ONLINE REGULATION OF LOW ORDER SYSTEMS UNDER BOUNDED CONTROL

A Thesis

by

SUMIT ARORA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2003

Major Subject: Mechanical Engineering

ONLINE REGULATION OF LOW ORDER SYSTEMS UNDER BOUNDED CONTROL

A Thesis

by

SUMIT ARORA

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Approved as to style and content by:

Suhada Jayasuriya
(Chair of Committee)

Reza Langari
(Member)

Dennis L. O'Neal
(Head of Department)

Jay R. Walton
(Member)

December 2003

Major Subject: Mechanical Engineering

ABSTRACT

Online Regulation of Low Order Systems under Bounded Control. (December 2003)

Sumit Arora,

B.E., Delhi College of Engineering, India

Chair of Advisory Committee: Dr. Suhada Jayasuriya

Time-optimal solutions provide us with the fastest means to regulate a system in presence of input constraints. This advantage of time-optimal control solutions is offset by the fact that their real-time implementation involves computationally intensive iterative techniques. Moreover, time-optimal controls depend on the initial state and have to be recalculated for even the slightest perturbation. Clearly time-optimal controls are not good candidates for online regulation. Consequently, the search for alternatives to time-optimal solutions is a very active area of research.

The work described here is inspired by the simplicity of optimal-aim concept. The "optimal-aim strategies" provide online regulation in presence of bounded inputs with minimal computational effort. These are based purely on state-space geometry of the plant and are inherently adaptive in nature. Optimal-aim techniques involve aiming of trajectory derivative (or the state velocity vector) so as to approach the equilibrium state in the best possible manner.

This thesis documents the efforts to develop an online regulation algorithm for systems with input constraints. Through a number of hypotheses focussed on trying to reproduce the exact time-optimal solution, the difficulty associated with this task is demonstrated.

A modification of optimal-aim concept is employed to develop a novel regulation

algorithm. In this algorithm, aim directions are chosen in a special manner to generate the time-optimal control *approximately*. The control scheme thus developed is shown to be globally stabilizing for systems having eigenvalues in the CLHP (closed left half-plane). It is expected that this method or its modifications can be extended to higher dimensional systems as a part of future research. An alternative control algorithm involving a simple state-space aiming concept is also developed and discussed.

To my loving parents and the late Dr. Sherif Noah.

ACKNOWLEDGMENTS

I wish to express my gratitude to Dr. Suhada Jayasuriya for his advice, guidance and support. I sincerely appreciate his time and effort through the duration of my thesis project.

I wish to thank Dr. Reza Langari and Dr. Jay R. Walton for their counsel and support.

I would also like to thank the talented bunch of colleagues in Mechanical Engineering Controls & Robotics Lab (TAMU) for their suggestions.

TABLE OF CONTENTS

CHAPTER Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

We are dependent on numerous dynamic systems in our daily lives. We encounter them everywhere, be it in the form of automobiles, aeroplanes, refrigerators or air conditioners. As we grow more dependent on these modern conveniences, we expect higher standards of performance and reliability. The increased expectations translate into more stringent performance and quality specifications for the engineers who design, build and/or operate these systems. This in turn means that all components of these systems have to perform in the expected manner.

Control systems are required to ensure satisfactory performance from dynamic systems. "The successful operation of a system under changing conditions often requires a control system" [1]. We can observe control systems in action all around us. For example, a car in automatic cruise control mode is an example of a control system that maintains a constant speed. When cruise control mode is engaged, a control system takes over the operation of the automobile. It gathers data from its sensors about speed, steering wheel position etc. and compares it against the desired set point speed. Depending on the result of this comparison, control system decides whether to increase or decrease the amount of fuel entering the engine. The control system may also decide to quit cruise control mode altogether in cases brakes are applied suddenly or the steering wheel is rotated by an appreciable amount. Figure 1 depicts the schematic of a typical control system with the dynamical system (or plant) being controlled and the controller. Control systems may be utilized to satisfy a wide variety of requirements. Common control system design specifications could

---

The journal model is *IEEE Transactions on Automatic Control.*

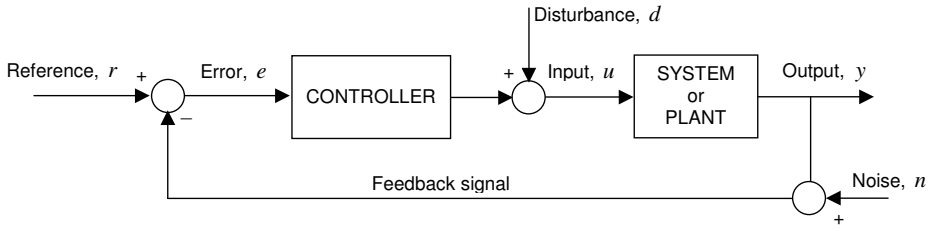

Fig. 1. A typical feedback control system

be a combination of or any of the following:

1. system stabilization,
2. negate the effect of noise and disturbances in the system output,
3. ensure satisfactory operation by preventing large rise times and overshoots,
4. maintain output variable at a desired reference value (regulation) or force output variable to follow a preset path (reference tracking),
5. and, adapt itself and the system, to unpredictable changes in environment or the system structure.

Reference tracking is one of the primary objectives of control systems. Consider the system shown in Figure 1. Say, both output and reference signal are functions of time denoted by $y(t)$ and $r(t)$ respectively. If the controller causes the output to follow reference signal or

$$y(t) = r(t),$$

then the controller is said to be performing reference tracking. If on the other hand, the controller drives the output to an equilibrium state σ and maintains it there

thereafter, reference tracking will imply

$$\lim_{t\to\infty} y(t) = \sigma.$$

This is more commonly known as regulation. Tracking or regulation is performed via appropriate control inputs. The controller receives information about the error e and utilizes it to calculate the control input u.

Control systems incorporate actuators to apply the computed input to the system. Often control effort, as calculated by control algorithm, may exceed the capacity of actuator(s). In such cases the actuators operate at their limits. It results in limited control action on the part of control system and this situation is known as actuator saturation. Actuator bandwidth limitations and the onset of nonlinearities are other reasons for constrained inputs (or bounded controls) apart from actuator saturation.[1]

A. Effects of actuator saturation

Bounded controls can lead to undesirable effects on control system performance and stability. Some of them are:

- Actuator saturation may destabilize control systems through a decrease in feedback gain.
- Severe performance degradation may occur when integrators are used in a linear compensator. After actuator saturation occurs the error signal is continuously integrated, by the integrators present in the loop, leading to *reset-windup*. This drives up the control effort requirements from the compensator even though instantaneous error is small, leading to serious problems like large overshoots

[1]The terms "actuator saturation", "constrained inputs" and "bounded controls" are used interchangeably in the same context in this work.

and very long settling times [2].

- Another serious problem with actuator saturation in a multi-variable relates to control directions. Actuator saturation may alter the direction of the control input vector. Consider a system with two actuators corresponding to two inputs $u_1 \in [-1, 1]$ and $u_2 \in [-1, 1]$. In such a case the maximum input from either actuator is $+1$ and minimum input is -1, so any inputs with magnitudes outside this range will cause actuator saturation. Suppose the control signal at any time is $u = \{2.5, 1.2\}$ then the saturated signal will be $u_s = \{1, 1\}$, which causes the control input direction to be altered. Figure 2 shows the effect of saturation on different control vectors in the two-dimensional control space [2]. Area enclosed

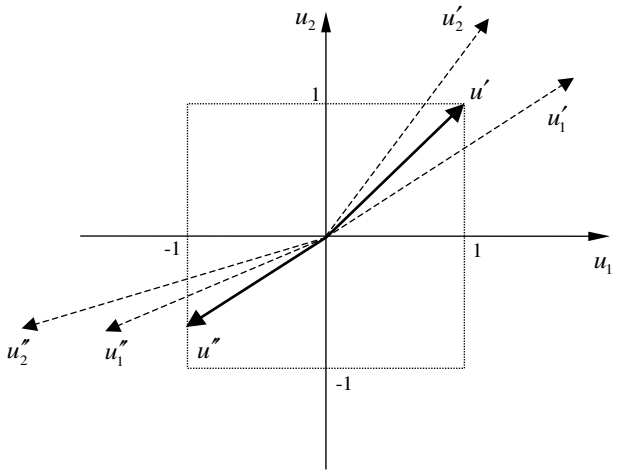

Fig. 2. Effect of actuator saturation on control directions

by dotted lines in Figure 2 depicts the operating region for the actuator discussed above. Control vectors u_1' and u_2' both have magnitudes greater than 1 in both u_1 and u_2 directions, so both are mapped to the vector $u' = [1, 1]$, whereas u_1'' and u_2'' have magnitude greater than 1 only in the u_1 direction, they are

both mapped to the vector u'' with their magnitude in u_2 direction remaining unchanged.

So, actuator saturation is generally not desirable in control system design practice. But, it is not always easy or cost effective to find actuators with high enough operating range and bandwidth so as to avoid saturation. As a result, the hunt for better methods to deal with bounded controls continues to be an area of active research.

B. Dealing with bounded controls

An appreciable amount of research in the last few decades has sought to characterize, formulate and find solutions to the bounded control problem for different classes of dynamical systems. Some well-known facts about systems with constrained inputs can be summarized as:

- Only linear stabilizable systems with having no open-loop unstable modes can be globally asymptotically stabilized by a bounded control [3].
- In general, nonlinear control laws are required to stabilize linear systems subject to input saturation [4],[5].
- If the constraint set contains the origin in its interior and a stabilizing control exists, there also exists a domain of attraction to the origin where no constraints violations occur [6].

In designing control systems with actuator saturation, following broad approaches have been followed [7],[8]:

- The first approach focuses on preventing adverse effects of actuator saturation on the system. This leads to a two-step design methodology. First step involves

synthesizing controllers, without taking saturation into account, to satisfy the control requirements, while the second step involves some sort of *anti-windup* scheme. Bulk of the research in this area is conducted on open-loop stable systems. These methods allow actuator saturation and therefore introduce nonlinearity in the closed loop system.

- The second approach is more concerned with stability issues in presence of control signal constraints. The goal is to design controllers that are either globally stabilizing, or locally stabilizing *for a known region*. All null-controllable linear systems can be globally stabilized using smooth constrained control laws. For systems that are not null-controllable, or for systems having non-globally stabilizing control laws, knowledge of closed-loop stability regions for the control law is of paramount importance.

- Often control systems are designed that have actuator saturations incorporated *a priori* into control design. Such techniques generally allow actuators to saturate and may involve switching or bang-bang controllers.

Asymptotically null-controllable systems constitute an important class of systems. As stated above, these systems can be globally stabilized using bounded inputs and smooth control laws. A system

$$\dot{x} = Ax + Bu \tag{1.1}$$

is said to be asymptotically null-controllable if it satisfies the following two conditions [9]:

1. the matrix A has no eigenvalues with positive real parts, and

2. the pair (A, B) is stabilizable or all uncontrollable modes of the system described

by Equation (1.1) have strictly negative parts.

In general the 'null-controllable region' of a system (also known as the 'controllable set' or the 'reachable set') is defined to be the set of all states that can be steered to the origin in a time $T \in (0, \infty)$. For systems with constrained inputs that are not asymptotically null-controllable, the knowledge of null-controllable regions is a great aid in determining the appropriate regulation scheme.

This thesis documents the development of an algorithm that can perform regulation in linear systems with input constraints and lends itself easily to online implementation. A new control scheme distinct from existing techniques is proposed. It involves a modification of the "optimal-aim strategy" [10] and approximates time-optimal control solution [11] for regulation.

C. Thesis organization

The layout of this thesis is as follows. Chapter II describes in brief the Optimal-aim concept, the concept of normal systems and Pontryagin's minimum principle based time-optimal control problem and solution thereof. Later part of chapter II explains in detail the motivation for this research. Chapter III describes the efforts to reproduce the time-optimal control solution in real-time. The efforts leading up to the development of the algorithm in final form, associated results and proofs and some other related formulations are described in chapter IV . Results and related analysis are presented in chapter V. The final chapter lists the conclusions derived from the work and suggestions for future research.

CHAPTER II

BACKGROUND

The bounded control problem has been tackled in a number of different ways by different researchers. Of these, two techniques viz.

1. Optimal control solutions (from Pontryagin's minimum principle), and
2. Optimal-aim strategies

were investigated in detail. Optimal control theory provides us with the minimum time control for regulation in systems with actuator saturation. The optimal-aim concept provides us with a regulation method that can be easily implemented in real-time. It was postulated that minimum time control (or time-optimal control) may be generated from optimal-aim technique by selecting the aim directions in some special way. These techniques are described in the following sections.

A. Time-optimal control

In the presence of constrained input signals, time-optimal control solutions based on Pontryagin's minimum principle provide the fastest regulation. Time-optimal control solutions are determined by solving a "Two-point Boundary Value Problem" (TPBVP). A general time-optimal regulator problem for LTI continuous systems and its solution are described next [11].

Given the system

$$\dot{x}(t) = g(x, u, t) = f(x(t)) + Bu(t), \tag{2.1}$$

where $x \in \mathbb{R}^n$, $u(t) \in \Omega = \{\omega \in \mathbb{R}^m | \omega_j \in [\omega_{j\min}, \omega_{j\max}]\}$ and $j = 1, \cdots, m$. Assume that the system given by Equation (2.1) is completely controllable. Then, given that

at some initial time t_0, the initial state of the system is $x(t_0) = x_0$. Find the control u^* that transfers the system in Equation (2.1) from x_0 to the origin in minimum time.

To determine the solution to the above problem using Pontryagin's minimum principle we consider a cost functional,

$$J = \int_{t_0}^{t_f} L(x, u, t)\mathrm{d}t. \tag{2.2}$$

All the states lying on a time-optimal trajectory and the corresponding input sequence necessarily minimize the cost function as defined in Equation (2.2). For the time-optimal problem the integrand $L(x, u, t) = 1$, since the total time, $(t_f - t_0)$ is to be minimized. The Hamiltonian for the problem is

$$H(x, u, t) = L(x, u, t) + \Big\langle p(t), g(x, u, t) \Big\rangle = 1 + \Big\langle p(t), \dot{x}(t) \Big\rangle. \tag{2.3}$$

The variable $p \in \mathbb{R}^n$ is the *Lagrange multiplier* vector for this optimization problem, and is also known as the *costate vector*. Initial and final conditions are denoted by x_0 and $x_f(= 0)$. Following equations relate the state $x(t)$ and costate $p(t)$ vectors [11]:

$$\dot{x}(t) = \frac{\partial H(x(t), p(t), u(t), t)}{\partial p} \tag{2.4}$$

and

$$\dot{p}(t) = -\frac{\partial H(x(t), p(t), u(t), t)}{\partial x}. \tag{2.5}$$

Equation (2.4) is an alternative way of expressing Equation (2.1).

This formulation results in a Two-point Boundary Value Problem (TPBVP). Minimization of cost function J or the Hamiltonian H over a function space yields the time-optimal solution.

The next section describes 'normal systems'. Time-optimal solutions for this class of systems are much simpler than those for other systems.

1. Normal systems

Consider a system

$$\dot{x} = Ax + Bu,$$

where $x \in \Re^n$, $u \in \Re^m$. Also, say $B = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ b_1 & b_2 & \cdots & b_m \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$, where b_1, b_2 $\cdots$ etc. are column vectors of B [11].

Let us compute the controllability matrices G_1 through G_m defined as:

$$\begin{aligned} G_1 &= \begin{bmatrix} b_1 & Ab_1 & A^2b_1 & \cdots & A^{n-1}b_1 \end{bmatrix} \\ G_2 &= \begin{bmatrix} b_2 & Ab_2 & A^2b_2 & \cdots & A^{n-1}b_2 \end{bmatrix} \\ &\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ G_m &= \begin{bmatrix} b_m & Ab_m & A^2b_m & \cdots & A^{n-1}b_m \end{bmatrix}. \end{aligned} \tag{2.6}$$

Then, if all of these controllability matrices are nonsingular, the system is said to be normal. In other words, normality implies controllability with respect to each component of the control vector u.

Some general characteristics of time-optimal control solutions are [11]:

- Time-optimal trajectory depends on initial state.
- Time optimal control exists for all initial states in $\mathbb{R}^n$ when the origin is the final state (for systems with stable modes).
- For a given initial state x_0, the time-optimal control solution exists if and only if x_0 lies within the null-controllable region [12].
- It is bang-bang, unique (for normal systems) and has at most $(n-1)$ control

switchings.

- It minimizes the Hamiltonian. The optimal solutions give global minima, satisfying $\frac{\partial H}{\partial u} = 0$ at all points on the optimal trajectory.

- The time-optimal solution, in addition to satisfying $\frac{\partial H}{\partial u} = 0$, also satisfies $H = 0$ at all points on the trajectory.

As an illustrative example, the time-optimal control problem for a system having two time constants and no zeros, along with the solution is presented next.

2. Time-optimal control of a 'two time constant system'

This section illustrates a typical time-optimal control problem and its solution procedure. Consider the system [11]

$$\frac{y(s)}{u(s)} = G(s) = \frac{1}{(s-\lambda_1)(s-\lambda_2)}, \tag{2.7}$$

where $\lambda_1, \lambda_2 \in \Re$ and $\lambda_1 \neq \lambda_2 > 0$. This transfer function has two real poles. A state-space realization of this system is:

$$\dot{x} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} x + \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} u, \tag{2.8}$$

where $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$. The control input is constrained as $u \in [-1, 1]$. The goal is to transfer the above system from a given initial state at $t = 0$ to the origin in the shortest possible time.

Since the system has two stable modes and is normal, the time-optimal control exists, is unique and it minimizes the Hamiltonian. The Hamiltonian for the system

is

$$\begin{aligned} H(x,u,t) &= 1 + \langle p(t), \dot{x}(t) \rangle \\ &= 1 + \lambda_1 x_1(t) p_1(t) + \lambda_2 x_2(t) p_2(t) + u(t) \left\{ \lambda_1 p_1(t) + \lambda_2 p_2(t) \right\}. \end{aligned} \tag{2.9}$$

Only the input u can be varied freely in Equation (2.9), so the *H-minimal* control is given by

$$u(t) = -\text{sign} \left\{ \lambda_1 p_1(t) + \lambda_2 p_2(t) \right\} = \pm 1. \tag{2.10}$$

From Equation (2.5), the costate equations are

$$\dot{p}_1(t) = -\frac{\partial H}{\partial x_1(t)} = -\lambda_1 p_1(t), \tag{2.11}$$

and

$$\dot{p}_2(t) = -\frac{\partial H}{\partial x_2(t)} = -\lambda_2 p_2(t). \tag{2.12}$$

If the correct initial values of the costate variables $p(t) = \begin{pmatrix} p_1(t) \\ p_2(t) \end{pmatrix}$ are known, the state and costate equations can be solved and the input $u(t)$ determined. Suppose, the initial values of the costates are $p_1(0) = \pi_1$ and $p_2(0) = \pi_2$, then from Equations (2.11) and (2.12): $p_1(t) = \pi_1 e^{-\lambda_1 t}$, $p_2(t) = \pi_2 e^{-\lambda_2 t}$.

The relations can be obtained for the control sequence $u(t)$ and the state $x(t)$ as functions of time t. The solution is analytical, closed form and control sequences are bang-bang. From the solution to above problem the time-optimal control sequence is an element of the set $[\{+1\}, \{-1\}, (\{+1\}, \{-1\}), (\{-1\}, \{+1\})]$. One of the four alternatives will apply depending on initial state x_0.

Figures 3 and 4 show the time-optimal trajectory and the control for a two time constant system, the time constants being $\lambda_1 = -0.01$ and $\lambda_2 = -0.03$, from initial

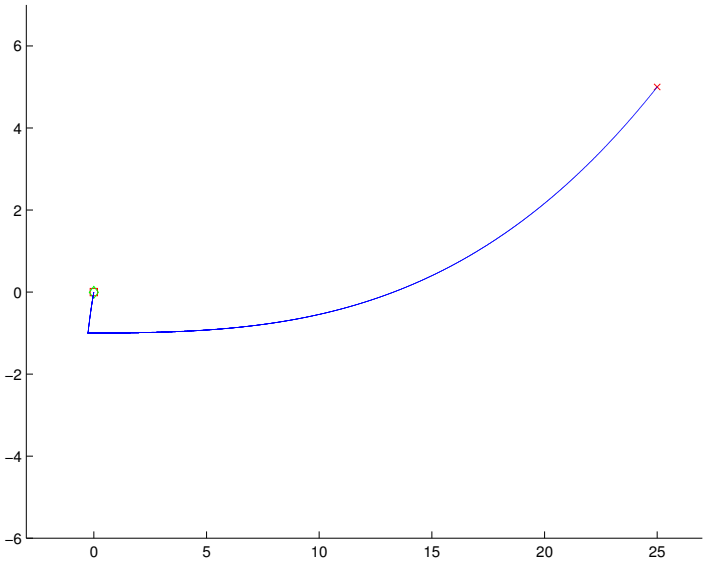

Fig. 3. Time-optimal trajectory of a two time constant system.

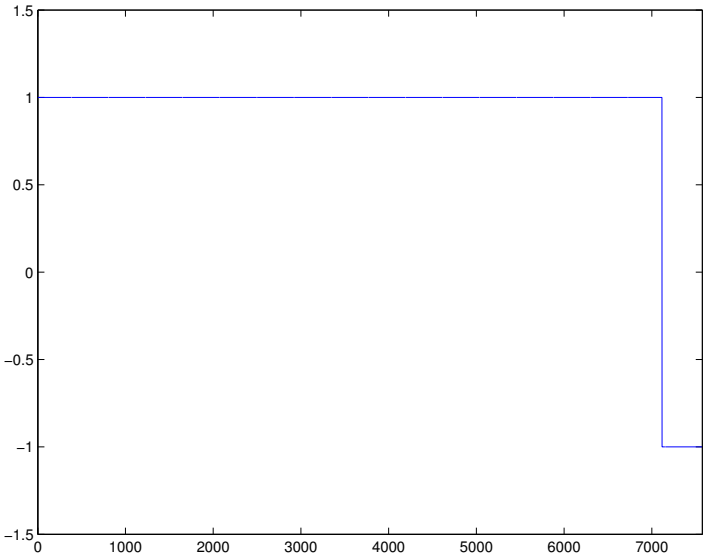

Fig. 4. Time-optimal input profile of a two time constant system.

state $x = \begin{pmatrix} 25 \\ 5 \end{pmatrix}$ to the origin. Bang-bang nature of control solution is obvious from the plot. The control sequence in this case is $(\{+1\}, \{-1\})$.

B. Optimal-aim concept

Optimal-aim concept was developed for performing regulation in transient nonlinear power systems. An optimal-aim strategy is a "control strategy based on the optimal constrained aim of trajectory derivatives in state space" [10]. This technique can provide online implementations for regulating systems having constrained input sets. Major desirable characteristics of the optimal-aim control strategy are [13],[14],[15]:

- It is a search scheme based on finite-dimensional optimization.
- Online implementation is possible even for unstable systems.

This strategy seeks to drive a system so that at each state on the resulting state space trajectory the corresponding *trajectory derivative* (or state velocity vector) points toward an equilibrium state as best as possible (by utilizing all available control effort). As a result of this, the system moves toward a condition of stability.

The basis for this concept is the fact that the direction of trajectory derivative at any state of a continuous-time system depends explicitly on instantaneous value of applied control. Input constraints will restrict trajectory derivatives to exist only in certain directions and in general it is not possible to point the derivative vector directly in direction of equilibrium states. So, control is applied so as to make the state velocity vector point toward equilibrium state in some optimal fashion. This is the optimal-aim condition.

Consider the system defined by the state equation

$$\dot{x}(t) = f(x(t)) + Bu(t), \tag{2.13}$$

where state vector $x(t) \in \mathbb{R}^n$, the control input constrained as $u(t) \in \Omega$ where $\Omega = \{\omega \in \mathbb{R}^m | \omega_j \in [\omega_{j\min}, \omega_{j\max}]\}$, $j = 1, \cdots, m$, σ_c is the equilibrium state, with $f(\sigma_c) = 0$. Figure 5 describes the state space geometry of optimal-aim concept. If $x(t)$ is current

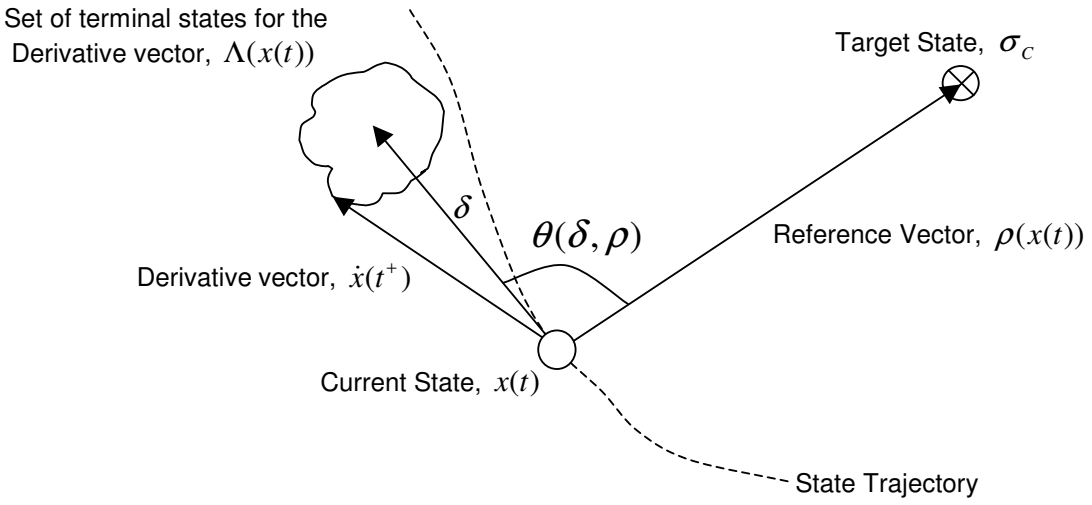

Fig. 5. The optimal aim geometry.

state vector at time t and σ_c the equilibrium state, there will exist:

$\Lambda(x(t))$	the constraint set consisting of points on which tip of the derivative vector $\dot{x}(t^+)$ will be constrained to terminate.
$\rho(x(t))$	the reference vector that emanates from current state and points at the equilibrium state σ_c.
δ	an arbitrary vector emanating from current state $x(t)$ and terminating in set $\Lambda(x(t))$.
$\theta(\delta, \rho)$	angle between vector δ and the target direction vector $\rho(x(t))$.

Referring to Figure 5, the optimal-aim condition can be described as choosing an input so that state velocity vector $\dot{x}(t^+)$ is oriented along reference vector ρ. If that is not possible then the input that achieves this in the best possible manner is chosen. We can see that trajectory derivative $\dot{x}(t^+)$ will point optimally toward equilibrium state σ_c when δ is oriented so that θ is minimized and $|\delta|$ is maximized. In case $\theta_{\min} > \frac{\pi}{2}$, we need to minimize $|\delta|$ in that direction to point $\dot{x}(t^+)$ along ρ in the best possible manner. This procedure is formalized below.

Optimal-aim technique seeks to determine the control input so that derivative vector $\dot{x}(t^+)$ satisfies two conditions:

1. Minimizes the angle θ over the set $\Lambda(x(t))$.

2. Either maximizes or minimizes the Euclidean norm $\|\dot{x}\|$ over the set $\Lambda(x(t))$ depending on whether θ is an acute or an obtuse angle.

Implementation of this technique consists of application of following three steps recursively:

1. Observing or estimating the system state $x(t)$ at time.

2. Computing the value of control $u(t)$ through real parameter optimization, to satisfy the two objectives stated above.

3. Apply the constant magnitude of the computed control to the actual system for a time interval Δt.

The optimal-aim conditions are satisfied only at discrete states (at a finite number of closely spaced states along the trajectory). Control input is recalculated at each of these states to satisfy the optimal-aim condition. For trajectory segments between these discrete states control input remains constant and optimal-aim conditions

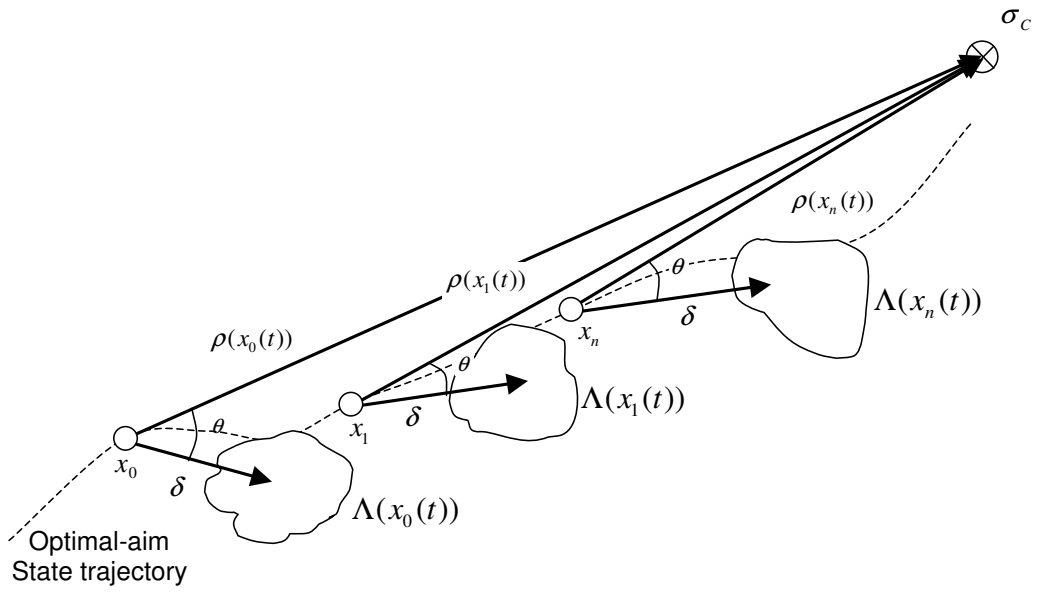

Fig. 6. State trajectory of a system evolving via optimal-aim strategy.

are satisfied only approximately. Figure 6 explains how the trajectory evolves with optimal-aim strategy in effect. The figure also shows optimal-aim variables defined at discrete points along the trajectory. Say, the initial state is x_0, then optimal-aim technique involves determining a reference vector ρ that minimizes θ and utilizing the corresponding input for system evolution. This will cause the system to move along the trajectory to a new state x_1, where the process will be repeated. This continues till the system reaches equilibrium state.

C. Implementation issues: Time-optimal controls vs. Optimal-aim methods

A time-optimal control solution is very desirable, as it requires the minimum amount of time for convergence to the equilibrium state. Although, analytical expression of the control sequence from optimal control theory is extremely useful, following factors

restrict application of time-optimal solutions for online regulation of more complex systems:

- Tedious solution procedure to solve the TPBVP, even for simple linear systems.
- Determination of the exact initial values of Lagrange multipliers (or costates) essential for the correct solution.
- Correct control sequence and the switching times have to be known in advance.
- Time-optimal solution corresponding to two close but distinct initial conditions can be significantly different.

Consequently, it is desirable to have alternatives to the cumbersome solution of TPBVP for regulation. Ideally, the alternative method should be amenable to online implementation and provide reasonably fast convergence. Many techniques have been explored to implement time-optimal solutions online. Optimal-aim strategies are good candidates for methods that approximate the time-optimal solution because of following characteristics:

- Calculations require very little memory and time.
- Implementation is based on closed-loop, online operations depending on current state alone.
- Control implementation is inherently adaptive in nature.

The state-space geometry of systems forms the basis of our expectation to approximate time-optimal control by modifying optimal-aim techniques. Optimal-aim techniques are based on state-space geometry and utilize the concept of aiming direction $\rho(x(t))$ and trajectory derivative $\dot{x}(t^+)$ (refer Figure 5) to calculate appropriate

control input for regulation. The Hamiltonian construction in optimal control theory has obvious geometrical interpretations too. The Hamiltonian for the time-optimal problem is defined as

$$H = 1 + \Big\langle p(x,t), \dot{x}(x,u,t) \Big\rangle. \tag{2.14}$$

Since, the Hamiltonian is an explicit function of the trajectory derivative $\dot{x}$, there exists a clear relation between geometry of the system and its Hamiltonian. It is postulated that a modified optimal-aim technique can generate the time-optimal trajectory, if aim directions can be chosen in a special manner.

Adaptation of optimal-aim concept to generate an approximate time-optimal solution will enable online implementation, without requiring the solution of a TPBVP. Also, this would ideally incorporate desirable characteristics of both techniques. To develop such an algorithm is the ultimate purpose of this research.

CHAPTER III

ATTEMPTS TO REPRODUCE EXACT TIME-OPTIMAL CONTROLS ONLINE

There have been many attempts to determine the time-optimal controls for systems and implement them in real-time [16],[17],[18]. Generally, these techniques are iterative in nature and require significant computational resources. Our endeavors try to avoid excessive computations.

Development of the required control algorithm involves establishing some assumptions and developing alternative control algorithms based on them. Application of these algorithms on the system, and subsequent simulation results provide an estimate of how closely a time-optimal solution is approximated.

This chapter illustrates some such assumptions and corresponding control schemes. Different control schemes thus developed, are applied to a double integrator system and computer simulations are carried out for validation. Since the double integrator has a simple time-optimal regulation solution, different control schemes are simulated and tested for their ability to successfully regulate the double integrator. Results and conclusions from simulations and implications of extending the algorithms to higher order or more complex systems are discussed.

Before describing the attempts to generate the time-optimal solutions online, we will explore the potential of state-space aiming methods to generate control solutions to meet different specifications. For example, the development of a globally stabilizing control via aiming is described in the next section.

A. Aiming in the presence of bounded controls

As shown by the optimal-aim concept, aiming techniques in state-space can prove to be faster, simpler and computationally lighter than the mathematically more elabo-

rate control algorithms. In this section we intend to utilize state-space aiming and explore the versatility of aiming techniques.

1. Null-controllable systems

Let us consider a general linear stable system

$$\dot{x} = Ax + Bu,$$

in the controllable canonical form. The state-space aiming formulation will involve an 'aim direction' and an 'aim state'. Let, ρ be the aim direction. By definition,

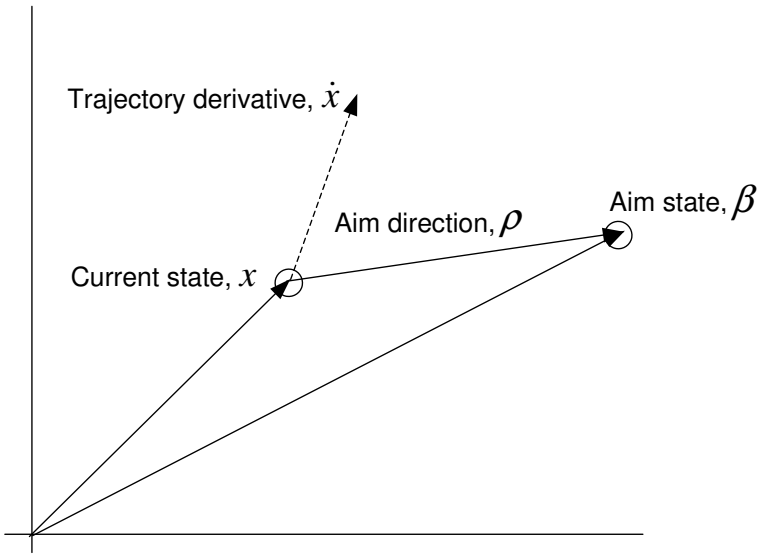

Fig. 7. Aiming solution geometry.

we want to point the trajectory derivative $\dot{x}$ in the aim direction ρ, so we seek to maximize the projection of trajectory derivative in the aim direction. In other words,

we try to maximize the inner product $\langle \dot{x}, \rho \rangle$. Clearly,

$$\begin{aligned} \langle \dot{x}, \rho \rangle &= (Ax + Bu)^T \rho \\ &= x^T A^T \rho + u^T B^T \rho. \end{aligned} \tag{3.1}$$

In Equation (3.1), only the input u can be affected, so to maximize $\langle \dot{x}, \rho \rangle$, we select

$$u_m^T = \text{sign}\Big[B^T \rho\Big]. \tag{3.2}$$

Equation (3.2) is the control law from the aiming strategy.

Next, we will employ the Lyapunov's theory to determine the stabilizing input for the same system. Let, the Lyapunov function be $V = x^T P x$. If matrix P be the positive definite solution of the Lyapunov matrix equation

$$\big(A^T P + PA\big) = -Q, \tag{3.3}$$

where Q is any positive definite matrix. For system stability:

$$\frac{dV(t)}{dt} = \dot{V} < 0.$$

It follows that,

$$\begin{aligned} \dot{V} &= \dot{x}^T P x + x^T P \dot{x} \\ &= (x^T A^T + u^T B^T) P x + x^T P (Ax + Bu) \\ &= \underbrace{x^T (A^T P + PA) x}_{<0} + 2 x^T P B u. \end{aligned} \tag{3.4}$$

To ensure $\dot{V} < 0$, we need to determine u so that $2x^T PBu < 0$. Suppose, $x \in \Re^n$ and $u \in [-1, 1]$. The stabilizing input can be selected as

$$u_s^T = -\text{sign}\Big[B^T P x\Big]. \tag{3.5}$$

This result is easily generalized to higher order systems.

For aiming control to be stabilizing, $u_s = u_m$. From Equations 3.5 and 3.2 we get

$$\rho = -Px. \tag{3.6}$$

Equation (3.6) provides us with a condition on aim direction ρ. Let the aim state be β. Then, by the definitions of aim direction and aim state (Figure 7)

$$\begin{aligned} & \rho = \beta - x \\ \Longrightarrow & \\ & \beta = x + \rho \\ & \quad = x + (-Px) \\ & \quad = (I - P)x. \end{aligned} \tag{3.7}$$

From Equations (3.6) and (3.7), regulation via state-space aiming concept will involve

1. Selection of a positive semi-definite matrix Q.
2. Solution of Laypunov's equation to determine the matrix P.
3. Computation of aim state $\beta = (I - P)x$.
4. Computation of aim direction $\rho = \beta - x$ and the stabilizing input $u^T = \text{sign}\left[B^T\rho\right]$.
5. Application of the input u for a predetermined time interval T. The quantities x, β and hence, u will change after this time.
6. Repeat steps 3 through 5.

Results: Figures 8 and 9 show the result of applying the above control algorithm on the system:

$$\dot{x} = \begin{bmatrix} -0.1 & 0 \\ 0 & -0.3 \end{bmatrix} x + \begin{bmatrix} -0.05 \\ 0.15 \end{bmatrix} u, \tag{3.8}$$

with $Q = \begin{bmatrix} 13 & 5 \\ 1 & 17 \end{bmatrix}$ and $P = \begin{bmatrix} 65 & 12.5 \\ 2.5 & 28.33 \end{bmatrix}$.

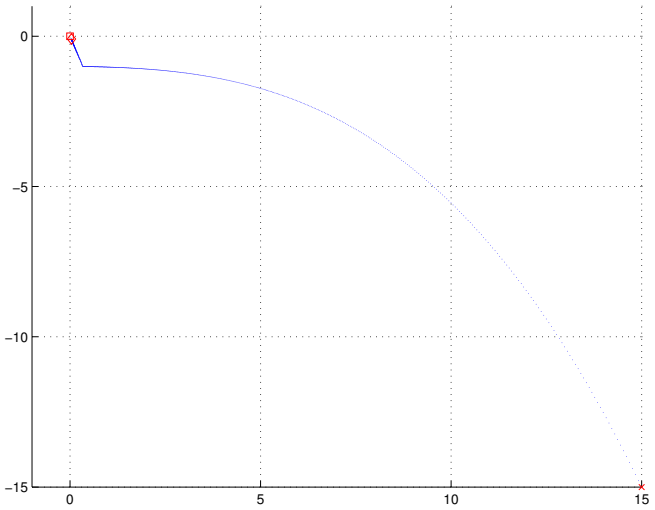

Fig. 8. State trajectory with aiming solution

2. Unstable systems

Consider a general linear unstable system

$$\dot{x} = Ax + Bu,$$

in the controllable canonical form. Since, the bounded input will definitely not be sufficient to stabilize the system, let us assume full state feedback for stabilization and adjust it so that it remains within the input bounds. The control law proposed

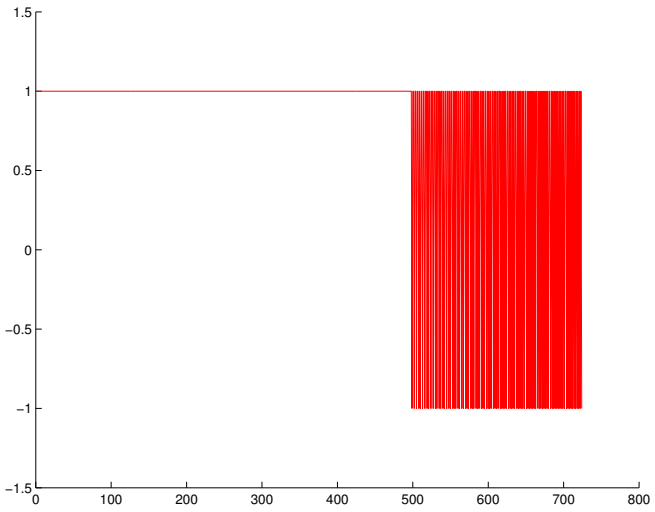

Fig. 9. Control input with aiming solution

is

$$u = Kx + u^*, \tag{3.9}$$

where u is the input to the system, u^* is the quantity to adjust for input constraints and K is the gain matrix for state feedback chosen to stabilize the nominal. The system equation can now be modified to

$$\dot{x} = (A + BK)x + Bu^*. \tag{3.10}$$

If $\overline{u}$ and $\underline{u}$ be the control bounds, with $\underline{u} < \overline{u}$, then

$$\underline{u} \leq (u = Kx + u^*) \leq \overline{u}, \tag{3.11}$$

everywhere on the eventual state trajectory. From Equation (3.11), the bounds on u^* are $\underline{u^*} = \underline{u} - Kx$ and $\overline{u^*} = \overline{u} - Kx$. We can see that if K be held constant, then the value of u^* will depend only on the state x.

Let us perform Lyapunov stability analysis on the above formulation. Let, the Lyapunov function be $V = x^T P x$. The matrix P is the positive semi-definite solution of the Lyapunov matrix equation

$$(A + BK)^T P + P(A + BK) = -Q, \tag{3.12}$$

where Q is any positive semi-definite matrix. To ensure system stability

$$\frac{dV(t)}{dt} < 0.$$

Since,

$$\begin{aligned} \dot{V} &= \dot{x}^T P x + x^T P \dot{x} \\ &= (x^T (A + BK)^T + (u^*)^T B^T) P x + x^T P((A + BK)x + Bu^*) \\ &= \underbrace{x^T((A + BK)^T P + P(A + BK))x}_{<0} + 2x^T P B u^*, \end{aligned} \tag{3.13}$$

to ensure $\dot{V} \leq 0$, and hence the Lyapunov stability of the modified system, we need to determine u^* so that $2x^T P B u^* \leq 0$. Suppose, $x \in \Re^n$ and $u \in [-1, 1]$. The stabilizing input can be computed as

$$u_s^* = -\text{sign}\Big[B^T P x\Big], \tag{3.14}$$

such that $u_s^* \in [\underline{u^*}, \overline{u^*}]$. Three cases will arise:

1. $\underline{u^*} < 0$ and $\overline{u^*} < 0$.
2. $\underline{u^*} < 0$ and $\overline{u^*} > 0$.
3. $\underline{u^*} > 0$ and $\overline{u^*} > 0$.

In addition to the above cases we also have to account for the sign of the quantity $B^T P x$, to determine the stabilizing control. The following selection table for stabilizing control can be generated using the above information:

Case	$B^T Px < 0$	$B^T Px > 0$
1	$u_s^* = \overline{u^*} \quad \otimes$	$u_s^* = \underline{u^*}$
2	$u_s^* = \overline{u^*}$	$u_s^* = \underline{u^*}$
3	$u_s^* = \overline{u^*}$	$u_s^* = \underline{u^*} \quad \otimes$

$\otimes$ - No stabilizing control inputs exist in these cases. This is the best that we can do as actuator saturation limits our choice.

If the plant is not null-controllable, the aiming approach involves full state feedback for favorable pole-placement, this shifts the allowable control set Ω by an amount Kx. This causes Ω become asymmetric with respect to the control $u = 0$, and as a result we cannot always determine the stabilizing input according to Equation (3.14).

Results: We choose an unstable two time constant system for observing the behavior under the above control scheme:

$$\dot{x} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.3 \end{bmatrix} x + \begin{bmatrix} -0.1 \\ -0.3 \end{bmatrix} u. \tag{3.15}$$

Then, we perform arbitrary pole-placement at the poles $\begin{pmatrix} -0.1 + 0.2i \\ -0.1 - 0.2i \end{pmatrix}$ via full-state feedback. The feedback gain being $K = \begin{bmatrix} -4 & 3.333 \end{bmatrix}$, with $Q = \begin{bmatrix} 13 & 2 \\ 1 & 8 \end{bmatrix}$ and $P = \begin{bmatrix} 360 & 563 \\ 558 & 966 \end{bmatrix}$. Figures 10 and 11 show the results for the initial state $\begin{pmatrix} 0.7 \\ 0.7 \end{pmatrix}$. Figures 12 and 13 show the results for the initial state $\begin{pmatrix} 0.75 \\ 0.75 \end{pmatrix}$.

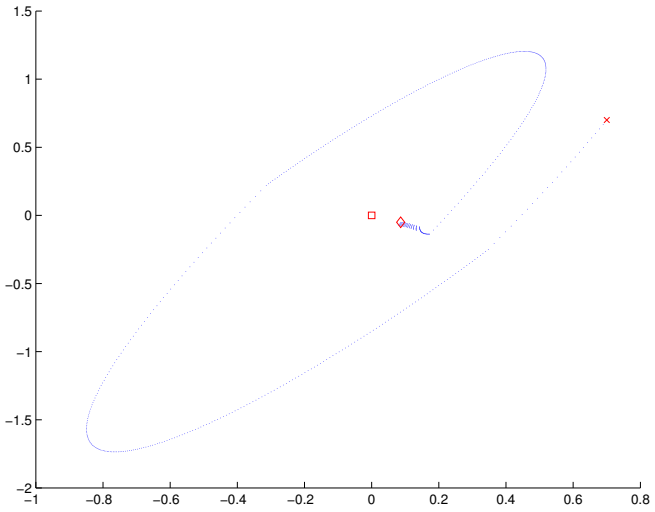

Fig. 10. State trajectory with aiming solution, initial state [0.7 0.7].

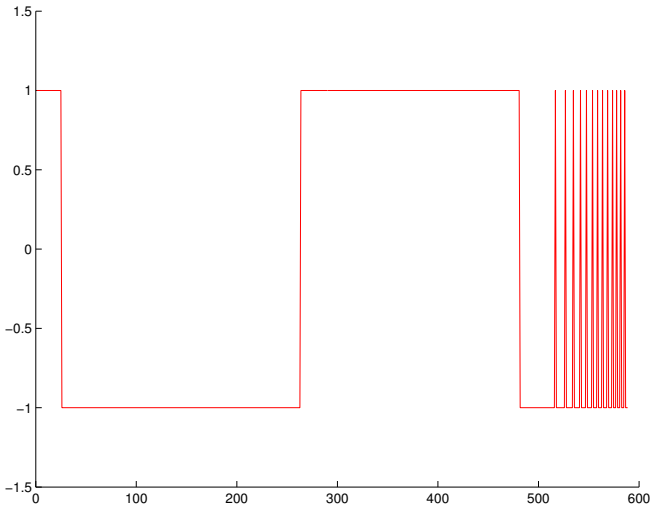

Fig. 11. Control input with aiming solution, initial state [0.7 0.7].

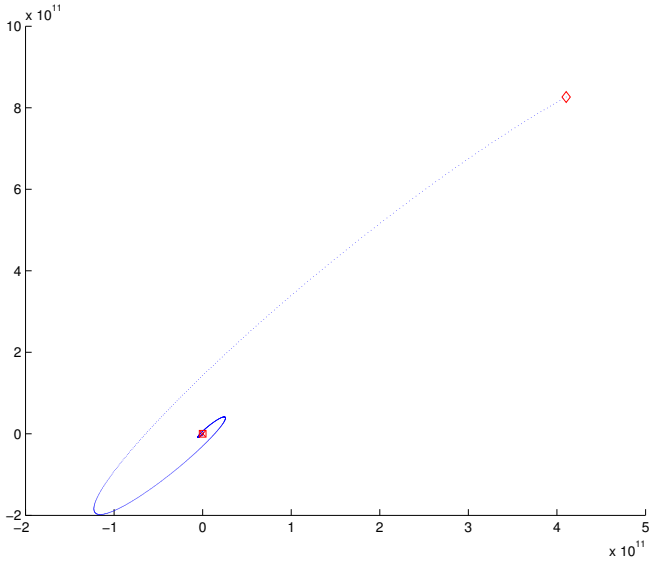

Fig. 12. State trajectory with aiming solution, initial state [0.75 0.75].

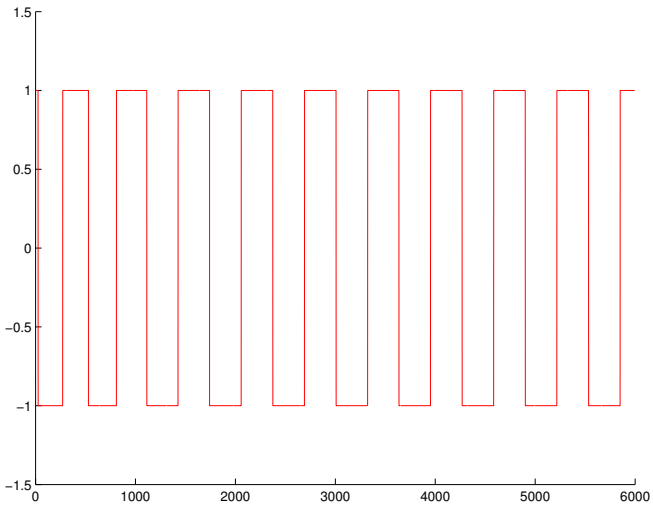

Fig. 13. Control input with aiming solution, initial state [0.75 0.75].

Thus we have seen that aiming methods are capable of generating globally asymptotically stabilizing controls in null-controllable systems provided the aiming direction is chosen in a special way (Equation (3.6)). This aiming solution is not successful in stabilizing marginally stable or unstable systems.

B. Time-optimal solution for double integrator

The time-optimal solution for a double integrator system is reproduced here for reference. State equation for a double integrator system is

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u = \begin{bmatrix} x_2 \\ u \end{bmatrix}, \tag{3.16}$$

where $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$. The control input $u(t) \in \Omega = \{\omega \in \mathbb{R}; \omega \in [-1, 1]\}$. The goal is to determine a control sequence that transfers the system from an initial state x_0 to the origin in minimum time.

The Hamiltonian for this problem is given by

$$H = 1 + x_2(t)p_1(t) + u(t)p_2(t). \tag{3.17}$$

Therefore, *H-minimal* control is

$$u(t) = -\text{sgn}\{p_2(t)\} = \pm 1. \tag{3.18}$$

Costate equations for the system are

$$\dot{p}_1(t) = -\frac{\partial H}{\partial x_1} = 0, \tag{3.19}$$

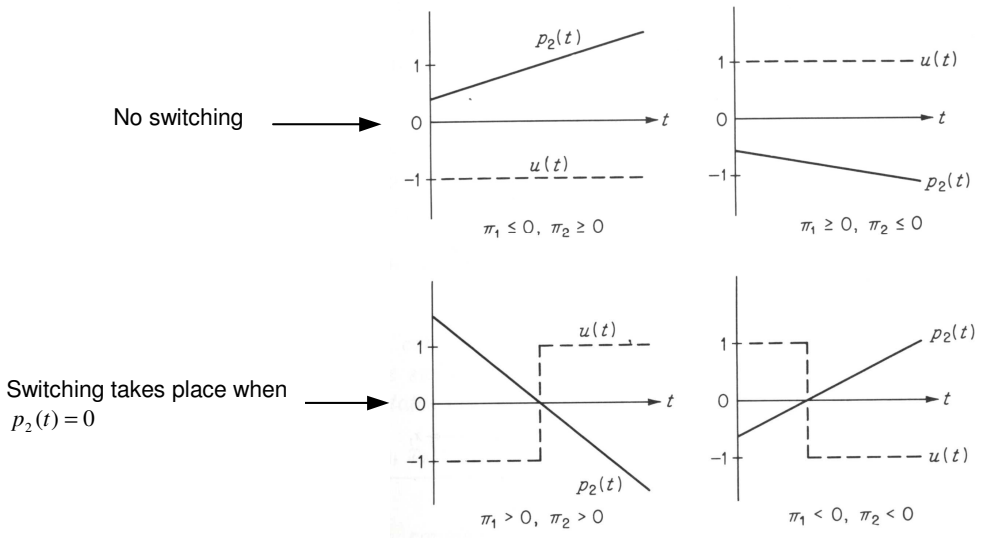

Fig. 14. The four possible modes for the evolution of the costate vector, $p(t)$.

and

$$\dot{p}_2(t) = -\frac{\partial H}{\partial x_2} = -p_1(t). \tag{3.20}$$

If initial values of costates are: $p_1(0) = \pi_1$, $p_2(0) = \pi_2$ $\left(\text{or } p(0) = \pi = \begin{pmatrix} \pi_1 \\ \pi_2 \end{pmatrix} \right)$, then the values of costates at any time can be given by $p_1(t) = \pi_1(\text{constant})$ and $p_2(t) = \pi_2 - \pi_1 t$. The plot of $p_2(t)$ w.r.t. time is a straight line. Four possible shapes of $p_2(t)$ and the corresponding shapes of H-minimal control are shown in Figure 14.

From this we can conclude that time-optimal control for a double integrator is piecewise constant and can switch at most, once, as the system is normal. So, $\{[+1], [-1], [+1, -1], [-1, +1]\}$ is the set containing all possible optimal control sequences. Figures 15 and 16 show the time-optimal solution with initial state $\begin{pmatrix} +5 \\ -5 \end{pmatrix}$.

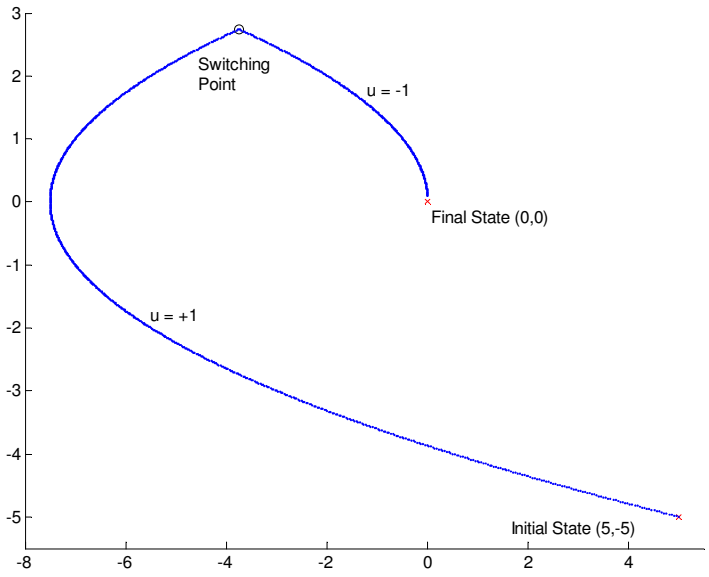

Fig. 15. Time-optimal trajectory of a double integrator.

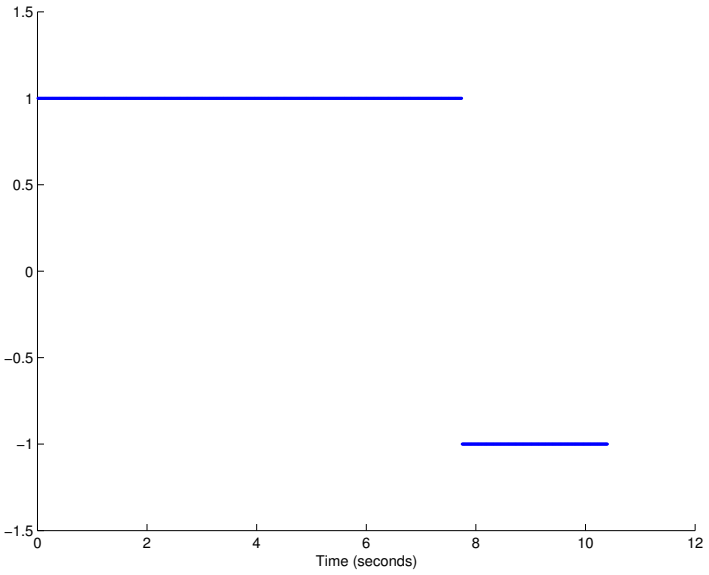

Fig. 16. Time-optimal control for a double integrator.

Seven alternative hypotheses are presented here along with accompanying assumptions and the derived control schemes. First five focus on reproducing the exact time-optimal control by estimating the values of costate variables corresponding to the initial state. The latter two hypotheses are related to the geometry of the system and do not attempt to reproduce the time-optimal solution.

Following facts relating to time-optimal solutions aid in the development of hypotheses 1 through 5:

- $\Omega = [-1, 1]$ is assumed to be the set containing all possible values of control u.
- Time-optimal control sequences are essentially bang-bang and the actuators always operate under saturation at either one of the extreme limits. Generally, the control solutions sought by following hypotheses will assume saturated actuators at all times i.e. $u = -1$ or $u = 1$.
- A time-optimal solution satisfies $H = 0$, at all points on the trajectory. As $H = 1 + \langle p, \dot{x} \rangle$, we must ensure $1 + \langle p, \dot{x} \rangle = 0$ or $\langle p, \dot{x} \rangle = -1$, in the search for costate p. Numerical calculations may not be able to make the dot product exactly equal to -1. So, we try to make the dot product as close to -1 as possible.

C. Hypothesis 1:

The time-optimal solution of a double integrator is straightforward once initial values of costate variables are known. This hypothesis relates initial state to initial values of costates in a straightforward way.

If $x(0) = \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix}$ *is the initial state.* π_1 *and* π_2 *are the initial values of the costates computed as* $\pi_1 = \frac{x_1(0)}{|x(0)|}$ *and* $\pi_2 = \frac{x_2(0)}{|x(0)|}$.

Control scheme corresponding to the above hypothesis is described by following sequence of steps:

1. Select a time step T over which applied input will remain constant.

2. Compute the initial values of costates from the initial state: $\pi_1 = \frac{x_1(0)}{|x(0)|}$ and $\pi_2 = \frac{x_2(0)}{|x(0)|}$.

3. The control input to be selected should always minimize the Hamiltonian. After costate variables are computed, we calculate two trajectory derivatives corresponding to extremities of the control set Ω. Since our search space is affine, we just have to investigate the extremities of the allowable control set Ω for the minimum. Here the extremities are $u_1 = -1$ and $u_2 = 1$. Corresponding values of trajectory derivatives are $\dot{x}_{(1)} = Ax + B(-1)$ and $\dot{x}_{(2)} = Ax + B(1)$ respectively.

4. Calculate Hamiltonian values corresponding to both trajectory derivatives: $H_{(1)} = 1 + \langle \pi, \dot{x}_{(1)} \rangle$ and $H_{(2)} = 1 + \langle \pi, \dot{x}_{(2)} \rangle$.

5. Next, we determine the quantity $\Big\{ \min \big(|H_{(1)}|, |H_{(2)}| \big) \Big\}$, or the minimum of two Hamiltonian values.

6. Select u_1 as the input if $\Big\{ \min \big(|H_{(1)}|, |H_{(2)}| \big) \Big\} = H_{(1)}$, or
   u_2 as input if $\Big\{ \min \big(|H_{(1)}|, |H_{(2)}| \big) \Big\} = H_{(2)}$.

7. Use the selected input to let the system evolve for one time step T.

8. Calculate new values of costate variables using the previous values from Equations (3.19) and (3.20).

9. Repeat steps 3 through 8 till the state converges to the origin.

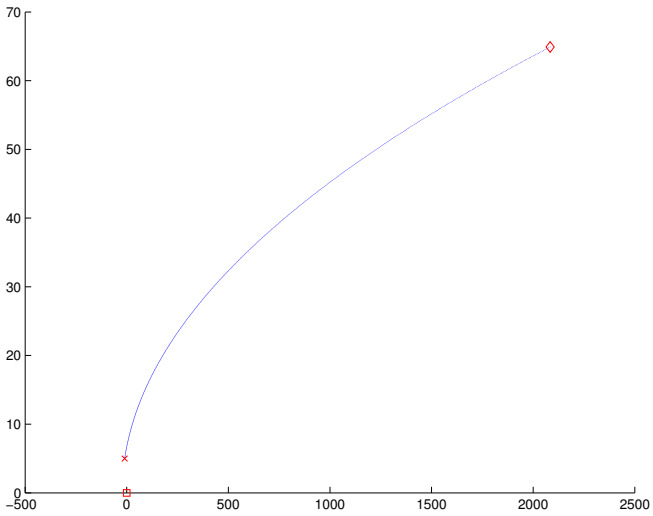

Fig. 17. System trajectory.

Validity of this hypothesis can be determined by applying the control algorithm to the double integrator. Simulation results are presented below for the initial state $\begin{pmatrix} -10 \\ +5 \end{pmatrix}$. Figure 17 shows the state-space trajectory for the system under this control algorithm. Control input and the Hamiltonian for above trajectory are shown in Figures 18 and 19, respectively.

No convergence is exhibited by the system. Moreover, Figure 17 shows that the value of the Hamiltonian differs from 0 all along the state trajectory. Hence, hypothesis 1 does not result in a time-optimal control. We need to seek alternatives to this algorithm.

D. Hypothesis 2:

Initial values of costate variables (π_1, π_2) are independent of each other. Instead of trying to relate the initial state and initial costate vectors, we can attempt to guess

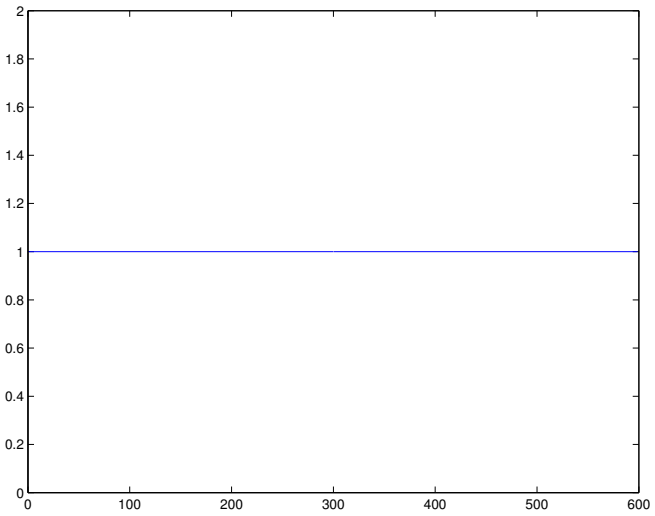

Fig. 18. Control input.

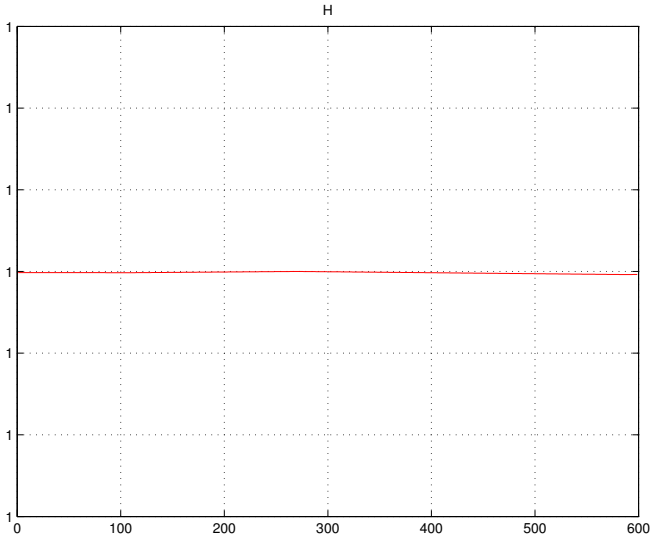

Fig. 19. Hamiltonian along the state trajectory.

the values of costate variables for this problem. For a simple system, such as the double integrator, we can confirm the accuracy of our guess by using simulations and observing if regulation is achieved.

A search in the two-dimensional space (π_1, π_2) *can be carried out to determine correct initial values of costate variables* $(p_1(0), p_2(0)) = (\pi_1, \pi_2)$.

We can perform the search along the circumference of a unit circle in (π_1, π_2) plane. Different orientation angles φ will have different components of π along the coordinate axes and will adjust for the *relative magnitude* of initial costates. Scaling will regulate the absolute magnitude of initial costates.

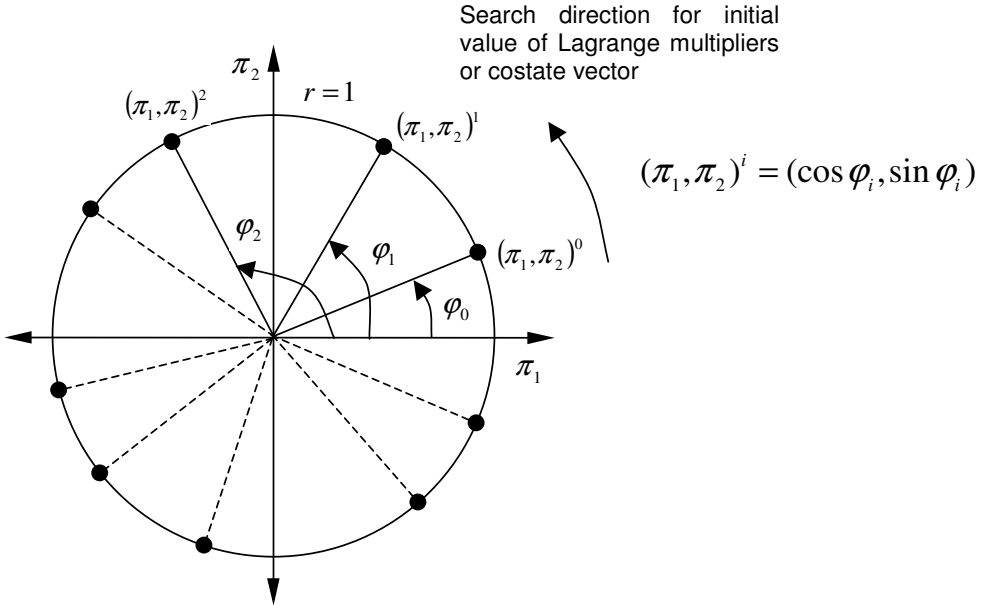

Fig. 20. Search for initial costates along the circumference of a unit circle in π_1, π_2 space.

Search for initial costate vector has to be performed off-line through a number of off-line trials. Figure 20 explains the search procedure. Different values of φ will provide different pairs of initial costates. After a particular costate $(\pi_1, \pi_2)^i$ is

chosen, off-line simulations are carried out with $u = \pm 1$ to determine if $H = 0$ along any one of the two trajectories thus generated. By selecting different values of initial costates (for different values of φ) the correct values of initial costate are decided using computer simulations. Correct values thus determined are used for the actual system. Corresponding control algorithm is described as follows:

1. Select a new costate as shown in Figure 20.
2. Calculate the two trajectory derivatives corresponding to the extremities of control set Ω. These are $\dot{x}_{(1)} = Ax + B(-1)$ and $\dot{x}_{(2)} = Ax + B(1)$.
3. Corresponding to these possible trajectory derivatives we calculate the Hamiltonian. $H_{(1)} = 1 + \langle p, \dot{x}_{(1)} \rangle$ and $H_{(2)} = 1 + \langle p, \dot{x}_{(2)} \rangle$
4. Next, we compute the min $\big(|H_{(1)}|, |H_{(2)}|\big)$.
5. Select the control that minimizes the Hamiltonian as input.
6. Using the selected input and let the system evolve for another time step.
7. Compute new values of costate variables.
8. Repeat steps 2 through 7 till convergence.

Figure 21 shows the simulations using results of the search process. These are some plots generated by selecting different values of π_1 and π_2. Search for π_1 and π_2 for Figure 21 was performed manually, i.e., the search for correct values was guided by visual observation of trajectory plots in each case. This can become tedious if there is no prior idea of magnitude of the values.

This procedure seems to be very simple in principle but with nothing to simplify the search process, it can prove to be very cumbersome to work with. Dimensionality

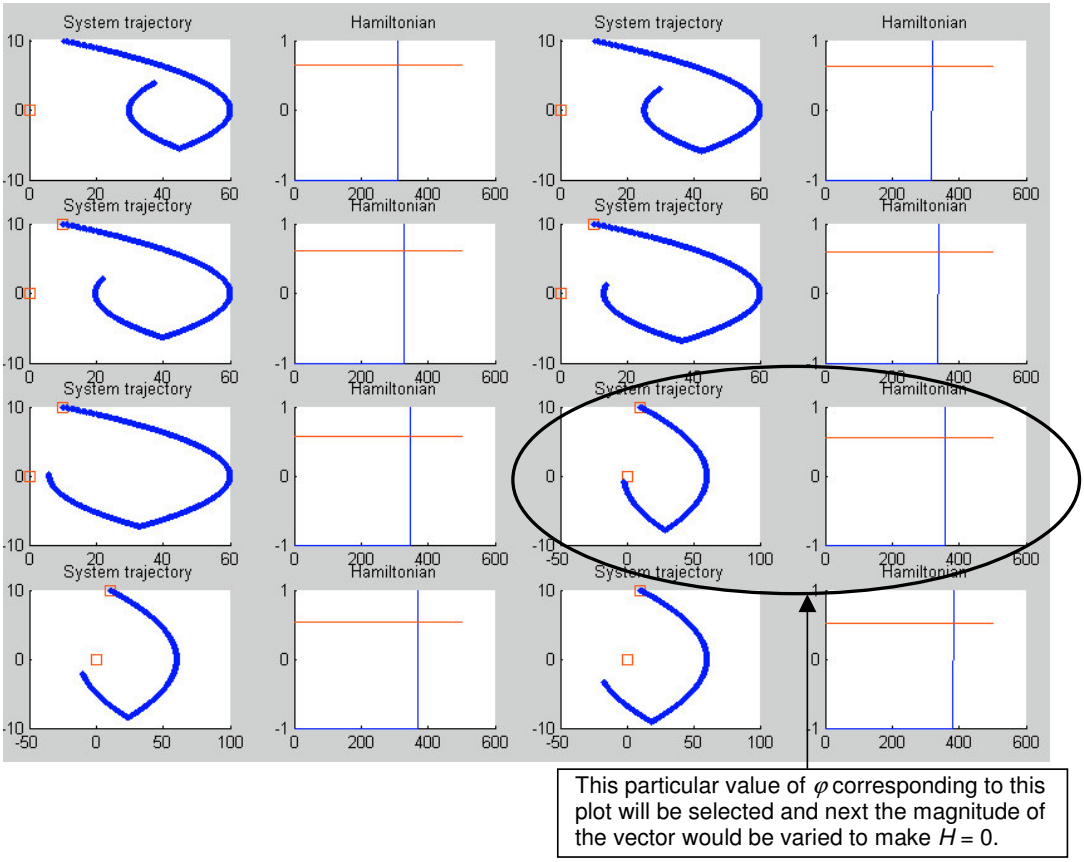

Fig. 21. System trajectories for the system using different values of π_1 and π_2 from a portion of the unit circle.

of the search space for a system depends on the order of the system. In case of higher order systems the complexity of search space and hence time required will increase unreasonably. This procedure is also computationally very intensive. For these reasons, we need to further modify this algorithm to simplify the search process.

E. Hypothesis 3:

Hypothesis 2 introduces the idea of estimating initial costate by performing a search in $\Re^2$. Hypothesis 3 seeks to reduce the size of the search space by utilizing geometrical implications of the fact that magnitude of the Hamiltonian H is zero along the time-optimal trajectory.

Hamiltonian $H = 1 + \langle p, \dot{x} \rangle = 0$ *at all points along the time-optimal trajectory. From this we can infer that* $\langle p, \dot{x} \rangle = -1$. *Now, a better guess for the initial values of costates can be made since we know,* $\begin{pmatrix} \pi_1 \\ \pi_2 \end{pmatrix}$ *lies in the half-space orthogonal to either* $\dot{x}(0)_{(u=1)}$ *or* $\dot{x}(0)_{(u=-1)}$ *depending on the correct value of input to be applied.*

This hypothesis if correct, will effectively reduce the search space from $\Re^2$ to $\mathbf{S} = \mathbf{S}_1 \bigcup \mathbf{S}_2$, where $\mathbf{S}_1 = \left\{ v : \langle \dot{x}(0)_{(u=1)}, v \rangle < 0 \right\}$ and $\mathbf{S}_2 = \left\{ v : \langle \dot{x}(0)_{(u=-1)}, v \rangle < 0 \right\}$. $\mathbf{S}$ represents a sector of the unit circle with scaling applied as appropriate. As for that sector, we can carry out the search for costates as before, with control scheme being similar to the one corresponding to hypothesis 2.

Figures 22 and 23 show actual initial costate values and the initial derivative vectors plotted together for different initial states. These plots show how the initial costates for the time-optimal control indeed lie in the half-space orthogonal to one of the $\dot{x}_{(1)}(0)$ or $\dot{x}_{(2)}(0)$. The control algorithm is the same as that in the previous case.

For the double integrator, hypothesis 3 reduces the size of search space to half, but we are still left with an appreciable portion of $\Re^2$. We can strive to simplify the search even further. Hypothesis 4 seeks to achieve this.

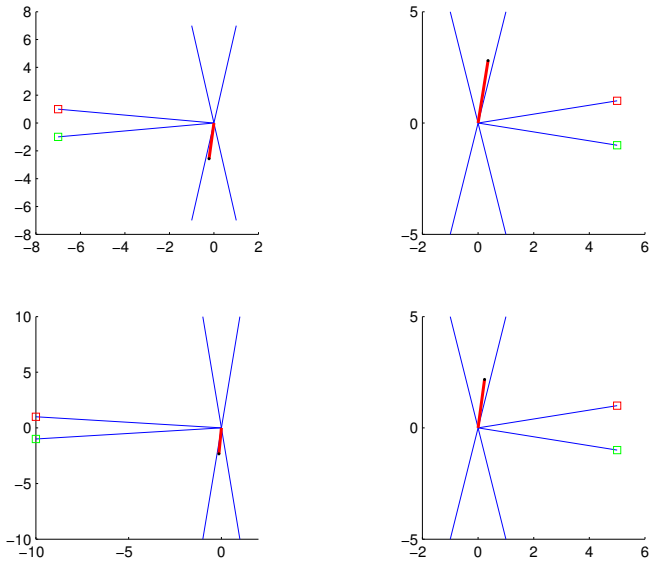

Fig. 22. Geometry of costate vector and initial trajectory derivative I

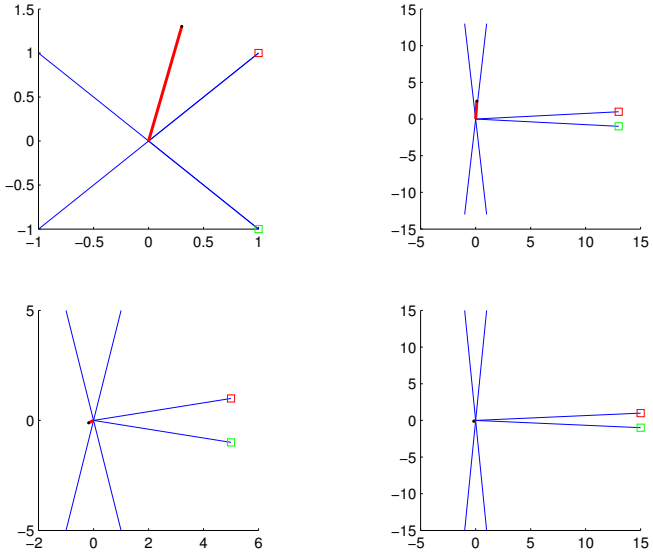

Fig. 23. Geometry of costate vector and initial trajectory derivative II

F. Hypothesis 4:

The preceding hypotheses show that searching for both costates independently is quite cumbersome. By invoking the Hamiltonian magnitude concept ($H = 0$), we can simplify the two-dimensional search to a single dimension.

We have known that the Hamiltonian $H = 1 + \langle p, \dot{x} \rangle = 0 \Rightarrow \langle p, \dot{x} \rangle = -1$. *This fact can help us reduce the search space for* $\begin{pmatrix} \pi_1 \\ \pi_2 \end{pmatrix}$ *even further. Recognizing that once* π_1 *is chosen,* π_2 *can be calculated using the above relation. This will reduce the 2-dimensional search to a 1-dimensional search.*

Now we only need to accurately select the value of one of the costate variables, the value of the other costate is calculated by using the $H = 0$ condition.

The control algorithm consists of the following steps.

1. Assume a particular value of π_1.
2. Calculate two extremal values for the trajectory derivative at the initial state corresponding to inputs $u = +1$ and $u = -1$ due to the actuator saturations in both directions. $\dot{x}_{(1)} = Ax + B(+1)$ and $\dot{x}_{(2)} = Ax + B(-1)$.
3. Compute the two alternative values of π_2 by utilizing the Hamiltonian definition $H = 1 + \pi_1 x_2 + \pi_2 u$ and substituting thus $\pi_{2(i)} = \frac{-1-\dot{x}_{1(i)}}{\dot{x}_{2(i)}}, i = 1, 2$.
4. Using the two sets of possible initial costates $\{\pi_1, \pi_{2(1)}\}$ and $\{\pi_1, \pi_{2(2)}\}$, off-line simulation is carried out to observe which of the sets maintains the Hamiltonian at zero, as required by the time-optimal solution.
5. Repeat steps 1 through 4 till the correct values for costate variables are selected.
6. The correct set of initial costates is then used for the regulation. The correct trajectory will have the Hamiltonian maintained at zero at all points on it.

The search space is thus reduced in dimension. But we still need to fix the value of one of the costates to three or four places of decimal to reproduce the time-optimal solution. After one of the costate variable has been guessed, we still need to know which value of input to use to be able to calculate the correct value for the second costate variable. This is acceptable for low order systems but will make the search process immensely intractable for higher order systems. And then again it will involve multiple off-line trials. Thus, this algorithm will not prove very suitable for online implementation.

G. Hypothesis 5:

Reverse-time evolution is an extremely useful means to backtrack the system trajectory from a known final state. In the regulation problem, final state is always the origin. Therefore, any trajectory that terminates at the origin driven by either $u = 1$ or $u = -1$, is an optimal trajectory. If negative-time evolution is performed with origin as initial state and input being either $u = 1$ or $u = -1$, a part or whole of the optimal trajectory can be retraced, depending on whether switching is involved or not.

Any reverse-time trajectory starting from the origin that is driven with $u = 1$ *or* $u = -1$ *as input, will always be along a time-optimal trajectory.*

The correct time-optimal trajectory can be reconstructed by ensuring that reverse-time trajectory from the origin and forward-time trajectory originating from initial state meet each other. The off-line simulations will involve the following steps.

1. The system is simulated to evolve in negative time from the origin till the *estimated* switching point position.

2. An initial value of input is estimated (either $u(0) = -1$ or $u(0) = 1$) the system

is simulated to evolve in forward time from the initial state.

3. The time-optimal trajectory is the one in which both forward-time and reverse-time curves just about meet each other before diverting from the exact time-optimal solution.

4. Selecting the input corresponding to the plot with most promising trajectory.

5. Using the selected input to let the system evolve for another time step.

6. Repeat steps 1 through 5.

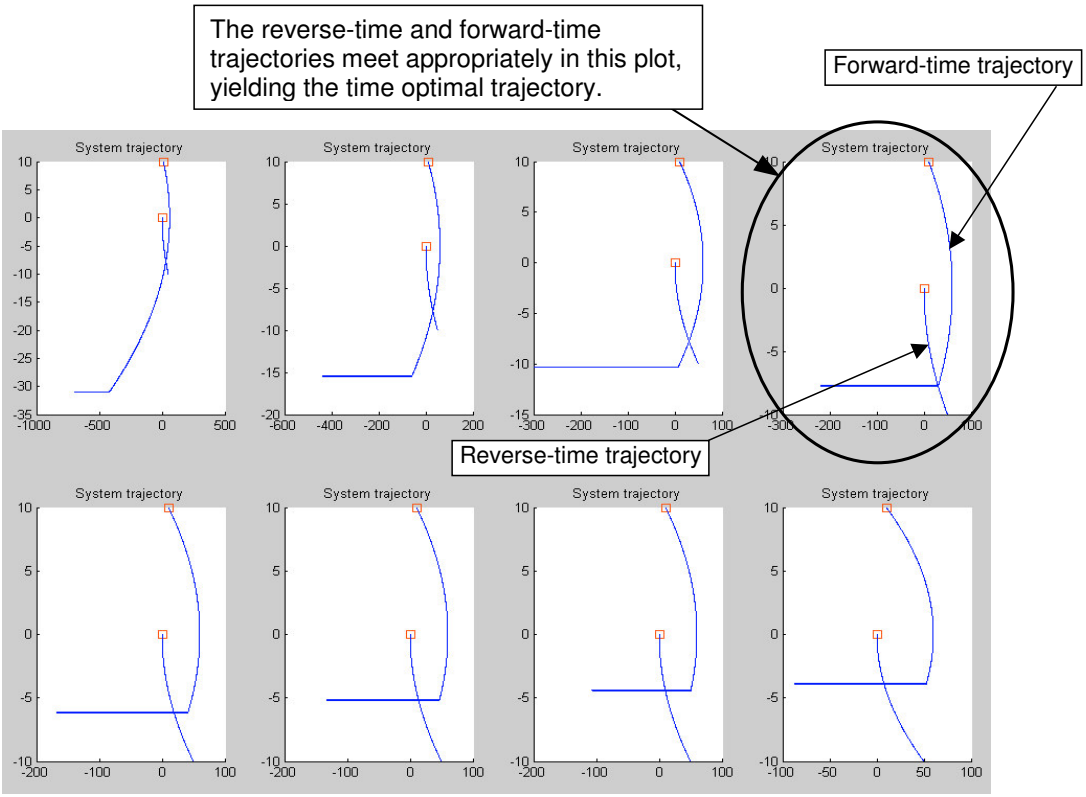

Fig. 24. Various plots showing the search procedure.

Figure 24 demonstrates how negative-time evolution may be utilized to determine the correct time-optimal trajectory through a number of off-line simulations.

As in the previous algorithms, this involves off-line simulations and figuring out the correct solution involves tedious observation as to where the two trajectories (forward- and reverse-time) meet. Hence, this algorithm holds limited promise for online implementation.

So far we have been trying to hypothesize methods to reproduce time-optimal control by avoiding solution of TPBVP. This inevitably involves either exact knowledge of costate variables or a number of off-line trials to determine them. Clearly, these cannot be implemented in real-time. Chapter IV describes some other methods that perform regulation with bounded inputs but not in a time-optimal manner.

CHAPTER IV

ONLINE APPROXIMATE TIME-OPTIMAL CONTROLS

Control schemes described in chapter III involve off-line trials to pin-point the exact values of costate variables. Such methods will prove to be abortive as they not only require more time but also more computing resources. Moreover, the off-line trials will have to be repeated for every distinct initial condition. With this realization, the initial goal of reproducing the true time-optimal solution was modified to developing an online control scheme that would *approximate* time-optimal solution. The approximate solution might require more time for convergence than the exact time-optimal control. This is acceptable as an online technique would dispense with off-line trials and high computational overheads.

This chapter describes some control schemes based on state-space geometry of the plant. Hypotheses 6 and 7 are very similar to each other and involve minimization of a scalar quantity. Later part of this chapter describes a state-space aiming technique very similar to the optimal-aim methods.

A. Hypothesis 6:

Superpose the two $\Re^2$ spaces corresponding to x and $\dot{x}$, and investigate the vector sum $(x+\dot{x})$. The sum $(x+\dot{x})$ can be interpreted as a predictive measure of where the state vector is going to terminate after the next integration step of the state equation.

The quantity $(x + \dot{x})$ is a predictive measure of the next integration step for a unit time interval. (If x be the "displacement", $\dot{x}$ is the velocity and the quantity $(x+\dot{x})$ then is the displacement after a unit time interval.) Figure 25 clearly describes the state-space geometry involved.

Intuitively, we can realize that, minimizing $(x + \dot{x})$ at every point along the tra-

jectory, will move the state closer to the origin and eventually $(x+\dot{x}) = 0$. Minimizing the norm of the *displacement after a unit time interval*, at a number of closely spaced points along the state trajectory, using the available values of input, in effect should bring the state closer to the origin as the system evolves. So we can write a tentative control law for the control calculation as

$$u = \left\{ u \middle| \min_{\Omega} \left\| x + \dot{x} \right\| \right\}. \tag{4.1}$$

This method can be slightly modified by investigating the quantity $(x + \alpha\dot{x})$ instead of $(x + \dot{x})$, yielding the following control law:

$$u = \left\{ u \middle| \min_{\Omega} \left\| x + \alpha\dot{x} \right\| \right\}. \tag{4.2}$$

Whereas, previously the prediction was for the unit time interval, we can change this time interval to any value by increasing or decreasing the value of the parameter $\alpha(> 0)$. In some cases this might give faster rate of convergence. This strategy seeks to regulate the system but not by reproducing the time-optimal control solution. To perform regulation the control strategy seeks to minimize the quantity $\|x + \alpha\dot{x}\|$ at a discrete number of points on the state-space trajectory of the system, over the input set Ω, and involves the following steps.

1. The input range is discretized into a finite number of segments. For example, $\Omega = [-1, 1] = \{u_j\} = \{-1.0, -0.9, -0.8 \cdots 0 \cdots 0.8, 0.9, 1.0\},\ j = 1, 2 \cdots p$, where p is the number of segments of the input set after discretization.[1]

2. Select an appropriate value for α. It has been found by experience, the results can be affected to an appreciable extent by varying the value of α.

[1]Instead of discretizing the input set Ω, we can also perform a mathematical optimization of the quantity $\|x + \dot{x}\|$ over this set. The optimization is a one-step process and the minimizing input u^* can be directly used in step 6.

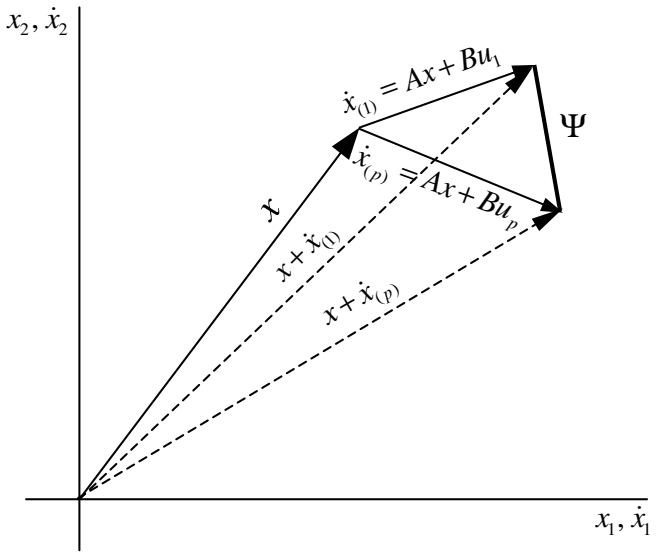

Fig. 25. Superposition of the x and $\dot{x}$ spaces.

3. Select a constant time interval over which the input will stay constant and after which the control will be recalculated according to the algorithm.

4. Calculate the allowable derivative vector set, using the current state value. $\Psi = \left\{\dot{x}_{(j)} : \dot{x}_{(j)} = Ax + Bu_j\right\}$.

5. Compute the sets $E = \{e\} = \left\{x + \alpha\dot{x}_{(j)}\right\}$, $j = 1, 2 \cdots p$ and $E_n = \{\|e\|\} = \left\{\|x + \alpha\dot{x}_{(j)}\|\right\}$.

6. Determine the minimum value of $\|x + \alpha\dot{x}_{(j)}\|$ from the set E_n and select the corresponding value of u_j, as the input, u^*. This is the minimizing input.

7. Use the selected input to let the system evolve for another time step.

8. Repeat steps 4 through 7.

This algorithm is advantageous over previous ones in the sense that it is completely online. The state-space trajectory described by the system under this control technique does hold the promise of eventual regulation, with some modifications. This modification is discussed in hypothesis 7.

The following results show the effect of changing the value of the parameter α on the resulting trajectory. Figures 26 and 27 show results of applying the control algorithm on the double integrator with $\alpha = 1$. Figures 28 and 29 are the results with $\alpha = 15$. In all these cases the initial state is $x(0) = \begin{pmatrix} 10 \\ -5 \end{pmatrix}$.

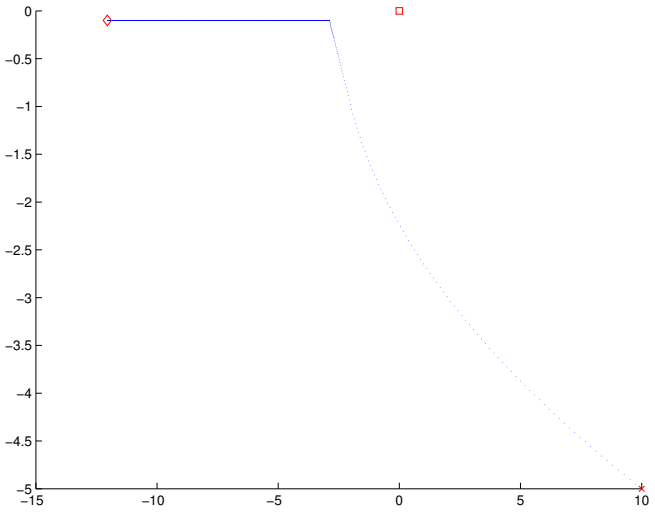

Fig. 26. System trajectory, $\alpha = 1$.

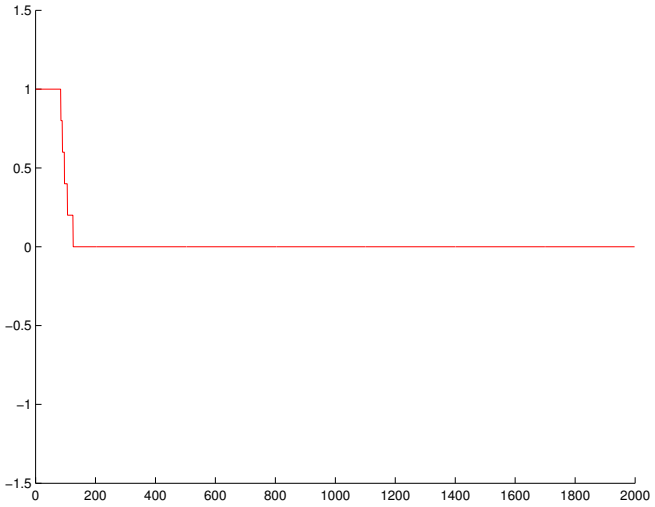

Fig. 27. Control input, $\alpha = 1$.

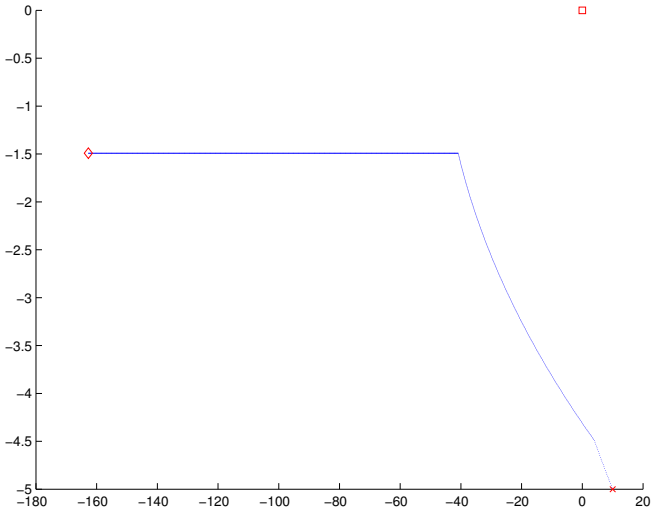

Fig. 28. System trajectory, $\alpha = 15$.

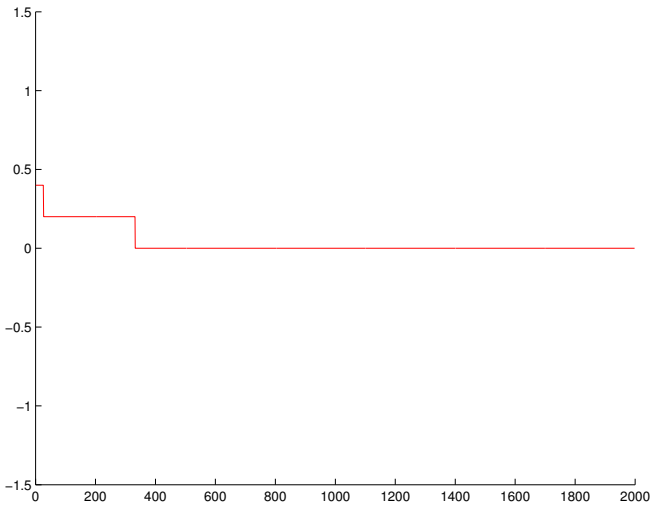

Fig. 29. Control input, $\alpha = 15$.

Studying the effect of changing the initial state but keeping α constant is also instructive. The following results are obtained by simulating the effect of controller developed using hypothesis 6 on the double integrator system. The plots correspond to two different initial states. The plots were generated with initial conditions $x(0) = \begin{pmatrix} 7 \\ -7 \end{pmatrix}$ and $x(0) = \begin{pmatrix} -10 \\ 12 \end{pmatrix}$, respectively. The value of the parameter $\alpha(=5)$ was kept constant for both simulations.

In Figures 30 and 31, we can clearly observe that notwithstanding the initial conditions, this algorithm will force the trajectory on similar (or symmetrical) paths. Noticeably all these trajectories appear to be heading toward the origin till they reach close to the horizontal axis. The control input plots (Figures 32 and 33) for these cases also show similar behavior.

A notable fact in the results following from hypothesis 6 is that the control signals never switch signs from positive to negative and vice versa. Also, the cost

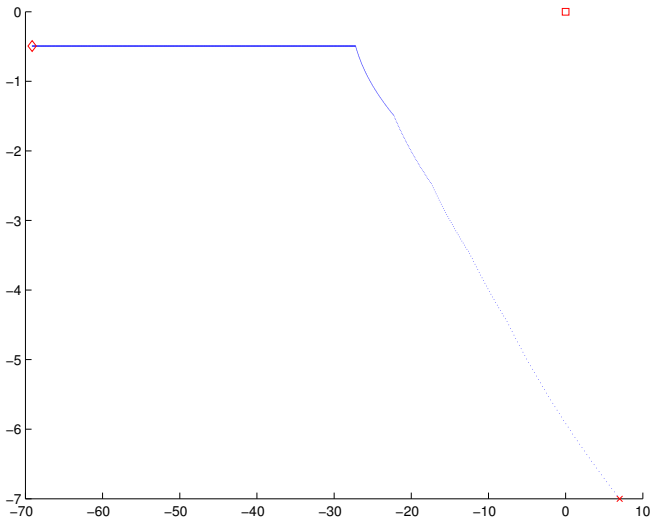

Fig. 30. System trajectory with initial state $[7 \quad -7]^T$.

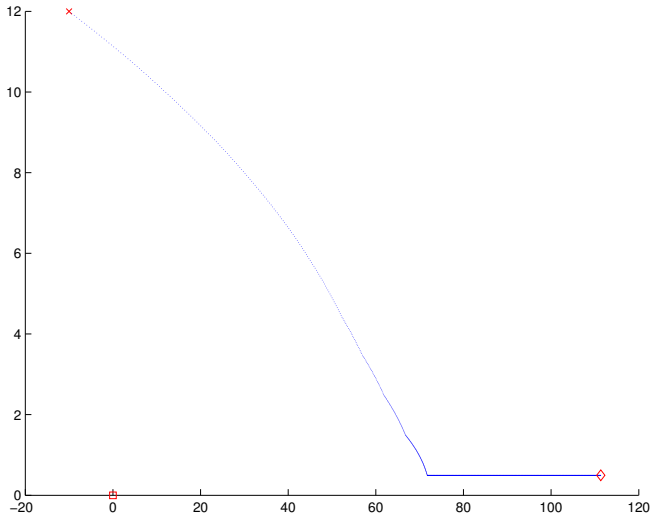

Fig. 31. System trajectory with initial state $[-10 \quad 12]^T$.

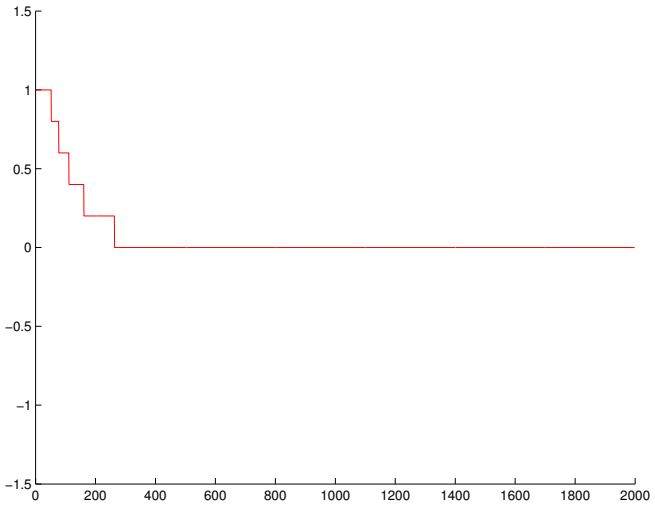

Fig. 32. Control input with initial state $[7 \quad -7]^T$.

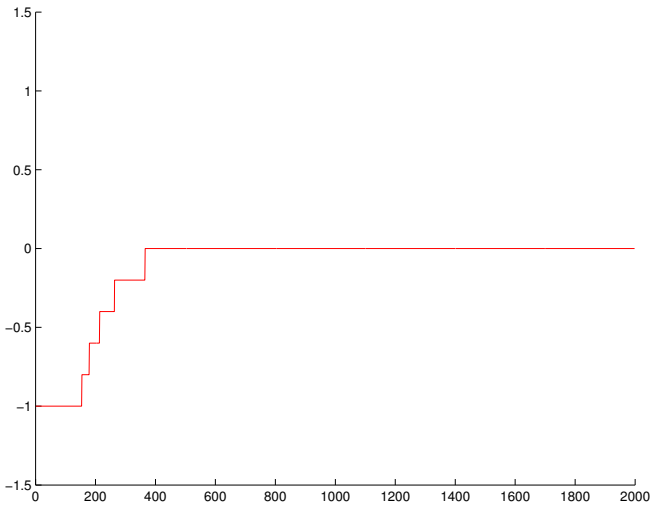

Fig. 33. Control input with initial state $[-10 \quad 12]^T$.

function $\|x + \alpha\dot{x}\|$ is always positive. This can be a reason for the control u to not switch signs. A modification of the cost function, so that it can take both negative and positive values, should be able to rectify this characteristic. We can try many different cost function alternatives that will take both positive and negative values. One such cost function is simply the product of elements of the vector $(x + \alpha\dot{x})$. This can be represented symbolically as $\prod_{i=1}^{n}(x_i + \alpha\dot{x}_i)$.

B. Hypothesis 7:

Now, the control scheme can be modified to incorporate the modified cost function. The modified control algorithm will seek to minimize the quantity $\prod_{i=1}^{n}(x_i + \alpha\dot{x}_i)$ at a discrete number of points on the state-space trajectory of the system, over the input set Ω.

At any point on state trajectory, control input can be calculated as

$$u = \left\{ u \middle| \min_{\Omega} \left[\prod_{i=1}^{n} (x_i + \alpha\dot{x}_i) \right] \right\}. \tag{4.3}$$

The controller thus developed is independent of the initial state. It is found that this control scheme successfully regulated not only the double integrator system, but also any general second-order stable system. In the following section we shall demonstrate the stabilizing nature of this control law.

1. Controller stability

Theorem: *If in a system*

$$\dot{x} = Ax + Bu, \tag{4.4}$$

where $x \in \Re^2$, the pair (A, B) is controllable and A has no eigenvalues with positive real parts, then the control law

$$u = \left\{ u \middle| \min_{\Omega} \left[\prod_{i=1}^{2} (x_i + \alpha \dot{x}_i) \right] \right\}, \tag{4.5}$$

globally asymptotically stabilizes the system.

Proof: Let $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ and $B = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$. This system can be transformed into a *controllable canonical* form using a similarity transformation T. We can always find a similarity transformation T, provided that the system is controllable. Say, the transformed state be q. Then, $q = T^{-1}x$. The transformed system is expressed as:

$$\dot{q} = A_C q + B_C u. \tag{4.6}$$

Here $A_C = TAT^{-1} = \begin{bmatrix} 0 & 1 \\ a_1 & a_2 \end{bmatrix}$ and $B_C = TB = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Longrightarrow \dot{q} = \begin{bmatrix} q_2 \\ a_1 q_1 + a_2 q_2 + u \end{bmatrix}$.

The value of control input is (Equation (4.3))

$$\begin{aligned} u &= \left\{ u \middle| \min_{\Omega} \left[\prod_{i=1}^{2} (x_i + \alpha \dot{x}_i) \right] \right\} \\ &= \left\{ u \middle| \min_{\Omega} \left[(q_1 + \alpha \dot{q}_1)(q_2 + \alpha \dot{q}_2) \right] \right\} \\ &= \left\{ u \middle| \min_{\Omega} \left[(q_1 + \alpha q_2)(q_2 + \alpha a_1 q_1 + \alpha a_2 q_2 + \alpha u) \right] \right\} \\ &= \left\{ u \middle| \min_{\Omega} \left[q_1 q_2 (1 + \alpha(a_1 + \alpha a_2)) + \alpha a_1 q_1^2 + \alpha(1 + \alpha a_2) + \alpha u (q_1 + \alpha q_2) \right] \right\} \end{aligned}$$

$\Longrightarrow$

$$u = -\text{sign}(q_1 + \alpha q_2). \tag{4.7}$$

We will employ the Lyapunov's stability theory to get conditions for stability [19],[20].

Lemma: Let $V(x)$ be continuously differentiable and positive definite function defined in $\Re^n$, with the property that $V(x) = 0$ if and only if $x = 0$. Then

- If $\frac{d[V(x)]}{dt}$ is negative semi-definite on $\Re^n$, the system is Lyapunov stable.
- If $\frac{d[V(x)]}{dt}$ is negative definite on $\Re^n$, the system is asymptotically stable.
- If $\frac{d[V(x)]}{dt}$ is positive definite on $\Re^n$, the system is unstable in both senses.

□

Therefore, it will suffice for us to compute the input u such that $\frac{d[V(x)]}{dt} \leq 0$ to ensure Lyapunov stability. The system being considered is semi-stable so it satisfies the matrix Lyapunov equation

$$\left(A_C^T P + P A_C\right) \leq -Q. \tag{4.8}$$

The matrix P is the positive semi-definite solution of the Lyapunov matrix equation and Q is any positive semi-definite matrix. Suppose the Lyapunov function is $V = q^T P q$, then

$$\begin{aligned} \dot{V} &= \dot{q}^T P q + q^T P \dot{q} \\ &= (q^T A_C^T + u^T B_C^T) P q + q^T P (A_C q + B_C u) \\ &= \underbrace{q^T (A_C^T P + P A_C) q}_{\leq 0} + 2 q^T P B u. \end{aligned} \tag{4.9}$$

Selecting $P = \begin{bmatrix} \lambda_1 & \alpha\lambda_2 \\ \alpha\lambda_2 & \alpha^2\lambda_2 \end{bmatrix}$, $\lambda_1 > \lambda_2 > 0$, we get:

$$
\begin{aligned}
2q^T PBu &= \begin{bmatrix} q_1 & q_2 \end{bmatrix} \begin{bmatrix} \lambda_1 & \alpha\lambda_2 \\ \alpha\lambda_2 & \alpha^2\lambda_2 \end{bmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \{-\text{sign}(q_1 + \alpha q_2)\} \\
&= -\alpha\lambda_2 \underbrace{(q_1 + \alpha q_2)\{\text{sign}(q_1 + \alpha q_2)\}}_{>0} \qquad (4.10) \\
&\Longrightarrow \\
2q^T PBu &< 0
\end{aligned}
$$

From Equations (4.9) and (4.10), $\dot{V} < 0$. Hence, global asymptotic stability proved.

As stated earlier, optimal-aim concept has been the inspiration behind the above development. Although, this algorithm has no obvious optimal-aim interpretation, in essence the control law is an aiming solution as it is based solely on the idea of directing the trajectory derivative in a direction that minimizes the quantity $\prod_{i=1}^{n}(x_i + \alpha\dot{x}_i)$.

2. Cost function and MPC analogy

Hypotheses 6 and 7 are inherently online and do not depend on quantities that have to be guessed. The corresponding control algorithms seeks to minimize the quantities $\|x + \alpha\dot{x}\|$ or $\prod_{i=1}^{n}(x_i + \alpha\dot{x}_i)$, respectively, at a number of discrete points along the trajectory over the input set. This can be likened to a moving horizon model predictive control (MPC) technique [21],[22] with both control horizon and the prediction horizon being equal to one time step T, and the norm $\|x + \alpha\dot{x}\|$ (or $\prod_{i=1}^{n}(x_i + \alpha\dot{x}_i)$) can be interpreted as a cost function for the optimization. This interpretation implies that this algorithm might share some of the advantages of the MPC method.

Chapter V presents the results corresponding to above control schemes when applied to different second order systems.

CHAPTER V

RESULTS AND DISCUSSION

A. Results

The results corresponding to control scheme developed in chapter IV are presented here. The results consist of plots of state-space trajectory and the control input when this control scheme is applied to the plant.

Figures 34 and 35 show the plots resulting from application of this control scheme to double integrator. The initial state is $x(0) = \begin{pmatrix} 10 \\ 15 \end{pmatrix}$, and $\alpha = 2$. In recording these results, the control set $\Omega = [-1, 1]$, was discretized as $\Omega = \{u_j\} = \{-1.0, -0.9, -0.8 \cdots 0 \cdots 0.8, 0.9, 1.0\}$. It is clear that the modified algorithm suc-

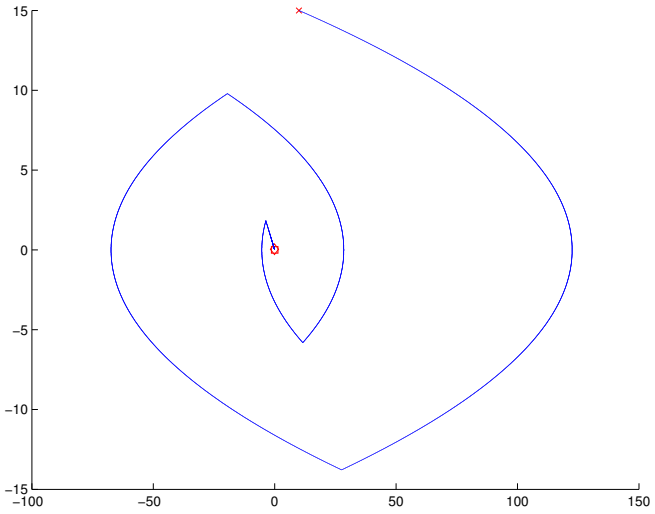

Fig. 34. State trajectory with the modified algorithm

ceeds in regulating the system. The state trajectory and the control plots display the switching behavior. The state trajectory also bears a close resemblance to the

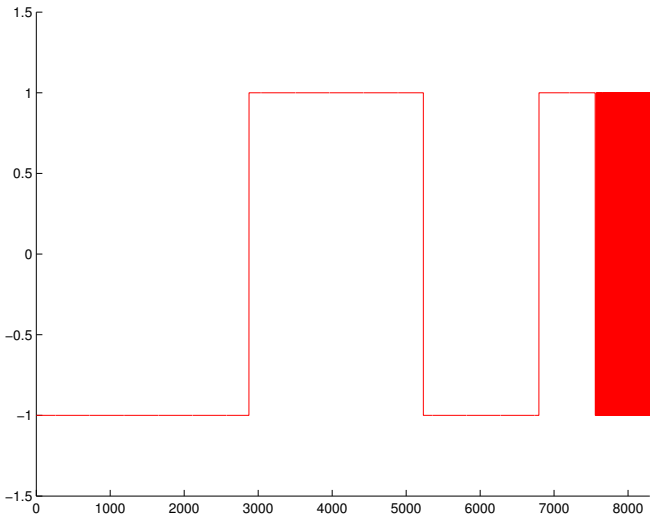

Fig. 35. Control input with the modified algorithm

time-optimal trajectory as well. In the control plot, we notice three distinct places where the control switching took place. The high density of the control plot towards the end denotes very rapid switching of the control input.

We can observe the effect of changing α, in the following plots, the initial state is the same as in the above figures. $\alpha = 6$ for Figures 36 and 37. $\alpha = 20$ for Figures 38 and 39. As is easily observed, α affects the results to a great extent and we cause the system to converge slower or faster by varying the value of α. High values of α require very fast input switching as the state approaches the origin and may not be desirable for that reason.

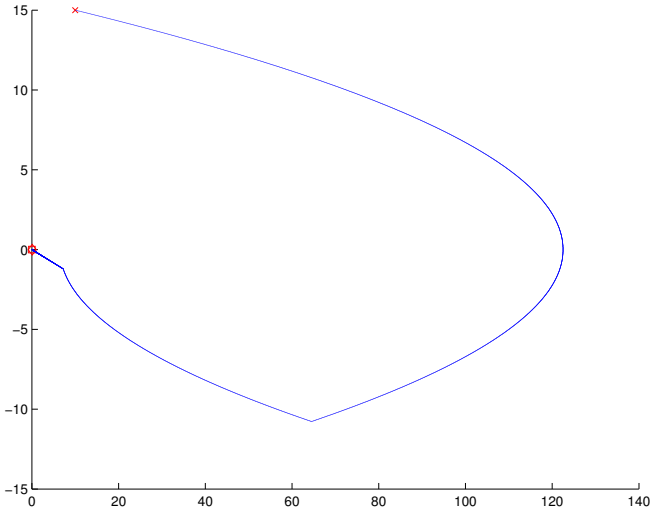

Fig. 36. State trajectory, $\alpha = 6$

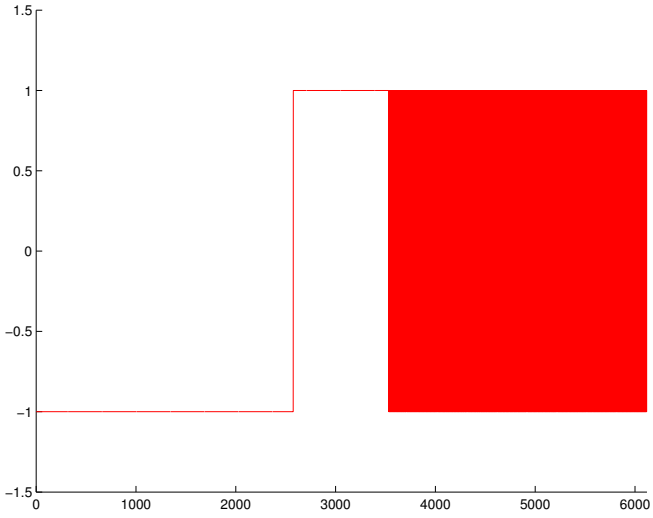

Fig. 37. Control input, $\alpha = 6$

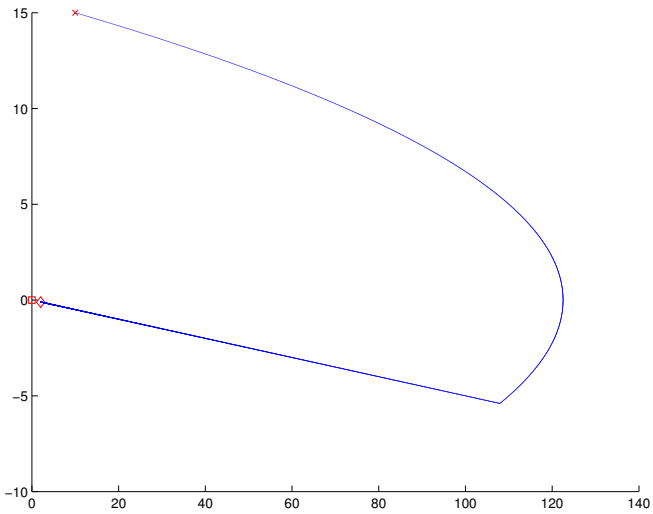

Fig. 38. State trajectory, $\alpha = 20$

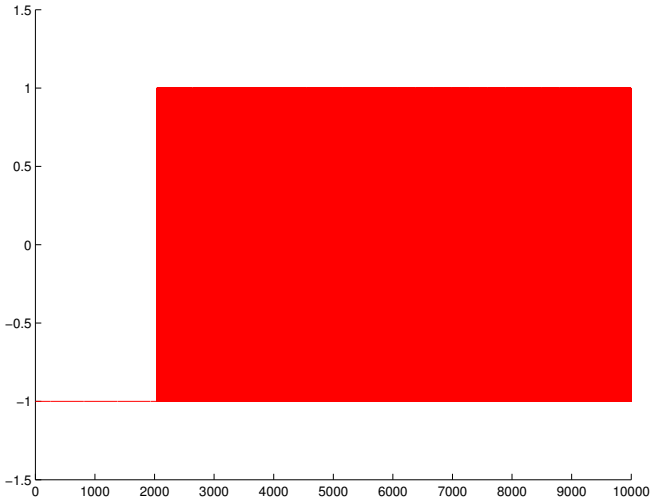

Fig. 39. Control input, $\alpha = 20$

As a proof of principle, we now apply the same to other second order systems like the harmonic oscillator and the 'two time constant equation' [11]. The harmonic oscillator is represented by the following equation:

$$\dot{x} = \begin{bmatrix} 0 & 2 \\ -2 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{5.1}$$

The simulation results for the harmonic oscillator are shown in Figures 40 and 41.

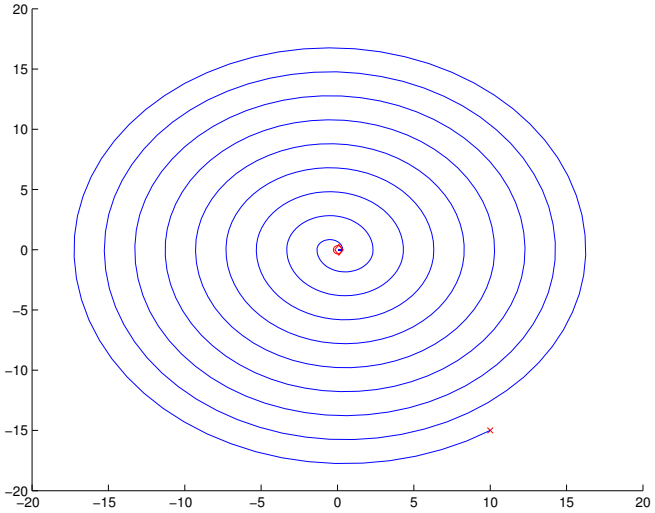

Fig. 40. State trajectory of a harmonic oscillator.

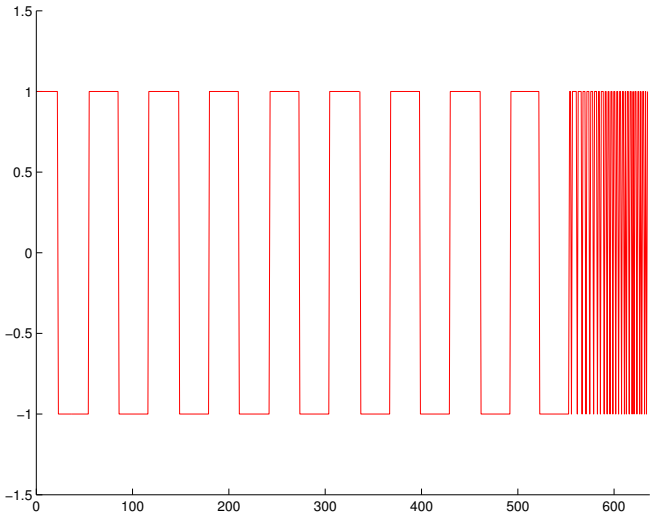

Fig. 41. Control input for a harmonic oscillator.

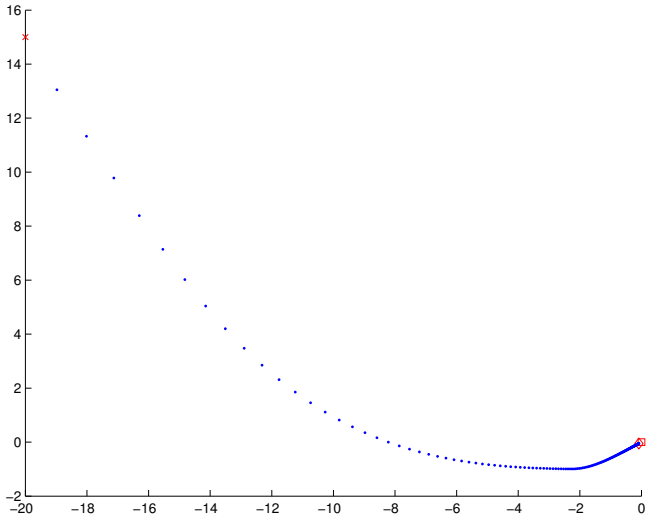

Fig. 42. State trajectory of a 'two time constant equation'.

The 'two time constant equation' [11] is represented as:

$$\dot{x} = \begin{bmatrix} -1 & 0 \\ 0 & -3 \end{bmatrix} x + \begin{bmatrix} 1 \\ 3 \end{bmatrix} u \tag{5.2}$$

The state trajectory and the control input plots for the above are shown in Figures 42 and 43.

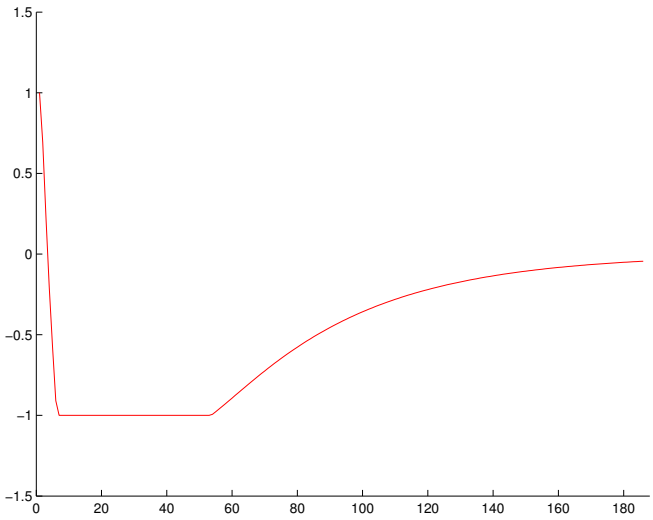

Fig. 43. Control input for a 'two time constant equation'.

B. Discussion

Hypotheses 1 through 5 focus on reproducing the exact time-optimal solution. In the course of this development it has been observed that prior knowledge of the costate variables is essential to reproduce the time-optimal solution. So, whereas our concern was to avoid solving the TPBVP, the focus on reproducing the exact time-optimal solution, and hence the dependence on costate variables made the TPBVP inevitable. Thus, none of the algorithms corresponding to hypotheses 1 through 5 are online and all involve off-line simulations to lock on to the correct values of costate variables.

Hypotheses 6 and 7 represent a departure from this focus on the exact time-optimal solution, and result in fully online controllers. Simulation results of different systems show that the control scheme of hypothesis 7 satisfies the regulation specification. The control signal is saturated most of the time and displays switching behavior. This is very close to the time-optimal solution. Hence, control scheme corresponding

to hypothesis 7 is our desired algorithm. This algorithm is simple but very powerful as it can globally stabilize not only asymptotically null-controllable systems but also systems having marginally stable modes. Double integrator is a system having two marginally stable modes and it is stabilized by this algorithm.

Most of the control input plots generated on applying this control scheme to different systems show a marked tendency to switch very fast in certain regions. This could be an issue for concern. Extremely fast switching in general is not desirable. Most of the physical systems do not exhibit the fast response times that are apparently required.

The proof of stability of this algorithm for second order systems is supplied, but as the control law (Equation (4.3)) becomes highly nonlinear for higher order systems, the stability properties of this algorithm with higher order systems are not so straightforward to characterize.

CHAPTER VI

CONCLUSIONS AND FUTURE WORK

This thesis describes the study of mainly second order systems having bounded inputs. Aiming methods were investigated and shown to be capable of generating globally asymptotically stabilizing online controls for null-controllable systems of any order, through the selection of aim states in a special manner. This suggested that aiming methods may prove to be good candidates for generating time-optimal controls online.

Initially the work was focussed on reproducing exact time-optimal solutions using purely state-space geometric method, but this task was not accomplished. Problem definition was then modified to generation of an online control scheme that approximates the time-optimal solution.

A novel regulation algorithm was developed for low order systems with inputs constraints. It is essentially a modification of optimal-aim concept where aim directions are selected in a special way to generate time-optimal control approximately.

The resulting control scheme is totally online in implementation and globally stabilizing for second order systems. This controller can have applications in the process industry where systems operate at equilibrium states and regulators are required to maintain equilibrium.

A. Future Work

One obvious direction the future work can explore is the successful application of the control schemes developed here to unstable systems.

The research can be continued to extend the results to higher order systems. The aiming solutions are apply to higher order systems as well but the stabilization achieved by control scheme corresponding to hypothesis 7 is only restricted to second

order systems.

The analogy of hypotheses 7 with model predictive control (MPC) was mentioned in chapter V. This can be investigated further and may lead to contributions to both MPC theory as well as regulators for low order plants.

REFERENCES

[1] W. J. Palm III, *Modeling, Analysis and Control of Dynamic Systems*, John Wiley & Sons, Inc., New York, 2nd edition, 1998.

[2] P. Kapasouris, M. Athans, and G. Stein, "Design of feedback control systems for stable plants with saturating actuators," in *Proc. 27th IEEE Conference on Decision and Control*, Austin, Texas, December 1988, pp. 469–479.

[3] E. D. Sontag and H. J. Sussmann, "Nonlinear output feedback design for linear systems with saturating controls," in *Proc. 29th IEEE Conference on Decision and Control*, Honolulu, Hawaii, December 1990, pp. 3414–3416.

[4] Z. Lin and A. Saberi, "Semi-global exponential stabilization of linear systems subject to "input saturation" via linear feedbacks," *Systems & Control Letters*, vol. 21, no. 3, pp. 225–239, 1993.

[5] H. J. Sussmann and Y. Yang, "On the stabilizability of multiple integrators by means of bounded feedback control," Sycon-91-01, Rutgers Center for Systems and Control, New Jersey, 1991.

[6] F. Blanchini and S. Miani, "Constrained stabilization of continuous-time linear systems," *Systems & Control Letters*, vol. 28, no. 2, pp. 95–102, 1996.

[7] D. Henrion, S. Tarbouriech, and V. Kučera, "Control of linear systems subject to input constraints: a polynomial approach," *Automatica*, vol. 37, no. 4, pp. 597–604, 2001.

[8] N. Kapoor and P. Daotidis, "Stabilization of systems with input constraints," *International Journal of Control*, vol. 66, no. 5, pp. 653–675, 1997.

[9] H. J. Sussmann, E. D. Sontag, and Y. Yang, "A general result on the stabilization of linear systems using bounded controls," *IEEE Transactions on Automatic Control*, vol. 39, no. 12, pp. 2411–2425, 1994.

[10] R. D. Barnard, "An optimal-aim control strategy for nonlinear regulation systems," *IEEE Transactions on Automatic Control*, vol. 20, no. 2, pp. 200–208, 1975.

[11] M. Athans and P. L. Falb, *Optimal Control - An Introduction to the Theory and Its Applications*, McGraw-Hill Book Company, New York, 1966.

[12] T. Hu, Z. Lin, and L. Qiu, "An explicit description of null controllable regions of linear systems with saturating actuators," *Systems and Control Letters*, vol. 47, no. 1, pp. 65–78, 2002.

[13] R. D. Barnard, "Optimal-aim control strategies applied to large-scale nonlinear regulation and tracking systems," *IEEE Transactions on Circuits and Systems*, vol. 23, no. 12, pp. 800–806, 1976.

[14] R. D. Barnard, "Optimal-proximity controls," *IEEE Transactions on Automatic Control*, vol. 23, no. 4, pp. 753–755, 1978.

[15] R. D. Barnard, "Continuous-time implementation of optimal-aim controls," *IEEE Transactions on Automatic Control*, vol. 21, no. 3, pp. 432–434, 1976.

[16] M. Athanassiades and J. M. Smith, "Theory and design of high-order bang-bang control systems," *IRE Transactions on Automatic Control*, vol. 6, no. 2, pp. 125–134, 1961.

[17] H. K. Knudsen, "An iterative procedure for computing time-optimal controls," *IEEE Transactions on Automatic Control*, vol. 9, no. 1, pp. 23–30, 1964.

[18] N. Chaiyaratana and A. M. S. Zalzala, "Hybridisation of neural networks and genetic algorithms for time-optimal control," in *Proc. 1999 Congress on Evolutionary Computation*, Washington, DC, July 1999, vol. 1, pp. 389–396.

[19] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, Inc., New Jersey, 2nd edition, 1996.

[20] J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, Inc., New Jersey, 1991.

[21] J. B. Rawlings, "Tutorial: Model predictive control technology," in *Proc. 1999 American Control Conference*, San Diego, California, June 1999, vol. 1, pp. 662–676.

[22] M. Morari and J. H. Lee, "Model predictive control: past, present and future," *Computers and Chemical Engineering*, vol. 23, no. 5, pp. 667–682, 1999.

VITA

Sumit Arora was born on July 9, 1977. He received his Bachelor of Engineering degree in Mechanical Engineering from Delhi University at Delhi College of Engineering, Delhi, India in 1998. He worked with Engineers India Limited, New Delhi from August 1998 to July 2000, where he was involved in detailed engineering design of process plant piping systems. He attended Texas A&M University from August 2000 to December 2003, where he received his Master of Science degree in Mechanical Engineering. He can be reached at 1-C Navkala Apartments, 14 I. P. Extension, Delhi 110092, India. E-mail: for_sumit@yahoo.com