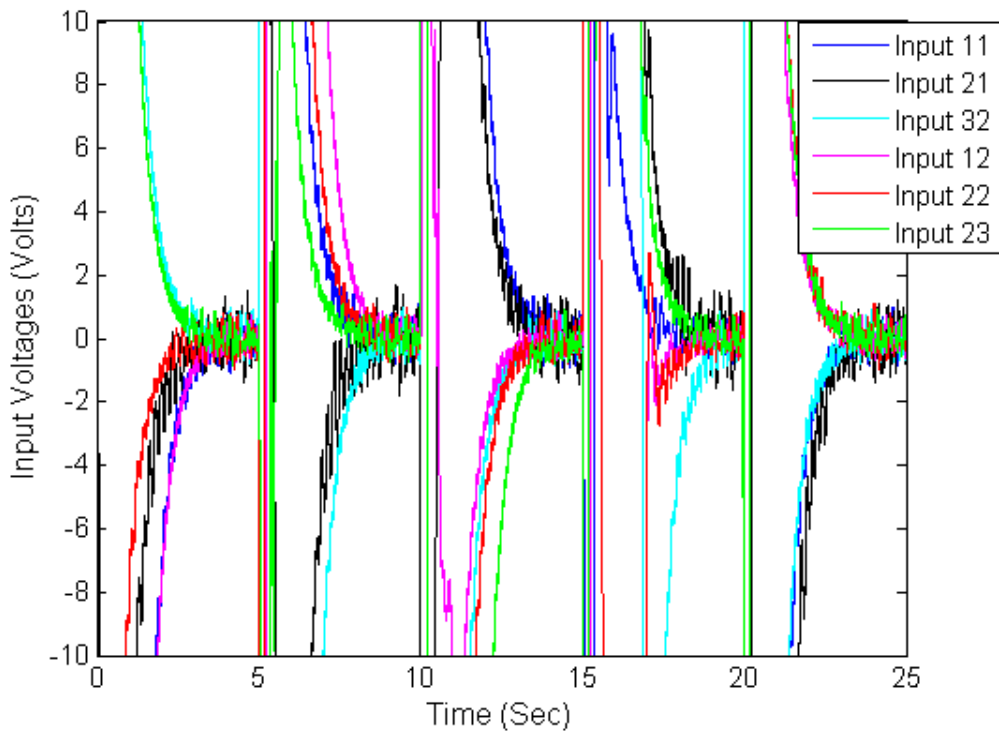
Figure 74 Input voltages for each subsystem resulting from row-column multiplication.

151
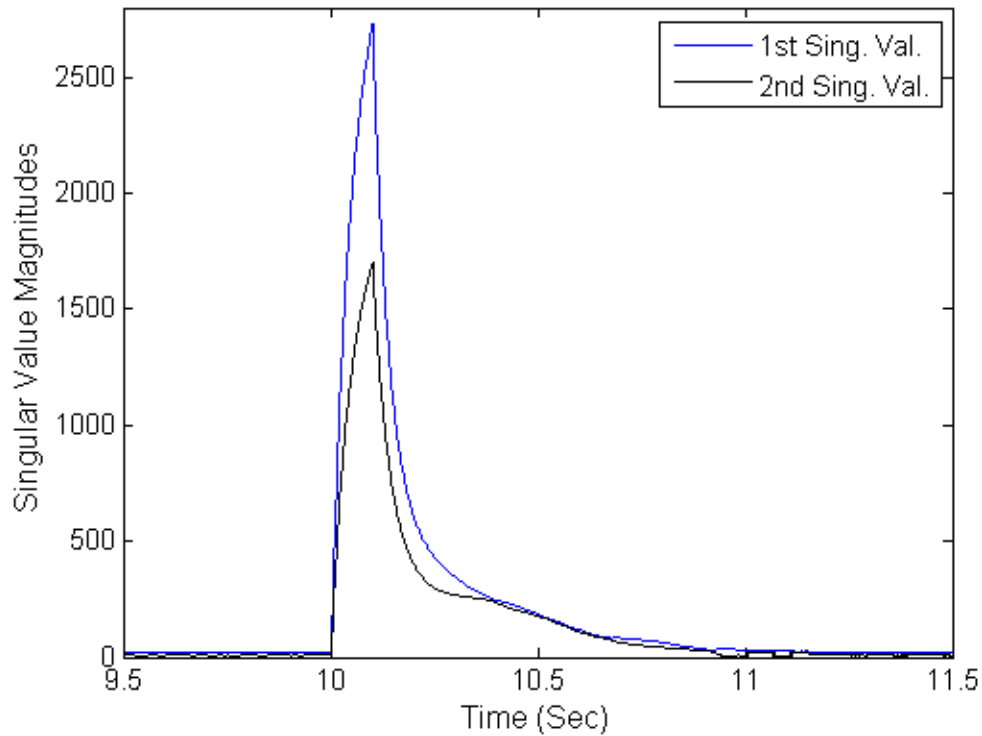
Figure 75 Singular values of the control input matrix, U, for the step at $t = 10$ of the sloped surface trajectory.

Power Method

The Power Method, discussed in Chapter 8, was used to determine the SVD rank-one approximation. The selected method capped the number of iterations at 20 and used random vectors for initialization. The resulting performance was essentially the same as the SVD System using the Golub-Reinsch algorithm. Figure 76 compares the singular values of the control input matrix as computed by the Golub-Reinsch algorithm and by the Power Method. Only the first singular value is shown for the Power Method. It is able to track the true first singular value, as computed by the Golub-Reinsch algorithm, except where it is nearly the same as the second singular value. However, the deviations observed were less than those observed in earlier simulations and they did not noticeably affect the system response.

152
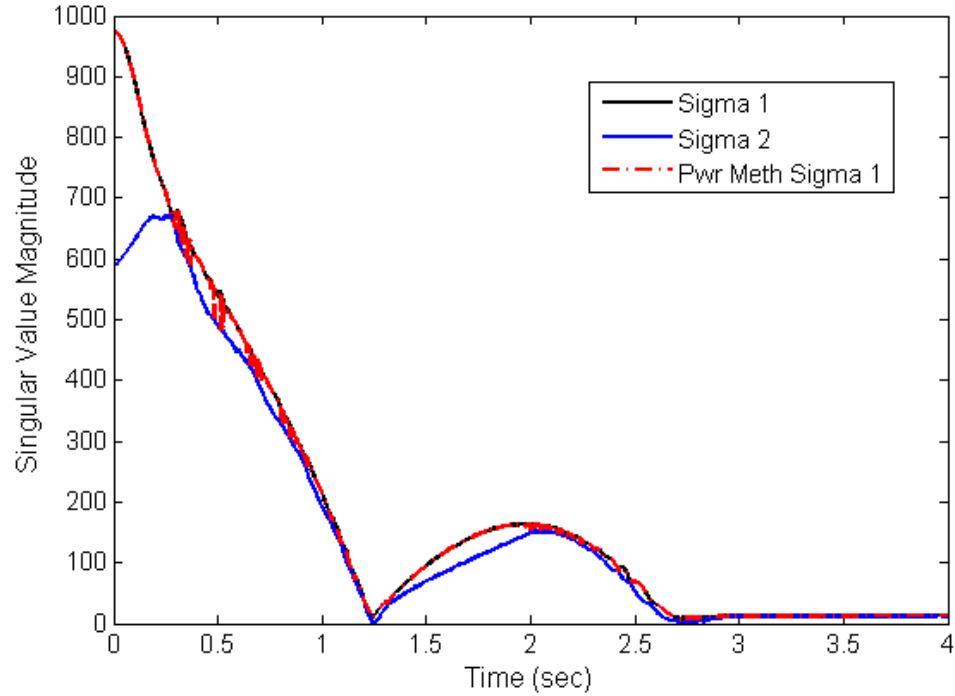
Figure 76 Singular values of the control input matrix as computed by the Golub-Reinsch algorithm and Power Method.

The SNMF System

The SNMF System was also tested using the DC motor grid, although nonnegativity was not a constraint of the system. The SNMF System was able to control the motors with nearly the same response characteristics as the SVD System, as shown in Figure 77. For this example, the linear controller was used with $k_p = 15, k_i = 7$, and $k_d = 10$. Notice that although the desired position of the motors contains both positive and negative values, the SNMF System is able to reach these positions in nearly the same time as the SVD System. However, regarding stability, the SNMF System incurred sustained oscillations with a saturated control input for $k_p = 35$, a lower gain than for the SVD System or IC system.
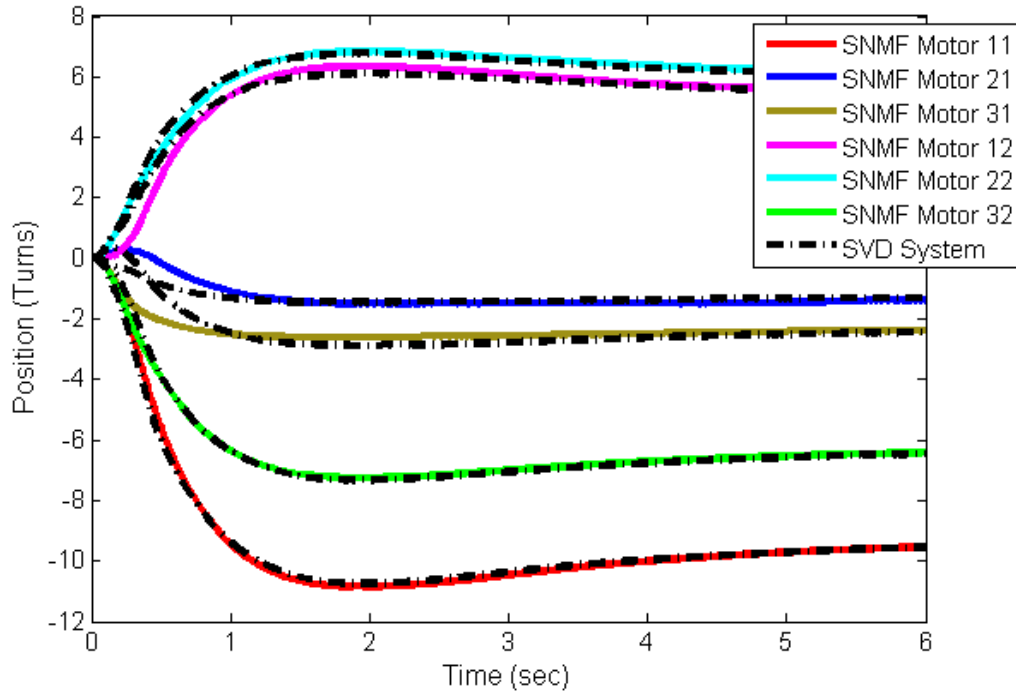
Figure 77 Step response of the SNMF System compared with the SVD System.

RC Circuit

A grid of RC circuits was used to demonstrate the use of the SVD and SNMF

Systems for a set of first-order LTI subsystems without a free integrator. As discussed in

Chapter 7, without the use of integral control the SVD and SNMF Systems cannot reduce

steady-state error in the same way as independent control when the subsystems

themselves do not have a free integrator. In addition to exploring this idea, this section

examines the concept of practical controllability discussed in Chapter 3. The use of the

row-column structure, while maintaining the theoretical controllability of the subsystems,

reduces the region of the state space that can be achieved in practice, due to limitations

such as saturation that affect the system response. Although this reduction is present in

many systems, tracking without a free integrator in the subsystems presents a clear

example of the reduction. First, however, the system will be described, and then results

from the experiments will be presented.

154

System Description

As mentioned, the subsystems are RC circuits. They are connected via row and column inputs that are combined by the same AD633 four-quadrant multipliers used for the DC motor demonstration. In order to visually demonstrate the results, the output voltage across the capacitor also controls the voltage across an incandescent light bulb. To power the bulb, a Texas Instruments OPA551 op amp is used in the non-inverting amplifier configuration. Figure 78 shows the RC circuit for one subsystem with row and column inputs and the power op-amp driving the light bulb.
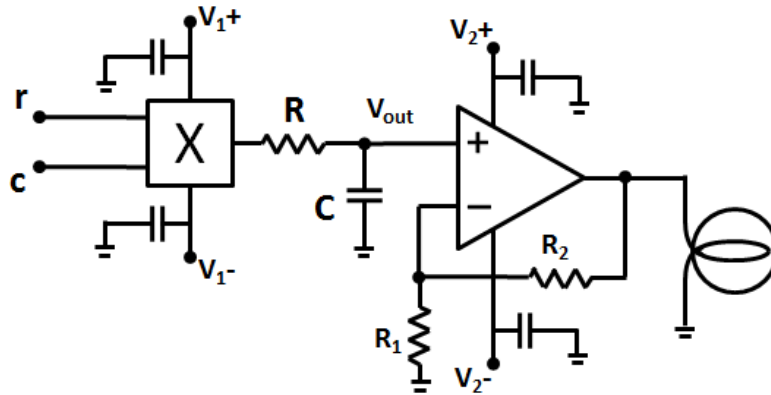
Figure 78 RC circuit and light bulb power circuit with row and column inputs, r and c, controlling the voltage, V_{out}, across capacitor C.

The dynamic model for the RC circuit subsystem is

$$V_{out}(s) = \frac{1}{10}\left(\frac{1}{RCs + 1}\right)r(s)c(s), \tag{229}$$

where $R = 20$ kohms and $C = 0.47$ µF. The gain, $1/10$, is due to the AD633 multiplier, as shown in (226). Further, as shown in Figure 78, $R_1 = 1$ kohm and $R_2 = 3.9$ kohm. The op amp and light bulb are for demonstration purposes only and, other than additional noise, have a minimal effect on the dynamics of the subsystems. Therefore, during the experiments, the light bulb was unplugged to ensure that the op amp would not be broken during testing due to overdrawing of the current. The response of a single subsystem to

an open loop step input is compared with the subsystem model in Figure 79. In addition to the linear dynamics of the subsystems, there are saturation limits of ± 10 V on both the inputs, r and c. It is also desired that V_{out} be restricted to ± 5 V continuous due to the limitations of the power op-amp. A 4×4 grid was used, requiring 8 inputs to control 16 outputs. Figure 80 shows the entire grid. The control of the grid was carried out using MATLAB XPC Target running at 1 kHz unless otherwise specified.
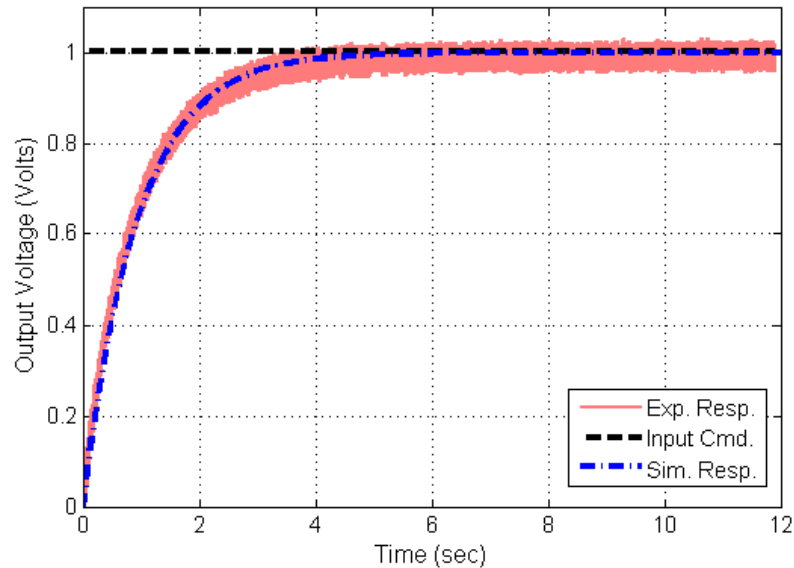
Figure 79 Open loop unit step response of a single subsystem and a simulated subsystem using the model given in (229).
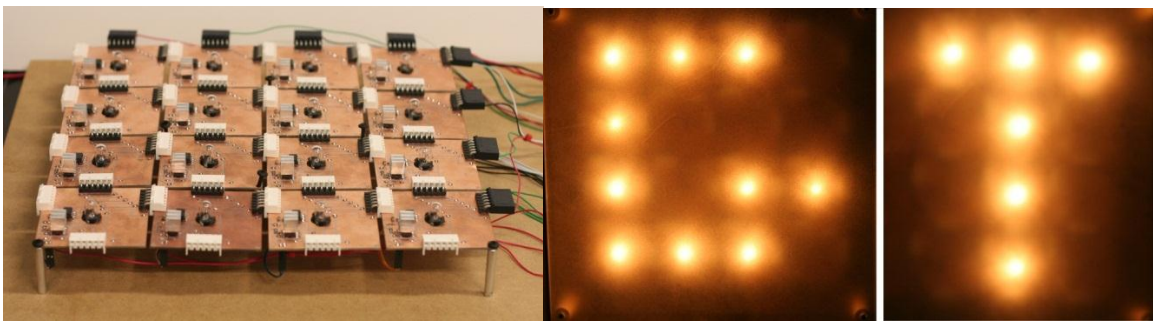
Figure 80 The RC circuit and light bulb grid (*left*) and the light bulbs on in two configurations (*right*).

The SVD System Response

Besides simply demonstrating that the SVD System works, this section focuses on its use for subsystems without free integrators. The challenge is that if a voltage input is

156

taken away from one subsystem, then the voltage output will return to zero. Therefore,

the output voltage of the entire system will converge to whatever rank-one voltage input

is given at any instant in time. To maintain a full-rank voltage output, the voltage input

must constantly change between the various rank-one approximations. The simplest

example is given by a diagonal matrix.

Consider a step input to the diagonal matrix,

$$V_{des} = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}. \tag{230}$$

To generate this desired output voltage with IC, the steady-state voltage input for the

subsystems on the diagonal is 4 V. However, the SVD System will switch between each

subsystem on the diagonal so that a voltage input to any one subsystem is only given for

1/4 of the time. Therefore, to maintain (230), the magnitude of the voltage input for the

SVD System must be 16 V. This is fine as long as there is no limit to the voltage input.

However, a saturation limit can make (230) unreachable for the SVD System even though

it is reachable using IC. For instance, using a simulation of the RC circuit grid with PI

control and an input voltage saturation of ± 10 V for both the row and column inputs, the

response to the step input results in an output voltage of 2.5 V for each subsystem on the

diagonal. The row and column inputs switch from $0 - 10$ V every 4 samples so that the

control input switches between each of the subsystems on the diagonal, as shown in

Figure 81. The subsystems off the diagonal receive no input. This is an example of the

reduction of the practically controllable state space discussed at the end of Chapter 3.

157
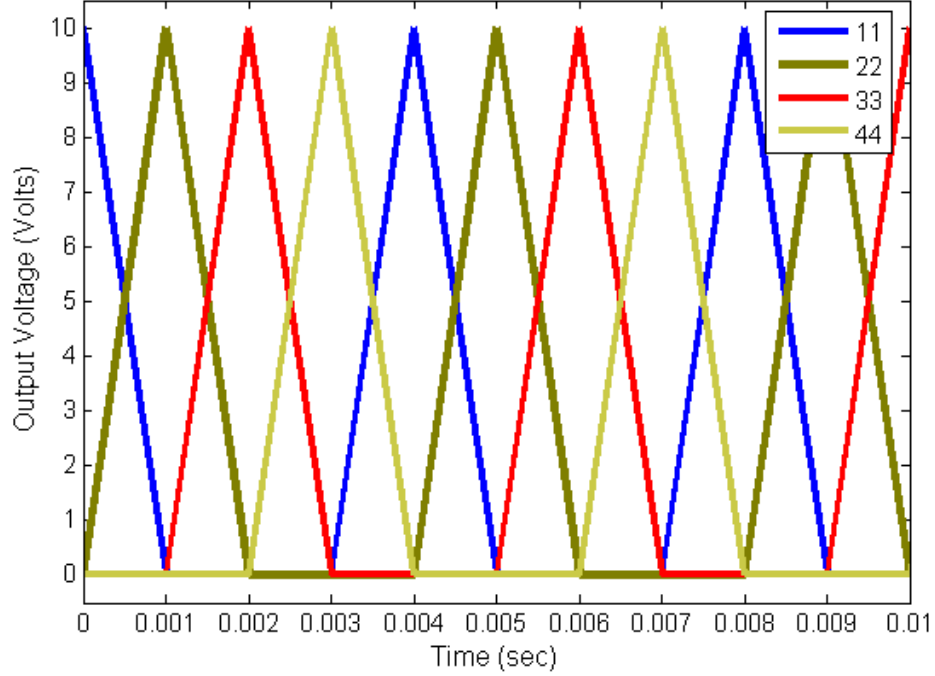
Figure 81 Control input for the subsystem on the diagonal for a step input to (230).

By the addition of random values of a small magnitude to the voltage reference

command,

$$V_{des} = \begin{bmatrix} 4.0040 & 0.0018 & 0.0090 & 0.0034 \\ 0.0008 & 4.0024 & 0.0094 & 0.0090 \\ 0.0024 & 0.0042 & 4.0049 & 0.0037 \\ 0.0012 & 0.0005 & 0.0049 & 4.0011 \end{bmatrix}, \tag{231}$$

the SVD System is able to track the ≈ 4 V commands along the diagonal while

maintaining the off-diagonal subsystems at ≈ 0 V, as shown in Figure 82. This can be

accomplished because of the ability of the SVD System to provide control input to the

subsystems off of the diagonal, as shown in Figure 83. In fact, even if the diagonal

elements are set to the saturation limit, the SVD System can track these signals within 2%

error. Of course, this result is dependent on the sample-time of the controller and the time

constant of the system.

158
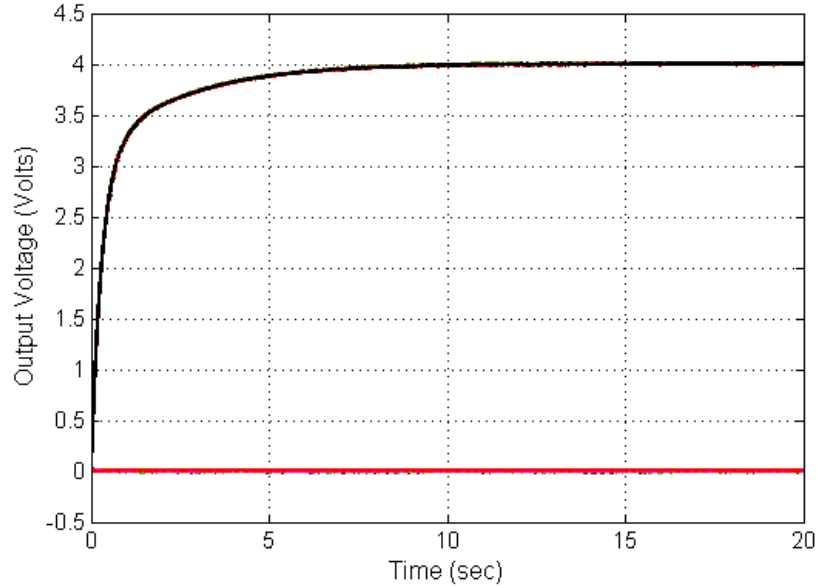
Figure 82 Step response to (231) for the SVD System.
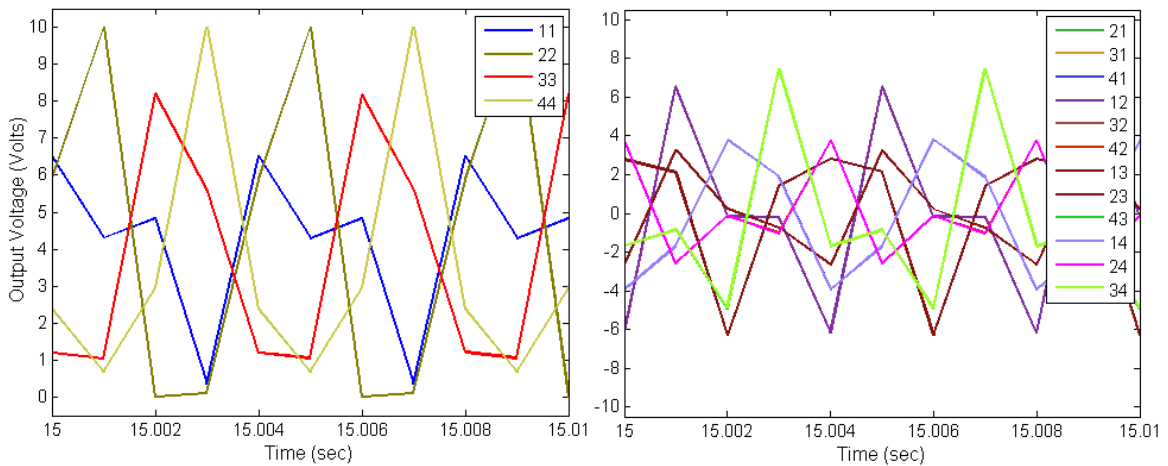
Figure 83 Control input for the subsystems on (*left*) and off (*right*) the diagonal for a step input to (231).

For the physical system, with saturation limits of ± 10 V on the row and column inputs, the noise in the system has been demonstrated experimentally to make it possible for the SVD System to maintain any arbitrary output voltage in the allowable range of $0 - 5$ V with varying degrees of accuracy. Figure 84 shows the response for the SVD System and IC to (230), for 5 V and 2 V along the diagonal, rather than for 4 V. Notice that the response to 5 V has significant oscillations at steady-state that are not seen for the 2 V response due to the input saturation. For these responses, $k_p = 10$ and $k_i = 50$.

159
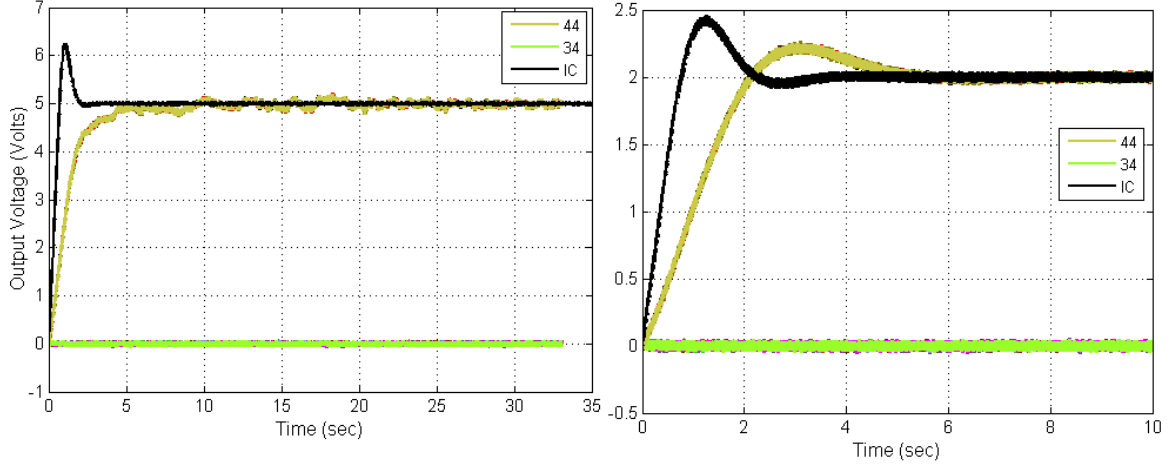
Figure 84 Step response to 5 V (*left*) and 2 V (*right*) along the diagonal for the SVD System as compared with IC. Subsystems 44 and 34 are labeled as they are the only subsystems visible for the SVD System.

The example discussed in the above paragraph is just one instance of the reduction of the practically controllable state-space because of the constraints of the row-column structure. However, through the use of the SVD System, there is less reduction than if a line scanning approach were used. For example, if the voltage reference command for every subsystem is 5 V, then the SVD System would respond exactly as IC. To examine the response for a range of commands, a set of test commands was used:

$$V1_{des} = \begin{bmatrix} 1.6870 & 2.6230 & 2.7149 & 2.6219 \\ 3.6629 & 0.1428 & 3.0310 & 0.6847 \\ 3.1688 & 3.3965 & 2.9725 & 2.8242 \\ 3.8380 & 3.7360 & 1.5689 & 0.1273 \end{bmatrix};$$

$$V2_{des} = \begin{bmatrix} 3 & 3 & 3 & 2 \\ 3 & 4 & 3 & 2 \\ 3 & 3 & 3 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}; V3_{des} = \begin{bmatrix} 3.2 & 3.8 & 4 & 4 \\ 3.5 & 2.6 & 1.4 & 2 \\ 3.7 & 2.5 & -3 & -3 \\ 4 & 2 & -3 & -4 \end{bmatrix}; \quad (232)$$

$$V4_{des} = \begin{bmatrix} 0 & 4 & 4 & 0 \\ 4 & 0 & 0 & 0 \\ 4 & 0 & 4 & 4 \\ 0 & 4 & 4 & 0 \end{bmatrix}; \text{and } V5_{des} = \begin{bmatrix} 0 & 4 & 4 & 4 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix}.$$

$V1_{des}$ is a set of random values, resulting in a full-rank command that has no designer bias. $V2_{des}$ and $V3_{des}$ represent relatively smoothly changing commands, similar to a surface that might be displayed by a pin array. $V3_{des}$ includes both positive and negative

160

values. The final two reference commands, $V4_{des}$ and $V5_{des}$, shown in Figure 80, represent sparse commands (and also a little school spirit). The SVD System is able to successfully generate all of these voltage references. For these tests, the values for the PI controller were $K_p = K_i = 150$.

The set of test commands in $V1_{des}$ will be used here for discussion. The responses for all the reference commands are given in Appendix C. Figure 85 shows the output voltage and error. Notice that the system is able to converge without any steady-state oscillations. The singular values for the error and control input are shown in Figure 86. The lower singular values of the control input gain in magnitude due to the integral control, while σ_1^{ci} is reduced in magnitude because the system is being initially driven in that direction. The convergence rate for the lowest singular values is heavily dependent on the integral gain. For example, the system does not reduce the error in the direction of what is initially σ_4^e until the integral control raises the control input in that direction to the same magnitude as the other singular values. This is seen in Figure 86 in that the lowest singular value of the error, as shown in the left plot, does not begin to converge until about 6 seconds when the control input in the direction of σ_4^{ci} is of equal magnitude with the other singular values, as shown in the plot on the right.
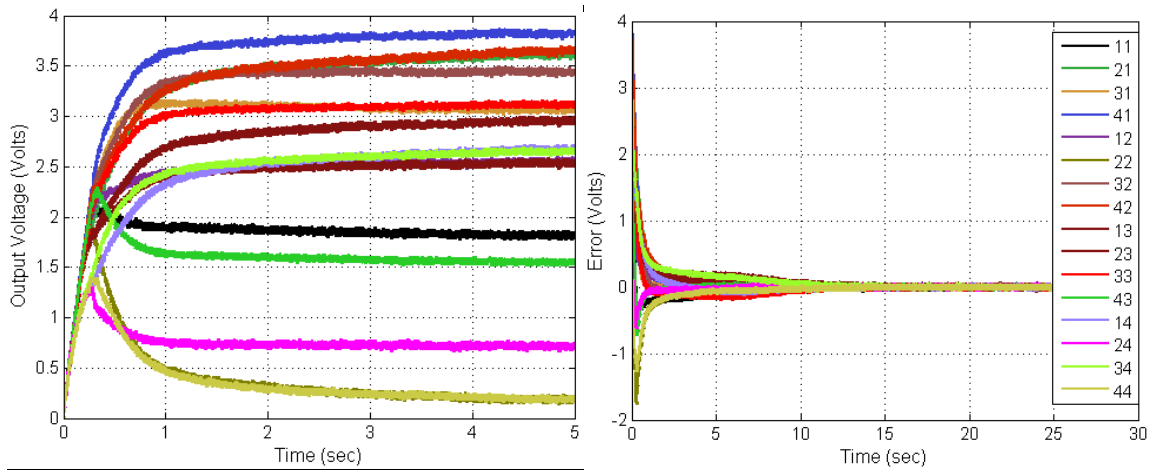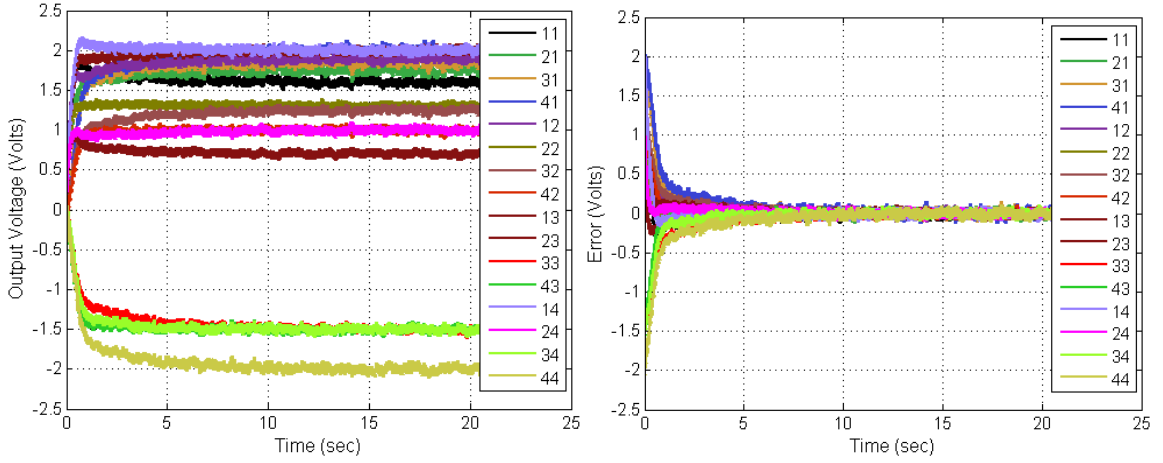
Figure 85 Response of the SVD System (*left*) and the error (*right*) for a step input to $V1_{des}$.
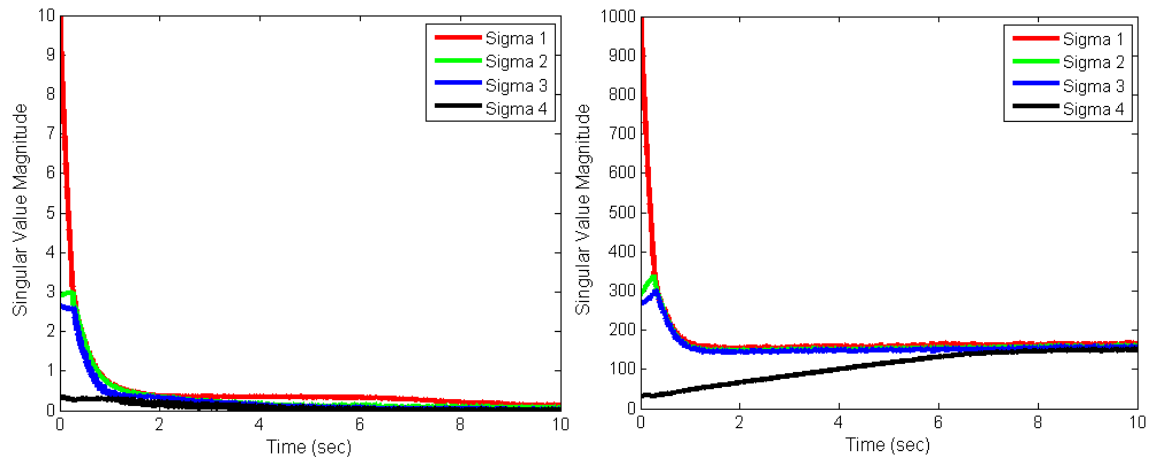
Figure 86 Singular values of the error (*left*) and control input (*right*) for a step input to $V1_{des}$.

The SNMF System Response

This section will extend the discussion of the previous section to the implementation of the SNMF System. The SNMF System is able to successfully control the grid in real-time at a 1 kHz sample rate using a tolerance of 0.01. The SNMF System is not able to reach the same outputs as the SVD System due to its additional constraints and numerical technique. Consider the diagonal matrix in (230). The SVD System is able generate this matrix, even if the voltage command on the diagonal is 5 V because of the noise in the system. The SNMF System is able to generate this matrix for 2 V just as well as the SVD System, but the performance deteriorates at 4 V, and at 5 V the SNMF

162

System cannot generate the commanded voltages. This is shown in Figure 87. For these responses, $K_p = K_i = 150$.
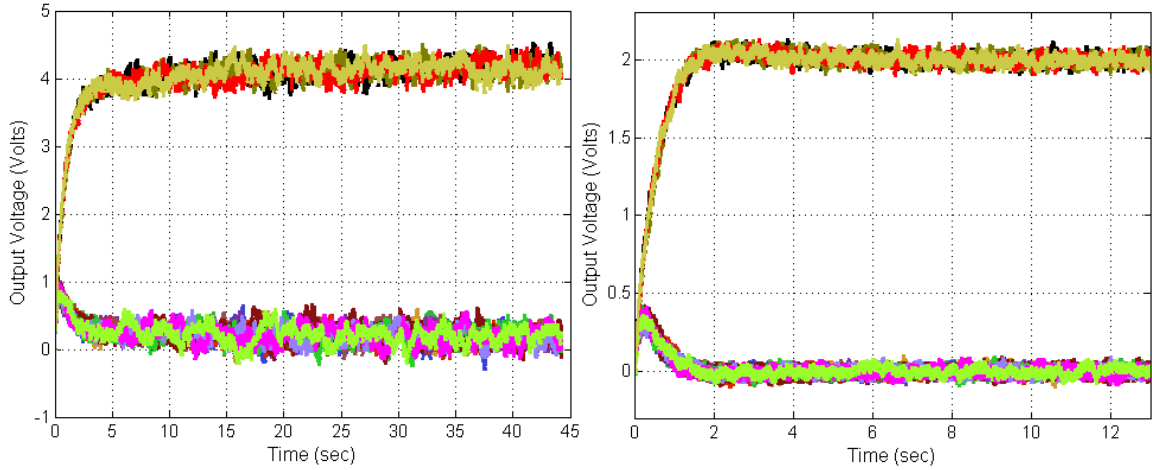
Figure 87 Step response to 5 V (*left*) and 2 V (*right*) along the diagonal for the SNMF System.

The test commands in (232) used for the SVD System were also tested with the SNMF System using a PI controller with $K_p = 100$ and $K_i = 50$. The results are shown in Appendix D. Notice that only the lower-rank and positively-valued matrices are achieved with few oscillations. In addition, $V3_{des}$ and $V4_{des}$, in particular, are not well achieved using the SNMF System. However, if the magnitude of the matrices is reduced, the SNMF System is able to achieve the command voltages. For example, if the magnitude of $V3_{des}$ is halved such that it becomes

$$V3\text{Half}_{des} = \begin{bmatrix} 1.6 & 1.9 & 2 & 2 \\ 1.75 & 1.3 & 0.7 & 1 \\ 1.85 & 1.25 & -1.5 & -1.5 \\ 2 & 1 & -1.5 & -2 \end{bmatrix}, \tag{233}$$

then the SNMF System can reach the desired voltage without significant oscillations, as shown in Figure 88. Notice in Figure 89 that the SNMF is not able to provide an input that is only in the direction of σ_1 due to the nonnegativity constraint. This is the reason that the SNMF System is not able to reach as much of the state space as the SVD System.

163

Figure 88 Response of the SNMF System (*left*) and the error (*right*) for a step input to $V3\text{Half}_{des}$.
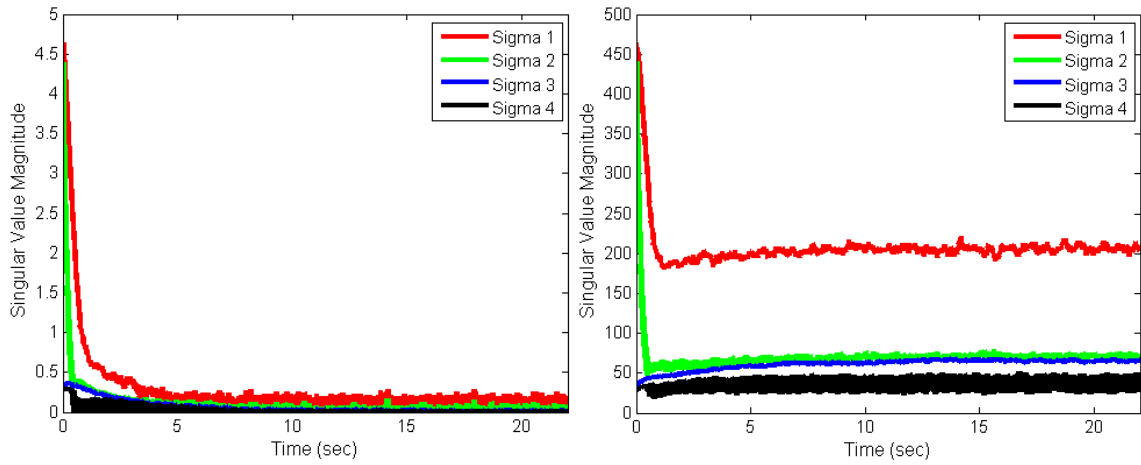
Figure 89 Singular values of the error (*left*) and control input (*right*) for a step input to $V3\text{Half}_{des}$.

164

CHAPTER 10

CONCLUSIONS AND FUTURE WORK

This thesis presents a new concept for controlling a set of systems coupled by the row-column structure by reducing the dimension of the control input using the SVD and the SNMF. In the process, the effect of the dimension reduction of a control signal has been explored. It has been proven that theoretical controllability is maintained despite a significant reduction in the number of inputs. It has also been shown experimentally that although theoretical controllability is maintained, the row-column structure's constraints reduce the space of states that are practically reachable.

The SVD and SNMF Systems have been demonstrated to be effective means of controlling subsystems that are coupled by the row-column structure. Theoretical conditions for the stability of these systems have been proven and validated through simulations and physical experiments. By relating the SVD System to independent control, guarantees for performance have been identified that reduce the control design task to the design of a controller for an individual subsystem. Those performance guarantees have been validated in simulations, and simulations and experiments have provided performance comparisons using the SVD and SNMF Systems and independent control and line scanning procedures. These have shown that the convergence rates of the new methods are much closer to independent control than they are to line scanning when using the same controller for each.

In addition to closed loop control techniques, open loop or command generation procedures have been examined. While these techniques improve on line scanning, they

have been shown to be significantly slower for dynamic subsystems than just using the SVD and SNMF Systems.

The physical implementation of the SVD and SNMF Systems has been explored using a row-column structure in multiple physical domains. A simulation examined a fluid power application through the use of a model of Digital Clay. That example shows the importance of the physical row and column signals being able to track the desired row and column inputs. It also demonstrates the performance improvement of the SNMF System for generating surfaces for Digital Clay in spite of system power constraints.

Experimental demonstrations were provided for electrical systems. The demonstrations validated the use of the SVD and SNMF Systems in real-time for systems with subsystem variation, saturation, and nonlinear and non-integrating dynamics. The real-time implementation of these examples was shown using only small grid sizes, but computational techniques were explored that permit the SVD System to be implemented in real-time for grid sizes on the order of $1,000 \times 1,000$ and larger. These computational techniques would also likely extend to the SNMF.

In summary, the SVD System provides an effective method to control numerous subsystems using only a few inputs when the subsystems can be coupled using the row-column structure. It offers orders of magnitude improvement over the traditional line scanning technique used to control subsystems in the row-column structure. The theoretical basis for this technique has been described in detail and validated. When additional nonnegativity constraints are present, the SNMF System provides a way to apply the same general concept of the SVD System while matching the constraints.

166

Future Work

This thesis provides the foundation for controlling many subsystems using input dimension reduction, which, in turn, raises a number of questions in various areas for future study. Theoretically, there is still no proof of stability or performance for general higher-order LTI subsystems with an arbitrary linear controller. Moreover, the proposed concepts have numerous possible theoretical extensions to other specific types of subsystems and control techniques.

In general, this thesis primarily focuses on subsystems that are all the same. Future work should examine situations where the subsystems have different dynamics. The DC motor demonstration provides strong evidence that some variation does not adversely affect the system response of either the SVD or SNMF Systems. Furthermore, in a simulation study using linear systems, it was observed that if the bound on stability for one subsystem is significantly tighter than for the other subsystems, then the stability bound of the SVD System is actually in between the different subsystems. This suggests a promising method for control design. If the controller is designed to provide stability for the subsystem that is least stable, then the SVD System will be guaranteed to be stable. Obviously, no proof of this conjecture has been given, but it is a promising area for future research. Future work also should consider subsystems that vary not just parametrically, but have significant variations from model to model— for example, if the grids of DC motors and RC circuits were combined into one large grid.

This thesis has also focused primarily on subsystems that are dynamically decoupled from one another, such as pin arrays like Digital Clay. However, there are many physical systems where the subsystems are dynamically coupled. For instance, one

proposal involving a pin array is to attach a flexible material, such as latex, to the pins for the purpose of providing a spatial low-pass filter of the surface. Such a filter would, of course, dynamically couple the pins. While coupled subsystems have been noted in connection with some of the theorems in this thesis, namely the small-gain, passivity, and circle criterion theorems, they have not been thoroughly explored, and none of the examples considered coupled subsystems. Similarly, although the small-gain and passivity theories apply to nonlinear subsystems, this thesis has focused on linear systems, particularly in examining system responses. Future work could extend the analyses and comparisons to nonlinear subsystems.

Further study is also needed regarding the various proposed adjustments to the computation of the SVD and the SNMF for use in real-time control. This thesis has shown that it is feasible to speed up the computation of the SVD without adversely affecting performance, but more work is needed to determine the best method. Additionally, to improve performance and to give the control designer more flexibility, the use of a weighted SVD or SNMF could be explored. This would allow the control designer to emphasize individual subsystems that are more important. Further research could examine the use of weights to more cleverly reduce the error of individual subsystems in cases with power limitations and input saturation. It would also be interesting to compare the selection of weights to the selection of control gains.

The use of the row-column structure, particularly in a manner permitting the use of the SVD and SNMF Systems, could be explored in other domains and applications. This thesis has used the pin array as a motivating example and focused on fluid power and electrical systems, but many other applications are possible. For example, large

arrays of MEMS devices have become more prevalent, and the techniques presented in this thesis could provide a useful way to control those devices. Moreover, as sensors and actuators drop in price, the number of applications for the control techniques presented in this thesis will continue to expand.

Finally, an interesting extension of this research is to approach the low-rank control problem from a more general perspective than just the SVD and SNMF. While the SVD provides an input that minimizes the difference between the low-rank input and the independent control input at each step, the problem can be generally stated as establishing a basis from which to select inputs and then selecting the best input from that basis at each sample time. For example, instead of using line scanning as an open loop procedure, it could be used within a feedback loop in a manner similar to the SVD System. At each iteration through the feedback loop, the control input for each line could be compared and the line with the greatest average input could be used for control. This is similar to the control input used in the proof of controllability in Chapter 3. It would not be difficult to prove stability and performance of this feedback control law using the same Lyapunov techniques as those used in Chapter 5. One use of this more general view would be the ability to include different types of constraints, such as limiting the column inputs to either 0 or 1. Also, the SVD System could provide an input that is optimally close to the independent input at each step, not one that is optimal with respect to the system response. Thus another method may provide an improved response. The concepts explored in this thesis have laid the groundwork for wide-ranging possibilities for future research.

SURFACES FOR KINEMATIC TESTS

"Identity x50":

$$S_{des} = \begin{bmatrix} 50 & 0 & 0 \\ 0 & 50 & 0 \\ 0 & 0 & 50 \end{bmatrix} \tag{234}$$

"Gradient":

$$S_{des} = \begin{bmatrix} 50 & 40 & 30 & 20 \\ 40 & 30 & 20 & 10 \\ 30 & 20 & 10 & 0 \\ 20 & 10 & 0 & 0 \end{bmatrix} \tag{235}$$

The other surfaces are 50×50 and will be shown using images:
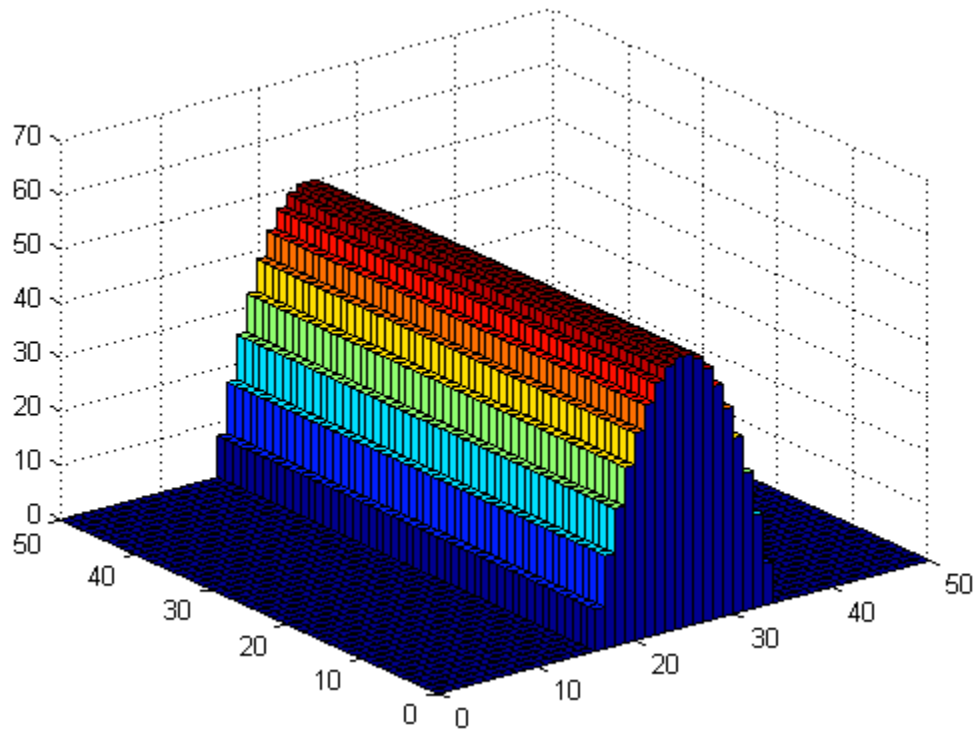
Figure 90 The "parabola" desired surface.
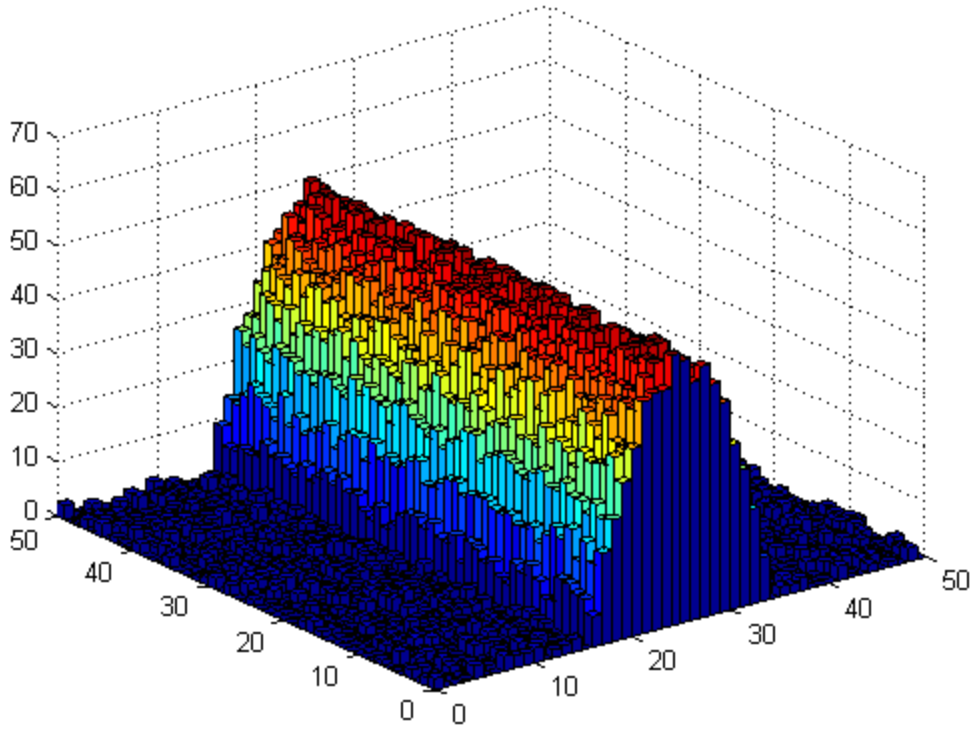
Figure 91 The "parabola with noise" desired surface.
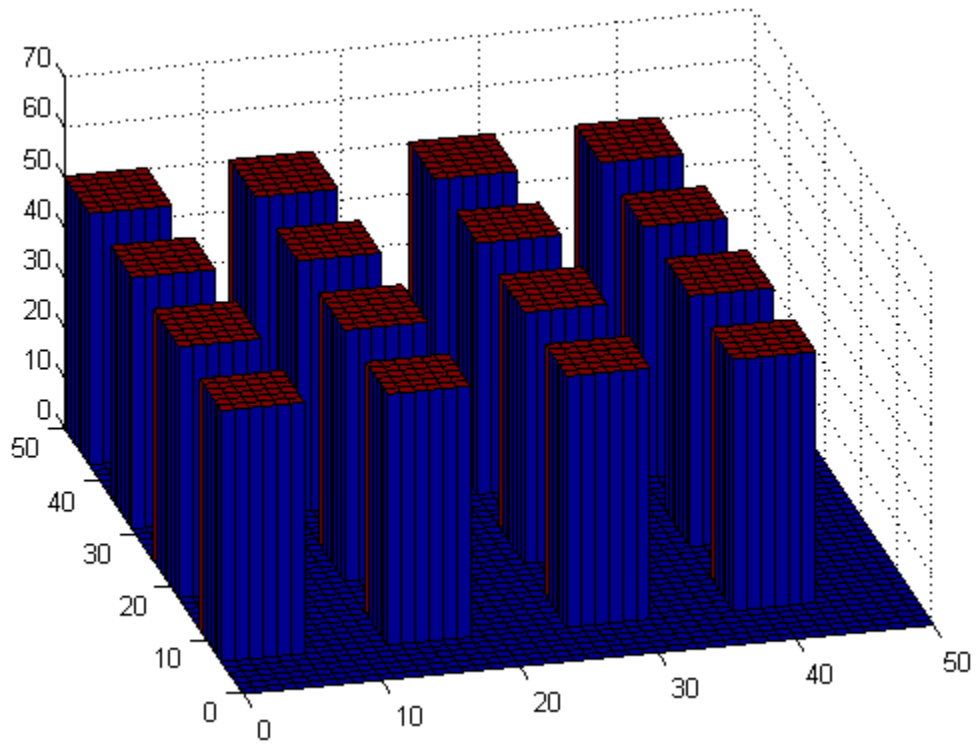
Figure 92 The "Grid of Squares" desired surface.
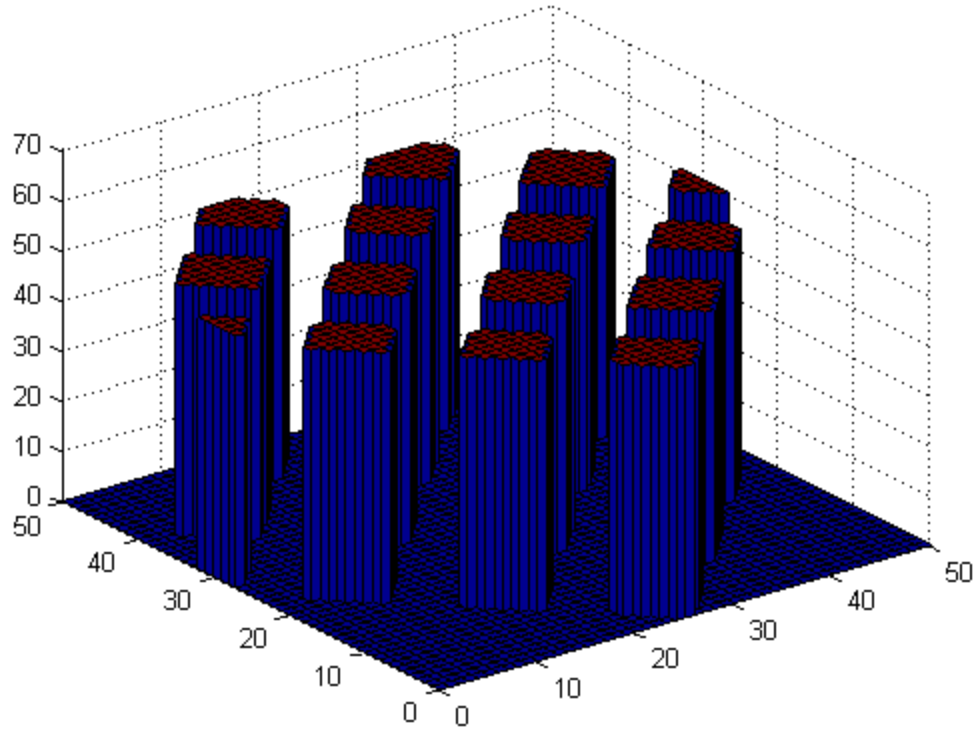
Figure 93 The "Grid of Squares Rotated" desired surface.
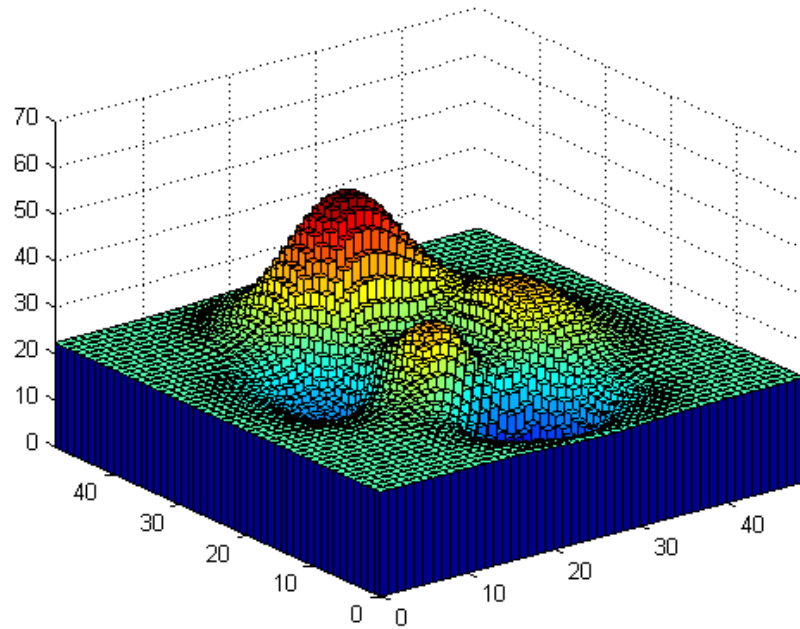
Figure 94 The "peaks" desired surface. The "peaks NZ" surface has the same shape but has a max value of 25 instead of 50 so that there are both positive and negative values.

172
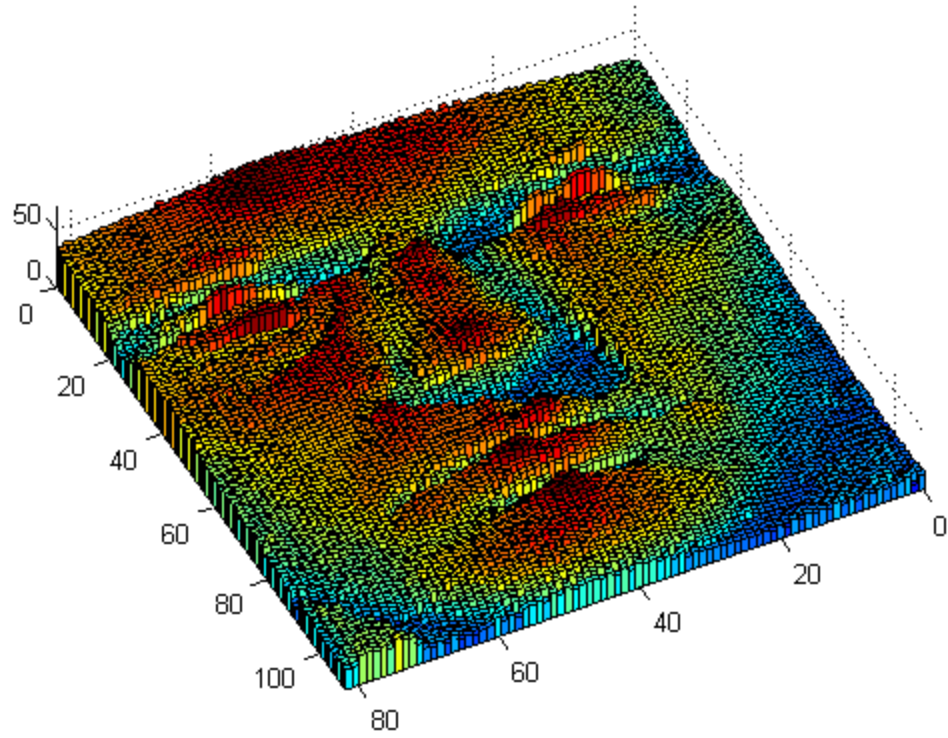
Figure 95 The "face" desired surface.
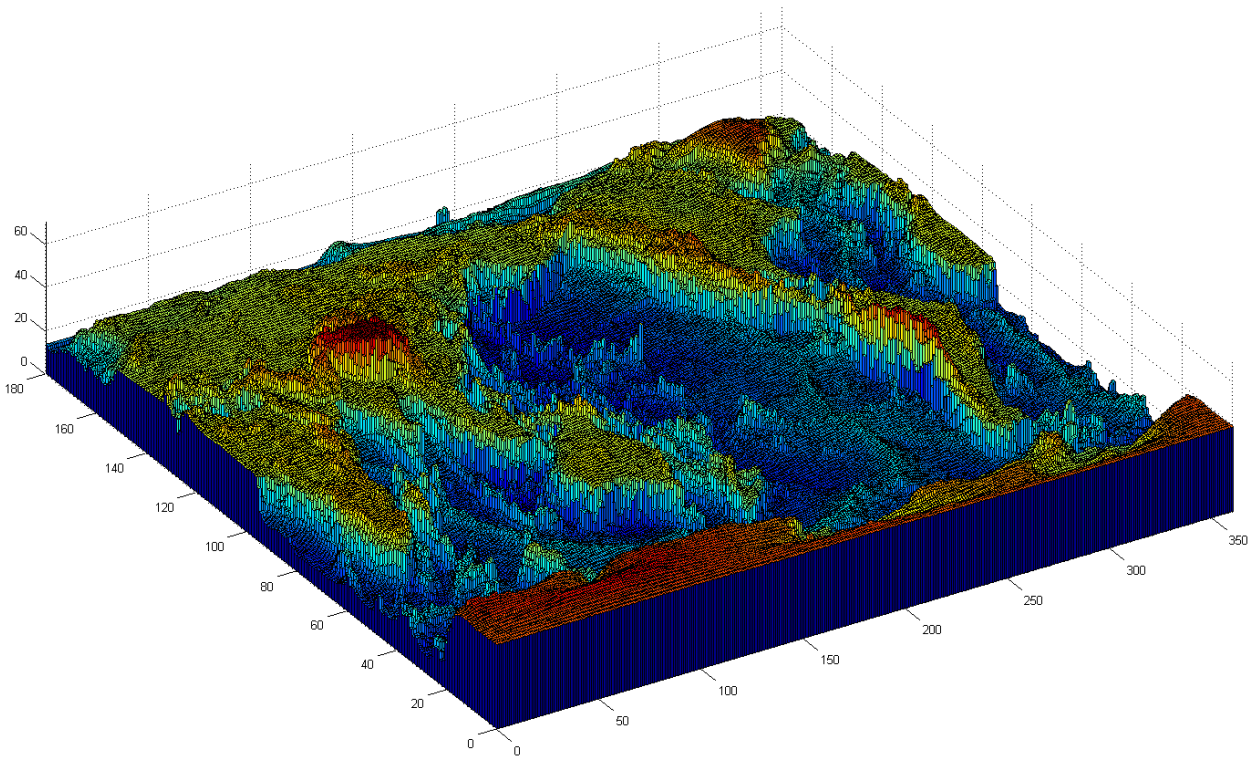
Figure 96 The "World Map" desired surface.

173

APPENDIX B

REFERENCES FOR COMMAND GENERATION TESTS
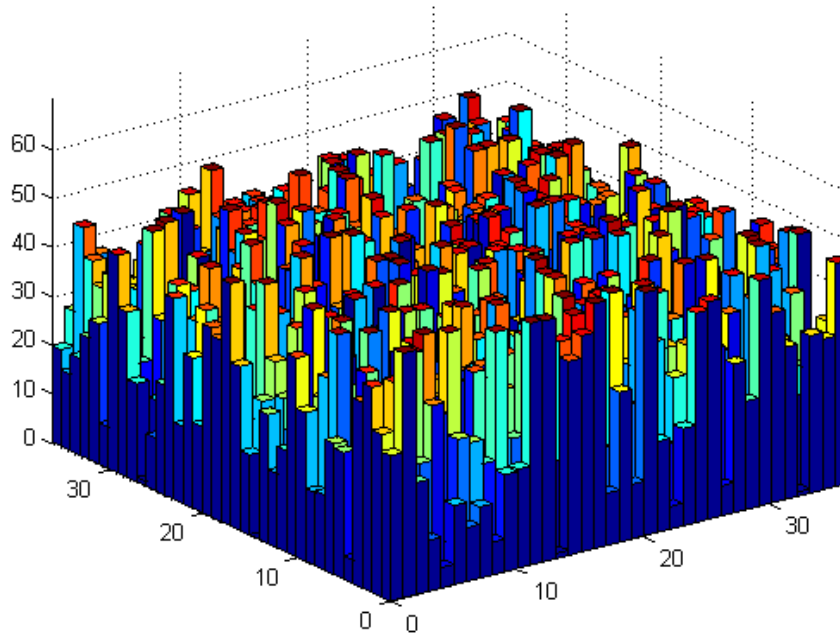
Figure 97 The "Random" reference is a command of random values between 0 and 50.
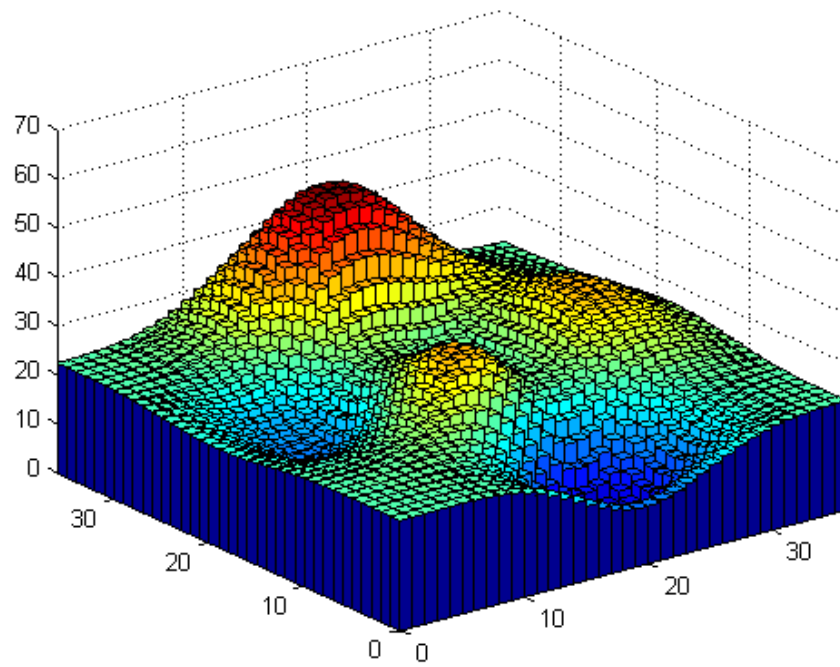
Figure 98 The "Peaks" reference command is a cropped version of the "peaks" surface in Figure 94.
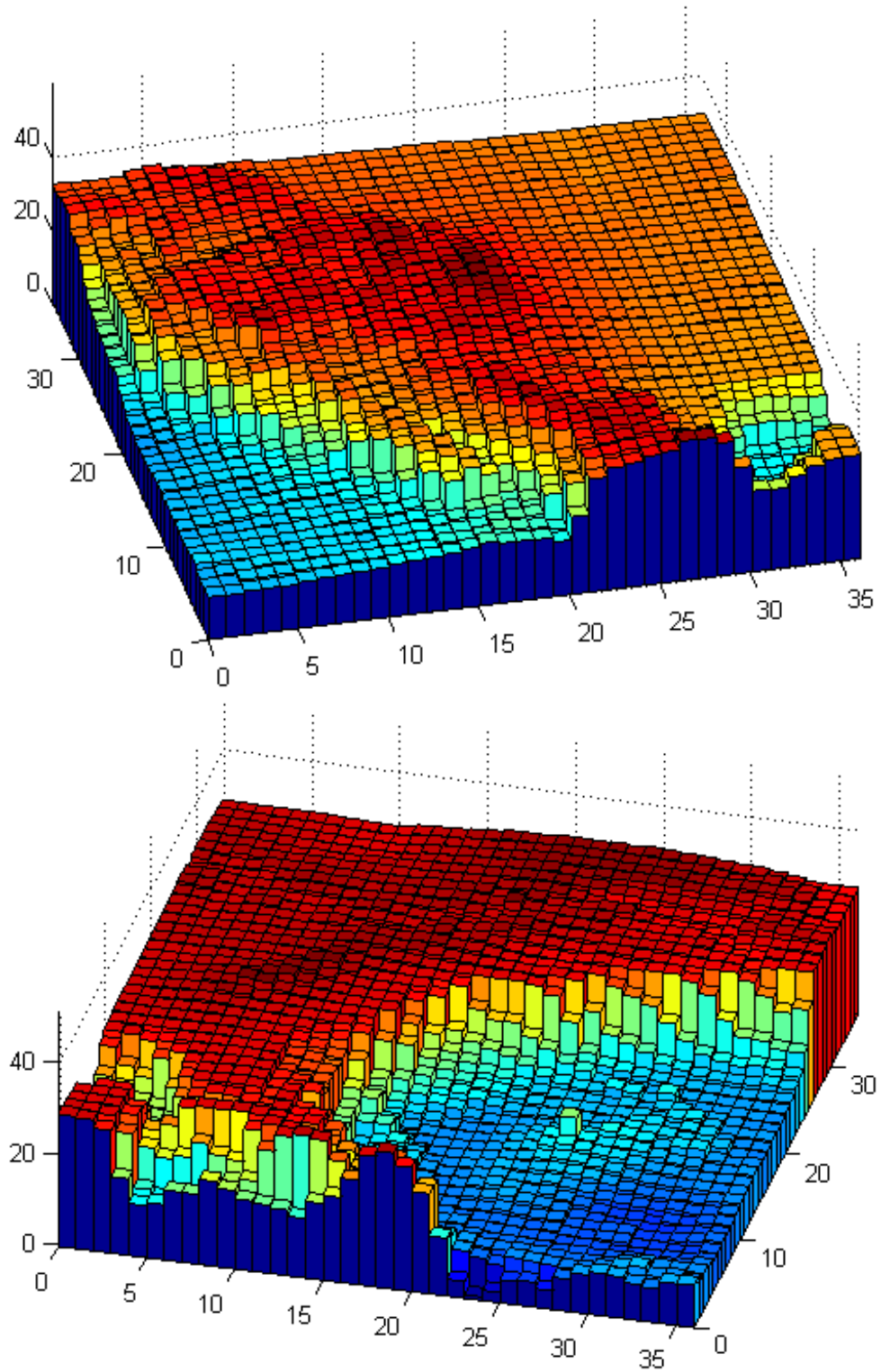
Figure 99 The Topo 1 and Topo 2 reference commands were created by cropping the "World Map" surface in Figure 96 into the western (*top*) and eastern (*bottom*) halves of the United States.

APPENDIX C

RESPONSE OF THE RC CIRCUIT FOR EXAMPLE COMMANDS USING THE SVD SYSTEM

Step Command to:

$$V2_{des} = \begin{bmatrix} 3 & 3 & 3 & 2 \\ 3 & 4 & 3 & 2 \\ 3 & 3 & 3 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}. \tag{236}$$
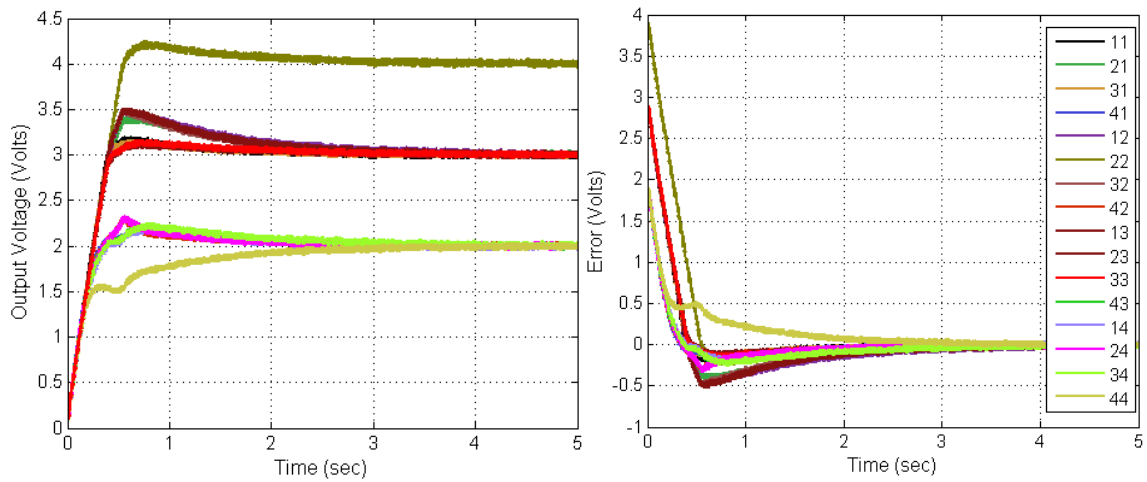
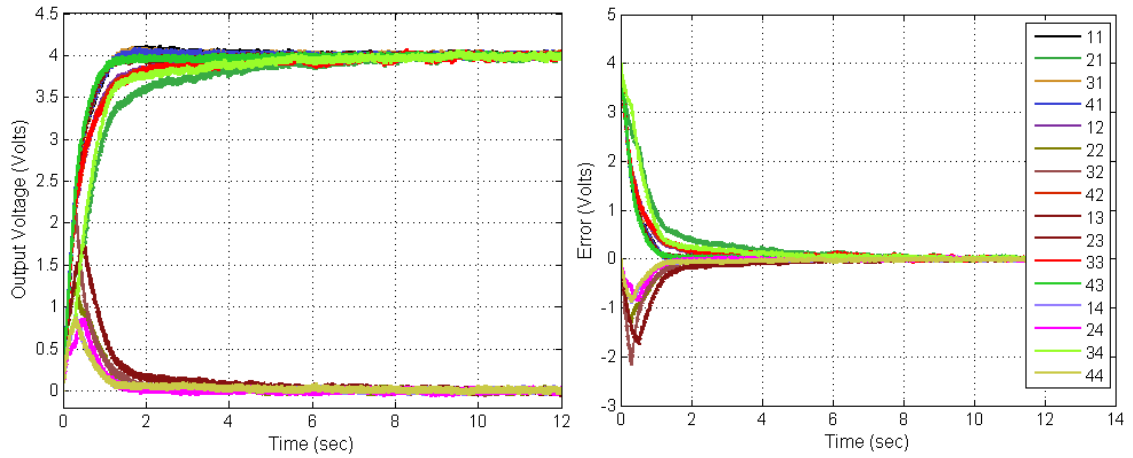**Figure 100** Response of the SVD System (*left*) and the error (*right*) for a step input to $V2_{des}$.
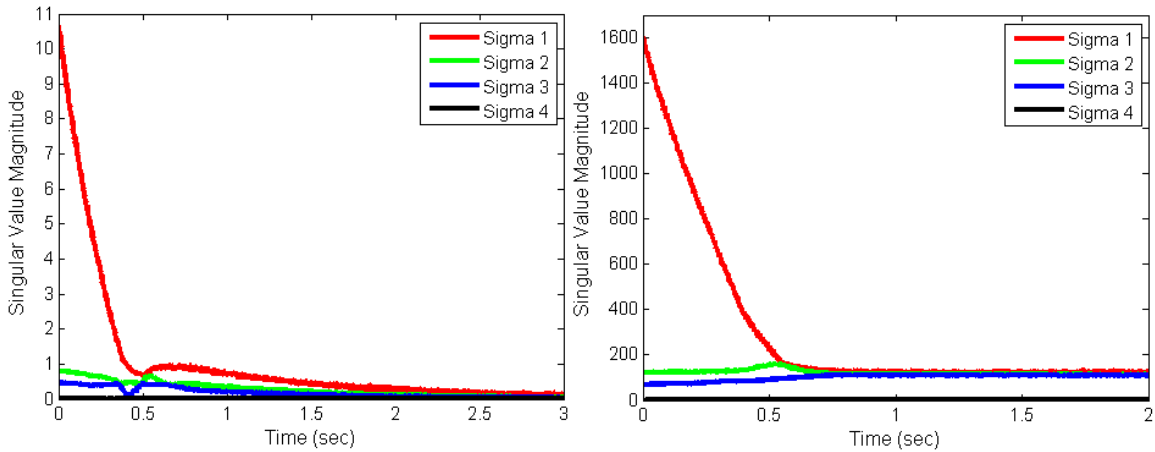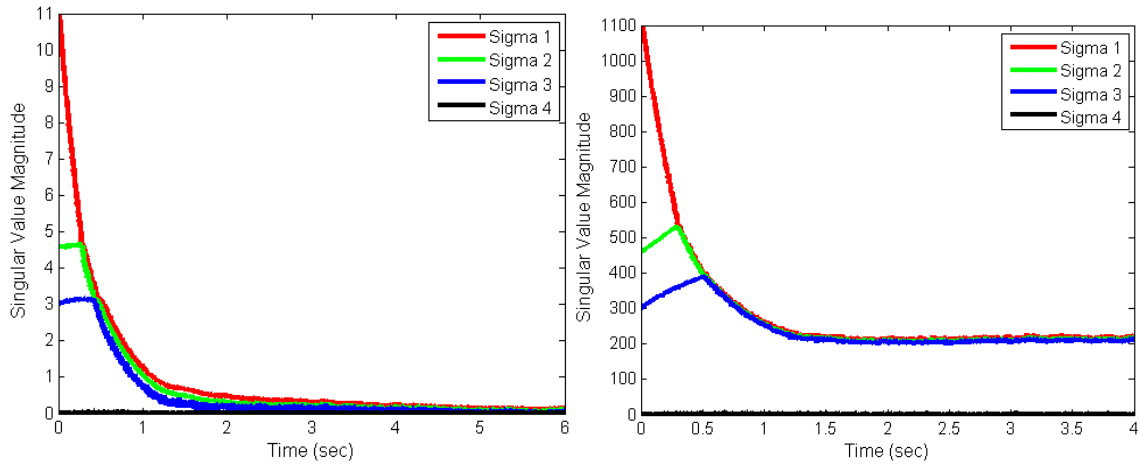

Figure 101 Singular values of the error (*left*) and control input (*right*) for a step input to $V2_{des}$.

Step Command to:

$$V3_{des} = \begin{bmatrix} 3.2 & 3.8 & 4 & 4 \\ 3.5 & 2.6 & 1.4 & 2 \\ 3.7 & 2.5 & -3 & -3 \\ 4 & 2 & -3 & -4 \end{bmatrix}. \qquad (237)$$
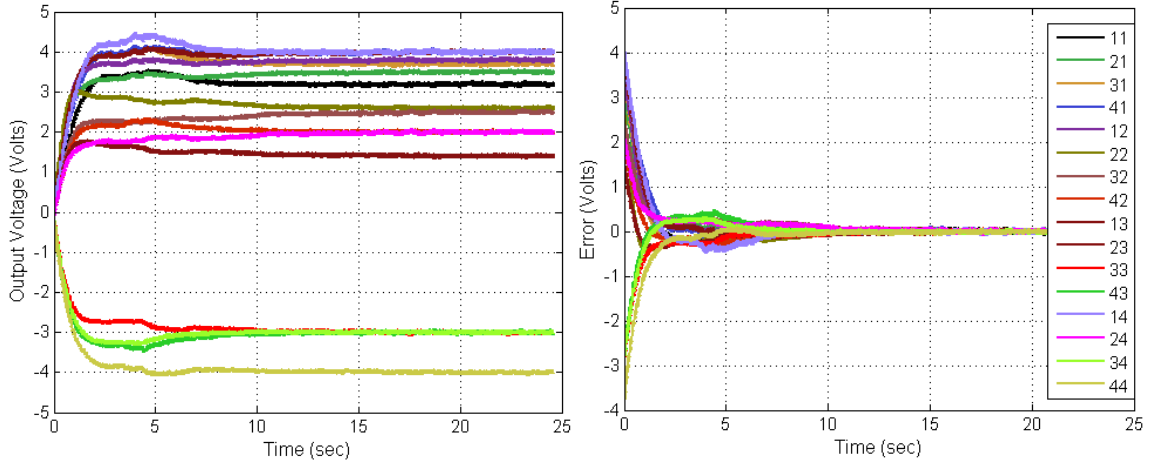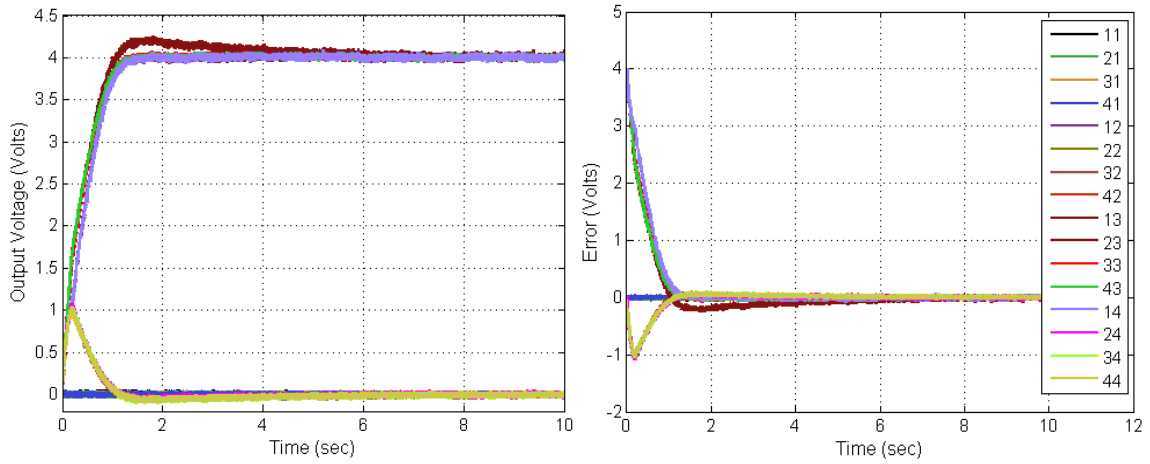

Figure 102 Response of the SVD System (*left*) and the error (*right*) for a step input to $V3_{des}$.
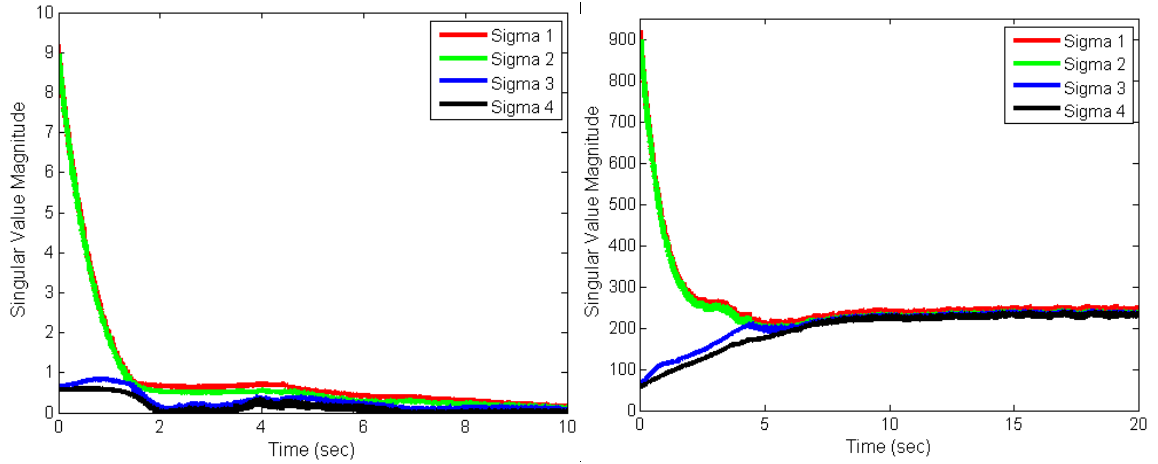
Figure 103 Singular values of the error (*left*) and control input (*right*) for a step input to $V3_{des}$.

Step Command to:

$$V4_{des} = \begin{bmatrix} 0 & 4 & 4 & 0 \\ 4 & 0 & 0 & 0 \\ 4 & 0 & 4 & 4 \\ 0 & 4 & 4 & 0 \end{bmatrix}. \tag{238}$$

Figure 104 Response of the SVD System (*left*) and the error (*right*) for a step input to $V4_{des}$.

Figure 105 Singular values of the error (*left*) and control input (*right*) for a step input to $V4_{des}$.

Step Command to:

$$V5_{des} = \begin{bmatrix} 0 & 4 & 4 & 4 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix}. \tag{239}$$

Figure 106 Response of the SVD System (*left*) and the error (*right*) for a step input to $V5_{des}$.
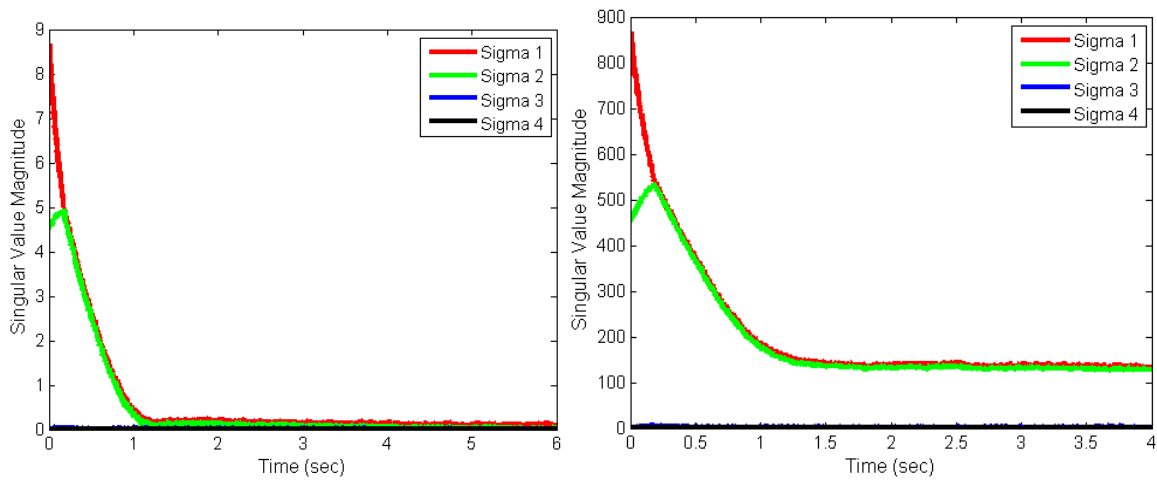

Figure 107 Singular values of the error (*left*) and control input (*right*) for a step input to $V5_{des}$.

179

APPENDIX D

RESPONSE OF THE RC CIRCUIT FOR EXAMPLE COMMANDS USING THE SNMF SYSTEM

Step Command to:

$$V2_{des} = \begin{bmatrix} 1.6870 & 2.6230 & 2.7149 & 2.6219 \\ 3.6629 & 0.1428 & 3.0310 & 0.6847 \\ 3.1688 & 3.3965 & 2.9725 & 2.8242 \\ 3.8380 & 3.7360 & 1.5689 & 0.1273 \end{bmatrix}. \tag{240}$$
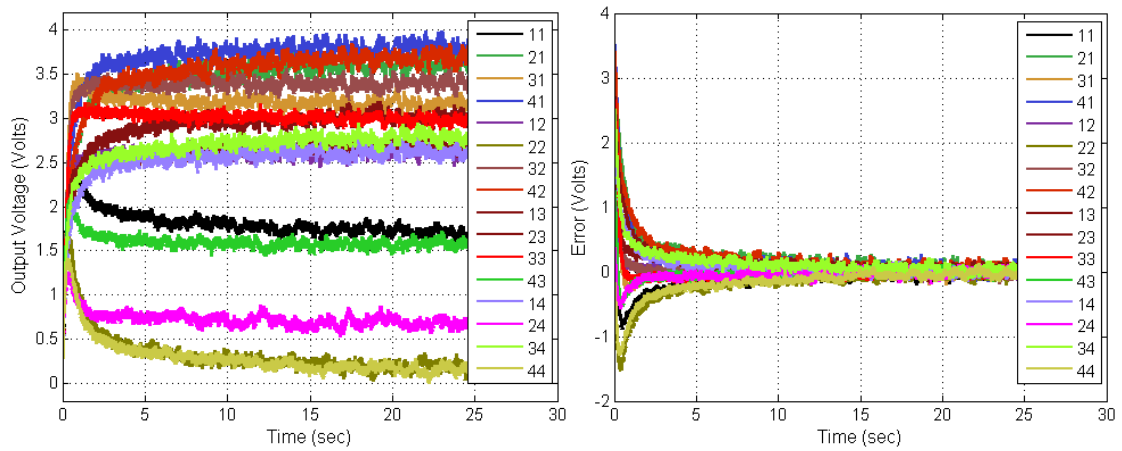

Figure 108 Response of the SNMF System (*left*) and the error (*right*) for a step input to $V1_{des}$.
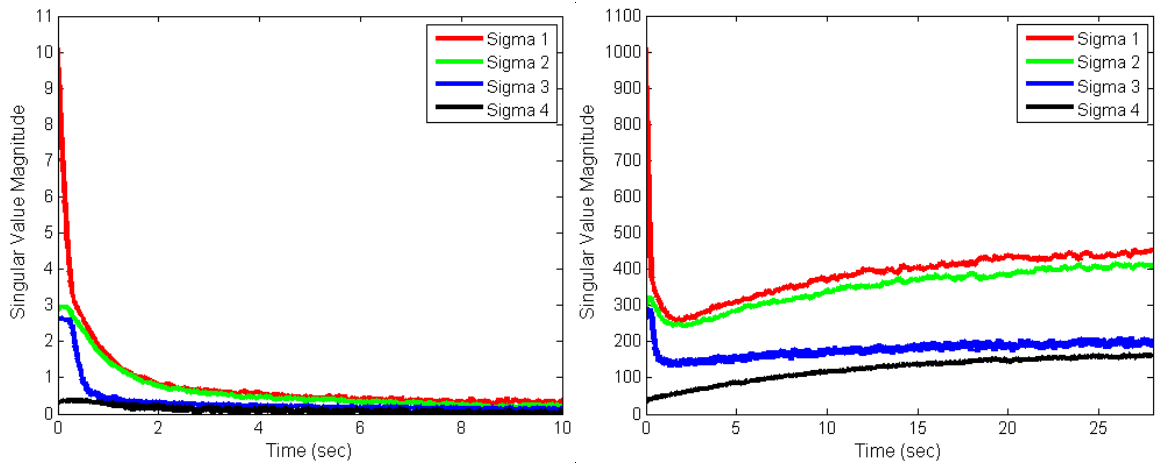

Figure 109 Singular values of the error (*left*) and control input (*right*) for a step input to $V1_{des}$.

Step Command to:

$$V2_{des} = \begin{bmatrix} 3 & 3 & 3 & 2 \\ 3 & 4 & 3 & 2 \\ 3 & 3 & 3 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}. \tag{241}$$
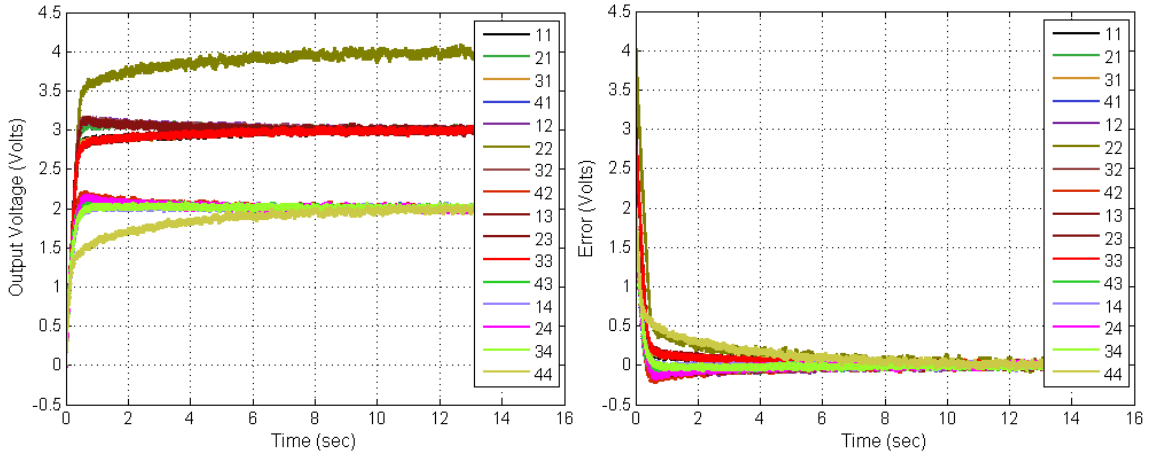

Figure 110 Response of the SNMF System (*left*) and the error (*right*) for a step input to $V2_{des}$.
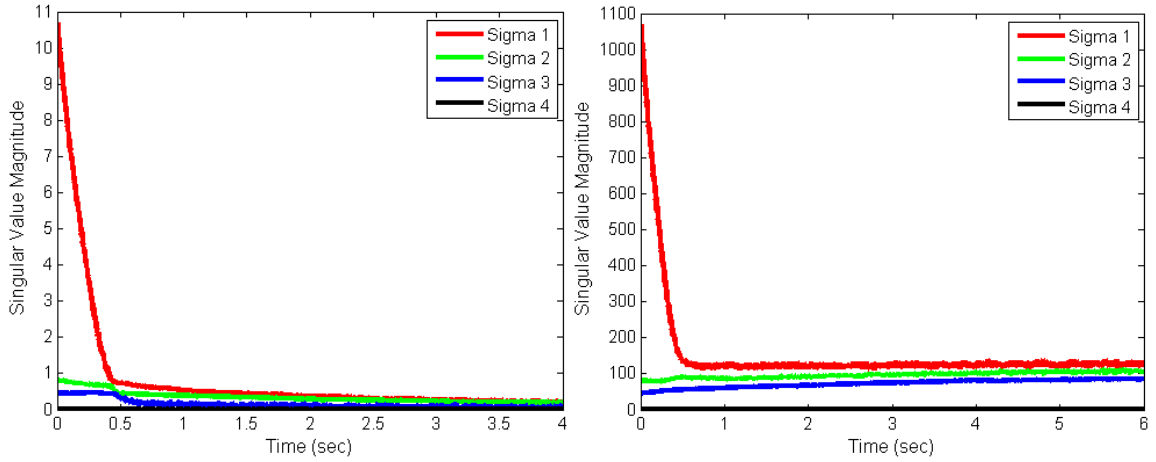

Figure 111 Singular values of the error (*left*) and control input (*right*) for a step input to $V2_{des}$.

181

Step Command to:

$$V3_{des} = \begin{bmatrix} 3.2 & 3.8 & 4 & 4 \\ 3.5 & 2.6 & 1.4 & 2 \\ 3.7 & 2.5 & -3 & -3 \\ 4 & 2 & -3 & -4 \end{bmatrix}. \tag{242}$$
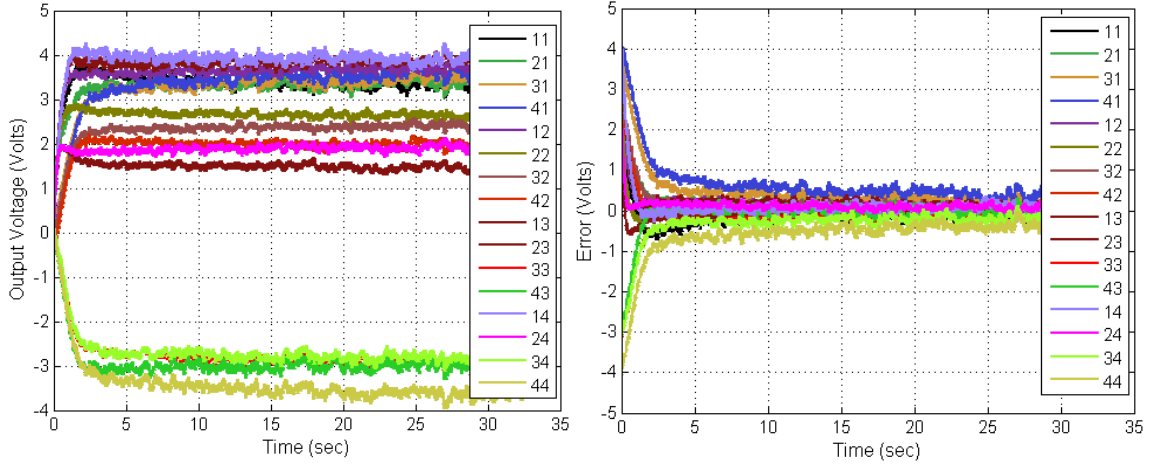

Figure 112 Response of the SNMF System (*left*) and the error (*right*) for a step input to $V3_{des}$.

Figure 113 Singular values of the error (*left*) and control input (*right*) for a step input to $V3_{des}$.

Step Command to:

$$V4_{des} = \begin{bmatrix} 0 & 4 & 4 & 0 \\ 4 & 0 & 0 & 0 \\ 4 & 0 & 4 & 4 \\ 0 & 4 & 4 & 0 \end{bmatrix}. \tag{243}$$

Figure 114 Response of the SNMF System (*left*) and the error (*right*) for a step input to $V4_{des}$.
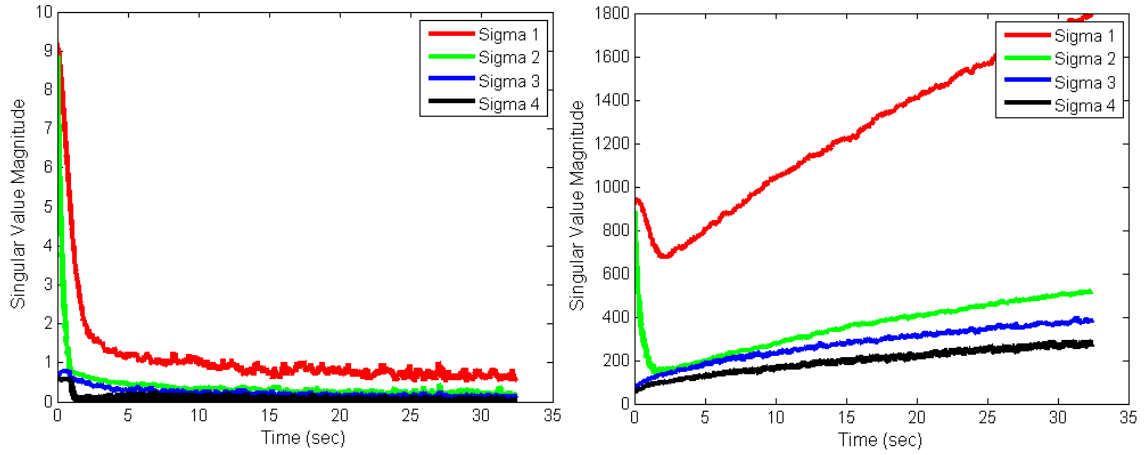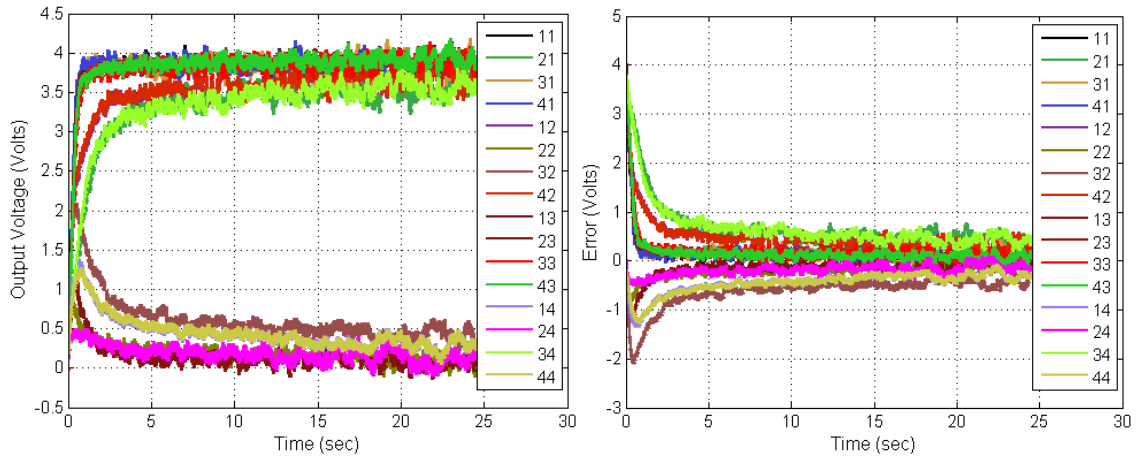

Figure 115 Singular values of the error (*left*) and control input (*right*) for a step input to $V4_{des}$.

Step Command to:

$$V5_{des} = \begin{bmatrix} 0 & 4 & 4 & 4 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix}. \tag{244}$$

Figure 116 Response of the SNMF System (*left*) and the error (*right*) for a step input to $V5_{des}$.
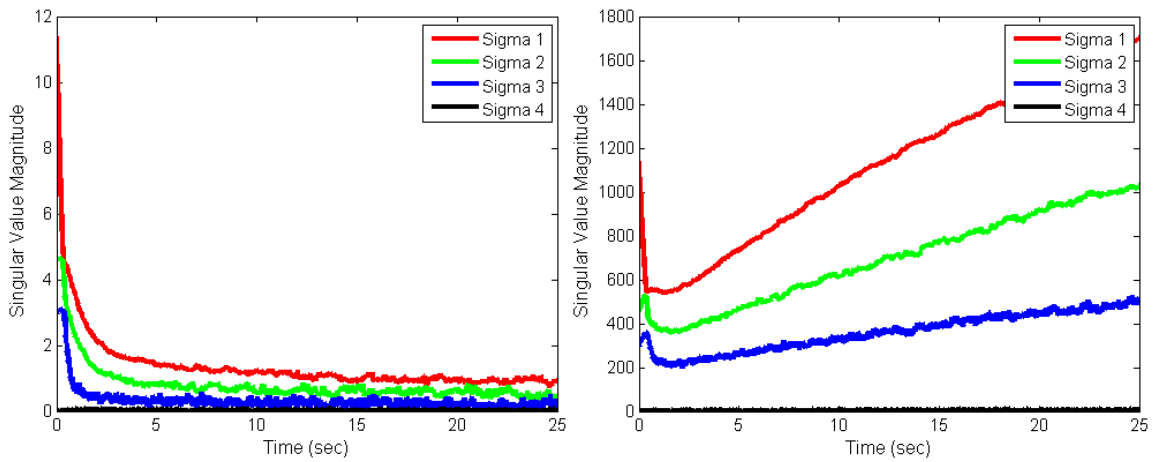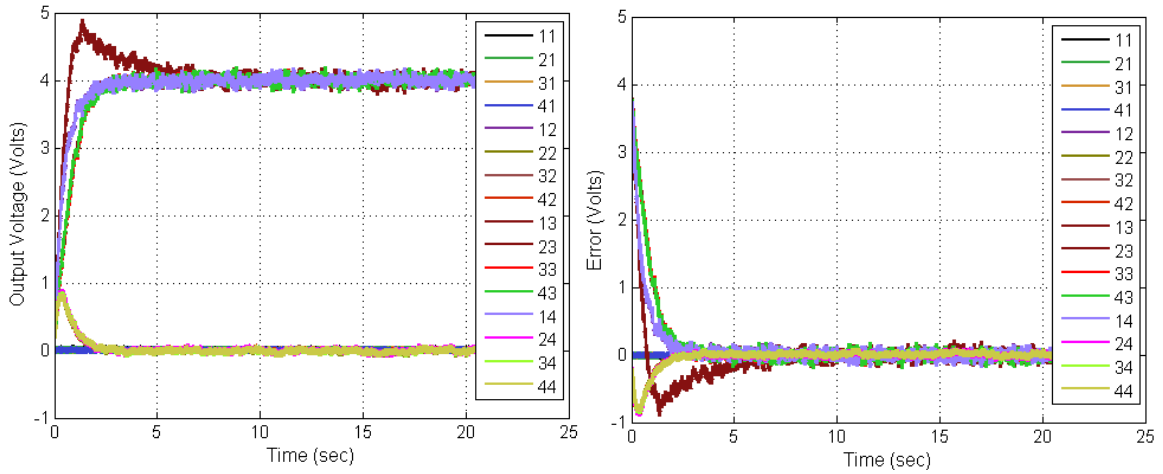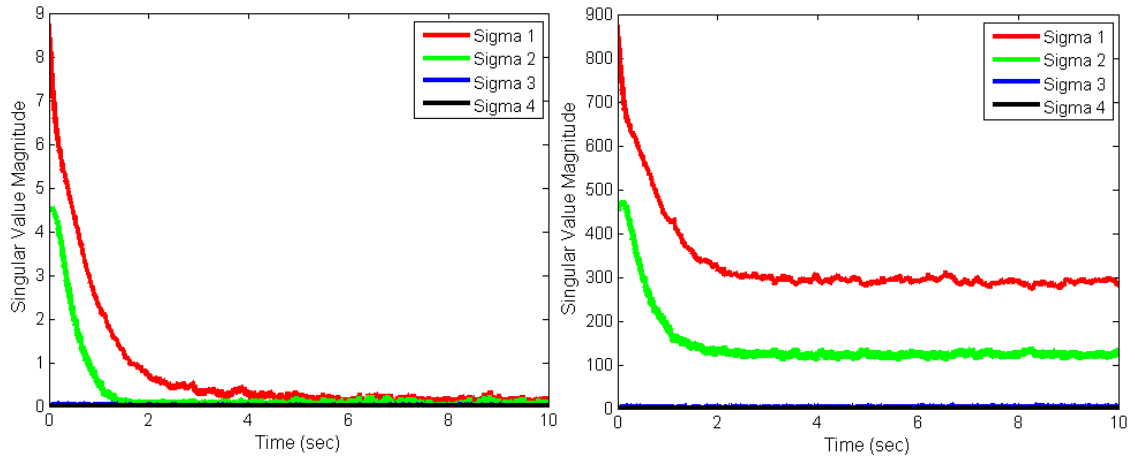

Figure 117 Singular values of the error (*left*) and control input (*right*) for a step input to $V5_{des}$.

184

REFERENCES

[1]   J. Fan, G.E. Stewart, and G.A. Dumont, "Two-dimensional frequency analysis for unconstrained model predictive control of cross-directional processes," *Automatica*, 40, 1981-1903, 2004.

[2]   M. Fardad, and M.R. Jovanovic, "Design of optimal controllers for spatially invariant systems with finite communication speed," *Automatica*, 47, pp. 880-889, 2011.

[3]   P. Massioni, and M. Verhaegen, "Subspace identification of circulent systems." *Automatica*, 44, pp. 2825-2833, 2008.

[4]   T.D. Murphy, "On multiple model control for multiple contact systems," *Automatica*, 44, pp. 451-458, 2008.

[5]   R.M. Murray, "Recent research in cooperative control of multi-vehicle systems," *ASME Journal of Dynamic Systems Measurement and Control*, 129, pp. 571-583, 2007.

[6]   F. Pasqualetti, R. Carli, and F. Bullo, "Distributed estimation via iterative projections with application to power monitoring," *Automatica*, 48, pp. 747-758, 2012.

[7]   K.J. Cho, S. Au, and H. Asada, "Large-scale servo control using a matrix wire network for driving a large number of actuators," *IEEE Int. Conf. on Robotics and Automation*, 1, pp. 646-651, 2003.

[8]   J. Baillieul, "Feedback designs for controlling device arrays with communication channel bandwidth constraints," *ARO Workshop on Smart Structures*, 1999.

[9]   H. Zhu, and W.J. Book, "Construction and Control of Massive Hydraulic Miniature Actuator-Sensor Array," IEEE Conf. on Computer Aided Control Systems Design, pp. 820-825, 2006.

[10]   J. Baillieul, and P.J. Antsaklis, "Control and Communication Challenges in Networked Real-Time Systems," *Proceedings of the IEEE*, 95(1), pp. 9-28, 2007.

[11]   J.K. Yook, D.M. Tilbury, and N.R. Soparkar, "Trading computation for bandwidth: reducing communication in distributed control systems using state estimators," *IEEE Trans. on Control Systems Technology*, 10(4), pp. 503-518, 2002.

[12]   D.J. Cristaldi, S. Pennisi, and F. Pulvirenti, *Liquid Crystal Display Drivers: Techniques and Circuits*, Springer, 2009.

[13] T. Keviczky, F. Borrelli, and G.J. Balas, "Decentralized receding horizon control for large scale dynamically decoupled systems," *Automatica*, 42, pp. 2105-2115, 2006.

[14] P. Massioni, and M. Verhaegen, "Subspace identification of circulent systems," *Automatica*, 44, pp. 2825-2833, 2008.

[15] G.H. Yang, and S.Y. Zhang, "Structural Properties of large-scale systems possessing similar structures," *Automatica*, 31(7), pp. 1011-1017, 1995.

[16] M. Farina, and R. Scattolini, "Distributed predictive control: a non-cooperative algorithm with neighbor-to-neighbor communication for linear systems," *Automatica*, 48, pp. 1088-1096, 2012.

[17] L. Flemming, and S. Mascaro, "Control of scalable wet SMA actuator arrays," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1350-1355, April, 2005.

[18] R. Mukherjee, T.F. Christian, and R.A. Thiel, "An actuation system for the control of multiple shape memory alloy actuators," *Sensors and Actuators*, 55, pp. 185-192, 1996.

[19] M. Nakatani, H. Kajimoto, K. Vlack, D. Sekiguchi, N. Kawakami, and S. Tachi, "Control method for a 3-D form display with coil-type shape memory alloy," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1332-1337, January, 2005.

[20] H. Zhu, "Practical structural design and control for Digital Clay," Ph.D. Thesis, School of Mech. Eng., Georgia Inst. of Tech., Atlanta, GA, 2005.

[21] R. Velazquez, and E. Pissaloux, "Design and optimization of crossbar architecture for shape memory alloy actuator arrays," *IEEE International Symposium on Micro-NanoMechatronics and Human Science*, pp. 1-5, February, 2006.

[22] K.J. Cho, B. Roy, S. Mascaro, and H. Asada, "A Vast DOF Robotic Car Seat using SMA Actuators with a Matrix Drive System," *IEEE International Conference on Robotics and Automation*, 4, pp. 3647-3652, July, 2004.

[23] Watkins, D. S., *Fundamentals of Matrix Computations*, 2nd ed., John Wiley & Sons, New York, Chap. 2 and 4, pp. 115-118 and 266-271, 2002.

[24] Eckart, C., and Young, G., "The approximation of one matrix by another of lower rank," *Psychometrika*, 1, pp. 211-218, 1936.

[25] Moore, B., 1981. "Principal component analysis in linear systems: Controllability, observability, and model reduction." *IEEE Transactions on Automatic Control*, 26(1), February, pp. 17-32.

[26] Moore, C., "Application of Singular Value Decomposition to the Design, Analysis, and Control of Industrial Processes," *IEEE American Control Conference*, pp. 643-650, June, 1986.

[27] Skogestad, S., and Morari, M., "Robust Control of Ill-Conditioned Plants: High-Purity Distillation," *IEEE Transactions on Automatic Control*, 33(12), pp. 1092-1105, December, 1988.

[28] Zou, K., and Doyle, J.C., *Essentials of Robust Control*, Prentice Hall, New Jersey, 1998.

[29] Skogestad, S., and Postlethwaite, I., *Multivariable Feedback Control: Analysis and Design*, John Wiley and Sons Ltd., New York, Chap. 3, pp. 72-91, 1996.

[30] Lau, H., Alvarez, J., and Jensen, K.F., "Synthesis of Control Structures by Singular Value Analysis: Dynamic Measures of Sensitivity and Interaction," *AIChE Journal*, 31(3), pp. 427-439, March, 1985.

[31] Hovd, M., Braatz, R.D., and Skogestad, S., "SVD Controllers for H_2-, H_∞-, and μ-Optimal Control," *Proceedings of the American Control Conference*, pp. 1233-1237, June, 1994.

[32] Brambilla, A., and D'Elia, L., "Multivariable Controller for Distillation Columns in the Presence of Strong Directionality and Model Errors," *Industrial and Engineering Chemistry Research*, 31(2), pp. 536-543, February, 1992.

[33] Anthonis, J., and Ramon, H., "Linear mechanical systems and dyadic transfer function matrices." *Automatica*, 39, pp. 1353-1363, March, 2003.

[34] H. Park and H. Kim, "One-Sided Non-Negative Matrix Factorization and Non-Negative Centroid Dimension Reduction for Text Classification," *Proceedings of the 2006 Text Mining Workshop in the Tenth SIAM International Conference on Data Mining*, 2006.

[35] C. Ding, T. Li, and M. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[36] H. Park, and H. Kim, "One-sided non-negative matrix factorization and non-negative centroid dimension reduction for text classification," *Text Mining Workshop in the Tenth SIAM International Conference on Data Mining*, 2006.

[37] J. Kim and H. Park, "Fast nonnegative matrix factorization: an active-set-like method and comparisons", *SIAM Journal on Scientific Computing*, 33 (6), pages 3261-3281, 2011.

[38] D. D. Lee, and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, pp. 788-791, 1999.

[39] K. Devarajan, "Nonnegative matrix factorization: an analytical and interpretive tool in computational biology," PLoS Computational Biology, Public Library of Science, 4, e1000029, 2008,

[40] A. Cichocki, R. Zdunek, A. H. Phan, and S. I. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*, Wiley, 2009.

[41] H. Kim and H. Park, "Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method", *SIAM Journal on Matrix Analysis and Applications*, 30, pp. 713-730, 2008.

[42] K. J. Cho, J. Rosmarin, and H. Asada, "Design of vast DOF artificial muscle actuators with a cellular array structure and its application to a five-fingered robotic hand," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2214-2219, 2006.

[43] H.K. Khalil, *Nonlinear Systems*, 3rd ed., New Jersey, Prentice Hall, 2002.

[44] C. S. Ngoo, "Admittance and impedance haptic control for realization of Digital Clay as an effective human machine interface (HMI) device," M.S. Thesis, School of Mech. Eng., Georgia Inst. of Tech., Atlanta, GA, 2009.

[45] G. Strang, *Linear Algebra and its Applications*, 4th ed., California, Thomson Brooks/Cole, 2006.

[46] G.H. Golub, and O. Reinsch, "Singular Value Decomposition and Least Squares Solutions," *Numerische Mathematik*, 14, pp. 403-420, 1970.

[47] J.C. Nash, "A one-sided transformation method for the singular value decomposition and algebraic eigenproblem," *The Computer Journal*, 18(1), pp. 74-76, 1975.

[48] A.A. Maciejewski, and C.A. Klein, "The Singular Value Decomposition: Computation and Application to Robotics," *International Journal of Robotics Research*, 8(6), pp. 63-79, 1989.

[49] J.C. Nash, and S. Shlien, "Simple Algorithms for the Partial Singular Value Decomposition," *The Computer Journal*, 30(3), pp. 268-275, 1987.