Modeling and Optimal Control of Tower Crane Motions

by

Ali R. Golafshani

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical Engineering

Waterloo, Ontario, Canada, 1999

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

The growing importance of tower cranes is apparent with their more widespread use on construction projects. The automation of tower crane operations is motivated by several factors, such as economic priorities, safety, reliability and speed. The research in this thesis investigates the application of optimal trajectories for tower cranes to improve the performance of their operations.

A dynamic study of a fixed-boom tower crane is initiated. Employing the Lagrangian method, a simplified mathematical description governing crane motion in state-space form is developed. To achieve fast operation with small load swing, the crane optimization problem is formulated as a special case of Lagrange optimal problem with inequality constraints on controls and terminal states. This is further extended to allow free end time and path (trajectory) constraints.

Due to computational difficultlies risen in finding optimal load trajectories for general motions using conventional optimization methods, an iterative algorithm is proposed in order to speed up the computation of optimal solutions. The algorithm is based on known second-order methods, which have been adapted and customized here for application to the required crane optimizations.

Finally, the proposed algorithm is successfully tested on the crane optimization problem to find optimal load transfers in two different cases. Comparing with typical load transfers of a conventional tower crane, the optimal load transfers offer significant time saving with limited swing and smooth motions.

Acknowledgements

First, I wish to express my appreciation to my supervisor Dr. J. D. Aplevich. His support and guidance helped me to overcome many difficulties in my work. It has been a privilege for me to be his student.

Thanks are also extended to Professors M. M. Bayoumi, G. Glinka, A. Vannelli and W. J. Wilson for their useful comments and suggestions.

I am deeply thankful to my parents, who brought me up with true love, perfect care and great expectation. Finally, I would like especially to thank my wife for her generous support, her mature understanding and her amazing patience during past years.

To my teachers.

No one perhaps deserves more than them.

Contents

List of Figures

List of Symbols and Units

$\square$	End of a proof.
$\boldsymbol{A}, \boldsymbol{B}, \ldots$	Capital boldface letters denote matrices.
$\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\beta}, \ldots$	Lowercase boldface letters denote (column) vectors. $\mathbf{F}$, $\mathbf{T}$ and $\mathbf{f}$ are also used in Figure 2.1 to denote force and torque vectors.
$\mathbf{0}$	Zero (column) vector.
$\boldsymbol{I}$	Identity matrix.
$x, M, \beta, \ldots$	Non-boldface letters (except capital Greek type) denote scalar-valued functions or scalars.
$\mathcal{Z}$	z-transform.
z	z-transform operator.
s	Laplace transform operator.
$\times$	Cross product if it applies to vectors. It is Cartesian product if applies to sets.
$u^{(k)}, \alpha^{(k)}, \ldots$	Variables evaluated at the k-th iteration. If the braces are removed from the superscripts, they just imply powered variables, such as u^k, α^2.
$\boldsymbol{v}^T, \boldsymbol{A}^T, \ldots$	Transpose of a vector or a matrix.

x_n	The n-th component of the vector $\boldsymbol{x}$.
$\boldsymbol{x}_{m:n}$	A vector formed by the $n-m+1$ components of $\boldsymbol{x}$, from x_m to x_n.
$\mathcal{U}, \Pi, \mathcal{T}, \ldots$	Capital Greek and script (calligraphic) letters denote sets.
$\Re$	The field of real numbers.
$\lvert\ \rvert$	Absolute value.
$\lVert\ \rVert$	Norm.
$\triangleq$	Equal by definition.
$\approx$	Approximately equal.
$\forall$	For all: for every.
$\in$	Belongs to.
$\sum$	Summation.
$\prod$	Product.
$\{e \mid P\}$	Set of all e having property P.
$[a, b]$	$\{x \in \Re \mid a \leq x \leq b\}$
$(a, b]$	$\{x \in \Re \mid a < x \leq b\}$
$[a, b)$	$\{x \in \Re \mid a \leq x < b\}$
max	Maximum.
min	Minimum.

$\arg\min$	Minimizing argument.
$\dot{\boldsymbol{x}} \triangleq \frac{d}{dt}\boldsymbol{x}$	Time derivative of a function.
$f_x \triangleq \frac{\partial}{\partial x} f$	First partial derivative.
$f_{xu} \triangleq \frac{\partial}{\partial u}\left(\frac{\partial f}{\partial x}\right)$	Second partial derivative.
$H_{\boldsymbol{x}}, H_{\boldsymbol{u}}, \ldots$	Gradient (row) vectors.
$H_{\boldsymbol{xx}}, H_{\boldsymbol{xu}}, \ldots$	Hessian matrices.
$\boldsymbol{f_x}, \boldsymbol{f_u}, \ldots$	Jacobian matrices.
m	Meter.
s	Second.
rad	Radian.
o	degree.
kg	Kilogram.
t	tonne.

Chapter 1

Introduction

Tower cranes are widely used for erection and material handling in dam, power plant and tall building construction projects. They are also very suitable for work in crowded work sites where access by mobile cranes is restricted. Among the different types, the fixed-boom tower crane (Figure 1.1), with an arm extended horizontally when it moves objects, is the subject of this work.

A tower crane is initially assembled in a construction site and anchored in the ground by a concrete foundation. As the building rises beneath it, the crane must rise in parallel, and will eventually get most of its support from several floors of the building instead of the concrete foundation in the ground. More on the tower crane structure is given in the next section.

Since mid 80's there has been a tendency toward automation and robotization in the domain of heavy construction and lifting equipment. Despite much work done on the automation of other kinds of cranes, little work has been recorded on the automation of tower cranes (see section 1.5 for detailed literature review). Economic priorities along with safety, reliability, precision and speed are among many reasons

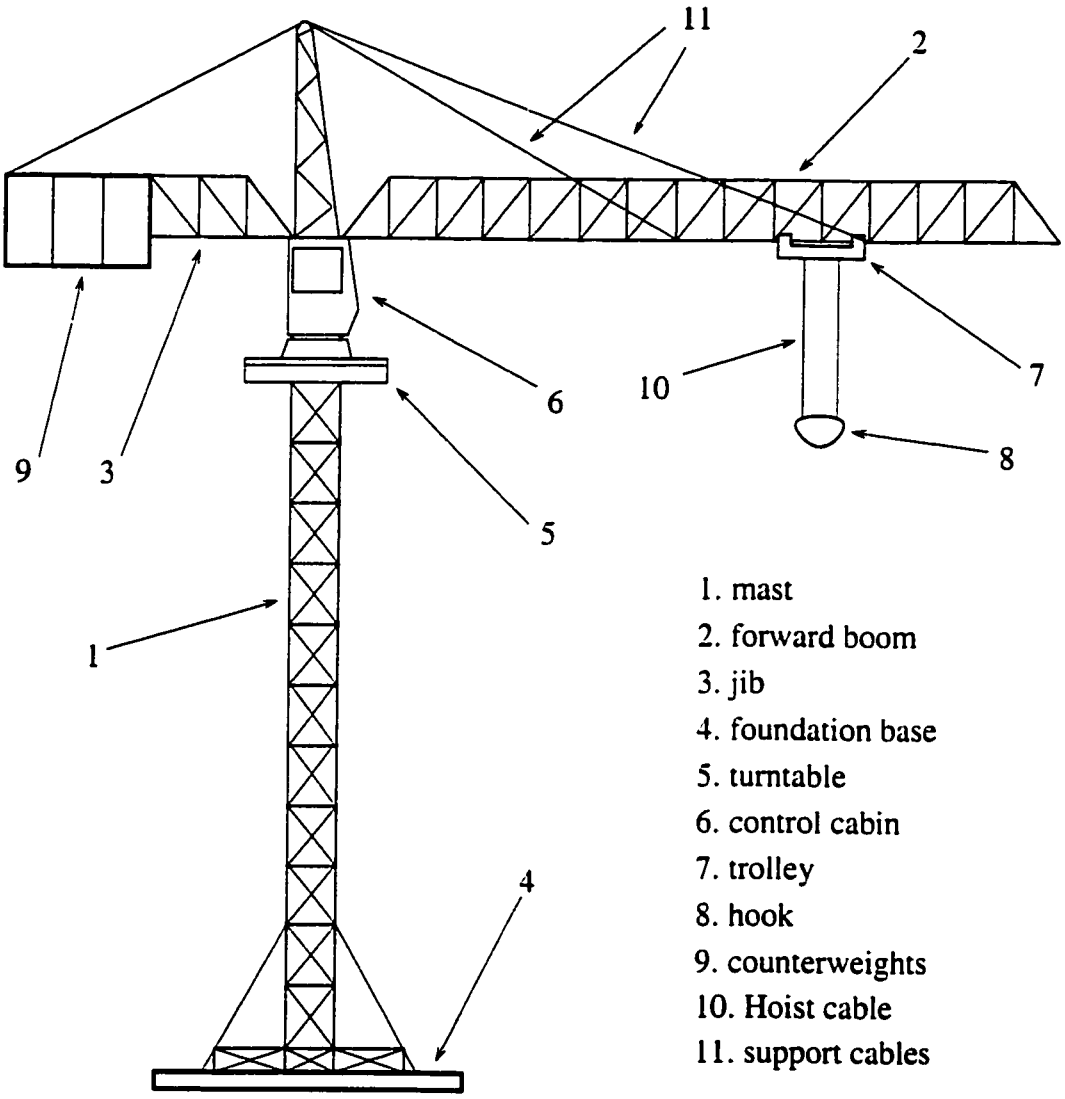


Figure 1.1: Tower crane with fixed boom

why the automation of tower crane operation is needed. These factors are explained in section 1.4.

An automatic control study of tower cranes starts from modeling. It is therefore one of the objectives of this thesis to establish a simplified mathematical representation of the crane motion. Then, the main goal of this work is to investigate the optimal trajectories of the crane for fast operation within operational limits. Since the crane model is nonlinear and coupled, finding optimal load trajectories for general motions by conventional optimization methods appeared to be computationally difficult [1], [2]. Therefore, an optimization algorithm is developed to compute optimal solutions of desired load displacements. Because of lack of a crane prototype, a simulation/animation program, linked with SIMULINK, has been coded in MATLAB to demonstrate the three dimensional operations of tower cranes.

Although today *automated tower cranes* (or *robot tower cranes*) are not in use, it is more than probable that they will be a practical operating reality in the near future. A robot crane is different from a remotely controlled crane as it is known today. All movements of an automated tower crane can be programmed in a computer device, and it only requires a supervisor rather than an operator.

In the context of this thesis, the word "crane" refers to tower crane.

1.1 Crane structure

The tower crane shown in Figure 1.1 comprises a vertical standing lattice-frame *mast* which projects a horizontal lattice-frame member in two parts. The longer section, called the *boom*, carries a *trolley* traveling along its length. On the opposite

side of the mast, a shorter part called the *jib* supports heavy slabs of concrete as counterweights to keep the crane in equilibrium. For example, on a tower crane of 70 t weight and 40 m tall, there are six blocks of concrete, each weighting 2.5 t, at the counterweight platform. However where space between buildings does not allow the jib movements, the forward boom can be stabilized by counterweights at the base or operating platform of the crane. Without a counterweight the crane will tip over if it picks up heavy loads.

The jib area also houses the hoist motor (winch) which is used to lift loads by a *hook* and a *hoist cable* passed through the trolley. The hoist cable begins in the motor area and passes along the boom to the trolley. At its end is the hook holding a load (See Figure 1.2).

The mast and the boom meet each other at the top of a *control cabin* located at heights from about 15 m to more than 80 m. This is the room where the crane operator sits and controls all of the crane movements. In modern tower cranes the control cabin is equipped with two main levers, called *joysticks*, and an easy-to-read display that updates the crane driver continuously on all relevant data such as load weight, trolley distance, slew angle and height under hook [3],[4]. The joysticks are moved to make the crane's boom rotate, to move the trolley along the boom, and to raise or lower the hook. Whenever the working site is not clear or the weather reduces visibility, the crane operation may be controlled by a remote control device on the ground.

There is a *turntable* on the main mast that allows a 360 degree rotation of all of the horizontal moving parts and the lifting parts of the tower crane including the control cabin. The slewing mechanism is mounted on the turntable.

Both sides of the boom are supported by cables from the top of the tower as

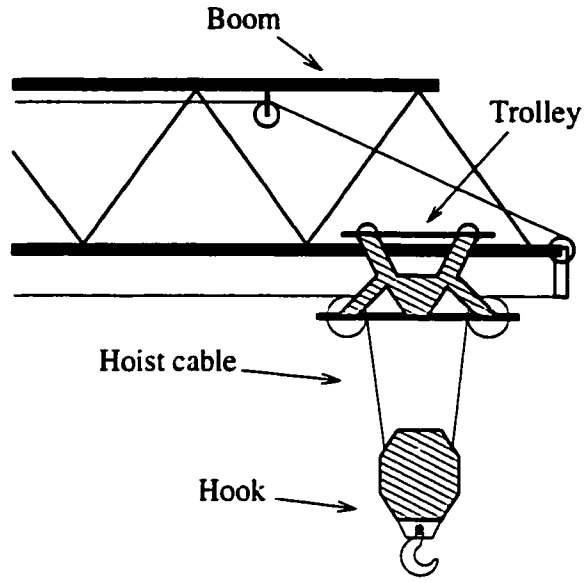


Figure 1.2: Tower crane: trolley-hook mechanism

shown in Figure 1.1. The horizontal-boom length may be longer than the mast height. The jib is typically 30-40% as long as the forward boom. All forces, including the weight of the crane and load, are transferred through the mast to a heavy foundation base [5].

1.2 Crane operation and limitations

The maximum lift capacity depends on how far the load is from the center of the mast. Information concerning this relationship between the maximum lifting capacity and the load radius distance is given by the crane manufacturers. For example, a tower crane that carries up to 8 tonnes at a maximum hook reach of 20 meters can reach out no further than 40 meters while carrying a load of 4 tonnes. Recent tower cranes are equipped with computerized systems that shut them off automatically if they lift loads heavier than their limit values.

There are speed limits on a tower crane operations. A tower crane having a higher motor horsepower can achieve higher operating speeds. The maximum hoist speed depends on the load. Hoist-cable configuration is another factor affecting lifting speed. Tower cranes are usually rigged with a two-part cable (as shown in Figure 1.2) or a four-part cable. Four-part hoist cable configuration provides greater lifting capacity with less hoist speed [6].

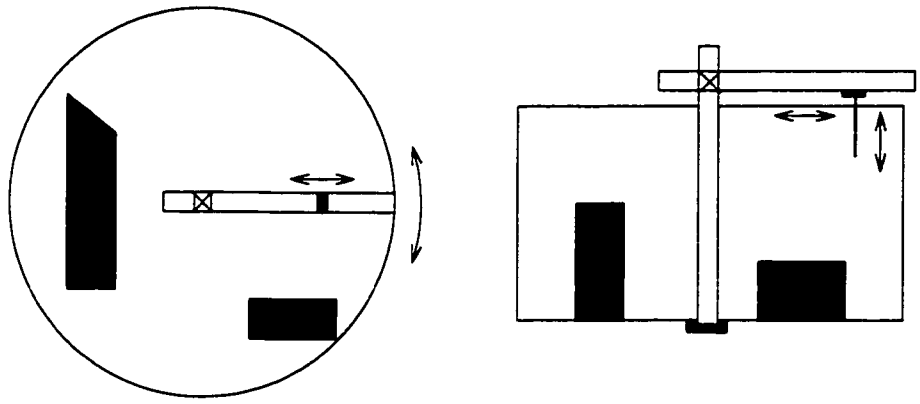
The speed of the trolley traveling along the boom can be varied up to 2 m/s. Also the boom can slew at a speed less than 0.15 rad/s [7]. This information is plotted on a chart that is unique for each crane style and model.

A tower crane operation is restricted within its *workspace* whose boundaries are defined by the following operational limits. The limits are automatically enforced by built-in devices in all tower cranes to provide safety during an operating cycle [6].

- **Trolley travel limit** The trolley travel is limited to predefined maximum in and maximum out positions. These limits prevent trolley from striking the mast and the boom head.

- **Hook height limit** There is a maximum height position for the load hook. implying a minimum length for the hoist cable. This limit provides sufficient striking distance between the hook and the trolley.

The maximum possible workspace for a tower crane is a cylinder with a radius almost equal to the length of the forward boom and the maximum height under the hook (Figure 1.3). The available workspace (operating area), however, excludes the "no-go" areas occupied by obstacles or buildings. Examples of no-go areas are shown in black in Figure 1.3. Today's modern tower cranes use this information to



(a) (b)

Figure 1.3: Tower crane workspace: **(a)** top-view **(b)** side-view

automatically shut down the operation if they are in danger of violating a no-go area [3].[8].

The tower crane workspace is accessible by:

1. moving the trolley back and forth along the forward boom (*traversing motion*),
2. lifting the hook, which is suspended from trolley, up and down (*lifting motion*) and
3. rotating the boom (*slewing motion*).

These functions, referred to as the *rigid-body motions* throughout this thesis, are controlled from the control cabin and may be supervised by persons on the ground communicating with the operator.

Finally, to ensure a safe load transfer, the following operating requirements are given by the crane manufacturers and handbooks. These requirements along with

other operational limits stated above will be considered in the derivation of an optimal trajectory in Chapter 4.

- Excessive load swing can cause the crane to collapse.
- Sudden speed change in the crane movements can produce forces well in excess of the weight being handled. These forces greatly increase the structural stress, and ultimately may lead to a structural failure.

1.3 Crane motors and power

Electric power in tower cranes is used for three functions. The first is for hoisting power. The second requirement for electric power is to operate the trolley or travel motor. The third power requirement is for the slewing motor. These power requirements for a large tower crane can all be provided, for example, by a 460 V, three-phase, 60 Hz source typically drawing current of 125 A up to 800 A.

Conventional tower cranes are equipped with motors that work at only two or three speeds. Change from one speed to another speed is usually provided by gear change which is not smooth. However, modern tower cranes use electronic controls to regulate motor voltages (for DC motors) or frequencies (for AC motors), providing infinitely variable speed without an electric or hydraulic retarder or coupling. This new generation of tower crane motors provides stepless speed and fine positioning, with even the smallest movements [8],[3]. This capability is crucial for automating tower cranes.

The regulator of a velocity controlled DC motor pulls in data from sensors that detect the speed requested by the operator and the actual motor speed. It uses this

information to precisely limit the output voltage and current to the motor so that the requested speed is achieved under any circumstances. But now, modern stepless frequency control drives are becoming more important to both crane manufacturers and customers. AC motors with frequency-variation control devices can provide continuous speed variation and progressive acceleration ramp. Stepless frequency control drives ensure rapid speed tracking performance and have become almost standard equipment on many tower crane models [9].

1.4 Crane automation

There are many answers to the obvious question of why the automation of tower crane operation is needed. They are as follows:

Safety. The automated tower crane will never over-torque its mast, exceed the safe-load limit, or slew faster than it should. Neither will it travel beyond its operating range. The prevention of over-stressing is the best possible guarantee against fatigue failure.

Reliability. Significant benefits from the elimination of over-stressing, and easy diagnosis and repair of electronic faults are expected. The repair job will be simple and fast, as circuit boards can be used in different cranes simply by reprogramming their microprocessors.

Efficiency. With the automation of tower crane operation the crane operation responsibility will be transferred to a person designated as supervisor of materials handling. The operator will program the crane's computer at the start of an operation, after planning which material would be needed where, and supervise its operation to make sure that orders are executed as planned. Manual assistance

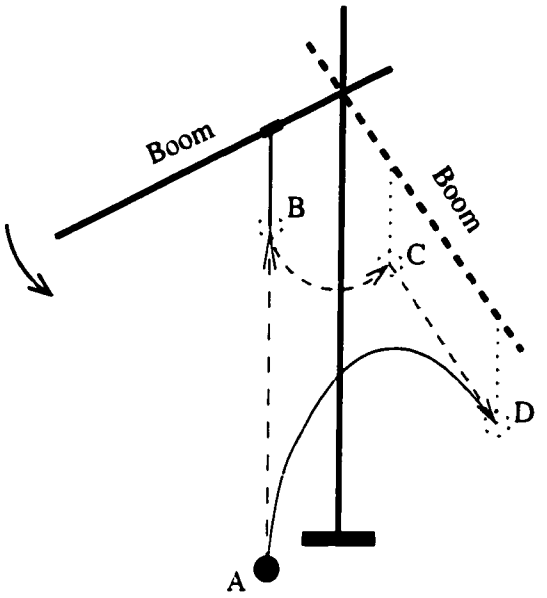


Figure 1.4: Two different load transfers between points A and D

may also be required for fine loading and unloading adjustments.

Speed. Even today's modern tower cranes are operated by a person who controls every function of the crane. A crane operator who wants more reliable control of the load sway motion uses only one of the crane functions (lifting, slewing and traversing) at a time, which is obviously inefficient and time consuming (e.g. dashed-line trajectory in Figure 1.4 typical of a load displacement from point A to point D). Experienced crane operators may use the crane functions simultaneously (usually load lifting while slewing or traversing). In the latter case, however, load movement has to be slow to avoid any undesired swing.

Another problem arises when the load arrives at its destination where it still swings. Thus the operator has to let all swings completely die out before unloading. The automated crane will take an optimal trajectory (not necessarily a straight path) between the initial point and the final point which requires the minimum

possible transfer time while satisfying all the operational constraints (e.g. solid line trajectory shown in Figure 1.4).

Precision. Exact load handling is very important in many projects that involve tower cranes. Current tower cranes are not built for precise load handling. Therefore, it takes a considerable time and effort to precisely transfer a load between two points. An automated crane can be programmed to perform such tasks quicker and with smoother motion.

Convenience. Even working with a sophisticated tower crane in a construction site. control of crane movement is difficult. Orders may come fast, either by radio or by hand signals from workers on the ground. A relatively simple operation may become complicated when two or more cranes work together in a construction site. It is expected that use of automated tower cranes will ease many of these difficulties.

Although the above reasons provide enough motivation toward crane automation. **economic priority** will perhaps be the biggest advantage. There will be plenty of room for cost cutting when an automated crane replaces an ordinary crane. The expected economic benefits resulting from the enhancement of crane performance will far exceed the cost of installing various control devices on a tower crane which may be worth more than half a million dollars.

Existing tower cranes can be furnished with extra devices for motion control. This may well be an economic advantage for existing modern tower cranes already computerized for monitoring purposes. A recent published paper by Yehiel Rosenfeld and Aviad Shapira explores the economic and technological feasibility of automation of existing tower cranes [10].

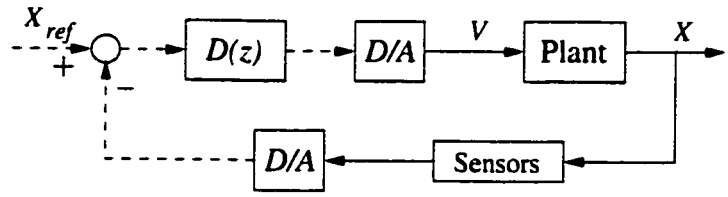


Figure 1.5: A feedback tracking control system for automated tower cranes

1.4.1 Control strategy

A feedback tracking control system such as one shown in Figure 1.5 which satisfies certain design specifications can be used in crane automation. The "Plant" in this system represents the crane dynamics and the driving mechanisms. As part of the driving mechanisms, the crane motors are coupled through gearing and cables to the moving parts. As mentioned in Section 1.3, regulated crane motors provide fast tracking performance of the requested speeds (V). V is defined by the crane operator in conventional cranes, or by the computer in an automated crane as shown in Figure 1.5. The digital controller $D(z)$ ensures that the system follows the desired (reference) trajectory ($\boldsymbol{x}_{ref}$).

The first step towards crane automation is to find the plant equations. Despite complexity of the crane dynamics due to nonlinearities, a simplified model of the plant may well work in the study of crane automation. Development of such a model is one of this work's objectives.

It is important to emphasize that, in Figure 1.5, the plant assumes the rigid-body velocities as the inputs. The output contains the rigid-body positions and the load swings with their corresponding speeds, labeled as X. The dynamics of the crane regulated motors can be ignored compared to that of the overall system. This is reasonable for a crane whose speed is determined by pendular motion of the suspended load. For a 20m hoist cable, for example, the period of natural frequency

is

$$T_n = \frac{2\pi}{\omega_n} = 2\pi\sqrt{\frac{l}{g}} = 9.0 \text{ sec.}$$

It is obvious that crane motor time constants are considerably smaller than 9.0 seconds. This along with the good tracking performance requirement for the crane motors justify ignoring the motor dynamics, as reflected in Figure 1.5.

An automated tower crane follows a given reference trajectory (path) X_{ref}. Perhaps the best reference trajectory is the optimal trajectory that provides fast operation with suppressed load swing between the initial (loading) point and the target (unloading) point, while satisfying all safety and operational limits. Working with an automated tower crane, both initial and target positions must be specified for the crane computer prior to an operation. This information along with operational and safety constraints are used to compute the optimal trajectory before the operation starts. This trajectory then becomes the reference path X_{ref} for the crane to follow. Specifically, the main goal of this work is to study this optimal trajectory.

The digital controller $D(z)$ uses the difference between the actual and reference trajectories to preserve satisfactory tracking performance. A robust closed-loop control system can overcome the effects of disturbances, such as wind, and model imperfection. This part is not covered in this work and is left for the future.

1.5 Literature review

With no previous academic work on the motion control of a fixed-boom tower crane, the related works include feasibility studies of crane automation and control of other

types of cranes such as overhead cranes, rotary cranes and jib cranes. Although there are some similarities among different types of cranes, there also exist many differences in their structures and behaviors that must be investigated individually.

Y. Rosenfeld and A. Shapira in [10] show that equipping existing tower cranes with electromechanical devices for motion control is technologically and economically feasible. While these improvements could in principle be applied to all types of cranes, they appear to be particularly suitable for tower cranes. As one application, Rosenfeld in a separate paper [11] describes the conversion of an existing full-scale 5-ton payload crane into a semi-automatic "handling robot". To do so, the crane was fitted with a programmable controller, speed regulators, encoders, several limit switches, a wireless remote control set, and a user-friendly MMI (man-machine-interface). He could demonstrate a 15-50% shortening of typical work cycles, high accuracy and repeatability, and a generally safer operation due to pre-tested paths and smoother movements with less sway and swing of the load [11].

The unloading of bulk materials from one point (e.g. ship) to another point (e.g. quayside) is usually done by an (overhead) *gantry crane*. Typically the crane (Figure 1.6) consists of a grab suspended from a moving trolley.

While the trolley moves along the rail, the suspended load can be lowered or raised by a hoist motor. Thus one can expect that the whole operation is done by a planar motion of the load hanging from the moving trolley. In the past 15 years, modeling and control study of an overhead crane has been the subject of many papers, including the following articles.

An adaptive controller for a simplified scale model of a gantry crane with a constant cable length is formulated in [12]. Then a new method of "reference model decomposition" as an extension of model reference adaptive control is presented.

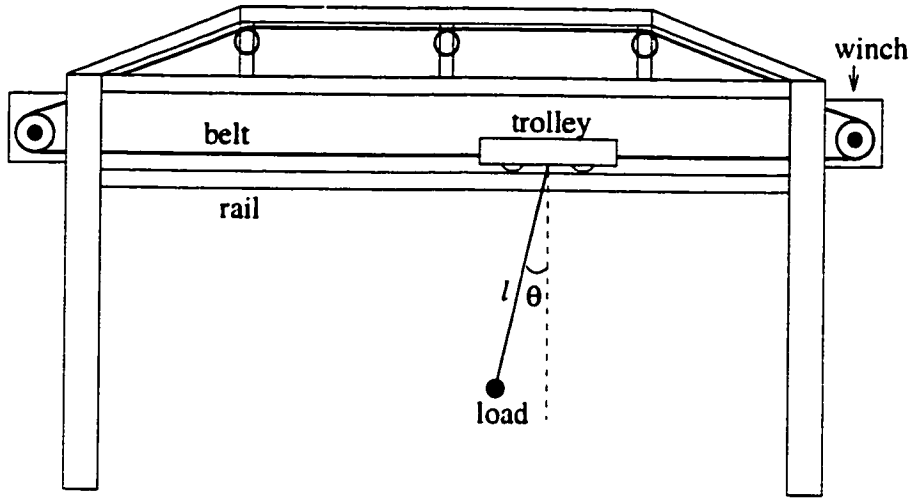


Figure 1.6: (Overhead) Gantry Crane

The proposed controller has not been applied to a real problem.

In [13] a more realistic model of a gantry crane is developed. The time-optimal control problem is analytically solved assuming both initial and terminal points to be at rest. The work is, however, limited to special cases where analytical solutions are available.

There exists another type of overhead crane, called a roof crane, studied in [14], [15] and [16]. The crane is widely used to transfer objects in factories and work places. Unlike the fixed rail of the gantry crane shown in Figure 1.6, the rail in a roof crane can move perpendicular to the page. As a result, the suspended load swings like a spherical pendulum which implies three-dimensional motion. Once nonlinear models are developed, different control strategies are applied to attenuate load swings. In [14] and [15] a feedback control based on a linearized model of the crane is developed and simulated on a digital computer. In [16] an optimal control strategy based on the derived nonlinear model of the crane is developed and tested.

A *rotary crane* is perhaps most similar to the tower crane among different types of construction cranes. A rotary crane makes three motions: rotation, load lifting and boom hoisting. In contrast to a tower crane with fixed boom, the boom of a rotary crane is not fixed due to lack of a trolley. In [17] and [18], Y. Sakawa and his students studied modeling and control of a rotary crane. After linearizing the nonlinear dynamical equations of the crane, the control scheme is set in two steps. First, an open-loop control is applied to transfer the system to a neighborhood of its equilibrium state. Then a feedback control moves the state of the system to the equilibrium state as quickly as possible.

Finally there is another type of crane similar to the tower crane except in the extending-boom section. The control study of this kind of crane was investigated in a paper published in 1989 [19]. Once the linearized model of the system is found, the control problem is solved by minimizing a cost function of second order. The goal is to suppress the load swing during the transfer process by extending or shrinking the length of the boom. Because of the difficulty caused by the required control constraints, a semi-optimal control strategy was proposed to make the control signal stay inside the given boundary condition. This work only deals with the especial case where the cable length and the boom angle are kept constant.

1.6 Industrial developments

With little academic work published on tower crane automation, there nevertheless have been signs of significant technological advances in the tower crane industry. In 1986, the technical director of Potain, the largest European tower crane manufacturer, claimed that a robot tower crane would be a practical operating reality in the near future [20]. This promises a future full of optimism in the field of crane

automation. Potain sees the automated crane as the logical extension of its several years of experience with the application of electronics to tower cranes. That is why it started making electronic control drives for all of the crane functions as the basic tools of any crane automation.

Liebherr's engineers are now developing an electronic module for automatic crane movements (AKB) to complete their modular crane control [21]. Computers are already used in modern tower cranes to monitor the exact position and speed of each moving part. Other crane manufacturers, such as Kroll, Comansa and MAN, are also gearing up in the same direction by offering fully progressive control drives on their entire range of tower cranes. Frequency-variation hoisting, trolley and slewing mechanisms have been added to the new models. Even existing models can be upgraded with new features due to customer demand and economic priority [9].

Almost every year new technological advances are introduced by the tower crane manufacturers in international exhibitions and trade fairs. One of the latest exhibitions was Bauma 98 in Germany [22]. In the evolution of tower crane technology, no one now believes that a tower crane is as primitive as 15 years ago.

1.7 Thesis outline

This thesis is organized into four chapters. The first chapter explains preliminaries which include an introduction to tower cranes, a comprehensive survey on previous work done on crane automation and their relation to the work of this thesis. It also establishes the needs for the proposed motion control and describes the technological developments occurring in the tower crane field. In particular, this chapter intends to portray a big picture of tower crane automation for its readers.

Chapter 2 introduces several assumptions in order to establish a simplified model of tower cranes. This model is then used to derive nonlinear differential equations of motion via the Lagrangian method. Using the rigid-body accelerations as the control inputs, a state-space representation of these equations is given. In addition, the driving forces and torque are found in order to compute the forces on the crane structure during a motion.

In Chapter 3 first the optimization criteria for a tower crane are proposed, and a cost function is defined to be minimized within the crane operational limits. Once the crane optimization is formulated, an optimization algorithm is developed to solve the constrained optimal problem. The optimization algorithm adopts an approach similar to the Han-Powell method in finite dimensional space, where the Hessian is replaced by a positive definite matrix updated by a certain rule.

Finally in Chapter 4 the optimal trajectories for two typical crane motions are illustrated. Then, the performance of these results will be measured as compared with some non-optimal trajectories, representative of a conventional manually controlled tower crane.

Chapter 5 summarizes the results and identifies some future research in this area.

1.8 Contributions

With the exception of this work, no previous academic work on motion control of tower cranes has been recorded. In the meantime, crane manufacturers have doubled their R&D efforts, upgrading existing crane models and introducing new models equipped with advanced technological innovations [9]. Due to high customer

demand it is now imperative for the crane manufacturers to keep up with new technologies. That is what makes this work new, significant and state of the art.

The contributions of this research can be summerized as follows:

Modeling

- The derivation of a simplified speed-controlled model for tower cranes. This mathematical model is given by a set of ordinary differential equations governing tower crane motions.
- The introduction of a state-space representation of the tower crane model. This contribution allows the application of state-space techniques for control design to this nonlinear multi-input system.
- The development of a crane simulator on MATLAB/SIMULINK to resemble the control cabin of a conventional tower crane. This application allows the simulation of 3D operations of a manually controlled tower crane in the absence of a crane prototype.

Optimization

- The formulation of a constrained crane optimization problem with a cost functional that preserves fast operation with small load swing.
- A modification to the second order method, with an approach similar to the Hann-Powell method in finite optimization, to solve continuous-time Lagrange optimal control problems.

- The implementation of the developed optimization algorithm for the crane optimization. This leads to computation of optimal trajectories for desired load displacements.

Chapter 2

System dynamics

The control study of any mechanical system starts from modeling. The mechanical model of a tower crane can be quite complex. However, a few assumptions are given to arrive at a simplified model of the crane in this Chapter.

In Section 2.1, the coordinate frames and variables are defined. Once the energy expressions have been formulated in Section 2.2, Hamilton's principle [23] is used to determine the Lagrange equations of crane motions.

The dynamics of the motor drives and industrial controllers can be very complex. As explained in Section 1.4, the motor dynamics is ignored in the equations of motion given in Section 2.3.

2.1 Definitions and assumptions

As shown in Figure 2.1 the crane model can be divided into two parts. The first part includes the slewing boom with the moving trolley. Shown in the "top" view of the boom (Figure 2.1(a)), the axis of rotation of the mast (tower) is perpendicular

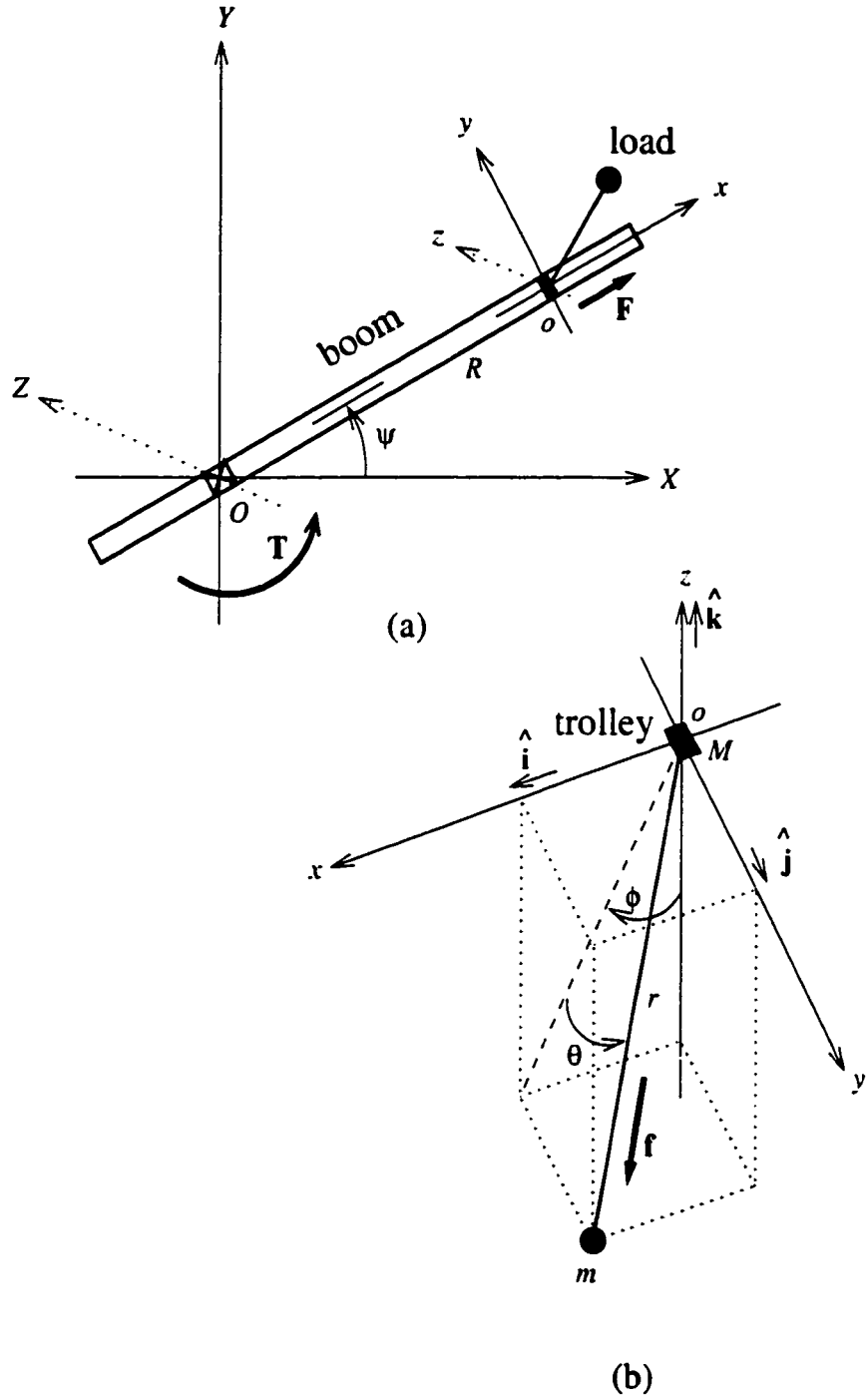


Figure 2.1: Crane model; **(a)** boom-trolley **(b)** trolley-load

to the page. The second part of the model involves a suspended load hanging from the trolley.

There are two coordinate frames used to represent the dynamics of the tower crane in Figure 2.1. The inertial frame of reference indicated by the XYZ coordinate frames is fixed to the ground. The xyz coordinate frame is a frame of reference that rotates with the boom and with origin coincident with the trolley. The x-axis of this moving frame is always collinear with the boom and the y-axis is normal to it.

A simplified model of the crane motions based on the following assumptions is developed in this work.

1. The crane members are rigid. The crane body consists of metal parts stiffly joined together which allows slight flexibility under normal working conditions.

2. Both the hook and the load together as well as the trolley are considered as point masses.

3. No friction is modeled. Friction, as an important characteristic of every highly-geared mechanism, must be considered in an enhanced model of the actual plant. Static and viscous frictions in the rigid-body mechanisms are present in a torque-controlled model of the crane studied in [2].

4. The hoist cable does not flex or stretch under load, but is assumed to be a rigid, twist-free massless rod. This is not an unusual assumption especially when heavy loads are moved.

5. The crane is driven by the motors that use external speed references as inputs, as shown in Figure 1.5.

There are five generalized coordinates (variables) used to describe the crane model in Figure 2.1. They are defined as follows:

R : distance between the trolley and the mast.

ψ : slewing angle of the boom measured from the X axis.

r : length of the cable connecting trolley to the load.

θ : angle between the suspended cable and its projection on the $x-z$ plane.

ϕ : angle between the z-axis and the projected line.

The first three variables are regarded as rigid body coordinates while the last two are seen as spherical coordinates. Note that the above variables are all functions of time. In future, the time argument, (t), will continue to be dropped for simplicity.

2.2 Energy expressions

The load position vector using the moving frame coordinates at o is

$$\boldsymbol{d}_m = (r\cos\theta\sin\phi)\,\hat{\boldsymbol{i}} + (r\sin\theta)\,\hat{\boldsymbol{j}} - (r\cos\theta\cos\phi)\,\hat{\boldsymbol{k}}\ . \tag{2.1}$$

The moving frame is rotating around the origin of the fixed frame with an angular velocity of $\boldsymbol{\omega} = \dot{\psi}\hat{\boldsymbol{k}}$. Therefore the velocity vector of the load is given by

$$\boldsymbol{v}_m = \boldsymbol{v}_o + \dot{\boldsymbol{d}}_m + \boldsymbol{\omega} \times \boldsymbol{d}_m \tag{2.2}$$

where $\boldsymbol{v}_o = \dot{R}\,\hat{\boldsymbol{i}} + R\dot{\psi}\,\hat{\boldsymbol{j}}$ is the trolley velocity vector and $\dot{\boldsymbol{d}}_m$ represents the relative velocity of the load (i.e. with respect to the moving frame). Employing the moving-frame unit vectors for simplicity, (2.2) can be expressed as

$$\boldsymbol{v}_m = (v_x)\,\hat{\boldsymbol{i}} + (v_y)\,\hat{\boldsymbol{j}} + (v_z)\,\hat{\boldsymbol{k}} \tag{2.3}$$

where

$$
\begin{aligned}
v_x &= \dot{R} + \dot{r}\cos\theta\sin\phi - r\dot{\theta}\sin\theta\sin\phi + r\dot{\phi}\cos\theta\cos\phi - r\dot{\psi}\sin\theta \\
v_y &= R\dot{\psi} + \dot{r}\sin\theta + r\dot{\theta}\cos\theta + r\dot{\psi}\cos\theta\sin\phi \\
v_z &= -\dot{r}\cos\theta\cos\phi + r\dot{\theta}\sin\theta\cos\phi + r\dot{\phi}\cos\theta\sin\phi \ .
\end{aligned}
$$

The total kinetic energy of the system (K) is equal to the sum of the kinetic energy of the load, the kinetic energy of the trolley and the kinetic energy of the boom. If M and m are the trolley and load masses, and I represents the moment of inertia of the boom about the center of mast (Z-axis), then the total kinetic energy is

$$
K = \frac{1}{2}m\left\langle \boldsymbol{v}_m, \boldsymbol{v}_m \right\rangle + \frac{1}{2}M\left(\dot{R}^2 + R^2\dot{\psi}^2\right) + \frac{1}{2}I\dot{\psi}^2. \tag{2.4}
$$

The potential energy of the system is due to the contribution of the suspended load which is

$$
P = -mgr\cos\theta\cos\phi \ . \tag{2.5}
$$

where g is the acceleration of gravity. The negative potential energy results from the choice of origin of the moving frame.

2.3 Lagrange equations

Using the Lagrangian approach, the equations of motions for the simplified model of the tower crane are derived from the kinetic and potential energy expressions (2.4) and (2.5). The Lagrangian function is defined to be

$$
L \triangleq K - P = \frac{1}{2}m\left\langle \boldsymbol{v}_m, \boldsymbol{v}_m \right\rangle + \frac{1}{2}M\left(\dot{R}^2 + R^2\dot{\psi}^2\right) + \frac{1}{2}I\dot{\psi}^2 + mgr\cos\theta\cos\phi. \tag{2.6}
$$

Lagrange equations are of the general form

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i$$

where q_i are the generalized coordinates and Q_i are the generalized forces or torques [23].

With no actuating force on the load, the Lagrange equations for the load swing motion are then given as

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = 0 \tag{2.7a}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}}\right) - \frac{\partial L}{\partial \phi} = 0 \tag{2.7b}$$

which follow

$$\begin{aligned} &\ddot{R}\sin\theta\sin\phi - \ddot{\psi}(R\cos\theta + r\sin\phi) - r\ddot{\theta} = \alpha \\ &(r\ddot{\psi}\sin\theta - \ddot{R})\cos\phi - r\ddot{\phi}\cos\theta = \beta \end{aligned} \tag{2.8}$$

where

$$\begin{aligned} \alpha &= 2\dot{R}\dot{\psi}\cos\theta + 2\dot{r}\dot{\psi}\sin\phi + 2\dot{r}\dot{\theta} + r\dot{\phi}^2\sin\theta\cos\theta - r\dot{\psi}^2\sin\theta\cos\theta\cos^2\phi \\ &\quad + 2r\dot{\psi}\dot{\phi}\cos^2\theta\cos\phi + R\dot{\psi}^2\sin\theta\sin\phi + g\sin\theta\cos\phi\ , \\ \beta &= 2\dot{r}\dot{\phi}\cos\theta - 2r\dot{\theta}\dot{\phi}\sin\theta - 2\dot{r}\dot{\psi}\sin\theta\cos\phi - 2r\dot{\theta}\dot{\psi}\cos\theta\cos\phi \\ &\quad - r\dot{\psi}^2\cos\theta\sin\phi\cos\phi - R\dot{\psi}^2\cos\phi + g\sin\phi\ . \end{aligned} \tag{2.9}$$

2.4 Choice of control inputs

The state-space model is widely used for analysis and design of a multi-input multi-output system (MIMO). A tower crane with three actuators is a MIMO system. The states of any lumped-parameter continuous-time dynamical system can be represented in state-space form by a set of first-order differential equations as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \tag{2.10}$$

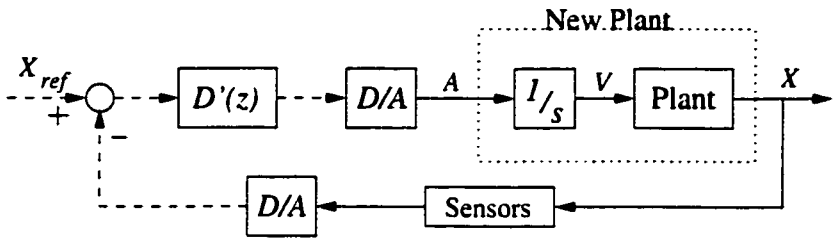


Figure 2.2: Feedback tracking control system of tower cranes

where $\boldsymbol{x}$ and $\boldsymbol{u}$ are the state and (control) input vectors respectively [24].

As shown in Figure 1.5, the plant inputs are velocities. They are $\dot{R}$, $\dot{\psi}$ and $\dot{r}$, the speeds of the rigid-body motions. These input variables and their derivatives $\ddot{R}$, $\ddot{\psi}$ and $\ddot{r}$, the rigid-body accelerations, appear in the load swing equations (2.8). Because the derivatives of the inputs are not allowed in the state-space equations (2.10) the rigid-body accelerations are defined as the control inputs, that is

$$\begin{cases} u_1(t) = \ddot{R}(t) \\ u_2(t) = \ddot{\psi}(t) \\ u_3(t) = \ddot{r}(t) \end{cases} \tag{2.11}$$

With this choice of input variables, the load swing equations (2.8) can now be represented in general state-space form of (2.10).

Above definition of the control inputs requires an adjustment to the closed-loop system shown in Figure 1.5, in which velocities are the plant inputs. Figure 2.2 offers an equivalent representation of the closed-loop system shown in Figure 1.5.

The new controller is

$$D'(z) = \mathcal{Z}[s]\ D(z)$$

where

$$\mathcal{Z}[s] = \frac{2}{T}\frac{z-1}{z+1}$$

using bilinear transformation. T is the sampling period and z is the z-transform operator. Once a digital controller $D'(z)$ is designed, the original controller of the system shown in Figure 1.5 is found by

$$D(z) = \frac{T}{2}\frac{z+1}{z-1} D'(z).$$

Here it is assumed that the digital controller is linear, otherwise the notation $D(z)$ would be meaningless.

2.5 Equations of motion

To derive the equations of motion, the following two assumptions are necessary.

$$\begin{cases} r & \neq & 0 \\ \theta & \neq & 90^o \end{cases}$$

The first condition is immediately known from the hook height limit given in Section 1.2. The second condition is justified under normal operation where the load swing is not large, i.e. $\theta \ll 90^o$.

Using the above assumptions, solutions of the equations (2.8) for $\ddot{\theta}$ and $\ddot{\phi}$ are

$$\ddot{\theta} = \frac{u_1}{r}\sin\theta\sin\phi - u_2(\frac{R}{r}\cos\theta + \sin\phi) - \frac{\alpha}{r} \quad (2.12a)$$

$$\ddot{\phi} = (u_2\sin\theta - \frac{u_1}{r})\frac{\cos\phi}{\cos\theta} - \frac{\beta}{r\cos\theta} \quad (2.12b)$$

where α and β are given in (2.9). The required Maple codes to obtain these equations are given in Appendix A. Equations (2.11) in conjunction with (2.12) form the following set of second-order differential equations describing the crane motions:

$$\ddot{R} = u_1$$

$$\begin{aligned}
\ddot{\psi} &= u_2 \\
\ddot{r} &= u_3 \\
\ddot{\theta} &= \frac{u_1}{r}\sin\theta\sin\phi - u_2(\frac{R}{r}\cos\theta + \sin\phi) - \frac{\alpha}{r} \\
\ddot{\phi} &= (u_2\sin\theta - \frac{u_1}{r})\frac{\cos\phi}{\cos\theta} - \frac{\beta}{r\cos\theta}
\end{aligned} \tag{2.13}$$

2.6 Driving forces and torque

Because tower cranes are used to hoist and move loads from one location to another, it is necessary to know the required forces and torques of the crane motors at any time during a motion. These are

$F(t)$: force generated by a travel motor with the direction shown in Figure 2.1(a),

$T(t)$: torque generated by a rotational motor as directed in Figure 2.1(a),

$f(t)$: force generated by a hoisting motor directed from trolley to load as shown in Figure 2.1(b).

Equations (2.7) are the Lagrange equations for the spherical coordinates θ and ϕ. Similarly, with F, T and f as the generalized forces, the Lagrange equations for the rigid body coordinates R, ψ and r are

$$\begin{aligned}
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{R}}\right) - \frac{\partial L}{\partial R} &= F \\
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\psi}}\right) - \frac{\partial L}{\partial \psi} &= T \\
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{r}}\right) - \frac{\partial L}{\partial r} &= -f
\end{aligned} \tag{2.14}$$

where L is defined by (2.6). Using equations of (2.12) to eliminate $\ddot{\theta}$ and $\ddot{\phi}$ in (2.14) results in

$$
\begin{aligned}
F &= Mu_1 + mu_1\cos^2\theta\sin^2\phi + mRu_2\cos\theta\sin\theta\sin\phi + mu_3\cos\theta\sin\phi \\
&\quad -mr\dot{\theta}^2\cos\theta\sin\phi + 2m\dot{R}\dot{\psi}\sin\theta\cos\theta\sin\phi + mr\dot{\psi}^2\cos^3\theta\cos^2\phi\sin\phi \\
&\quad -mr\dot{\phi}^2\cos^3\theta\sin\phi - mg\cos^2\theta\sin\phi\cos\phi + 2mr\dot{\psi}\dot{\phi}\sin\theta\cos^2\theta\sin\phi\cos\phi \\
&\quad -mr\dot{\psi}^2\cos\theta\sin\phi - mR\dot{\psi}^2\cos^2\theta\sin^2\phi - MR\dot{\psi}^2 - 2mr\dot{\psi}\dot{\theta}\cos\theta\sin^2\phi\,. \\
T &= mRu_1\sin\theta\cos\theta\sin\phi + \left(I + MR^2 + mR^2\sin^2\theta\right)u_2 + mRu_3\sin\theta \\
&\quad +2mRr\dot{\psi}\dot{\phi}\cos\theta\sin^2\theta\cos\phi - mgR\sin\theta\cos\theta\cos\phi - mRr\dot{\psi}^2\sin\theta \\
&\quad +2MR\dot{R}\dot{\psi} + 2mR\dot{R}\dot{\psi}\sin^2\theta - 2mrR\dot{\psi}\dot{\theta}\sin\theta\sin\phi - mrR\dot{\phi}^2\sin\theta\cos^2\theta \\
&\quad -mrR\dot{\theta}^2\sin\theta - mR^2\dot{\psi}^2\sin\theta\cos\theta\sin\phi + mrR\dot{\psi}^2\sin\theta\cos^2\theta\cos^2\phi\,. \\
f &= m\big(u_1\cos\theta\sin\phi + Ru_2\sin\theta + u_3 - r\dot{\theta}^2 - 2r\dot{\psi}\dot{\theta}\sin\phi + 2\dot{R}\dot{\psi}\sin\theta \\
&\quad -r\dot{\phi}^2\cos^2\theta - g\cos\theta\cos\phi + 2r\dot{\psi}\dot{\phi}\sin\theta\cos\theta\cos\phi + r\dot{\psi}^2\cos^2\theta\cos^2\phi \\
&\quad -r\dot{\psi}^2 - R\dot{\psi}^2\cos\theta\sin\phi\big)
\end{aligned}
$$

where u_1, u_2 and u_3 are the control inputs defined by (2.11).

A major application of the above results is to evaluate the applied forces to the crane structure during a motion. In an automated control this information can be used in the crane computer to prevent over-stressing and fatigue failure.

In addition, equations (2.14) joined with equations (2.7) describe the equations of motion for a tower crane with torque-controlled drives. In this case, the tower crane is driven by force and torque reference inputs generated by the crane motors. These inputs are F, T and f for travel, slew and hoist drives respectively.

2.7 Model simplification

Whereas the rigid-body positions may change drastically, the load swings remain small under normal working conditions. As stated in Section 1.2, large load swings must be avoided to ensure a safe load transfer. It is also assumed that the load swing is slow, especially for a long hoist cable. Therefore, the following nonlinear terms due to the swing angles (θ, ϕ) and their derivatives ($\dot{\theta}$, $\dot{\phi}$) are neglected from the system equations of (2.13).

$$-\dot{\phi}^2 \sin\theta\cos\phi,\ -\frac{R}{r}\dot{\psi}^2 \sin\theta\sin\phi,\ \frac{u_1}{r}\sin\theta\sin\phi,\ 2\dot{\theta}\dot{\phi}\frac{\sin\theta}{\cos\theta}$$

Using

$$\sin\theta \approx \theta,\ \sin\phi \approx \phi,\ \cos\theta \approx 1,\ \cos\phi \approx 1,$$

the resulting equations of motion are

$$\begin{aligned}
\ddot{R} &= u_1 \\
\ddot{\psi} &= u_2 \\
\ddot{r} &= u_3 \\
\ddot{\theta} &= -\frac{1}{r}\left(Ru_2 + ru_2\phi + 2\dot{R}\dot{\psi} + 2\dot{r}\dot{\psi}\phi + 2\dot{r}\dot{\theta} - r\dot{\psi}^2\theta + 2r\dot{\psi}\dot{\phi} + g\theta\right) \\
\ddot{\phi} &= \frac{1}{r}\left(ru_2\theta - u_1 - 2\dot{r}\dot{\phi} + 2\dot{r}\dot{\psi}\theta + 2r\dot{\theta}\dot{\psi} + r\dot{\psi}^2\phi + R\dot{\psi}^2 - g\phi\right)
\end{aligned} \tag{2.15}$$

Note that the above equations are still nonlinear, despite removing nonlinearities due to the load swing angles and their derivatives.

2.8 State-space model

The system of equations (2.15) is MIMO. Therefore, a state-space representation of the general form (2.10) is preferred in a control study of the system.

Introducing the state vector

$$\boldsymbol{x}(t) = \left(R(t), \psi(t), r(t), \theta(t), \phi(t), \dot{R}(t), \dot{\psi}(t), \dot{r}(t), \dot{\theta}(t), \dot{\phi}(t)\right)^T \quad (2.16)$$

and the control vector

$$\boldsymbol{u}(t) = (u_1(t), u_2(t), u_3(t))^T \quad (2.17)$$

the equations of motion (2.15) written in state-space form are

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \quad (2.18)$$

where

$$f_i = \begin{cases} x_{i+5} & \text{for } i = 1, 2, 3, 4, 5 \\ u_{i-5} & \text{for } i = 6, 7, 8 \end{cases}$$

$$\begin{aligned} f_9 = \; & x_4 x_7^2 - 2x_7 x_{10} - (x_5 + x_1/x_3)u_2 \\ & -(2x_6 x_7 + 2x_5 x_7 x_8 + 2x_8 x_9 + g x_4)/x_3 \\ f_{10} = \; & 2x_7 x_9 + x_5 x_7^2 - u_1/x_3 + x_4 u_2 \\ & +(2x_4 x_7 x_8 - 2x_8 x_{10} + x_1 x_7^2 - g x_5)/x_3 \; . \end{aligned}$$

In the above, $\boldsymbol{x} \in \mathcal{R}^{10}$ and $\boldsymbol{u} \in \mathcal{R}^3$. The domains of feasible states and controls are however tighter than $\mathcal{R}^{10}$ and $\mathcal{R}^3$ respectively, due to operational and safety limits. These domains will be defined in Section 3.2. In addition, $\boldsymbol{f} : \mathcal{R}^{10} \times \mathcal{R}^3 \rightarrow \mathcal{R}^{10}$ together with its partial derivatives with respect to each of the components of $\boldsymbol{x}$ and $\boldsymbol{u}$ are continuous.

Chapter 3

Optimal control problem

To achieve fast load handling, high-speed crane motions are required. The speeds are limited by manufacturers. High-speed motion can cause large load swings both during and after transfer, which are undesirable. Therefore, an optimal trajectory for the load motion that produces short travel time with suppressed load swing but that also satisfies operational constraints is needed. This optimal trajectory is the open-loop solution of the optimal control problem studied in this chapter.

Once the optimization criteria are specified, a certain (performance) cost functional can be defined. The crane optimization then is to minimize the selected cost functional subject to the dynamics of the crane and the constraints imposed by the operational and safety limits. This turns out to be a free-end-time optimal control problem with hard control constraints, end-point equality constraints and path (trajectory) constraints.

Because the system equations are nonlinear and coupled, analytical solutions to the optimal control problem only exist in special cases. Numerical methods must be used to find the optimal solution in general. However, finding optimal load

trajectories for general motions by conventional optimization methods appeared to be computationally difficult [1], [2]. Therefore, in Section 3.4.3, an iterative algorithm is proposed to solve a continuous-time optimal control problem where the control inputs and the terminal states are constrained. The algorithm is based on known second-order methods, which have been adapted and customized here for application to the required crane optimizations.

This chapter makes use of extensive notation. A list of symbols is included at the beginning of this thesis for reference.

3.1 Cost functional

Selection of a (performance) cost functional, J, is of prime importance in finding a meaningful solution of the optimal control problem. After optimization criteria (objectives) are identified a suitable cost function can be chosen in the formulation of the corresponding optimal control problem. In the following derivation, optimization is taken to mean *minimization* of the selected cost functional, subject to various constraints.

For the crane whose equations of motion are given in (2.18), the optimization criteria are considered to be

(a) short transfer time, and

(b) suppressed load swing during and after the transfer process.

Because there is always a trade-off between the duration of a transfer process and the amount of load swing during that process, the following cost functional is defined:

$$J \triangleq a(t_f - t_0) + b\int_{t_0}^{t_f} \Phi^2(r(t), \theta(t), \phi(t), \dot{r}(t), \dot{\theta}(t), \dot{\phi}(t))\, dt \qquad (3.1)$$

Constants a and b are the weighting factors, and Φ represents a function of the load swing in the transfer process. Here the final time t_f is unspecified (free). It is noted that $b = 0$ in (3.1) corresponds to a pure *time optimal* problem. Intuitively, a bigger a (relative to b) results in a faster motion, whereas a smaller a leads to a longer motion but more damped load swing during the motion.

Among many possible choices for Φ, consider

$$\Phi^2(t) = r^2(t)\left(\theta^2(t) + \phi^2(t) + \dot{\theta}^2(t) + \dot{\phi}^2(t)\right). \qquad (3.2)$$

The above function grows rapidly as $r(t)$ increases. Note that the load distance from the z-axis of the moving frame in Figure 2.1(b) is proportional to $r(t)$. Intuitively, excessive load swing angles with long hoist cable are more undesirable than when the hoist cable is short. A load swing of 10° translates to more than 17 m of displacement when the hoist cable length is 50 m. This, however, compares with a load sway of less than two meters when the hoist cable is 5 m long.

In terms of the state variable definition (2.16), function (3.2) can be expressed as

$$\Phi^2(\boldsymbol{x}(t)) = x_3^2(t)\left(x_4^2(t) + x_5^2(t) + x_9^2(t) + x_{10}^2(t)\right). \qquad (3.3)$$

Substituting (3.3) for Φ in (3.1) yields

$$J = \int_{t_0}^{t_f} \left[a + bx_3^2(t)\left(x_4^2(t) + x_5^2(t) + x_9^2(t) + x_{10}^2(t)\right)\right] dt \qquad (3.4)$$

It is then expected that minimizing the cost functional (3.4) along with a certain end-point constraint satisfies the given optimization criteria.

3.2 Operational and safety constraints

The following constraints are to be met. During a load transfer, the trolley travel and the hook height are limited by

$$\begin{array}{rcl} R_{min} & \leq & x_1(t) \leq R_{max} \\ r_{min} & \leq & x_3(t) \leq r_{max} \end{array} \qquad \forall t \in [t_0, t_f]. \tag{3.5}$$

In addition, there are limits on the speeds of the rigid-body motions that yield

$$\begin{array}{rcl} |x_6(t)| & \leq & \dot{R}_{max} \\ |x_7(t)| & \leq & \dot{\psi}_{max} \\ |x_8(t)| & \leq & \dot{r}_{max} \end{array} \qquad \forall t \in [t_0, t_f]\,. \tag{3.6}$$

Finally, as stated in Section 1.2, sudden speed change in the crane movements as well as large load swings must be avoided to preserve a safe load transfer. This requires limitations on the rigid-body accelerations (control variables) and the load swings, as

$$\begin{array}{rcl} |u_1(t)| & \leq & \ddot{R}_{max} \\ |u_2(t)| & \leq & \ddot{\psi}_{max} \\ |u_3(t)| & \leq & \ddot{r}_{max} \end{array} \qquad \forall t \in [t_0, t_f] \tag{3.7}$$

and

$$\begin{array}{rcl} |x_4(t)| & \leq & \theta_{max} \\ |x_5(t)| & \leq & \phi_{max} \end{array} \qquad \forall t \in [t_0, t_f]\,. \tag{3.8}$$

The state variables of the dynamical system (2.18) belong to the set of *admissible states* defined by

$$\mathcal{S} \triangleq \{\boldsymbol{x} \,|\, \boldsymbol{x} : [t_0, t_f] \to \Omega \text{ is continuous}\}\,, \tag{3.9}$$

$$\Omega = \left\{\boldsymbol{\nu} \in \Re^{10} \,|\, \underline{x}_i \leq \nu_i \leq \overline{x}_i,\ i = 1, \ldots, 10\right\} \tag{3.10}$$

with constant vectors $\underline{x}$ and $\overline{x}$ known by the limit values in the state-variable inequality constraints (3.5), (3.6) and (3.8). These constraints are usually called path (trajectory) constraints in the formulation of an optimal control problem.

Similarly, the control variables of the dynamical system (2.18) belong to the set of *admissible controls* defined by

$$\mathcal{U} \triangleq \{\boldsymbol{u} \,|\, \boldsymbol{u} : [t_0, t_f] \to \Pi \text{ is piecewise continuous}\}, \tag{3.11}$$

$$\Pi = \left\{\boldsymbol{\mu} \in \Re^3 \,|\, \underline{u}_i \leq \mu_i \leq \overline{u}_i,\ i = 1, 2, 3\right\} \tag{3.12}$$

where constant vectors $\underline{u}$ and $\overline{u}$ are identified by the limit values in the control inequality constraints (3.7).

3.3 Crane optimization

Consider the dynamical system (2.18), on a free end-time interval $[t_0, t_f]$ which is subject to the initial and final state conditions

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0 \in \Omega \tag{3.13a}$$

$$\boldsymbol{x}(t_f) = \boldsymbol{x}_f \in \Omega. \tag{3.13b}$$

In a typical load displacement, while the initial point may not be in equilibrium, the final (target) point must be in equilibrium to ensure a safe unloading condition. In an equilibrium point the load is at rest (i.e. all of the velocities, accelerations and swing angles are zero). Specifically, assume that

$$\boldsymbol{x}_0 = (R_0, \psi_0, r_0, \theta_0, \phi_0, 0, 0, 0, 0, 0)^T \tag{3.14a}$$

$$\boldsymbol{x}_f = (R_f, \psi_f, r_f, 0, 0, 0, 0, 0, 0, 0)^T, \tag{3.14b}$$

where R_0 and ψ_0 are the initial positions of the trolley and the boom respectively, and r_0 is the initial length of the hoist cable. Similarly, R_f and ψ_f are the final positions of the trolley and the boom respectively, and r_f is the desired length of the hoist cable at the target point. Nonzero swing angles θ_0 and ϕ_0 indicate that the load is not initially located straight under the trolley, as may be the case in practice.

The crane optimization problem can now be formulated as follows:

- Subject to the dynamical system (2.18) and the initial and final conditions (3.13), find a control $\boldsymbol{u}(t) \in \mathcal{U}$ such that the cost functional

$$J(\boldsymbol{u}, t_f) = \int_{t_0}^{t_f} \left[a + b\Phi^2(\boldsymbol{x}(t))\right] dt \tag{3.15}$$

  is minimized over $\mathcal{S}$ with $\Phi^2(\boldsymbol{x}(t))$ as defined in (3.3).

Because, for a fixed t_f, the state $\boldsymbol{x}$ is uniquely determined by (2.18) when the initial condition $\boldsymbol{x}_0$ and the control $\boldsymbol{u}$ are given [25], the cost functional J depends only on $\boldsymbol{u}$.

The control strategy proposed in Section 1.4.1 requires pre-computation of a reference trajectory ($\boldsymbol{x}_{ref}$) as a benchmark path for the crane to follow. This reference trajectory is indeed the solution to the above optimal problem.

3.4 Constrained Lagrange problem

The optimization problem described in the previous section is a special case of the following general optimal control problem.

Consider the dynamical system, on a free end-time interval $\mathcal{T} = [t_0, t_f]$:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \tag{3.16a}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{3.16b}$$

which is subject to the terminal state constraints

$$g_i(\boldsymbol{x}(t_f)) \leq 0 \quad i = 1, \ldots, r\,, \tag{3.17}$$

the state path constraints

$$\varphi_j(\boldsymbol{x}(t), t) \leq 0 \quad j = 1, \ldots, l \quad \forall t \in \mathcal{T}\,. \tag{3.18}$$

and the control constraints

$$\boldsymbol{u} \in \Pi = \{\boldsymbol{\mu} \in \Re^m \,|\, \underline{u}_i \leq \mu_i \leq \overline{u}_i,\ i = 1, \ldots, m\} \tag{3.19}$$

with $\overline{\boldsymbol{u}}$ and $\underline{\boldsymbol{u}}$ two constant vectors in $\Re^m$, representing upper and lower limits on the control inputs. In the above, $\boldsymbol{x}(t) \in \Re^n$ is the state vector of the system at time $t \in \mathcal{T}$ which corresponds to the control vector $\boldsymbol{u}(t) \in \Re^m$.

Let the set of admissible controls be

$$\mathcal{U} = \{\boldsymbol{u} \,|\, \boldsymbol{u} : \mathcal{T} \to \Pi \text{ is piecewise continuous}\}\,, \tag{3.20}$$

and let the set of feasible controls be

$$\mathcal{F} = \{\boldsymbol{u} \,|\, \boldsymbol{u} \in \mathcal{U},\ g_i(\boldsymbol{x}(t_f)) \leq 0,\ \varphi_j(\boldsymbol{x}(t), t) \leq 0,\ i = 1, ..., r,\ j = 1, ..., l\} \tag{3.21}$$

One of the standard forms of a constrained optimal control problem, known as the *constrained Lagrange problem* [26], is formulated as follows.

- Subject to the dynamical system (3.16), find a control $\boldsymbol{u}(t) \in \mathcal{U}$ such that the cost functional

$$J(\boldsymbol{u}, t_f) = \int_{t_0}^{t_f} L\left(\boldsymbol{x}(t), \boldsymbol{u}(t), t\right) dt \tag{3.22}$$

  is minimized over $\mathcal{F}$.

Throughout, it is understood that the norm of any vector $\boldsymbol{w} \in \Re^p$ for some dimension p is

$$\|\boldsymbol{w}\| = \max_{1 \le i \le p} |w_i| ,$$

and the norm of any $\boldsymbol{w} : \mathcal{T} \to \Re^p$ is

$$\|\boldsymbol{w}\| = \max_{t_0 \le t \le t_f} \|\boldsymbol{w}(t)\| .$$

The following conditions are assumed to be satisfied.

1. $\boldsymbol{f} : \Re^n \times \Re^m \times \mathcal{T} \to \Re^n$, $L : \Re^n \times \Re^m \times \mathcal{T} \to \Re$.

2. The functions $\boldsymbol{f}$ and L together with their partial derivatives up to the second order with respect to each of the components of $\boldsymbol{x}$ and $\boldsymbol{u}$ are piecewise continuous and uniformly bounded for all $(\boldsymbol{x}, \boldsymbol{u}, t) \in \Re^n \times \Re^m \times \mathcal{T}$ (*smoothness requirement*).

3. There exists a non-negative piecewise continuous function $k(.)$ on $\mathcal{T}$ such that

$$\|\boldsymbol{f}(\boldsymbol{x}_1, \boldsymbol{u}, t) - \boldsymbol{f}(\boldsymbol{x}_2, \boldsymbol{u}, t)\| \le k(t)\|\boldsymbol{x}_2 - \boldsymbol{x}_1\|$$

   for all $(\boldsymbol{x}_1, \boldsymbol{u}, t), (\boldsymbol{x}_2, \boldsymbol{u}, t) \in \Re^n \times \Pi \times \mathcal{T}$ (*Lipshitz condition*).

From the theory of differential equations [27],[28], the system (3.16) has a unique continuous solution $\boldsymbol{x}(t)$ on the interval $\mathcal{T}$ corresponding to each $\boldsymbol{u} \in \mathcal{U}$.

In what follows in the rest of this chapter, an effective algorithm will be developed to solve the above Lagrange problem. But, first a brief survey on the techniques and algorithms for solving optimal control problems is given.

3.4.1 Computational challenges

Clearly, except for some special cases [2], finding closed-form optimal control solutions is almost impossible. A numerical approach then has to be adopted in most cases. Computational techniques for solving optimal control problems, like optimization in finite-dimensional space, are iterative in nature. Expectations of a good numerical optimal control algorithm can be summarized as follows.

1. It should be convergent, at least locally and preferably globally.
2. It is "efficient", which means that it has a fast (local) convergence rate.
3. It must handle terminal and path constraints.

Because a general optimal control problem can be viewed as an optimization of a functional in a general control space subject to system dynamics and some functional equality/inequality constraints, there is a great deal of similarity in computational techniques between optimization in finite dimensional space and optimal control problems. Although there is a wide variety of methods for the computation of optimal controls, they can be classified into different types according to their updating schemes for the controls. Most well known methods update only the control variables, and the update scheme is such that, when at the k-th iteration,

$$u^{(k+1)} = P\left(u^{(k)} + \alpha^{(k)} v^{(k)}\right)$$

where P is the transformation mapping within the admissible control space $\mathcal{U}$, $v^{(k)}$ a search direction, and $\alpha^{(k)}$ a suitably chosen stepsize. Different constructions of the mapping P, search direction $v^{(k)}$ and stepsize $\alpha^{(k)}$ result in many different methods (algorithms): the gradient method, the projection method [29], the conjugate gradient method [30], the quasi-Newton methods [31], etc.

Some methods approximate the control trajectories by orthogonal functions of polynomials. That is

$$u(t) \approx \sum_{i=1}^{N} \mu_i \phi_i(t)$$

where $\{\phi_i(t)\}$ is a family of N orthogonal functions or polynomials. Sometimes, both the control and the state variables are approximated in this way. Consequently, the original optimal control problem is converted into an optimization problem, where the updatings are on the parameters $\{\mu_i\}$. Different selections of the $\{\phi_i(t)\}$ yield many different methods, such as the Chebyshev method [32], [33], the Fourier method [34], [35], and the Taylor method [36]. In practice, the most common parameterization methods approximate the orthogonal functions by piecewise polynomials, in particular spline functions [37] and piecewise constant functions [38], [39].

Computational methods for optimal control problems can also be classified into two classes: first-order methods and second-order methods. Despite the emergence of many successful first-order methods, the development of second-order methods has been relatively slow. This is perhaps because of two major reasons. One reason is that second-order methods require not only the evaluation of the second derivatives of the Hamiltonian function at every sampling time, but also their storage. Another reason is that the minimization of the summation of the first and second

variations is itself a complex problem. However, because second-order methods generally enjoy rapid, usually quadratic, convergence around the solution, many attempts have been made to solve optimal control problems using second-order methods.

Motivated by the success of the trust region approach in finite dimensional optimization, Fukushima and Yamamoto [40] proposed an algorithm to minimize the summation of the first and second variations. The algorithm is shown to be globally convergent, but the control variable is assumed to be unconstrainted.

In late 70's, a promising algorithm, developed by Han [41], [42] and Powell [43], emerged as a general purpose algorithm for solving optimization problems in finite dimensional space. The method replaces the original problem by a sequence of quadratic programming problems, where the original cost function is approximated by a quadratic function, with the Hessian being replaced by a positive definite matrix, and the inequality constraints approximated by a linear function.

In the following, an algorithm, with an approach similar to the Han-Powell method in finite dimensional optimization, will be developed to solve fixed end-time optimal control problems where the control variables and the terminal states are constrained. The first step is to establish the optimality conditions according to the first variation technique of the calculus of variations [44]. The next step is to produce an upper bound on a second-order convex functional approximation due to a change in the control. Consequently, the original optimal problem is replaced by a series of simpler subproblems. It is then shown that the solution of this simplified problem generates a descent direction of the original cost functional. Finally, the original problem will be extended to allow free end time and to include path constraints.

3.4.2 Optimality conditions

To arrive at the necessary conditions for optimality of the Lagrange problem defined in Section 3.4, consider the case where the final time t_f is known and also no path constraints are involved. In Sections 3.4.7 and 3.4.8 these restrictions will be removed. What follows in this section is a specialized derivation of the optimality conditions according to the first variation technique of the calculus of variations [44].

Introducing n continuous *costates* (Lagrange multipliers) $p_1(t).....p_n(t)$, the *Hamiltonian* function H is defined by

$$H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t) \triangleq L(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{p}^T(t)\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \tag{3.23}$$

where

$$\boldsymbol{p} = (p_1.....p_n)^T \in \Re^n \ .$$

with L and f as given in (3.22) and (3.16(a)) respectively.

Introduction of the undetermined costates $p_i(t)$ enables us to treat $x_i(t)$ as though they were independent whereas they are indeed related by the system equations (3.16(a)). The augmented cost functional is then

$$\tilde{J}(\boldsymbol{u}) = \int_{t_0}^{t_f} \left[H(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}(t), t) - \boldsymbol{p}^T(t)\dot{\boldsymbol{x}}(t)\right] dt \, . \tag{3.24}$$

Since the system equations (3.16(a)) must be satisfied, minimizing the cost functional J is equivalent to minimizing the augmented functional $\tilde{J}$.

Let $\boldsymbol{u}^*$ and $\boldsymbol{x}^*$ be the optimal pair that minimizes (3.24) and satisfies the system (3.16). Introducing a small parameter ϵ, then

$$\boldsymbol{u}(t, \epsilon) = \boldsymbol{u}^*(t) + \delta\boldsymbol{u}(t) \tag{3.25a}$$

$$\boldsymbol{x}(t, \epsilon) = \boldsymbol{x}^*(t) + \delta\boldsymbol{x}(t) \tag{3.25b}$$

with

$$\delta \boldsymbol{u}(t) = \left.\frac{\partial \boldsymbol{u}}{\partial \epsilon}\right|_{\epsilon=0} \epsilon$$

$$\delta \boldsymbol{x}(t) = \left.\frac{\partial \boldsymbol{x}}{\partial \epsilon}\right|_{\epsilon=0} \epsilon$$

as the first-order approximations of the functions $\boldsymbol{u}(t,\epsilon)$ and $\boldsymbol{x}(t,\epsilon)$ about their optimums, for which $\epsilon = 0$. Similarly,

$$\dot{\boldsymbol{x}}(t,\epsilon) = \dot{\boldsymbol{x}}^*(t) + \delta\dot{\boldsymbol{x}}(t) \tag{3.26}$$

with

$$\delta\dot{\boldsymbol{x}}(t) = \left.\frac{\partial \dot{\boldsymbol{x}}}{\partial \epsilon}\right|_{\epsilon=0} \epsilon .$$

It is assumed above that $\boldsymbol{u}(t,\epsilon)$, $\boldsymbol{x}(t,\epsilon)$ and $\dot{\boldsymbol{x}}(t,\epsilon)$ lie in a small neighborhood of their optimums $\boldsymbol{u}^*$, $\boldsymbol{x}^*$ and $\dot{\boldsymbol{x}}^*$.

Because $\left.\frac{\partial x_i}{\partial \epsilon}\right|_{\epsilon=0}$ is only a function of time, differentiating equation (3.25) with respect to time and then comparing the result with equation (3.26) yields

$$d\frac{\delta x_i(t)}{dt} = \delta\frac{dx_i(t)}{dt} \qquad i = 1, 2, \ldots, n . \tag{3.27}$$

This result shows that the δ operation and the derivative operation are interchangeable.

For the variational functional

$$\tilde{J}(\epsilon) = \int_{t_0}^{t_f} \left[H(\boldsymbol{x}(t,\epsilon), \boldsymbol{u}(t,\epsilon), \boldsymbol{p}(t), t) - \boldsymbol{p}^T(t)\dot{\boldsymbol{x}}(t,\epsilon)\right] dt \tag{3.28}$$

the Taylor expansion about $\epsilon = 0$ is

$$\mathcal{J}(\epsilon) = \mathcal{J}(0) + \left.\frac{d\mathcal{J}}{d\epsilon}\right|_{\epsilon=0} \frac{\epsilon}{1!} + \left.\frac{d^2\mathcal{J}}{d\epsilon^2}\right|_{\epsilon=0} \frac{\epsilon^2}{2!} + \ldots \tag{3.29}$$

where $\tilde{J}(0)$ is an extremum value of $\tilde{J}$. For $\tilde{J}(0)$ to be a minimum, the necessary condition is that the first variation in (3.29) vanishes, i.e.

$$\delta\tilde{J} = \left.\frac{d\tilde{J}}{d\epsilon}\right|_{\epsilon=0} = 0\,.$$

Then, using a result from the calculus of variations [45], one finds

$$\delta\mathcal{J} = \int_{t_0}^{t_f} \delta(H - \boldsymbol{p}^T\dot{\boldsymbol{x}})\,dt + (H - \boldsymbol{p}^T\dot{\boldsymbol{x}})dt\Big|_{t_0}^{t_f} = 0\,. \tag{3.30}$$

Since $\mathbf{x}$, $\dot{\mathbf{x}}$ and $\mathbf{u}$ in the integrand (3.28) depend on ϵ, the first variation of $\delta(H - \boldsymbol{p}^T\dot{\boldsymbol{x}})$ is given by

$$\delta(H - \boldsymbol{p}^T\dot{\boldsymbol{x}}) = H_{\boldsymbol{x}}^{*}\delta\boldsymbol{x} - \boldsymbol{p}^{*T}\delta\dot{\boldsymbol{x}} + H_{\boldsymbol{u}}^{*}\delta\boldsymbol{u} \tag{3.31}$$

where

$$H_{\boldsymbol{u}}^{*} = \left.\frac{\partial H}{\partial \boldsymbol{u}}\right|_{\epsilon=0} = \left[\frac{\partial H}{\partial u_1} \;\cdots\; \frac{\partial H}{\partial u_m}\right]_{\epsilon=0}$$

$$H_{\boldsymbol{x}}^{*} = \left.\frac{\partial H}{\partial \boldsymbol{x}}\right|_{\epsilon=0} = \left[\frac{\partial H}{\partial x_1} \;\cdots\; \frac{\partial H}{\partial x_n}\right]_{\epsilon=0}.$$

Substituting (3.31) in (3.30) yields

$$\delta\mathcal{J} = \int_{t_0}^{t_f} \left(H_{\boldsymbol{x}}^{*}\delta\boldsymbol{x} - \boldsymbol{p}^{*T}\frac{d\delta\boldsymbol{x}}{dt}\right) dt + \int_{t_0}^{t_f} H_{\boldsymbol{u}}^{*}\delta\boldsymbol{u}dt = 0 \tag{3.32}$$

where the last two terms in (3.30) are removed due to fixed initial and final times, and also the δ and the derivative operations have been interchanged using (3.27). Finally, integrating the second term of the first integrand in (3.32) once by parts yields

$$\delta\mathcal{J} = -\,\boldsymbol{p}^{*T}\delta\boldsymbol{x}\Big|_{t_0}^{t_f} + \int_{t_0}^{t_f} \left(\frac{d}{dt}\boldsymbol{p}^{*T} + H_{\boldsymbol{x}}^{*}\right)\delta\boldsymbol{x}dt + \int_{t_0}^{t_f} H_{\boldsymbol{u}}^{*}\delta\boldsymbol{u}dt = 0\,. \tag{3.33}$$

The last two terms in (3.33) indicate the changes in the $\mathcal{J}$ caused by varying x_i and u_i during the process time. Therefore the coefficients of $\delta\boldsymbol{x}$ and $\delta\boldsymbol{u}$ in (3.33)

must each be zero, resulting in the well-known *Euler equations*

$$\frac{d}{dt}\boldsymbol{p}^{*T} + H^*_{\boldsymbol{x}} = 0 \tag{3.34a}$$

$$H^*_{\boldsymbol{u}} = 0. \tag{3.34b}$$

Using the definition (3.23) in (3.34) gives

$$-\dot{\boldsymbol{p}}^*(t)^T = H_{\boldsymbol{x}}\left(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{p}^*(t), t\right) \tag{3.35a}$$

$$\mathbf{0} = H_{\boldsymbol{u}}\left(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{p}^*(t), t\right). \tag{3.35b}$$

These are $n+m$ ordinary differential equations. Equation (3.35(a)) is called the *costate equation*, while (3.35(b)) is usually referred to the *control equation* since it normally results in determination of the optimal controls u_i^* provided that they do not appear linearly in the augmented integrand $\mathcal{L}$. Otherwise the control variables are *bang-bang*, determined by their extrema.

A necessary condition for optimality of $\boldsymbol{x}^*(t)$ and $\boldsymbol{u}^*(t)$ requires existence of a non-zero continuous costate $\boldsymbol{p}^*(t)$, such that the costate and control equations (3.35) are satisfied for $t \in \mathcal{T}$ subject to the system (3.16).

For the problem with terminal state constraints (3.17) not only must equations (3.35) be satisfied, but in addition the remaining term in (3.33).

$$-\boldsymbol{p}^{*T}\delta\boldsymbol{x}\Big|_{t_0}^{t_f},$$

must also be equal to zero to ensure the vanishing of the first variation. This condition can be reduced to

$$-\boldsymbol{p}^*(t_f)^T\delta\boldsymbol{x}(t_f) = 0 \tag{3.36}$$

because of fixed initial value (3.16(b)). This condition, along with the r terminal state constraints (3.17), provides the additional relations, termed as the *transversality condition*, required for the optimality conditions.

Through the use of an underdetermined constant η_i, the differential of each of the r constraints $g_i(\boldsymbol{x}(t_f))$ evaluated at the optimal state $\boldsymbol{x}^*(t_f)$ can be introduced into (3.36) to form

$$-\boldsymbol{p}^*(t_f)^T\delta\boldsymbol{x}(t_f) + \sum_{i=1}^{r}\eta_i^* g_{i_{\boldsymbol{x}}}(\boldsymbol{x}^*(t_f))\delta\boldsymbol{x}(t_f) = 0$$

where

$$g_{i_{\boldsymbol{x}}} = \left[\frac{\partial g_i}{\partial x_1} \cdots \frac{\partial g_i}{\partial x_n}\right].$$

Then, the following transversality condition holds

$$\boldsymbol{p}^*(t_f) = \sum_{i=1}^{r}\eta_i^* g_{i_{\boldsymbol{x}}}(\boldsymbol{x}^*(t_f))^T .$$

with

$$\eta_i^* \geq 0 \quad \text{and} \quad \eta_i^* g_i(\boldsymbol{x}^*(t_f)) = 0$$

for all $i = 1, \ldots, r$.

3.4.3 Description of the optimization process

In this section, an iterative algorithm, with an approach similar to the Han-Powell method in finite dimensional optimization, is developed to solve fixed end-time optimal control problems with constraints on the control and terminal state variables. At every iteration, the algorithm minimizes a cost functional which is an upper bound on the summation of the first and second variations of the original cost functional, defined in (3.24), due to a change in the control.

Let $\boldsymbol{v}(t)$ be the variation of control $\boldsymbol{u}'(t) - \boldsymbol{u}(t)$, and let $\boldsymbol{y}(t)$ satisfy the following linearized system equations

$$\dot{\boldsymbol{y}}(t) = \boldsymbol{f}_{\boldsymbol{x}}(t)\boldsymbol{y}(t) + \boldsymbol{f}_{\boldsymbol{u}}(t)\boldsymbol{v}(t) \tag{3.37}$$

$$\boldsymbol{y}(t_0) = \boldsymbol{0} \tag{3.38}$$

where the gradients $\boldsymbol{f}_{\boldsymbol{x}}(t)$ and $\boldsymbol{f}_{\boldsymbol{u}}(t)$ are evaluated at $(\boldsymbol{x}(t), \boldsymbol{u}(t))$.

The first and second variations of the augmented cost functional are defined [46] as

$$\Delta\tilde{J}(\boldsymbol{v}) \triangleq \int_{t_0}^{t_f} H_{\boldsymbol{u}}(t)\boldsymbol{v}(t)dt \tag{3.39}$$

and

$$\Delta^2\tilde{J}(\boldsymbol{v}) \triangleq \frac{1}{2}\int_{t_0}^{t_f} \left[\boldsymbol{y}(t)^T, \boldsymbol{v}(t)^T\right] \begin{bmatrix} H_{\boldsymbol{xx}}(t) & H_{\boldsymbol{xu}}(t) \\ H_{\boldsymbol{ux}}(t) & H_{\boldsymbol{uu}}(t) \end{bmatrix} \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{v}(t) \end{bmatrix} dt \tag{3.40}$$

where

$$H_{\boldsymbol{xx}}(t) = \frac{\partial}{\partial \boldsymbol{x}}\left(\frac{\partial H}{\partial \boldsymbol{x}}\right)^T, \quad H_{\boldsymbol{xu}}(t) = \frac{\partial}{\partial \boldsymbol{u}}\left(\frac{\partial H}{\partial \boldsymbol{x}}\right)^T, \quad \text{etc.}$$

In the above, $\boldsymbol{x}(t)$ and $\boldsymbol{p}(t)$ are the state and costate corresponding to the control $\boldsymbol{u}(t)$.

Typically, for most computational techniques seeking the optimal control, an initial control $\boldsymbol{u}^{(0)}$ is selected and a sequence of new controls $\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}, \ldots, \boldsymbol{u}^{(k)}, \ldots$, is generated, each improving upon its predecessor. A natural and convenient way to find an improving control $\boldsymbol{u}^{(k+1)}$ at the k-th iteration is to minimize $\Delta\tilde{J}^{(k)}(\boldsymbol{v}^{(k)})$ (first-order method) or $\Delta\tilde{J}^{(k)}(\boldsymbol{v}^{(k)}) + \Delta^2\tilde{J}^{(k)}(\boldsymbol{v}^{(k)})$ (second-order method) subject to the linearized system

$$\dot{\boldsymbol{y}}^{(k)}(t) = \boldsymbol{f}_{\boldsymbol{x}}^{(k)}(t)\boldsymbol{y}^{(k)}(t) + \boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\boldsymbol{v}^{(k)}(t) \tag{3.41a}$$

$$\boldsymbol{y}^{(k)}(t_0) = \boldsymbol{0}, \tag{3.41b}$$

and the linearized terminal constraints

$$g_i(\boldsymbol{x}^{(k)}(t_f)) + g_{i_{\boldsymbol{x}}}(\boldsymbol{x}^{(k)}(t_f))\boldsymbol{y}^{(k)}(t_f) \leq 0 \qquad i = 1, \ldots, r \tag{3.42}$$

with $\boldsymbol{v}^{(k)} = \boldsymbol{u} - \boldsymbol{u}^{(k)}$. All the components of the gradient and Hessian terms in $\Delta\tilde{J}^{(k)}$ and $\Delta^2\tilde{J}^{(k)}$ are evaluated at $\boldsymbol{x}^{(k)}$ and $\boldsymbol{p}^{(k)}$ corresponding to $\boldsymbol{u}^{(k)}$. In what

follows, only the second-order method is considered. However, the second-order problem is difficult to solve, and further simplification is needed.

Proposition

- Let $\boldsymbol{M}(t)$ be the Hessian matrix in (3.40),

$$\boldsymbol{M}(t) \triangleq \begin{bmatrix} H_{\boldsymbol{xx}}(t) & H_{\boldsymbol{xu}}(t) \\ H_{\boldsymbol{ux}}(t) & H_{\boldsymbol{uu}}(t) \end{bmatrix}. \tag{3.43}$$

Then there exists a positive time-varying scalar $\tilde{\lambda}(t)$ such that, for any $t \in \mathcal{T}$,

$$\left[\boldsymbol{y}(t)^T, \boldsymbol{v}(t)^T\right] \boldsymbol{M}(t) \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{v}(t) \end{bmatrix} \leq \tilde{\lambda}(t) \left(\boldsymbol{y}(t)^T \boldsymbol{y}(t) + \boldsymbol{v}(t)^T \boldsymbol{v}(t)\right). \tag{3.44}$$

Proof: Because $\boldsymbol{M}(t)$ is real and symmetric, its spectral decomposition (Section 2.2.5.4 of [47]) is

$$\boldsymbol{M}(t) = \boldsymbol{U}(t)^T \boldsymbol{\Lambda}(t) \boldsymbol{U}(t),$$

where $\boldsymbol{U}$ is an orthonormal matrix satisfying $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$, and $\boldsymbol{\Lambda}$ is a diagonal matrix containing the eigenvalues of the matrix $\boldsymbol{M}(t)$. Then,

$$\left[\boldsymbol{y}(t)^T, \boldsymbol{v}(t)^T\right] \boldsymbol{M}(t) \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{v}(t) \end{bmatrix} = \boldsymbol{z}(t)^T \boldsymbol{\Lambda}(t) \boldsymbol{z}(t).$$

where

$$\boldsymbol{z}(t) = \boldsymbol{U}(t) \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{v}(t) \end{bmatrix}.$$

Assuming that $\lambda_{max}(t)$ is the largest eigenvalue of the matrix $\boldsymbol{M}(t)$,

$$\boldsymbol{\Lambda}(t) \leq \lambda_{max}(t) \boldsymbol{I}. \tag{3.45}$$

Note that, based on assumption 2 on page 40 and the Hamiltonian definition by (3.23), all the elements of the matrix $\boldsymbol{M}(t)$ are uniformly bounded. Therefore, $\lambda_{max}(t)$ in (3.45) is always well defined, for any $t \in \mathcal{T}$.

From (3.45),

$$\boldsymbol{z}(t)^T \boldsymbol{\Lambda}(t) \boldsymbol{z}(t) \;\leq\; \lambda_{max}(t) \boldsymbol{z}(t)^T \boldsymbol{z}(t) \;\leq\; \lambda_{max}(t) \left(\boldsymbol{y}(t)^T \boldsymbol{y}(t) + \boldsymbol{v}(t)^T \boldsymbol{v}(t) \right)$$

since

$$\left[\boldsymbol{y}(t)^T, \boldsymbol{v}(t)^T \right] \boldsymbol{U}(t)^T \boldsymbol{U}(t) \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{v}(t) \end{bmatrix} \;=\; \left(\boldsymbol{y}(t)^T \boldsymbol{y}(t) + \boldsymbol{v}(t)^T \boldsymbol{v}(t) \right) .$$

Thus the following inequality holds:

$$\left[\boldsymbol{y}(t)^T, \boldsymbol{v}(t)^T \right] \boldsymbol{M}(t) \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{v}(t) \end{bmatrix} \;\leq\; \lambda_{max}(t) \left(\boldsymbol{y}(t)^T \boldsymbol{y}(t) + \boldsymbol{v}(t)^T \boldsymbol{v}(t) \right) .$$

Because the Hessian matrix $\boldsymbol{M}(t)$ is not necessarily positive definite, implying that $\lambda_{max}(t)$ is not necessarily strictly positive, consider

$$\tilde{\lambda}(t) \;\triangleq\; \max\{1 \,,\, \lambda_{max}(t)\} \,, \tag{3.46}$$

which is always positive and satisfies

$$\lambda_{max}(t) \;\leq\; \tilde{\lambda}(t)$$

for any $t \in \mathcal{T}$. Hence,

$$\left[\boldsymbol{y}(t)^T, \boldsymbol{v}(t)^T \right] \boldsymbol{M}(t) \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{v}(t) \end{bmatrix} \;\leq\; \tilde{\lambda}(t) \left(\boldsymbol{y}(t)^T \boldsymbol{y}(t) + \boldsymbol{v}(t)^T \boldsymbol{v}(t) \right) . \qquad \square$$

The above proposition provides an upper bound on the integrand function in (3.40). While the Hessian matrix may be difficult to compute at each iteration, the right hand side in (3.44) can be computationally simple.

Consider now the following terminal condition

$$\boldsymbol{p}(t_f) \;=\; \mathbf{0}\,, \tag{3.47}$$

and the new cost functional

$$\bar{J}(\boldsymbol{v}) \;\triangleq\; \int_{t_0}^{t_f} \bar{L}\,(\boldsymbol{y}(t), \boldsymbol{v}(t), t)\,dt \tag{3.48}$$

where

$$\bar{L} \;\triangleq\; H_{\boldsymbol{u}}(t)\boldsymbol{v}(t) + \frac{1}{2}\tilde{\lambda}(t)\left(\boldsymbol{y}(t)^T\boldsymbol{y}(t) + \boldsymbol{v}(t)^T\boldsymbol{v}(t)\right)\,. \tag{3.49}$$

The above functional is indeed an upper bound of a second-order approximation to the change of the original cost functional $\tilde{J}$ due to a change in the control. With the fact that the integrand term in (3.48) is a convex function. $\bar{J}$ is easier to compute than the second-order approximation $\Delta\tilde{J}(\boldsymbol{v}) + \Delta^2\tilde{J}(\boldsymbol{v})$.

The above results are similar to the case of finite-dimensional optimization handled by the Han-Powell method where the Hessian is iteratively replaced by an updated positive definite matrix to facilitate both the computational and convergence analysis [41], [42], [43]. Hence, instead of solving the original problem. the following subproblem is solved repeatedly.

Direction finding subproblem (DFS):

- Given $\boldsymbol{u}^{(k)}$, find an optimal control $\bar{\boldsymbol{u}}^{(k)} \in \mathcal{U}$, such that

$$\bar{\boldsymbol{u}}^{(k)} \;=\; \arg\min_{\boldsymbol{u}\in\mathcal{U}} \bar{J}^{(k)}(\boldsymbol{u} - \boldsymbol{u}^{(k)}) \tag{3.50}$$

  subject to the linearized equations (3.41) and the linearized terminal inequality constraints (3.42).

This is usually referred to as "direction finding" in the literature (see [47] for example), because its optimal solution at each iteration, $\bar{\boldsymbol{u}}^{(k)}$, helps an optimization algorithm to make progress towards its solution. The following algorithm is then adapted from [48] to find the optimal input $\boldsymbol{u}^*(t)$.

Optimization algorithm:

1. Select an initial control $\boldsymbol{u}^{(0)} \in \mathcal{U}$. Set $k = 0$.
2. Compute the state $\boldsymbol{x}^{(k)}$ by integrating (3.16) forward.
3. Compute the costate $\boldsymbol{p}^{(k)}$ by integrating (3.35(a)) backward from the terminal condition (3.47).
4. Solve the above direction-finding subproblem (**DFS**) to find $\bar{\boldsymbol{u}}^{(k)}$.
5. If $\bar{\boldsymbol{u}}^{(k)} \simeq \boldsymbol{u}^{(k)}$ within a certain accuracy range, then set $\boldsymbol{u}^* = \bar{\boldsymbol{u}}^{(k)}$ and stop. Otherwise, go to Step 6.
6. Compute a suitable stepsize $\alpha^{(k)}$, according to a stepsize rule.
7. Set $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \alpha^{(k)} \left(\bar{\boldsymbol{u}}^{(k)} - \boldsymbol{u}^{(k)} \right)$. Set $k = k + 1$ and return to Step 2.

Clearly the set Π of the constrained inputs, defined in (3.19), is convex. Because $\bar{\boldsymbol{u}}^{(k)}$, $\boldsymbol{u}^{(k)} \in \Pi$ and

$$\begin{aligned} \boldsymbol{u}^{(k+1)}(t) &= \boldsymbol{u}^{(k)}(t) + \alpha^{(k)} \left(\bar{\boldsymbol{u}}^{(k)} - \boldsymbol{u}^{(k)}(t) \right) \\ &= \left(1 - \alpha^{(k)} \right) \boldsymbol{u}^{(k)}(t) + \alpha^{(k)} \bar{\boldsymbol{u}}^{(k)} \end{aligned}$$

is a convex combination of $\bar{\boldsymbol{u}}^{(k)}$ and $\boldsymbol{u}^{(k)}$, $\boldsymbol{u}^{(k+1)} \in \Pi$ for any $t \in \mathcal{T}$ and for any $\alpha^{(k)} \in [0, 1]$. Therefore the above algorithm always generates a sequence of admissible controls, as long as the initial control $\boldsymbol{u}^{(0)}$ is admissible.

An advantage of the above algorithm is that the original constrainted nonlinear problem is solved by solving a sequence of constrained linear quadratic optimal control subproblems, which are much simpler than the original one. Moreover, at the kth iteration, the existence and uniqueness of the solution of the constrained linear quadratic problem is always guaranteed [49]. Also, as will be seen in the next section, the algorithm generates a descent direction of the cost functional $\tilde{J}(\boldsymbol{u})$.

Another advantage of the above algorithm is that, for each subproblem at the k-th iteration, its Hamiltonian function, which is quadratic in both the state $\boldsymbol{x}(t)$ and the control $\boldsymbol{u}(t)$ for any $t \in \mathcal{T}$, is a strictly convex function in control $\boldsymbol{u}(t)$. Therefore the optimal control of every subproblem can never be singular.

However, one disadvantage of the above algorithm is that after the Hessian matrix $\boldsymbol{M}(t)$ is replaced by the diagonal matrix $\tilde{\lambda}(t)\boldsymbol{I}$, the second-order convergence rate may not hold any longer.

3.4.4 Descent property

In this section it will be shown that at each iteration, the algorithm given in the previous section generates a descent direction of the cost function $\tilde{J}(\boldsymbol{u})$. Such a descent direction helps the optimization algorithm to improve the next control update by reducing the cost functional at each iteration. Before showing the descent property, the optimality conditions of the direction-finding subproblem (**DFS**) are given. But, first the following property is required.

Property

- For any $\boldsymbol{u}(t), \boldsymbol{u}'(t) \in \mathcal{U}$, there exists a positive constant c such that

$$\tilde{J}(\boldsymbol{u}') = \tilde{J}(\boldsymbol{u}) + \Delta\tilde{J}(\boldsymbol{v}) + E(\boldsymbol{v}) , \tag{3.51a}$$

where the remainder $E(\boldsymbol{v})$ is upper bounded by

$$E(\boldsymbol{v}) \leq c\|\boldsymbol{v}\|^2, \tag{3.51b}$$

with $\Delta\tilde{J}(\boldsymbol{v})$ defined as (3.39) and $\boldsymbol{v}(t) = \boldsymbol{u}'(t) - \boldsymbol{u}(t)$.

Proof: Equation (3.51(a)) is a multivariable case of the Taylor polynomial of degree 1 with the reminder E at $\boldsymbol{u}(t)$. From the Proposition of page 50, the second variation $\Delta^2\tilde{J}(\boldsymbol{v})$ defined in (3.40) is upper bounded. Hence, according to the Calculus of Taylor polynomials [50], there exists a positive constant c to satisfy (3.51(b)). □

Now the optimality conditions of **DFS** are given. Let $\bar{\boldsymbol{u}}^{(k)}(t)$ be the solution of **DFS** at the k-th iteration and $\bar{\boldsymbol{v}}^{(k)}(t) = \bar{\boldsymbol{u}}^{(k)}(t) - \boldsymbol{u}^{(k)}(t)$. Also, let $\bar{\boldsymbol{y}}^{(k)}(t)$ be the solution of the linearized system (3.41) with input $\bar{\boldsymbol{v}}^{(k)}(t)$. Then, similar to the optimality conditions derived in Section 3.4.2, there exists a costate function $\bar{\boldsymbol{q}}^{(k)}(t)$ of the direction-finding subproblem satisfying

$$-\dot{\bar{\boldsymbol{q}}}^{(k)}(t) = \bar{H}_{\boldsymbol{y}}(\bar{\boldsymbol{y}}^{(k)}(t), \bar{\boldsymbol{v}}^{(k)}(t), \bar{\boldsymbol{q}}^{(k)}(t), t) \tag{3.52}$$

where $\bar{H}(\boldsymbol{y}^{(k)}(t), \boldsymbol{v}^{(k)}(t), \boldsymbol{q}^{(k)}(t), t)$ is the Hamiltonian function of the direction-finding subproblem, defined as

$$\begin{aligned}\bar{H} &\triangleq \bar{L}(\boldsymbol{y}^{(k)}(t), \boldsymbol{v}^{(k)}(t), t) + \boldsymbol{q}^{(k)}(t)^T \left(\boldsymbol{f}_{\boldsymbol{x}}^{(k)}(t)\boldsymbol{y}^{(k)}(t) + \boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\boldsymbol{v}^{(k)}(t)\right) \\ &= H_{\boldsymbol{u}}(\boldsymbol{x}^{(k)}(t), \boldsymbol{u}^{(k)}(t), \boldsymbol{p}^{(k)}(t), t)\boldsymbol{v}^{(k)}(t) + \frac{1}{2}a^{(k)}(t)\boldsymbol{y}^{(k)}(t)^T\boldsymbol{y}^{(k)}(t) \\ &\quad + \frac{1}{2}a^{(k)}(t)\boldsymbol{v}^{(k)}(t)^T\boldsymbol{v}^{(k)}(t) + \boldsymbol{q}^{(k)}(t)^T \left(\boldsymbol{f}_{\boldsymbol{x}}^{(k)}(t)\boldsymbol{y}^{(k)}(t) + \boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\boldsymbol{v}^{(k)}(t)\right)\end{aligned} \tag{3.53}$$

with $\bar{L}$ substituted by (3.49). The costate equation (3.52) then becomes

$$-\dot{\bar{\boldsymbol{q}}}^{(k)}(t) = \boldsymbol{f}_{\boldsymbol{x}}^{(k)}(t)^T\bar{\boldsymbol{q}}^{(k)}(t) + a^{(k)}(t)\bar{\boldsymbol{y}}^{(k)}(t). \tag{3.54}$$

Moreover, there exist undetermined constants $\bar{\eta}_i^{(k)}$ such that the following transversality condition holds,

$$\bar{\boldsymbol{q}}^{(k)}(t_f) = \sum_{i=1}^{r} \bar{\eta}_i^{(k)} g_{i\boldsymbol{x}}(\boldsymbol{x}^{(k)}(t_f))^T \tag{3.55}$$

where

$$\bar{\eta}_i^{(k)} \geq 0 \quad \text{and} \quad \bar{\eta}_i^{(k)} \left(g_i(\boldsymbol{x}^{(k)}(t_f)) + g_{i\boldsymbol{x}}(\boldsymbol{x}^{(k)}(t_f)) \boldsymbol{y}^{(k)}(t_f) \right) = 0 \tag{3.56}$$

for all $i = 1, \ldots, r$. The function $g_{i\boldsymbol{x}}(\boldsymbol{x}^{(k)}(t_f))$ is the gradient of the terminal state constraint g_i, defined in (3.17), evaluated at $\boldsymbol{x}^{(k)}(t_f)$.

In addition, similar to (3.35(b)), $\bar{\boldsymbol{v}}^{(k)}(t)$ is the solution of the control equation

$$\bar{H}_{\boldsymbol{v}}(\bar{\boldsymbol{y}}^{(k)}(t), \bar{\boldsymbol{v}}^{(k)}(t), \bar{\boldsymbol{q}}^{(k)}(t), t) = \boldsymbol{0} .$$

which results in

$$H_{\boldsymbol{u}}(\boldsymbol{x}^{(k)}(t), \boldsymbol{u}^{(k)}(t), \boldsymbol{p}^{(k)}(t), t)^T + a^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t) + \boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)^T \bar{\boldsymbol{q}}^{(k)}(t) = \boldsymbol{0} \tag{3.57}$$

using (3.53). By the Kuhn-Tucker Theorem [51], there exist nonnegative vector functions $\beta^{(k)}(t)$ and $\gamma^{(k)}(t)$, such that, for any $t \in \mathcal{T}$,

$$\begin{aligned} H_{\boldsymbol{u}}(\boldsymbol{x}^{(k)}(t), \boldsymbol{u}^{(k)}(t), \boldsymbol{p}^{(k)}(t), t)^T + a^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t) \\ + \boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)^T \bar{\boldsymbol{q}}^{(k)}(t) + \boldsymbol{\beta}^{(k)}(t) - \boldsymbol{\gamma}^{(k)}(t) = \boldsymbol{0} \end{aligned} \tag{3.58}$$

where

$$\boldsymbol{\beta}^{(k)}(t)^T \left(\bar{\boldsymbol{u}}^{(k)}(t) - \overline{\boldsymbol{u}} \right) = 0 \tag{3.59a}$$

$$\boldsymbol{\gamma}^{(k)}(t)^T \left(\underline{\boldsymbol{u}} - \bar{\boldsymbol{u}}^{(k)}(t) \right) = 0 . \tag{3.59b}$$

The functions $\overline{\boldsymbol{u}}$ and $\underline{\boldsymbol{u}}$ are the upper and lower limits on the control inputs, specified in (3.19).

Next, it will be shown that $\bar{\boldsymbol{v}}^{(k)}(t)$, the result of the direction-finding subproblem, turns out to be a descent direction of the cost functional $\tilde{J}(\boldsymbol{u})$. That is, if $\bar{\boldsymbol{v}}^{(k)}(t) \neq 0$ and $\alpha^{(k)} \in (0, 1]$, then

$$\tilde{J}(\boldsymbol{u}^{(k)} + \alpha\bar{\boldsymbol{v}}^{(k)}) - \tilde{J}(\boldsymbol{u}^{(k)}) \; < \; 0$$

for all $0 < \alpha \leq \alpha^{(k)}$.

From (3.51) and (3.39),

$$\begin{aligned} \tilde{J}(\boldsymbol{u}^{(k)} + \alpha\bar{\boldsymbol{v}}^{(k)}) - \tilde{J}(\boldsymbol{u}^{(k)}) \; &\leq \; \Delta\tilde{J}(\alpha\bar{\boldsymbol{v}}^{(k)}) + c\|\alpha\bar{\boldsymbol{v}}^{(k)}\|^2 \\ &\leq \; \alpha\int_{t_0}^{t_f} H_{\boldsymbol{u}}^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)dt + c\alpha^2\|\bar{\boldsymbol{v}}^{(k)}\|^2 \end{aligned} \quad (3.60)$$

where, for simplicity,

$$H_{\boldsymbol{u}}^{(k)}(t) = H_{\boldsymbol{u}}(\boldsymbol{x}^{(k)}(t), \boldsymbol{u}^{(k)}(t), \boldsymbol{p}^{(k)}(t), t)\,.$$

Substituting $H_{\boldsymbol{u}}^{(k)}(t)$ from (3.58) in (3.60) yields

$$\begin{aligned} \tilde{J}(\boldsymbol{u}^{(k)} + \alpha\bar{\boldsymbol{v}}^{(k)}) - \tilde{J}(\boldsymbol{u}^{(k)}) \; \leq \\ -\alpha\int_{t_0}^{t_f} \left(a^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)^T\bar{\boldsymbol{v}}^{(k)}(t) + \bar{\boldsymbol{q}}^{(k)}(t)^T\boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)\right) dt \\ -\alpha\int_{t_0}^{t_f} \left(\boldsymbol{\beta}^{(k)}(t)^T\bar{\boldsymbol{v}}^{(k)}(t) - \boldsymbol{\gamma}^{(k)}(t)^T\bar{\boldsymbol{v}}^{(k)}(t)\right) dt + c\alpha^2\|\bar{\boldsymbol{v}}^{(k)}\|^2. \end{aligned} \quad (3.61)$$

Note that

$$\begin{aligned} \boldsymbol{\beta}^{(k)}(t)^T\bar{\boldsymbol{v}}^{(k)}(t) \; &= \; \boldsymbol{\beta}^{(k)}(t)^T\left(\bar{\boldsymbol{u}}^{(k)}(t) - \boldsymbol{u}^{(k)}(t)\right) \\ &= \; \boldsymbol{\beta}^{(k)}(t)^T\left(\bar{\boldsymbol{u}}^{(k)}(t) - \overline{\boldsymbol{u}}\right) + \boldsymbol{\beta}^{(k)}(t)^T\left(\overline{\boldsymbol{u}} - \boldsymbol{u}^{(k)}(t)\right) \geq 0 \end{aligned}$$

because

$$\begin{aligned} \boldsymbol{\beta}^{(k)}(t)^T\left(\bar{\boldsymbol{u}}^{(k)}(t) - \overline{\boldsymbol{u}}\right) \; &= \; 0 \\ \boldsymbol{\beta}^{(k)}(t) \; &\geq \; 0 \\ \left(\overline{\boldsymbol{u}} - \boldsymbol{u}^{(k)}(t)\right) \; &\geq \; 0\,. \end{aligned}$$

Similarly,

$$\begin{aligned}\boldsymbol{\gamma}^{(k)}(t)^T\bar{\boldsymbol{v}}^{(k)}(t) &= \boldsymbol{\gamma}^{(k)}(t)^T\left(\bar{\boldsymbol{u}}^{(k)}(t)-\boldsymbol{u}^{(k)}(t)\right)\\ &= \boldsymbol{\gamma}^{(k)}(t)^T\left(\bar{\boldsymbol{u}}^{(k)}(t)-\underline{\boldsymbol{u}}\right)+\boldsymbol{\gamma}^{(k)}(t)^T\left(\underline{\boldsymbol{u}}-\boldsymbol{u}^{(k)}(t)\right)\le 0\end{aligned}$$

because

$$\begin{aligned}\boldsymbol{\beta}^{(k)}(t)^T\left(\bar{\boldsymbol{u}}^{(k)}(t)-\underline{\boldsymbol{u}}\right) &= 0\\ \boldsymbol{\gamma}^{(k)}(t) &\ge 0\\ \left(\underline{\boldsymbol{u}}-\boldsymbol{u}^{(k)}(t)\right) &\le 0\,.\end{aligned}$$

Therefore, the inequality (3.61) can be reduced to

$$\begin{aligned}\tilde{J}(\boldsymbol{u}^{(k)}+\alpha\bar{\boldsymbol{v}}^{(k)})-\tilde{J}(\boldsymbol{u}^{(k)}) \le\ & -\alpha\int_{t_0}^{t_f}\left(a^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)^T\bar{\boldsymbol{v}}^{(k)}(t)\right)dt+c\alpha^2\|\bar{\boldsymbol{v}}^{(k)}(t)\|^2\\ & -\alpha\int_{t_0}^{t_f}\left(\bar{\boldsymbol{q}}^{(k)}(t)^T\boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)\right)dt\,. \qquad (3.62)\end{aligned}$$

To further simplify (3.62), consider

$$\begin{aligned}\frac{d}{dt}\left(\bar{\boldsymbol{q}}^{(k)}(t)^T\bar{\boldsymbol{y}}^{(k)}(t)\right) &= \dot{\bar{\boldsymbol{q}}}^{(k)}(t)^T\bar{\boldsymbol{y}}^{(k)}(t)+\bar{\boldsymbol{q}}^{(k)}(t)^T\dot{\bar{\boldsymbol{y}}}^{(k)}(t)\\ &= \bar{\boldsymbol{q}}^{(k)}(t)^T\boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)-a^{(k)}(t)\bar{\boldsymbol{y}}^{(k)}(t)^T\bar{\boldsymbol{y}}^{(k)}(t)\,, \qquad (3.63)\end{aligned}$$

where $\dot{\bar{\boldsymbol{y}}}^{(k)}(t)$ and $\dot{\bar{\boldsymbol{q}}}^{(k)}(t)$ are replaced by (3.41(a)) and (3.54). So,

$$\begin{aligned}\int_{t_0}^{t_f}\bar{\boldsymbol{q}}^{(k)}(t)^T\boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)dt &= \bar{\boldsymbol{q}}^{(k)}(t)^T\bar{\boldsymbol{y}}^{(k)}(t)\Big|_{t_0}^{t_f}+\int_{t_0}^{t_f}a^{(k)}(t)\bar{\boldsymbol{y}}^{(k)}(t)^T\bar{\boldsymbol{y}}^{(k)}(t)dt\\ &> \bar{\boldsymbol{q}}^{(k)}(t_f)^T\bar{\boldsymbol{y}}^{(k)}(t_f) \qquad (3.64)\end{aligned}$$

given the fact that $\bar{\boldsymbol{y}}^{(k)}(t_0)=0$, and $a^{(k)}(t)\bar{\boldsymbol{y}}^{(k)}(t)^T\bar{\boldsymbol{y}}^{(k)}(t)$ is always positive. Using (3.55) to replace $\bar{\boldsymbol{q}}^{(k)}(t_f)$ in (3.64) reads

$$\int_{t_0}^{t_f}\bar{\boldsymbol{q}}^{(k)}(t)^T\boldsymbol{f}_{\boldsymbol{u}}^{(k)}(t)\bar{\boldsymbol{v}}^{(k)}(t)dt > \sum_{i=1}^{r}\bar{\eta}_i^{(k)}g_{i\boldsymbol{x}}(\boldsymbol{x}^{(k)}(t_f))\bar{\boldsymbol{y}}^{(k)}(t_f)\,,$$

following

$$\int_{t_0}^{t_f} \bar{q}^{(k)}(t)^T f_u^{(k)}(t)\bar{v}^{(k)}(t)dt \;>\; -\sum_{i=1}^{r} \bar{\eta}_i^{(k)} g_i(x^{(k)}(t_f)) \;\geq\; 0 \qquad (3.65)$$

from (3.56) and (3.17). This result can now be used to simplify the inequality (3.62) as

$$\tilde{J}(u^{(k)} + \alpha\bar{v}^{(k)}) - \tilde{J}(u^{(k)}) < -\alpha \int_{t_0}^{t_f} a^{(k)}(t)\bar{v}^{(k)}(t)^T \bar{v}^{(k)}(t)dt + c\alpha^2 \|\bar{v}^{(k)}(t)\|^2.$$

Hence, by choosing a small enough stepsize $\alpha^{(k)}$, such that

$$0 \;<\; \alpha^{(k)} \;\leq\; \frac{1}{c\|\bar{v}^{(k)}(t)\|^2} \int_{t_0}^{t_f} a^{(k)}(t)\bar{v}^{(k)}(t)^T \bar{v}^{(k)}(t)dt\,. \qquad (3.66)$$

the following relation always holds

$$\tilde{J}(u^{(k)} + \alpha\bar{v}^{(k)}) - \tilde{J}(u^{(k)}) \;<\; 0$$

for all $0 < \alpha \leq \alpha^{(k)}$ and $\bar{v}^{(k)} \neq 0$. This inequality shows that whenever $\bar{u}^{(k)} \neq u^{(k)}$, $\bar{v}^{(k)} = \bar{u}^{(k)} - u^{(k)}$ is a descent direction of the cost functional $\tilde{J}(u)$ at the k-th iteration. □

The following important concluding remarks are in order:

- Because $\tilde{\lambda}(t)$ is positive and $\bar{v}^{(k)} \neq 0$, the upper bound in (3.66) is always well defined and positive. This ensures existence of a positive stepsize $\alpha^{(k)}$ at each iteration.

- The descent property shown in this section will always hold as long as the function $\tilde{\lambda}(t)$ is positive, regardless of whether the inequality relationship (3.44) is satisfied. However, this inequality is important for the rate of the convergence of the algorithm. Intuitively, as an analog to the Han-Powell method in finite-dimensional optimization [41],[43], tighter approximation by (3.44) makes the rate of convergence closer to second-order, while looser approximations would make the algorithm behave more like a first-order algorithm.

3.4.5 Computation of descent directions

According to the descent property, the solution of the direction-finding subproblem (**DFS**), which is a linearly constrained quadratic optimal control, is always a descent direction at each iteration. If there were neither control nor state constraints, **DFS** would be just a classical time-varying LQR which could be solved by integrating two Riccati equations [52],[24]. A convenient way to solve **DFS**, with both control and terminal state constraints, is to convert it into a standard quadratic programming form in finite dimension by discretizing the control variables.

Let the time interval $[t_0, t_f)$ be divided into N equal subintervals $[t_{j-1}, t_j)$, with $t_j = t_0 + jT$ and $T = (t_f - t_0)/N$ $(j = 0, \ldots, N)$. During each subinterval, control variables are approximated by constant vectors

$$\boldsymbol{u}(t) \approx \boldsymbol{u}(t_{j-1}). \quad t \in [t_{j-1}, t_j) \tag{3.67}$$

for $j = 1, \ldots, N$. Note that each control parameter $\boldsymbol{u}(t_j) \in \Pi$, according to (3.19). The above approximation leads to a spline representation of the controls [53] as

$$\boldsymbol{u}(t) \approx \sum_{j=0}^{N-1} \boldsymbol{u}_j \phi_j(t)$$

with $\boldsymbol{u}_j = \boldsymbol{u}(t_j)$ and the pulse functions $\phi_j(t)$ defined by

$$\phi_j(t) \triangleq \begin{cases} 1, & \text{when } t \in [t_j, t_{j+1}) \\ 0, & \text{otherwise} \end{cases}.$$

Assuming that the Jacobians $\boldsymbol{f_x}$ and $\boldsymbol{f_u}$ are approximately constant over each subinterval $[t_j, t_{j+1})$, from the theory of linear systems [24], the continuous state equations (3.41) can be converted into the discrete state equations

$$\boldsymbol{y}_{j+1}^{(k)} = \boldsymbol{A}_j^{(k)} \boldsymbol{y}_j^{(k)} + \boldsymbol{B}_j^{(k)} \left(\boldsymbol{u}_j - \boldsymbol{u}_j^{(k)} \right) \tag{3.68a}$$

$$\boldsymbol{y}_0^{(k)} = \boldsymbol{0} \tag{3.68b}$$

where

$$\boldsymbol{A}_j^{(k)} = e^{T\boldsymbol{f}_{\boldsymbol{x}}^{(k)}(t_j)} \quad , \quad \boldsymbol{B}_j^{(k)} = \boldsymbol{f}_{\boldsymbol{u}}(t_j) \int_{t_j}^{t_{j+1}} e^{\boldsymbol{f}_{\boldsymbol{x}}^{(k)}(t_{j+1}-\tau)} d\tau \ ,$$

and $\boldsymbol{u}_j^{(k)} = \boldsymbol{u}^{(k)}(t_j)$, $\boldsymbol{y}_j^{(k)} = \boldsymbol{y}^{(k)}(t_j)$ are all evaluated at $t = t_j$ for $j = 0, \ldots, N-1$. Consequently, the solution of the above difference equation for $\boldsymbol{y}_j^{(k)}$ can be expressed as

$$\boldsymbol{y}_j^{(k)} = \sum_{i=1}^{j} \boldsymbol{\Phi}_{j,i}^{(k)} \boldsymbol{B}_{i-1}^{(k)} \left(\boldsymbol{u}_{i-1} - \boldsymbol{u}_{i-1}^{(k)} \right) \tag{3.69}$$

with the *transition matrix*

$$\boldsymbol{\Phi}_{j,i}^{(k)} = \prod_{p=i}^{j-1} \boldsymbol{A}_p^{(k)} \quad , \quad \boldsymbol{\Phi}_{j,j}^{(k)} = \boldsymbol{I} . \tag{3.70}$$

Introducing the *control parameter vector*

$$\boldsymbol{\xi} \triangleq \left[\boldsymbol{u}_0^T, \ldots, \boldsymbol{u}_{N-1}^T \right]^T \in \Re^{mN}. \tag{3.71}$$

the solution (3.69) can be written in vector form of

$$\boldsymbol{y}_j^{(k)} = \boldsymbol{D}_j^{(k)} \left(\boldsymbol{\xi} - \boldsymbol{\xi}^{(k)} \right) \tag{3.72}$$

where $\boldsymbol{\xi}^{(k)}$ consists of $u_i^{(k)}$ $(i = 0, \ldots, N-1)$, and the matrix $\boldsymbol{D}_j^{(k)}$ is defined by

$$\boldsymbol{D}_j^{(k)} = \left[\boldsymbol{\Phi}_{j,1}^{(k)} \boldsymbol{B}_0^{(k)}, \ldots, \boldsymbol{\Phi}_{j,j}^{(k)} \boldsymbol{B}_{j-1}^{(k)}, \boldsymbol{0}, \ldots . \boldsymbol{0} \right] \in \Re^n \times \Re^{mN} \tag{3.73}$$

for $j = 1, \ldots, N$. Thus, by applying the rectangular integration rule to (3.48), the cost functional $\bar{J}^{(k)}$ becomes the following quadratic function

$$\bar{J}^{(k)}(\boldsymbol{\xi} - \boldsymbol{\xi}^{(k)}) = \frac{1}{2} \left(\boldsymbol{\xi} - \boldsymbol{\xi}^{(k)} \right)^T \boldsymbol{Q}^{(k)} \left(\boldsymbol{\xi} - \boldsymbol{\xi}^{(k)} \right) + \boldsymbol{q}^{(k)} \left(\boldsymbol{\xi} - \boldsymbol{\xi}^{(k)} \right) \tag{3.74}$$

where

$$\begin{aligned} \boldsymbol{q}^{(k)} &= T \left[H_{\boldsymbol{u}}^{(k)}(t_0), \ldots, H_{\boldsymbol{u}}^{(k)}(t_{N-1}) \right] \\ \boldsymbol{Q}^{(k)} &= T \sum_{j=1}^{N-1} a^{(k)}(t_j) \left(\boldsymbol{I} + \boldsymbol{D}_j^{(k)T} \boldsymbol{D}_j^{(k)} \right) . \end{aligned} \tag{3.75}$$

In the above, the time-varying scalar $\tilde{\lambda}(t)$ and gradient $H_{\boldsymbol{u}}(t)$ are parameterized over the time interval $[t_0, t_f)$. By removing the constant terms in (3.74), the quadratic cost function can be simplified as

$$\bar{J}^{(k)}(\boldsymbol{\xi}) = \frac{1}{2}\boldsymbol{\xi}^T \boldsymbol{Q}^{(k)} \boldsymbol{\xi} + \left(\boldsymbol{q}^{(k)} - \boldsymbol{\xi}^{(k)^T} \boldsymbol{Q}^{(k)}\right) \boldsymbol{\xi} \tag{3.76}$$

Similarly, the terminal constraints (3.42) turn into the following linear inequality constraint

$$\boldsymbol{g}(\boldsymbol{x}^{(k)}(t_f)) + \boldsymbol{g}_{\boldsymbol{x}}(\boldsymbol{x}^{(k)}(t_f)) \boldsymbol{D}_N^{(k)} \left(\boldsymbol{\xi} - \boldsymbol{\xi}^{(k)}\right) \leq \mathbf{0} \tag{3.77}$$

with $\boldsymbol{g} \triangleq [g_1, \ldots, g_l]^T$.

The original direction finding subproblem (**DFS**) can now be formulated in finite optimization as follows:

Linear quadratic programming (LQP):

- Given $\boldsymbol{\xi}^{(k)}$, find a control parameter vector $\boldsymbol{\xi}$ such that the quadratic cost function (3.76) is minimized subject to the linear constraint (3.77).

Notice that the matrix $\boldsymbol{Q}^{(k)}$ is always symmetric and positive definite at each iteration due to (3.75). Therefore, **LQP** is a standard quadratic programming problem with linear constraints, which can be solved by many well-known techniques in finite optimization [47],[51]. Another advantage of the above parameterization technique is that there are many good codes available to solve a quadratic programming problem such as **LQP**. Some of them are available free of charge, for example, the Fortran code "E04NAF", a NAG library routines [54], and the MATLAB function "QP" [55]. In this research, the latter, which uses an active set method [47], has been employed to solve the above optimization problem. It apeared to converge

satisfactorily in the three cases studied in Chapter 4. However, comparison with other routines, such as "E04NAF" and "LOQO" [56], could be useful.

Once the optimal solution $\bar{\xi}^{(k)}$ of **LQP** is found, its elements can be used to construct the optimal control $\bar{u}^{(k)}$, similarly to (3.67). This optimal control is in fact an approximate solution for **DFS** at the k-th iteration.

3.4.6 Stepsize rule

Ideally, the best stepsize at each iteration is the one which minimizes the cost functional $\tilde{J}(u^{(k)} + \alpha\bar{v}^{(k)})$. That is, at the k-th iteration,

$$\alpha^{(k)} = \arg\min_{\alpha \geq 0} \tilde{J}(u^{(k)} + \alpha\bar{v}^{(k)}) . \tag{3.78}$$

Consequently, according to Step 7 of the algorithm given in Section 3.4.3, the next control update will be

$$u^{(k+1)}(t) = u^{(k)}(t) + \alpha^{(k)}\bar{v}^{(k)}(t)$$

where $\bar{v}^{(k)}(t) = \bar{u}^{(k)}(t) - u^{(k)}(t)$.

From the descent property, whenever $\bar{v}^{(k)} \neq 0$, the stepsize defined in (3.78) is a positive number. Moreover, as mentioned in Section 3.4.3, $\alpha^{(k)} \leq 1$, so that the algorithm generates a sequence of admissible controls. However, the calculation of the above exact stepsize is very expensive. In practice, an inaccurate line search is used.

Consider now the stepsize

$$\alpha^{(k)} = \max\left\{1 \, , \, \frac{1}{2c\|\bar{v}^{(k)}(t)\|^2}\int_{t_0}^{t_f} a^{(k)}(t)\bar{v}^{(k)}(t)^T\bar{v}^{(k)}(t)dt\right\} \tag{3.79}$$

where c is the constant of the remainder term defined in (3.51(b)). The above rule generates a stepsize equal to half of the upper bound limit in (3.66), unless it returns 1. Therefore, $\alpha^{(k)} \in (0, 1]$ while always satisfying condition (3.66).

3.4.7 Free final time

The optimization algorithm given in Section 3.4.3 assumes a fixed final time t_f. However, the free end-time optimal control problem of Section 3.4 can be easily converted to a fixed end-time optimal control problem with an additional state. The idea is to specify a nominal time interval, $[0, 1]$, for the problem and to use a scale parameter, adjustable by the optimization algorithm, to scale the system dynamics and, in effect, scale the duration of the time interval. The scale parameter is represented by the extra state. To perform this, a transformation by G. Leitmann is adopted below.

Let the time variable t be a linear transformation of the new time variable τ as

$$t(\tau) \;=\; w\tau + t_0 \;. \tag{3.80}$$

with

$$dt(\tau) \;=\; w d\tau$$

where w is the *duration scale parameter*. Then, the time interval $[t_0, t_f]$ transforms to the new time interval $[0, 1]$ with $t_f = w + t_0$. Usually in real applications of free end-time optimal control, such as the crane problem, a good engineering estimate for the range of t_f is available, that is

$$T_1 \;\leq\; t_f \;\leq\; T_2$$

or

$$T_1 - t_0 \;\leq\; w \;\leq\; T_2 - t_0 \tag{3.81}$$

for some given $T_2 > T_1 > t_0$.

Consider the augmented dynamical system

$$\dot{\hat{\boldsymbol{x}}}(\tau) = \hat{\boldsymbol{f}}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}, \tau) \triangleq \begin{pmatrix} \hat{x}_{n+1} \boldsymbol{f}(\hat{\boldsymbol{x}}(\tau), \hat{\boldsymbol{u}}(\tau), \tau \hat{x}_{n+1} + t_0) \\ 0 \end{pmatrix} \;,\; \hat{\boldsymbol{x}}(0) = \begin{pmatrix} \boldsymbol{x}_0 \\ w \end{pmatrix} \tag{3.82}$$

with $\hat{\boldsymbol{x}}(\tau) \triangleq (\boldsymbol{x}(w\tau + t_0)^T, w)^T$, $\hat{\boldsymbol{u}}(\tau) \triangleq \boldsymbol{u}(w\tau + t_0)$, and $\boldsymbol{f}$ and $\boldsymbol{x}_0$ given in (3.16). Because the scale factor w is an unknown constant, unlike x_0, the initial value $\hat{x}_{n+1}(0) = w$ is not fixed. The inequality constraint (3.81) in conjunction with (3.17) form a new set of terminal state constraints as

$$\hat{g}_i(w, \hat{\boldsymbol{x}}(t_f)) \leq 0 \quad i = 1, \ldots, r+1 . \tag{3.83}$$

The free end-time optimal control problem of Section 3.4 can now be converted into the following equivalent fixed end-time optimal control problem.

- Find a control $\hat{\boldsymbol{u}}(\tau) : [0, 1] \rightarrow \Pi$ and a scale parameter w such that the cost functional

$$\hat{J}(\hat{\boldsymbol{u}}, w) = \int_0^1 \hat{L}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}, \tau) d\tau = \int_0^1 \hat{x}_{n+1} L(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}, \tau \hat{x}_{n+1} + t_0) d\tau \tag{3.84}$$

  is minimized subject to the equations (3.82) and the constraints (3.83) and (3.18).

According to the optimality conditions derived in Section 3.4.2, with the costate vector $\hat{\boldsymbol{p}}$, the associated costate equation is

$$\dot{\hat{\boldsymbol{p}}}(\tau) = -\hat{H}_{\hat{\boldsymbol{x}}}(\hat{\boldsymbol{x}}(\tau), \hat{\boldsymbol{u}}(\tau), \hat{\boldsymbol{p}}(\tau), \tau)^T \tag{3.85}$$

where

$$\hat{H}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}, \hat{\boldsymbol{p}}, \tau) \triangleq \hat{L}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}, \tau) + \hat{\boldsymbol{p}}^T \hat{\boldsymbol{f}}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}, \tau) .$$

Similar to the case of finding the optimal control $\hat{\boldsymbol{u}}^*$, the optimal duration w^* may be found through the iterative process of an optimization procedure. The algorithm of Section 3.4.3 has been modified to accommodate updating the scale parameter w at every iteration, as follows.

Modified optimization algorithm:

1. Select an initial control $\hat{\boldsymbol{u}}^{(0)} \in \Pi$ and an initial scale parameter $w^{(0)} \in [T_1 - t_0, T_2 - t_0]$. Set $k = 0$.

2. Compute state $\hat{\boldsymbol{x}}^{(k)}$ by integrating (3.82) forward.

3. Compute costate $\hat{\boldsymbol{p}}^{(k)}$ by integrating (3.85) backward.

4. Solve a corresponding direction-finding subproblem to find $\bar{\boldsymbol{u}}^{(k)}$ and $\bar{w}^{(k)}$.

5. If $\bar{\boldsymbol{u}}^{(k)} \simeq \hat{\boldsymbol{u}}^{(k)}$ and $\bar{w}^{(k)} \simeq w^{(k)}$ within a certain accuracy range, then set $\hat{\boldsymbol{u}}^* = \bar{\boldsymbol{u}}^{(k)}$, $w^* = \bar{w}^{(k)}$,and stop. Otherwise, go to Step 6.

6. Compute a suitable stepsize $\alpha^{(k)}$, according to a stepsize rule.

7. Set $\hat{\boldsymbol{u}}^{(k+1)} = \hat{\boldsymbol{u}}^{(k)} + \alpha^{(k)} \left(\bar{\boldsymbol{u}}^{(k)} - \hat{\boldsymbol{u}}^{(k)}\right)$ and $w^{(k+1)} = w^{(k)} + \alpha^{(k)} \left(\bar{w}^{(k)} - w^{(k)}\right)$. Then set $k = k + 1$ and return to Step 2.

Once the optimal solutions $\hat{\boldsymbol{u}}^*$, $\hat{\boldsymbol{x}}^*$ and w^* are found, the original controls and states u_i^* and x_j^* will be available by reverse scaling as

$$\begin{aligned} u_i^*(t) &= \hat{u}_i^* \left(\frac{t - t_0}{w^*}\right) \quad \text{for } i = 1, \ldots, m \\ x_j^*(t) &= \hat{x}_j^* \left(\frac{t - t_0}{w^*}\right) \quad \text{for } j = 1, \ldots, n, \end{aligned} \tag{3.86}$$

with the final time $t_f = w^* + t_0$.

The main disadvantage of this method, in general, is that it converts linear systems into nonlinear systems. However, this may not be a concern because the crane dynamics presented in Chapter 2 is already nonlinear.

3.4.8 Path constraints

The definition of the general optimal control problem of Section 3.4 allows path constraints of the form (3.18) to be handled directly. However, constraints of this form are quite burdensome computationally. This is mainly due to the fact that a separate gradient calculation must be performed for each point at which the path constraint is evaluated. In what follows, path constraints are converted into end-point constraints which are computationally much easier to handle.

Let the system equations be augmented with an extra state variable x' satisfying

$$\dot{x}'(t) = \sum_{i=1}^{l} \mu_i \max\{0, \varphi_i(\boldsymbol{x}(t), t)\}^2 \tag{3.87}$$

where the positive scalar μ_i serves as a *weighting factor* for every constraint φ_i. The right-hand side is squared so that it is differentiable with respect to $\boldsymbol{x}$. Then it is clear that the end-point condition

$$x'(t_f) \leq 0 \tag{3.88}$$

holds if and only if all of the path constraints $\varphi_i(\boldsymbol{x}(t), t) \leq 0$ $(i = 1, \ldots, l)$ are satisfied. The end-point condition (3.88) is in the form of (3.17).

In the implementation of the above technique to replace the crane trajectory constraints, the weighting factors μ_i $(i = 1, \ldots, l)$ in (3.87) have been considered equal to 1, assuming the same weights for all of the constraints.

3.5 Summary

This chapter started with the definition of a cost function for the crane optimization problem based on specific criteria to achieve fast load transfer with suppressed load

swing. Combined with a set of operational and safety constraints given in Section 3.2, the crane optimization problem was then described in Section 3.3.

The crane optimization problem was considered as a special case of a constrained Lagrange problem introduced in Section 3.4. Because of the complex nature of the crane problem due to its nonlinear dynamics, a fast and efficient algorithm was needed to find the solution of the optimization problem. This task was accomplished through the rest of this chapter, beginning with the derivation of the optimality conditions in Section 3.4.2. The developed iterative algorithm updates the control variables by solving a simple linear quadratic problem at each iteration. This is due to an upper limit approximation on the second variation of the original cost functional, which was adapted from the Han-Powell method in finite-dimensional optimization. It was then shown, in Section 3.4.4, that the proposed algorithm generates a descent direction at each iteration. To compute such a descent direction, a standard quadratic programming problem in finite optimization has to be solved at each iteration.

This chapter is concluded by inclusion of the cases of free final time and path constraints in Sections 3.4.7 and 3.4.8.

Chapter 4

Crane optimal trajectories

In Chapter 2 the state-space equations of a simplified model for the tower crane dynamics were derived. In Chapter 3 an optimization algorithm for solving a particular class of constrained optimal control problems was developed. In this Chapter, the developed algorithm will be used to find the optimal trajectories for two different types of crane motions.

While the optimal solutions provide fast motions with small load swings, it is desired to compare quantitatively their performance with some non-optimal trajectories representative of a conventional manually controlled tower crane. The comparisons are expected to show whether, in terms of the transport time and load swing, the optimal load transfers can offer significant improvement in the operations of a tower crane.

First a minimum-time load transfer between two given points is studied in Section 4.2. No boom slew or change in the hoist cable length is considered in this basic motion. The numerical result reasonably matches the analytical result given in [2] [57], showing how well the optimization algorithm works. Despite small load

swing of the time-optimal solution, by adding a function of load swing into the cost function, the swing is further reduced to meet the safety requirement of a manually controlled tower crane. Finally, the resulting optimal solution is compared with a typical motion of a conventional crane to illustrate the performance of the optimal load displacement.

The next simulation, in Section 4.4, involves all of the three crane motions: the trolley movement, the boom rotation and the load lift. While no analytical solution is available for this complex motion, the optimization algorithm is applied to compute the optimal trajectory. This result will then be compared with a non-optimal trajectory, typical of a manually controlled tower crane, showing significant reduction in both transfer time and load swing.

In this Chapter, all system responses are simulated and generated in the MATLAB environment. Also, all codes for the optimization algorithm were written in MATLAB. A simulation/animation program, linked with SIMULINK, was coded in MATLAB to facilitate the simulation of a conventional tower crane operations, mainly due to lack of a crane prototype. The next page picture shows the animated tower crane simulating the control joysticks and data monitoring system of a manually controlled tower crane.

4.1 Computational characteristics of the optimization program

There are three simulation results in this Chapter that have been obtained using the optimization algorithm of Section 3.4.7. The characteristics of the optimization program that implements the proposed algorithm include the following:

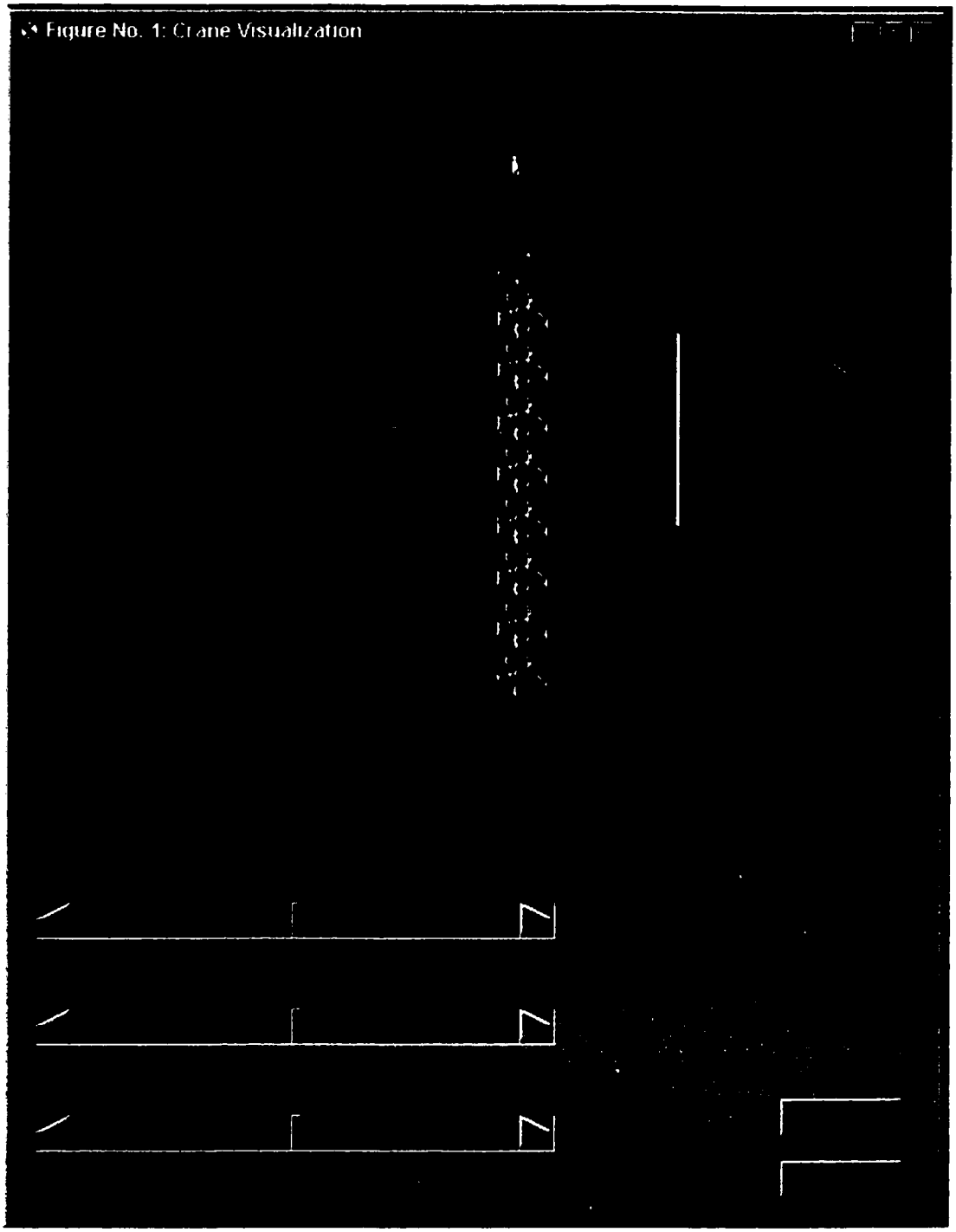


Figure 4.1: Tower crane simulator

- For simplicity, the classical fourth-order Runge-Kutta formula, with fixed time stepsizes, is used for integrating both state and costate differential equations [58].

- For simplicity, the time sequence used to integrate the state and costate equations is the same as the one used to solve the direction-finding subproblem (**DFS**).

- During simulations of the optimization algorithm, it was found that solving a constrained optimal control problem with the end-point condition (3.14(b)) would require a considerable increase in the number of iterations to obtain optimal solutions. Therefore, at the expense of slight constraint violation and reduced accuracy, the end-point condition (3.14) was replaced by

$$(\boldsymbol{x}(t_f) - \boldsymbol{x}_f)^T (\boldsymbol{x}(t_f) - \boldsymbol{x}_f) \leq \zeta \tag{4.1}$$

  where ζ is a small positive number.

4.2 Time-optimal planar motion

To illustrate the crane optimization formulated in Section 3.3, a simple case, for which an analytical (exact) solution exists [2] [57], is studied in this section. The analytical solution will be used to verify computation of the optimal trajectory. The author believes that this special case can provide a clear and quick understanding of some of the advantages of an optimal trajectory over a non-optimal trajectory.

For the crane represented by the state equations (2.18), suppose that a load hanging from the trolley at a constant length (r_0) is initially at a motionless position,

i.e.

$$\boldsymbol{x}_0 = (R_0, \psi_0, r_0, 0, 0, 0, 0, 0, 0, 0)^T \quad \text{at } t = 0 \tag{4.2}$$

$$\begin{cases} x_8(t) = \dot{r}(t) = 0 \\ u_3(t) = \ddot{r}(t) = 0 \end{cases} \quad \text{for } 0 \le t \le t_f \,. \tag{4.3}$$

The task is to move the trolley along the boom toward a target point $x_1(t_f) = R_f$ where the system comes to rest again as quickly as possible. that is

$$\boldsymbol{x}_f = (R_f, \psi_0, r_0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0)^T \quad \text{at } t = t_f. \tag{4.4}$$

The final time is to be minimized, expressed in integral form as

$$t_f = \int_0^{t_f} d\tau \,. \tag{4.5}$$

The end-point condition (4.4) is required for a motionless unloading configuration at the target point.

No slewing motion is allowed. so that

$$\begin{cases} x_7(t) = \dot{\psi}(t) = 0 \\ u_2(t) = \ddot{\psi}(t) = 0 \end{cases} \quad \text{for } 0 \le t \le t_f \,. \tag{4.6}$$

The load therefore acts as a simple pendulum while the trolley is traveling along the boom. This minimum-time optimization is a special case of the crane optimization described in Section 3.3, with $b = 0$ and $a = 1$.

Using (4.3) and (4.6), the state equations (2.18) are reduced to

$$\begin{aligned} \dot{x}_1(t) &= x_6(t) \\ \dot{x}_5(t) &= x_{10}(t) \\ \dot{x}_6(t) &= u_1(t) \\ \dot{x}_{10}(t) &= -\left(g\, x_5(t) + u_1(t)\right)/r_0 \end{aligned} \qquad \text{for } 0 \le t \le t_f \tag{4.7}$$

where g is the gravity acceleration. With the help of four costate variables p_1, p_5, p_6 and p_{10}, the corresponding costate equations are

$$\begin{aligned} \dot{p}_1(t) &= 0 \\ \dot{p}_5(t) &= g\,p_{10}(t)/r_0 \\ \dot{p}_6(t) &= -p_1(t) \\ \dot{p}_{10}(t) &= -p_5(t) \end{aligned} \qquad \text{for } 0 \le t \le t_f\ . \tag{4.8}$$

The following limits on trolley speed and control (acceleration) are assumed:

$$\dot{R}_{min} \le x_6(t) \le \dot{R}_{max} \tag{4.9a}$$

$$\ddot{R}_{min} \le u_1(t) \le \ddot{R}_{max} \tag{4.9b}$$

with

$$\dot{R}_{max} = -\dot{R}_{min} = 1.5\,\mathrm{m/s}\quad .\quad \ddot{R}_{max} = -\ddot{R}_{min} = 1.5\,\mathrm{m/s^2}.$$

These numbers were chosen to be identical to the time-optimal trolley movement problem studied in [2] [57], for which the analytical (exact) solution will be used for comparison in the following section.

4.2.1 Analytical solution

Because of the simple structure of the above minimum time optimization, it is possible to find its analytical (continuous) solution. State equations (4.7) and costate equations (4.8) along with the boundary conditions (4.2) and (4.4) form a *two-point boundary-value problem* subject to the constraints (4.9) and the following transversality condition

$$u_1(t_f)\left(p_6(t_f) - \frac{1}{r_0}p_{10}(t_f)\right) = 1\ .$$

The detailed derivation of the two-point boundary-value problem and its exact solution for two different cases, unconstrained trolley speed and constrained trolley speed, are given in [2].

For the constrained motion with

$$r_0 = 39.24\,\text{m},\ R_0 = 10\,\text{m},\ R_f = 40\,\text{m}, \tag{4.10}$$

the minimum elapsed time is found to be equal to $t_f = 23.50\,\text{s}$. Figure 4.2 shows the optimal trajectories of the trolley and its speed and control variations. The load swing angle during the motion is also shown in the same figure.

From Figure 4.2, the following observations can be made.

- Either the control variable or the speed variable is saturated during the motion, providing fast trolley movement.
- The load swing angle during the motion is bounded within $\pm 4.3°$. This corresponds to a maximum of 3 m load oscillation, which is reasonable for the size of crane considered.
- At the target position, the trolley stops with no load swing, satisfying the end-point condition (4.4).
- The optimal solution is symmetric about $t = t_f/2$ (the optimal control and speed are symmetric while the load swing is skew-symmetric). Intuitively, this makes sense where the states of the system are the same at its two ends.

4.2.2 Numerical solution

In the previous section, the exact analytical solution of the minimum time optimization was given. In this section, the optimization algorithm developed in Chapter 3 is

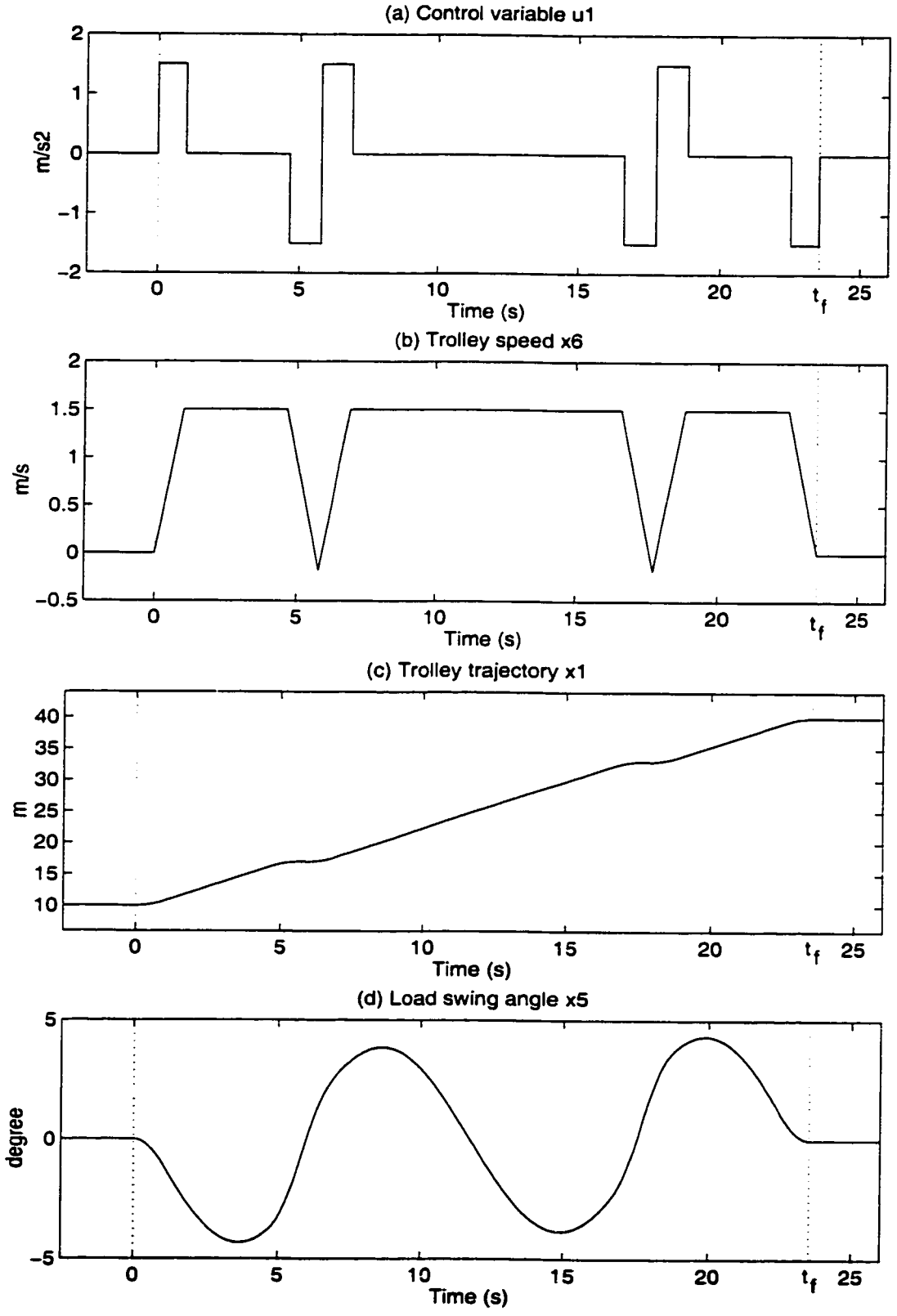


Figure 4.2: Minimum-time (analytical) solution for the planar motion

applied to solve the same time-optimal problem. Its numerical solution is expected to be an approximation of the analytical solution.

In order to use the optimization algorithm of Section 3.4.7, the following steps were taken.

According to Section 3.4.7, the new state variable $x_{11}(t) = t_f$ is defined to convert the free end-time optimal trolley motion into a fixed end-time optimal control problem. This new state satisfies

$$\dot{x}_{11}(t) = 0 \quad \text{with} \quad T_1 \leq x_{11}(0) \leq T_2 \,. \tag{4.11}$$

Because the minimum travel time (t_f) is greater than $(R_f - R_0)/\dot{R}_{max} = 20\,\text{s}$. T_1 is set to 20. Also. T_2 is arbitrarily chosen large enough ($T_2 = 50$).

As suggested in Section 3.4.8. another extra state variable is needed to include the path constraint (4.9a) into the optimization process. The new state $\dot{x}_{12}$ satisfies

$$\dot{x}_{12}(t) \;=\; \max\{0, x_6(t) - \dot{R}_{max}\}^2 + \max\{0, -x_6(t) + \dot{R}_{min}\}^2 \,. \tag{4.12}$$

and the end-point condition is

$$x_{12}(t_f) \;=\; 0 \,. \tag{4.13}$$

The cost functional (4.5) can now be converted into

$$t_f = \int_0^1 x_{11}(t)dt \,,$$

which has a fixed time interval. Similar to (3.82), the state equations (4.7), (4.11)

and (4.12) are scaled as

$$\begin{aligned}
&\dot{\hat{x}}_1(t) = \hat{x}_{11}(t)\,\hat{x}_6(t)\\
&\dot{\hat{x}}_5(t) = \hat{x}_{11}(t)\,\hat{x}_{10}(t)\\
&\dot{\hat{x}}_6(t) = \hat{x}_{11}(t)\,\hat{u}_1(t)\\
&\dot{\hat{x}}_{10}(t) = -\hat{x}_{11}(t)\,(g\,\hat{x}_5(t) + \hat{u}_1(t))/r_0 \qquad \text{for } 0 \le t \le 1\\
&\dot{\hat{x}}_{11}(t) = 0\\
&\dot{\hat{x}}_{12}(t) = \hat{x}_{11}(t)\,(\max\{0, \hat{x}_6(t) - 1.5\}^2 + \max\{0, -\hat{x}_6(t) - 1.5\}^2)
\end{aligned}$$

with $\hat{x}(t) = x(t_f t)$ and $\hat{u}_1(t) = u_1(t_f t)$.

Similar to (4.1), the terminal state equality constraints (4.4) and (4.13) are approximated by

$$(\hat{x}_1(1) - 40)^2 + \hat{x}_5^2(1) + \hat{x}_6^2(1) + \hat{x}_{10}^2(1) + \hat{x}_{12}^2(1) \le 0.005$$

where the original state variables x_i are replaced by the new scaled state variables $\hat{x}_i$.

Using 100 equal time subintervals, the optimization algorithm is applied to calculate the approximate time-optimal solution of the planar motion. The results (after reverse scaling) are given in Figure 4.3.

The computed travel time is 23.7 s which is slightly bigger than 23.5 s, the minimum travel time obtained from the exact solution. Although the numerical solution resembles the analytical solution, there are a few discrepancies, especially when the speed variable is saturated. Figures 4.3(a),(b) show that the control variable oscillates about zero whenever the trolley travels at its maximum speed (1.5 m/s). This jittery phenomenon may be reduced by a nonuniform redistribution of the time subintervals.

Unlike the optimal control in Figure 4.2(a), the control sequence in Figure 4.3(a)

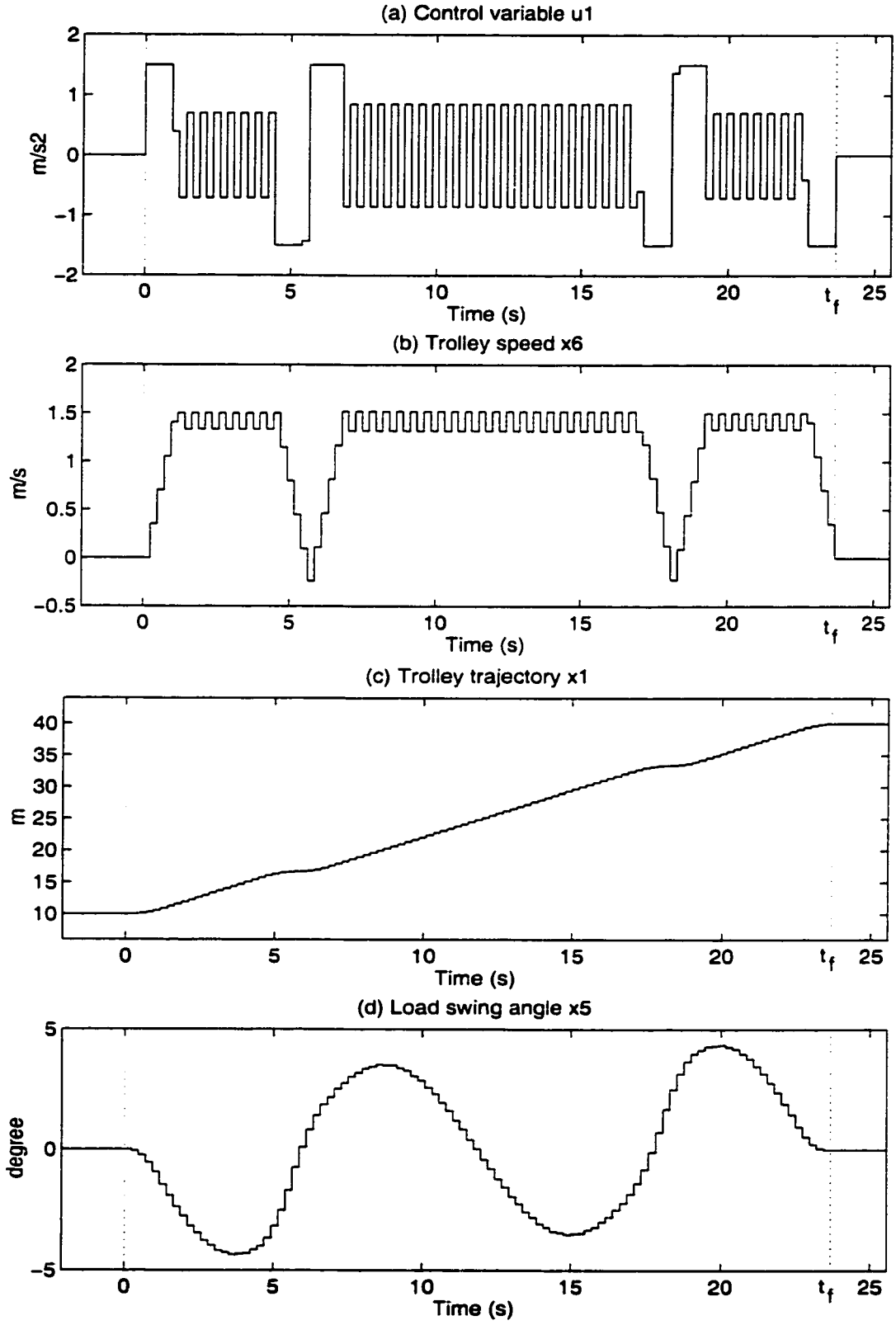


Figure 4.3: Time-optimal (numerical) solution for the planar motion

is not quite symmetric. This is perhaps due to the parameterization scheme in the optimization algorithm.

Despite these differences, the optimal trajectory obtained by applying the optimization algorithm closely follows the exact optimal trajectory. Nevertheless, it is expected that the numerical results will improve as the number of discretization levels increases. This would, however, come at the expense of an increase in the optimization process time.

Finally, it is important to note that except in special cases, like the one studied here, an exact (analytical) optimal solution may not be always available. Therefore, numerical methods, such as the one presented in this research, are the only resort.

4.3 Optimal planar motion

Exact and approximate time-optimal planar motions were given in the previous section. It is obvious by definition that the time-optimal transport, regardless of its 6 m load swing, is faster than any other transports. Working with a real tower crane, however, a load swing of this magnitude appeared to be very difficult to control. A crane operator always keeps the load swing very small during motion to avoid undesired swings at the destination.

To reduce the load swing of the time-optimal solution, a function of the load swing similar to (3.2) is added to the cost functional (4.5) as

$$\int_0^{t_f} \left[1 + \frac{1}{4} r^2 \left(x_5^2(t) + x_{10}^2(t)\right)\right] dt$$

where r is the hoist cable length. The constant factor 1/4 is chosen by trial and error to reduce the maximum load swing to less than 2 m (or about 3°), typical

in practice. For the system of (4.7), an optimal load displacement between the two given positions (4.2) and (4.4) is then required to minimize the above cost functional subject to the constraints (4.9).

Except the cost functional, the formulation of the time-optimal problem given in Section 4.2.2 still holds here. Similarly, 100 equal time subintervals were used to compute the optimal solution of the desired displacement. The results are shown in Figure 4.4.

From Figure 4.4(d), the load swing is bounded within $\pm 2.3°$ or $\pm 1.6\,\mathrm{m}$, which is smaller than the maximum load swing of the time-optimal motion. However, the computed travel time is 31.4 s, almost 35% more than the minimum travel time of the time-optimal solution. This was anticipated as the new cost function weighs on the load swing as well as the travel time.

From the practical design point of view, the optimal motion of Figure 4.4 is to be preferred to the time-optimal motion. As shown in Figures 4.2(a,b), the speed and control variables of the time-optimal motion are pushed towards their limits with fast transitions throughout the motion. However, the speed and control variables of the optimal motion in Figures 4.6(a,b) appear to be much smoother, as a result of adding a function of load swing into the cost function. It can be easily verified from the dynamical equations (4.7) that the control input $u_1(t)$ is proportional to the load swing and the speed of trolley [2] [57].

4.4 Non-optimal planar motion

It is obvious by definition that the optimal solution offers faster load displacement than any other non-optimal load transfer with the same load swing, including those

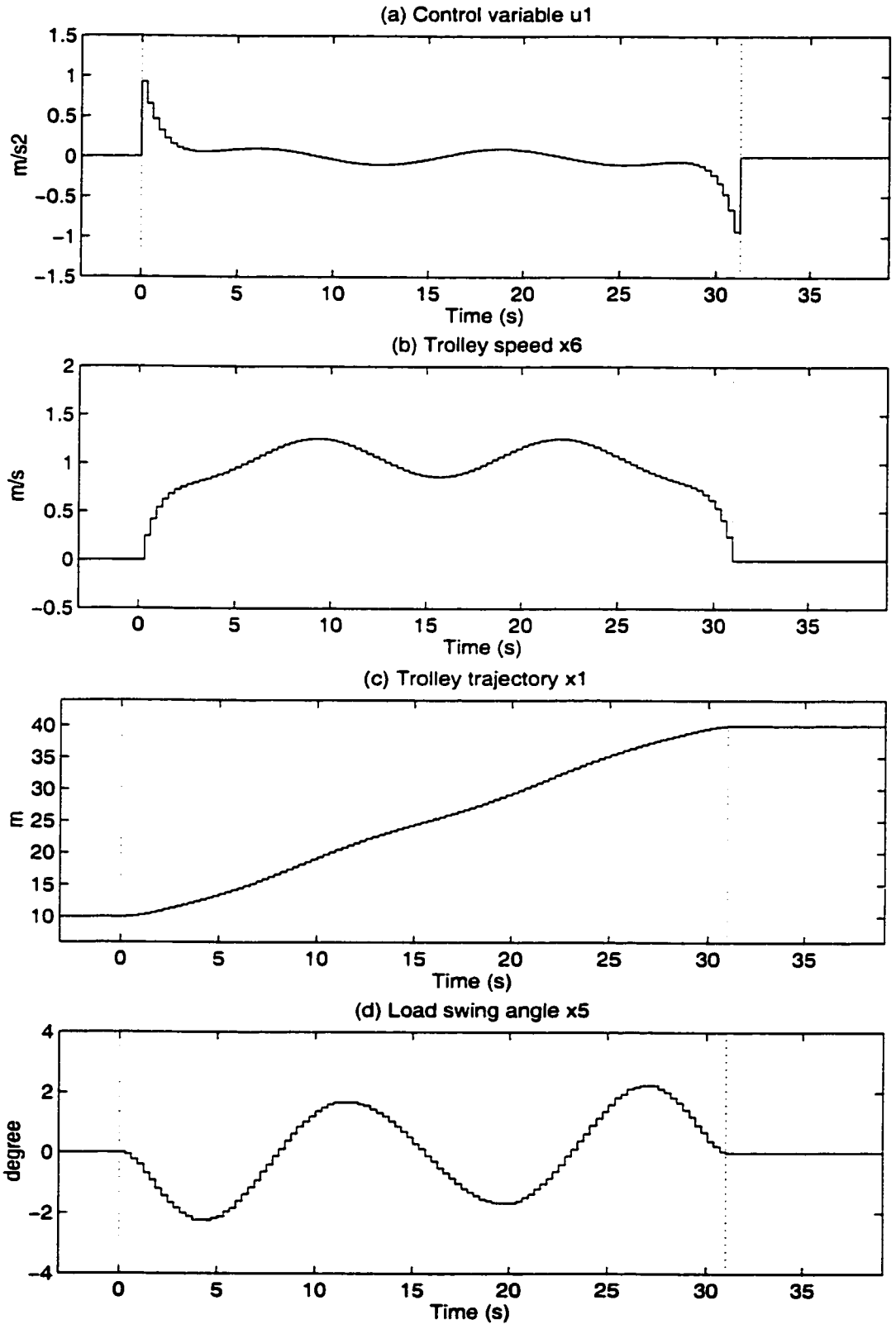


Figure 4.4: Optimal planar motion

given by a crane operator. However, it is not clear if the time saving is significant for small load swings. Therefore, the performance of the optimal motion is compared with the performance of a non-optimal motion typical of a manually controlled tower crane.

In this section a typical load transfer of a conventional tower crane for the planar motion described in Section 4.2 is simulated. This result is obtained after some informal communication with an experienced crane operator and several hours of examining crane operations in a construction site [59]. But first, the following hypothetical case is studied to show that excessive load swing at the end of desired displacement is unavoidable if the crane operator moves the trolley very fast.

Suppose that the trolley travels as quickly as possible towards its final position at R_f. Because of speed and acceleration limits, it takes at least 21 s for the trolley to complete this task. The trolley trajectory as well as its speed and control variations are shown in Figure 4.5. Note that, because no friction has been considered in our simplified crane model, the load swing in Figure 4.5(d) does not decay after the trolley movement is completed.

While the trolley rests at the target point, the load swing continues with a magnitude of more than 8° (see Figure 4.5(d)). With a 40 m long hoist cable, the load swings as much as 11 m (peak-to-peak), which is clearly unacceptable in practice.

To have a safe unloading condition, the load swing has to be considerably reduced before reaching the target position. Practically speaking, this ultimately requires gradual speed changes both at the beginning and at the end of the trolley motion. This is illustrated in Figure 4.6 which is representative of a manually controlled tower crane. The trolley speeds up in three steps within the first 10 seconds

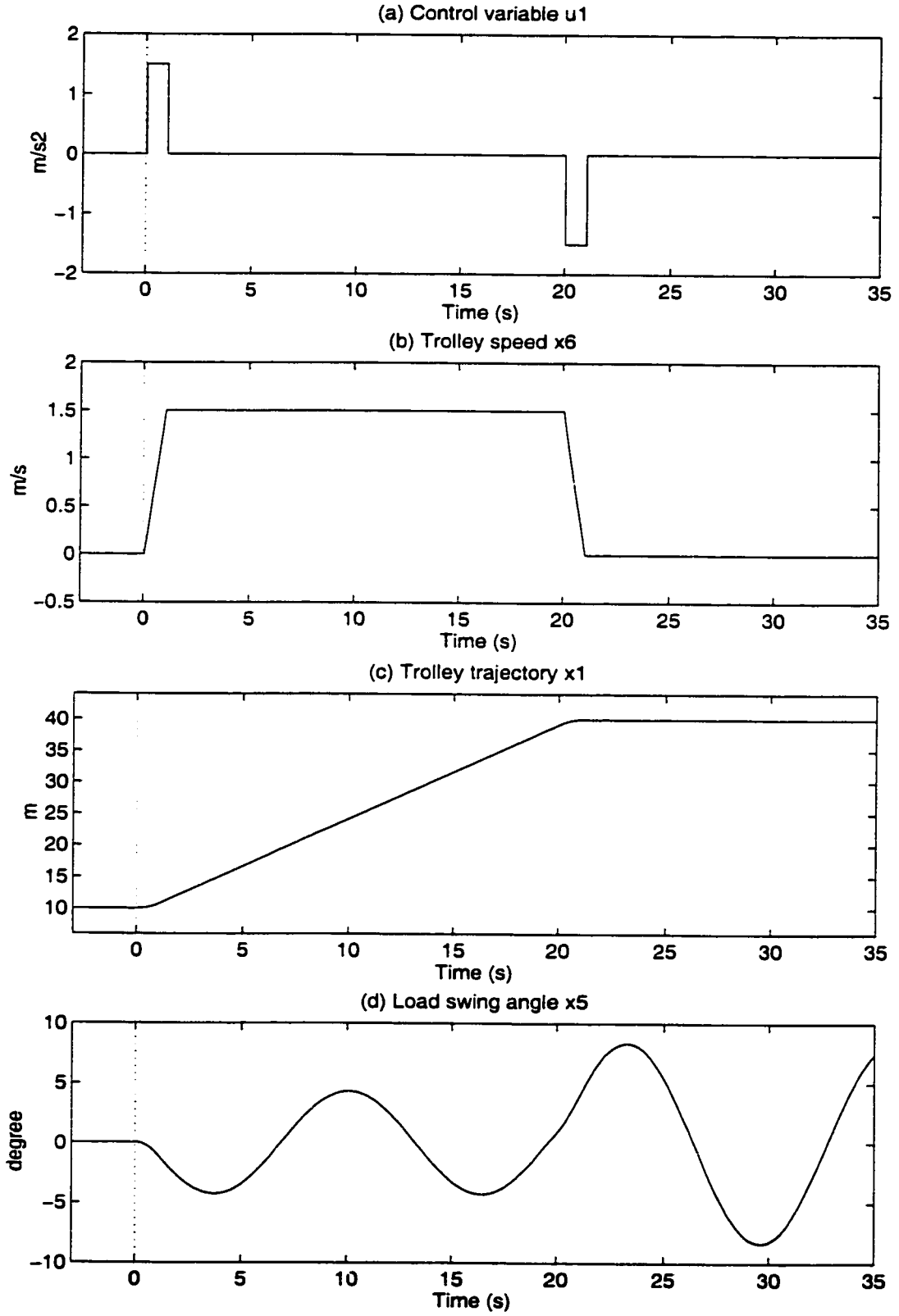


Figure 4.5: Non-optimal planar motion with fast trolley movement

of the motion (Figure 4.6(b)). The trolley then moves at a constant high speed for about 13 seconds before it gradually slows down close to the target position.

As shown in Figure 4.6(d), the load swing is limited to 2.5° or 1.7 m during the motion, but it will be as small as 0.5° or 40 cm at the end of motion which is fairly reasonable for a safe unloading condition. This is compared with a similar load swing of the optimal trajectory during the motion, but a motionless target condition for the load (Figure 4.4(d)).

The final transport time is measured at about 42 s when the trolley rests at the target point. This is 33% more than the final time of the optimal motion. The optimal solution may even be further shortened if, similar to the non-optimal motion, the exact zero load swing requirement (4.4) is relaxed.

It is always possible to shorten the optimal solution by allowing larger load swings. While a human operator loses control of load swing if it is not very small, an automated tower crane equipped with robust control system may well handle larger load swings within the crane safety requirements. The minimum transport time of the given planar motion is 23.5 s belonging to the time-optimal solution. If maximum load swing of 5° is allowed, the time-optimal trajectory can offer significant time saving.

By comparing Figures 4.4(d) and 4.6(d), it is clear that the optimal load trajectory is much smoother than the non-optimal load trajectory. Working with a conventional tower crane, the crane operator carefully controls the trolley movements to avoid undesirable load swings.

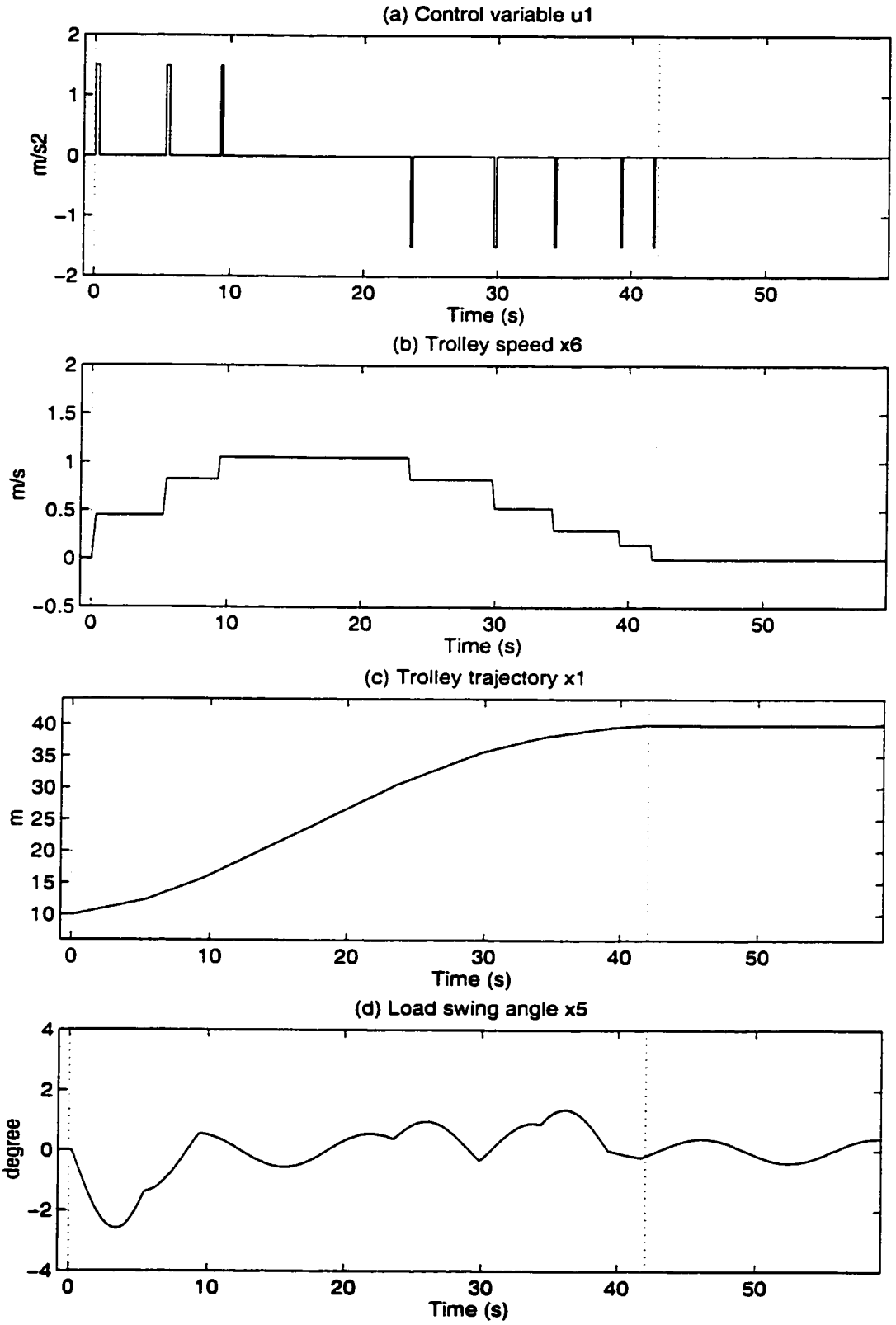


Figure 4.6: Typical planar motion of a conventional tower crane

4.5 A three dimensional motion

In the previous sections, a simple case of the optimal crane motion was studied. It was shown that the optimal trajectory could provide fast and smooth load transfer with small swing during the motion and zero swing at the end. In this section, a complex and more realistic case that involves all of the crane rigid-body motions is considered. These motions are the trolley travel, the boom slew and the load lift. Unlike the planar motion, the load in this case travels in a three dimensional space.

For the crane described by (2.18), consider a load hanging from the trolley at the following initial position:

$$\boldsymbol{x}_0 = (10, 0, 30, 0, 0, 0, 0, 0, 0, 0)^T \quad \text{at } t = 0 \,. \tag{4.14}$$

This is a motionless position because the load swing angles and all of the speed components in (4.14) are set to zero. The load is subject to transfer from the above initial position to the stationary target position

$$\boldsymbol{x}_f = (40, 1, 20, 0, 0, 0, 0, 0, 0, 0)^T \quad \text{at } t = t_f \,. \tag{4.15}$$

The speed and control limits on the rigid-body motions are considered to be

$$\begin{cases} |x_6(t)| \leq 1.5\,\text{m/s} \\ |x_7(t)| \leq 0.15\,\text{rad/s} \quad \text{for } 0 \leq t \leq t_f \\ |x_8(t)| \leq 1.5\,\text{m/s} \end{cases} \tag{4.16}$$

$$\begin{cases} |u_1(t)| \leq 1.5\,\text{m/s}^2 \\ |u_2(t)| \leq 0.15\,\text{rad/s}^2 \quad \text{for } 0 \leq t \leq t_f \,. \\ |u_3(t)| \leq 1.5\,\text{m/s}^2 \end{cases} \tag{4.17}$$

In what follows, first a non-optimal solution to the above displacement, representative of a conventional tower crane, is presented. This is followed by computation of an optimal solution of the desired displacement. Then, the performance of the optimal motion is measured as compared with the manually controlled motion. It will be shown that, with smaller swing, the optimal load trajectory is much faster and smoother. This well illustrates why study of optimal crane motions is worth consideration.

4.5.1 Non-optimal Solution

The load transfer just defined is a typical task of a tower crane working in a construction site. It requires

- 30 m of trolley travel,
- 1 rad ($\simeq 57°$) of boom slew, and
- 10 m of load lift.

With a conventional tower crane, which is manually operated, the above task may be done in segments to have a safe and more reliable load transfer [59]. By doing so, the crane operator has better control on the load swings.

Among many different possibilities, consider the following sequence of crane motions to complete the prescribed three dimensional task.

First the boom rotates 1 rad (57°) while the load is being raised 10 m, from $r_0 =$ 30 m to $r_f = 20$ m (Figures 4.9(a,b)). Their speeds gradually increase and decrease in several steps to avoid large load swings. Once these motions are completed, both trolley and boom are locked for the rest of operation.

Next, the trolley travels 30 m forward, from $R_0 = 10\,\text{m}$ to $R_f = 40\,\text{m}$ (Figure 4.9(c)). The trolley accelerates in the first few seconds of its motion. It then travels at a constant rate before it gradually stops at the target point (Figure 4.8(c)). The speed variation of the trolley is similar to the basic motion given in Figure 4.5(b).

While these three motions can be performed in different orders, there are two main reasons for the above choice. One is to bring the load close to the crane at the beginning of load transfer. A close load is more easily controlled by the crane operator. Secondly, slewing the boom while the load is close to the mast is to be preferred, to minimize load swing and energy. Therefore the boom is rotated before moving the trolley 30 m away outward.

From Figure 4.10, the maximum load swing during the load transfer is 3° or 1.2 m. Once the crane motions are completed (after $t = 54\,\text{s}$), the load swings as much as ±2° or ±50 cm which is small enough for a safe unloading condition. Note that the load swings in Figure 4.10 do not decay over time, since no friction is considered in the simplified model of crane.

Finally, Figure 4.11 shows the projection of the load sway on the $x - y$ plane of the moving frame in Figure 2.1. It shows the horizontal load movement relative to the trolley. The trajectory leaves the origin as the boom starts slewing. It ends up in a periodic motion as the load transfer is completed. The maximum load distance from the origin is 1.3 m. The periodic solution is a result of our frictionless model. In reality, however, this trajectory moves toward the origin as the load swings gradually die out.

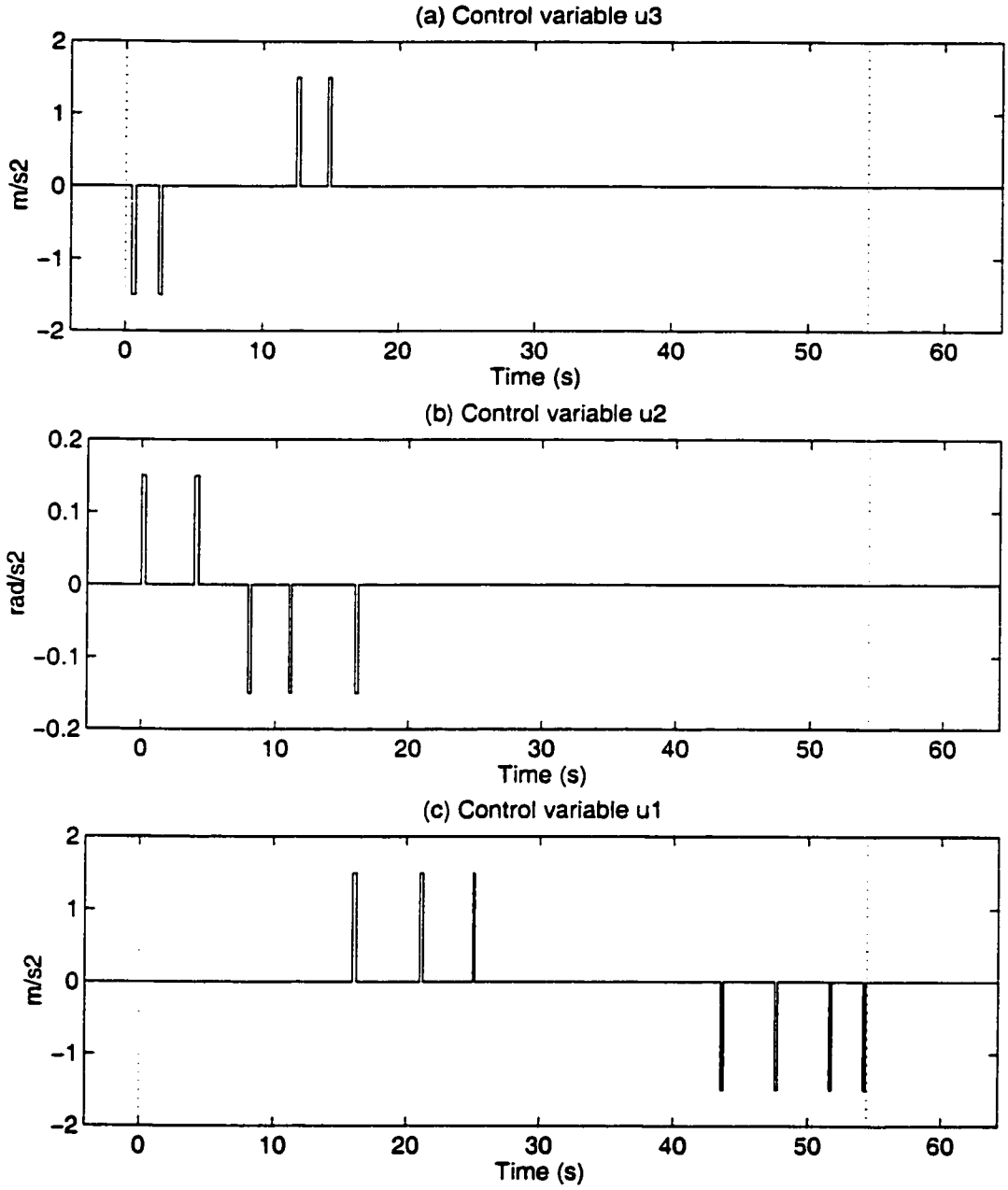


Figure 4.7: Control variations of the non-optimal 3D motion

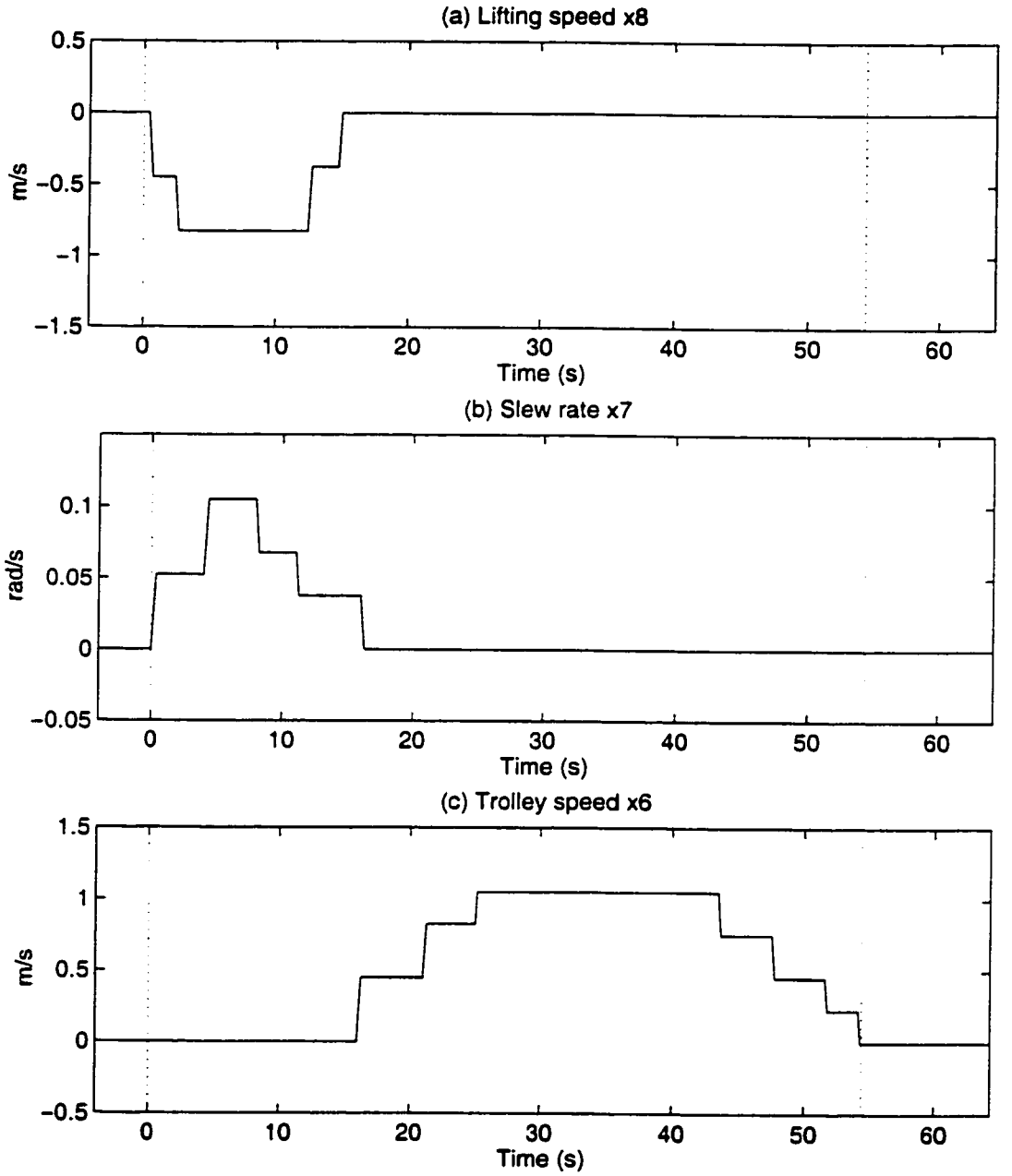


Figure 4.8: Rigid-body speeds of the non-optimal 3D motion

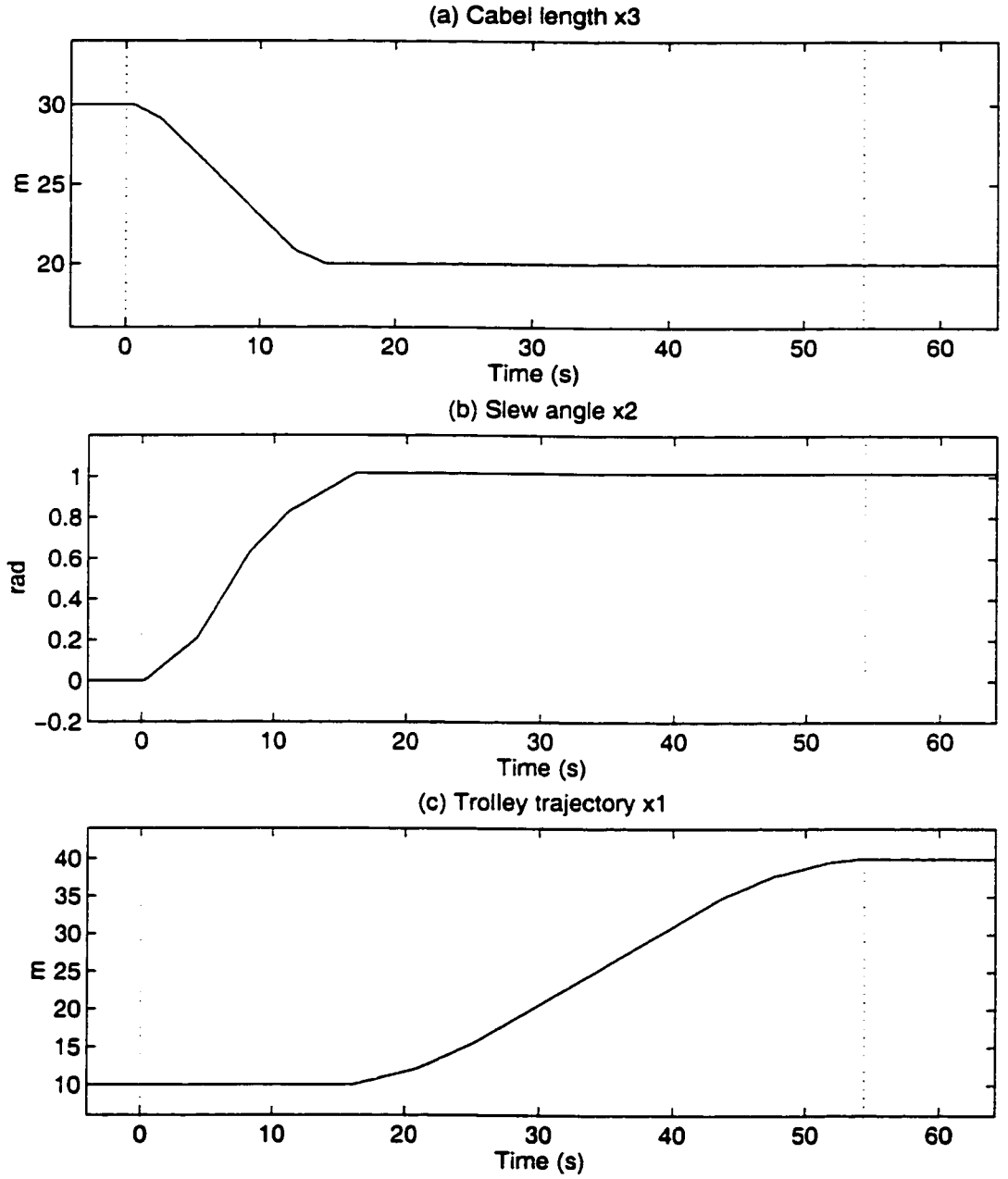


Figure 4.9: Rigid-body positions of the non-optimal 3D motion

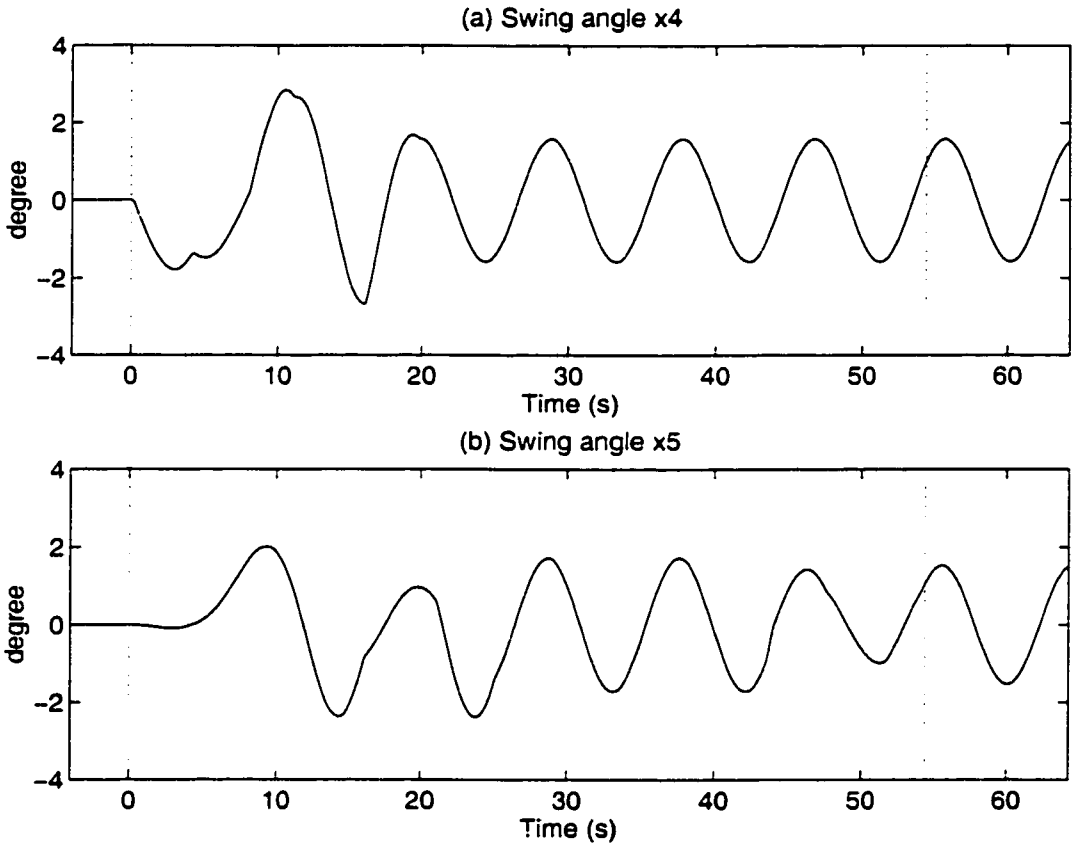


Figure 4.10: Load swings of the non-optimal 3D motion

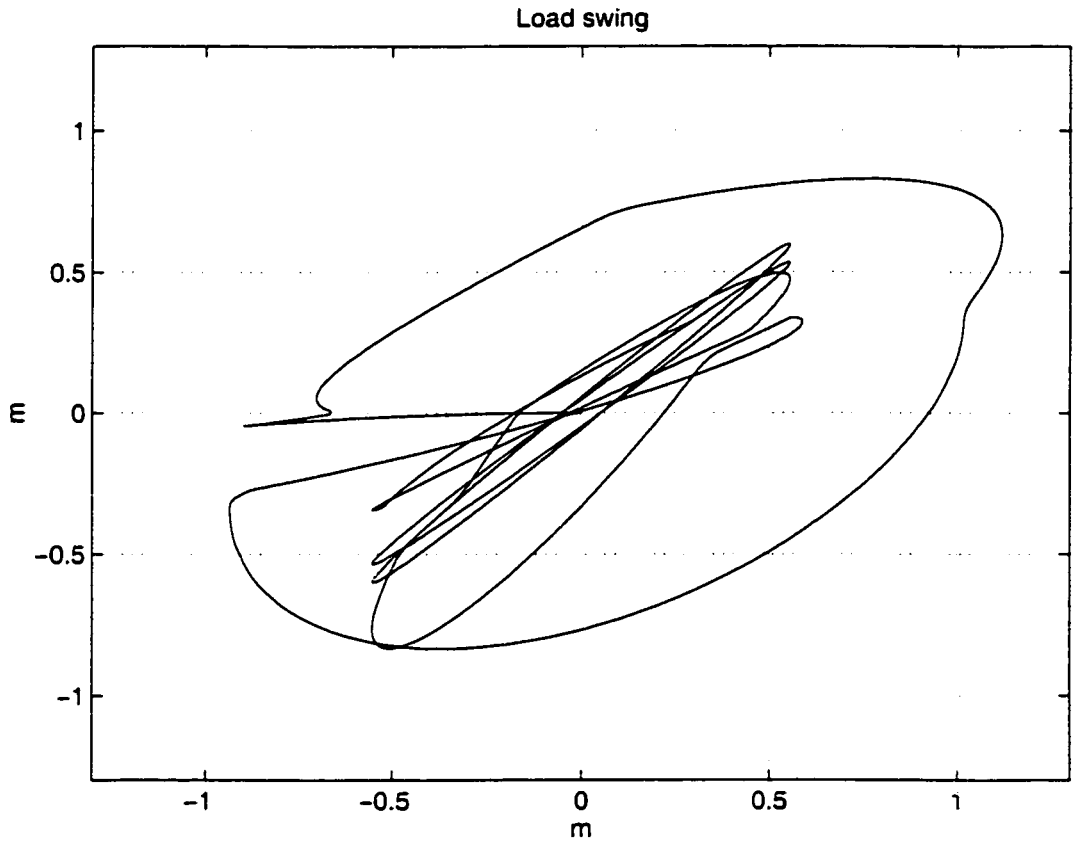


Figure 4.11: Horizontal load displacement relative to the trolley during the non-optimal 3D motion

4.5.2 Optimal Solution

In the previous section a manually controlled (non-optimal) load displacement for the three dimensional task was given. In this section, the optimization algorithm of Chapter 3 is employed to find an optimal load transfer that minimizes a cost function of the form (3.4). The result will then be compared with the non-optimal motion of Section 4.5.1.

An optimal load displacement between the two given positions (4.14) and (4.15) is considered to minimize the cost functional

$$\int_0^{t_f} \left[1 + x_3^2(t)\left(x_4^2(t) + x_5^2(t) + x_9^2(t) + x_{10}^2(t)\right)\right] dt \tag{4.18}$$

subject to the constraints (4.16) and (4.17). The cost functional (4.18) is a special case of the general form (3.4) with $a = b = 1$. The relative values of a and b were chosen by try and error such that the maximum load swing of the resulting optimal solution would not exceed that of the non-optimal trajectory. One would expect a faster load transfer by increasing a. In contrast, increasing b would result in a slower motion with more damped load swings.

To convert the above optimization into a fixed end-time optimization, let the new state $x_{11} = t_f$ satisfy

$$\dot{x}_{11}(t) = 0 \quad \text{with} \quad T_1 \leq x_{11}(0) \leq T_2 \; . \tag{4.19}$$

Because the final time (t_f) must be at least 21 s, the minimum required time to transfer the trolley, the parameter T_1 is set to 21. Also, $T_2 = 100$ is arbitrarily chosen large enough.

Another extra state, x_{12}, is needed to include the path constraints (4.16) into the optimization process. The new state satisfies

$$\dot{x}_{12}(t) \;=\; \max\{0, x_6(t) - 1.5\}^2 + \max\{0, -x_6(t) - 1.5\}^2 +$$

$$\max\{0, x_7(t) - 0.15\}^2 + \max\{0, -x_7(t) - 0.15\}^2 +$$
$$\max\{0, x_8(t) - 1.5\}^2 + \max\{0, -x_8(t) - 1.5\}^2 , \qquad (4.20)$$

and the end-point conditions

$$x_{12}(t_f) \;=\; 0 \,. \qquad (4.21)$$

To form a fixed end-time optimization, the state equations (2.18) with (4.19) and (4.20) are scaled as

$$\dot{\hat{\boldsymbol{x}}}(t) = \hat{\boldsymbol{f}}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}) = \begin{pmatrix} \hat{x}_{11} \boldsymbol{f}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}) \\ 0 \\ \hat{f}_{12} \end{pmatrix}$$

$$\begin{aligned} \hat{f}_{12} \;=\; & \hat{x}_{11} \Big(\max\{0, \hat{x}_6(t) - 1.5\}^2 + \max\{0, -\hat{x}_6(t) - 1.5\}^2 + \\ & \max\{0, \hat{x}_7(t) - 0.15\}^2 + \max\{0, -\hat{x}_7(t) - 0.15\}^2 + \\ & \max\{0, \hat{x}_8(t) - 1.5\}^2 + \max\{0, -\hat{x}_8(t) - 1.5\}^2 \Big) \end{aligned}$$

with $\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t_f t)$, $\hat{\boldsymbol{u}}(t) = \boldsymbol{u}(t_f t)$, and $\boldsymbol{f}$ defined in (2.18).

The minimization (4.18) is now converted into the minimization

$$\int_0^1 \hat{x}_{11}(t) \left[1 + \hat{x}_3^2(t) \left(\hat{x}_4^2(t) + \hat{x}_5^2(t) + \hat{x}_9^2(t) + \hat{x}_{10}^2(t) \right) \right] dt \qquad (4.22)$$

over the fixed time interval $[0, 1]$.

In calculating the optimal solution, the terminal state equality constraints (4.15) and (4.21) are approximated by

$$(\hat{\boldsymbol{x}}_{1:10}(1) - \boldsymbol{x}_f)^T (\hat{\boldsymbol{x}}_{1:10}(1) - \boldsymbol{x}_f) + \hat{x}_{12}^2(1) \;\leq\; 0.01$$

where $\boldsymbol{x}_f$ is given in (4.15), and the original state $\boldsymbol{x}$ is replaced by the new scaled state $\hat{\boldsymbol{x}}$.

Once the three dimensional optimization is formulated, the optimization algorithm of Chapter 3 is employed to calculate the approximate optimal state and control variables. The results, using 100 equal time subintervals, are given in Figures 4.12 to 4.16.

Figures 4.12, 4.13 and 4.14 show the rigid-body controls, speeds and positions respectively. Figure 4.15 shows the swing angles of the optimal motion. Finally, Figure 4.16 displays the horizontal load trajectory relative to the trolley during the load displacement. From these Figures, the following observations are provided.

- The computed load transfer time is 27 s which is half of 54 s suggested by the manually controlled motion. The time saving is clearly a big advantage for the optimal load displacement.

- As shown in Figure 4.15, the optimal solution can offer exact zero load swing at the end of transfer process. This is, however, almost impossible in a manually controlled tower crane. Precise load handling is another advantage of the optimal load displacement.

- Comparing Figures 4.15 and 4.16, the optimal load motion is much smoother than the non-optimal load motion.

- As shown in Figure 4.14(c), the hoist cable is initially shortened to less than 7 m before it returns to 20 m at the end of motion. This helps to reduce the horizontal load distance (relative to the trolley) during the transfer process. Note that the integrand function in (4.18) is quadratic in the hoist cable length.

- The computed results may not be "globally" optimum. A different set of initial values in the optimization algorithm may result in a different "locally"

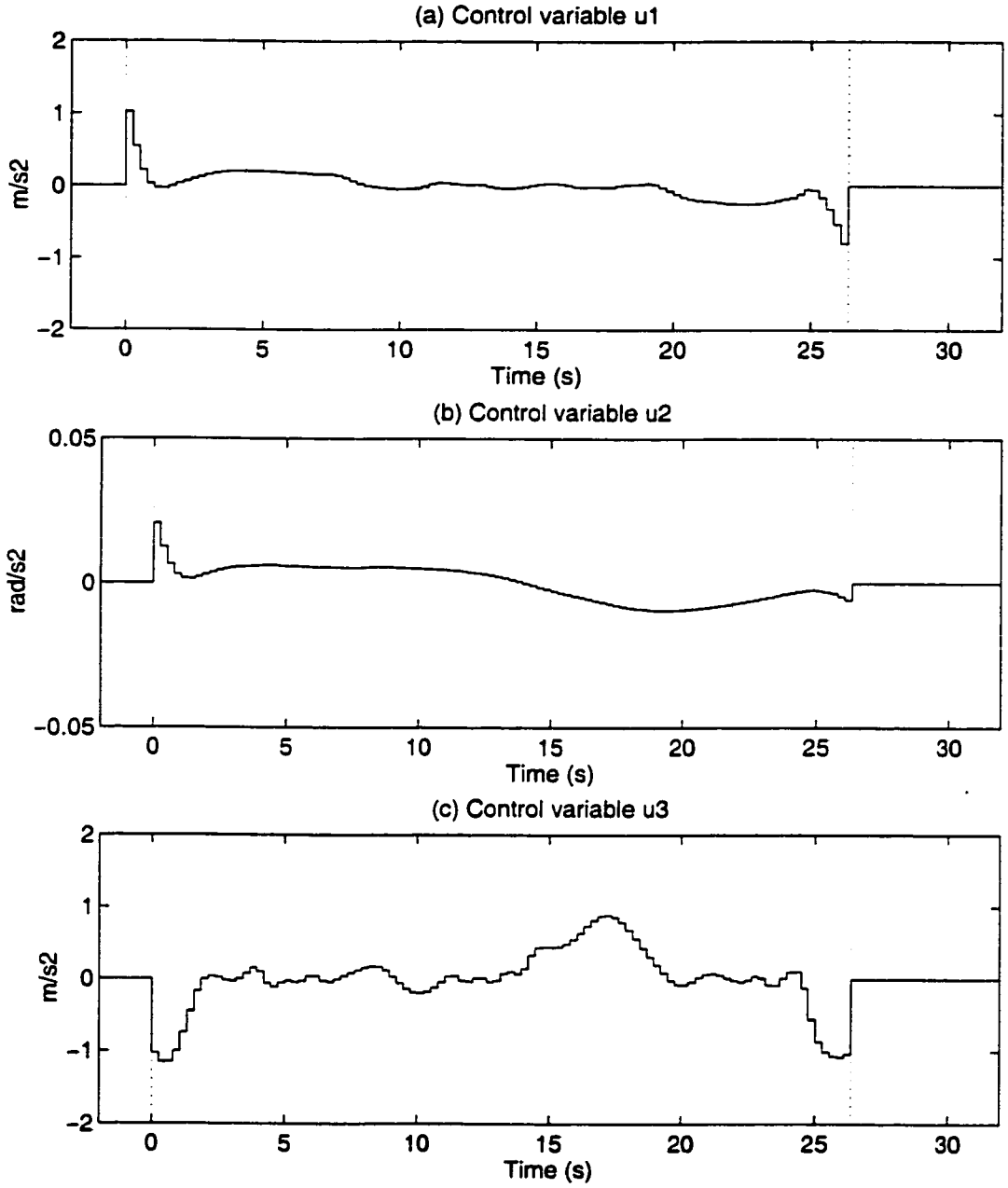


Figure 4.12: Control variations of the optimal 3D motion

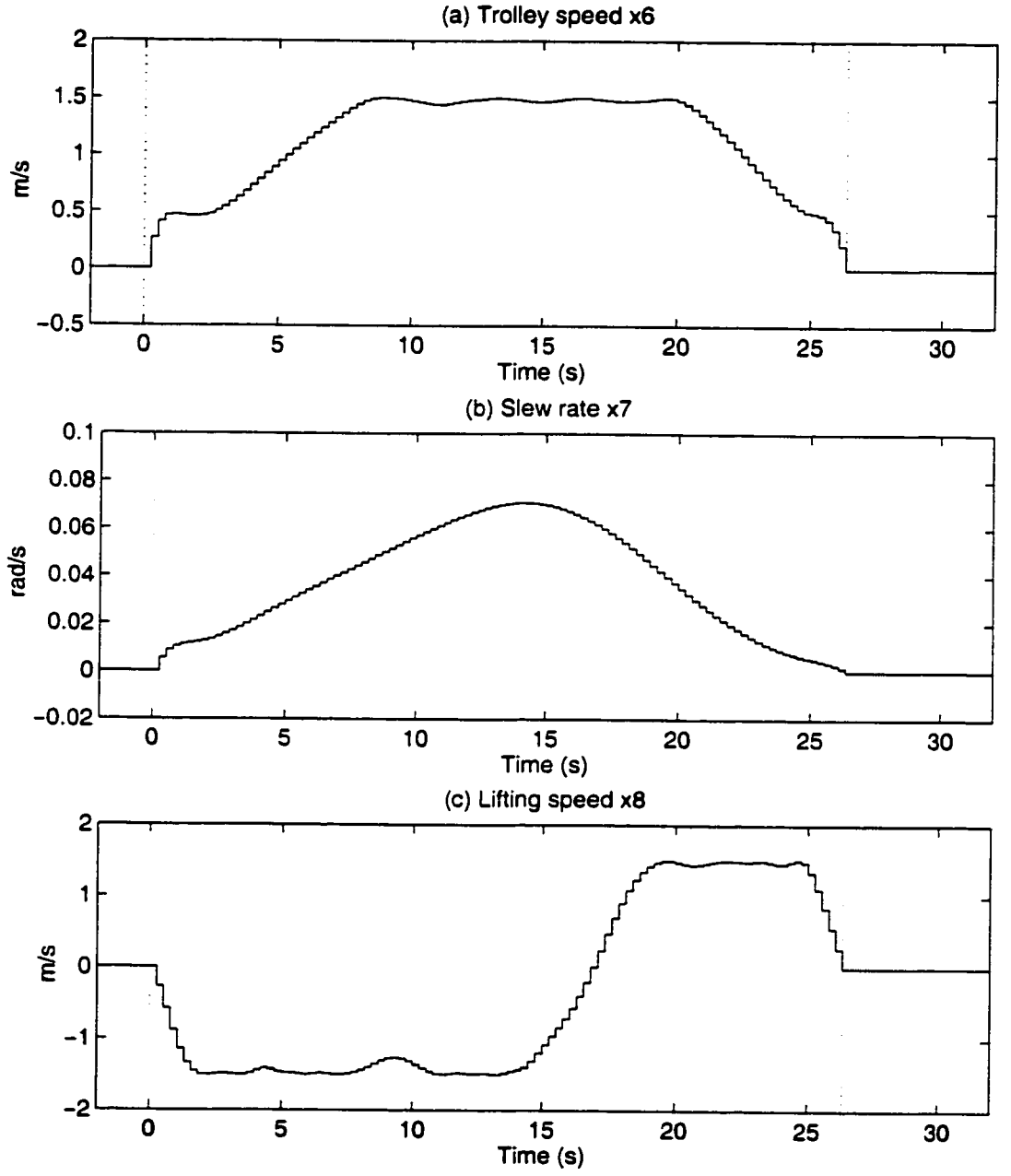


Figure 4.13: Rigid-body speeds of the optimal 3D motion

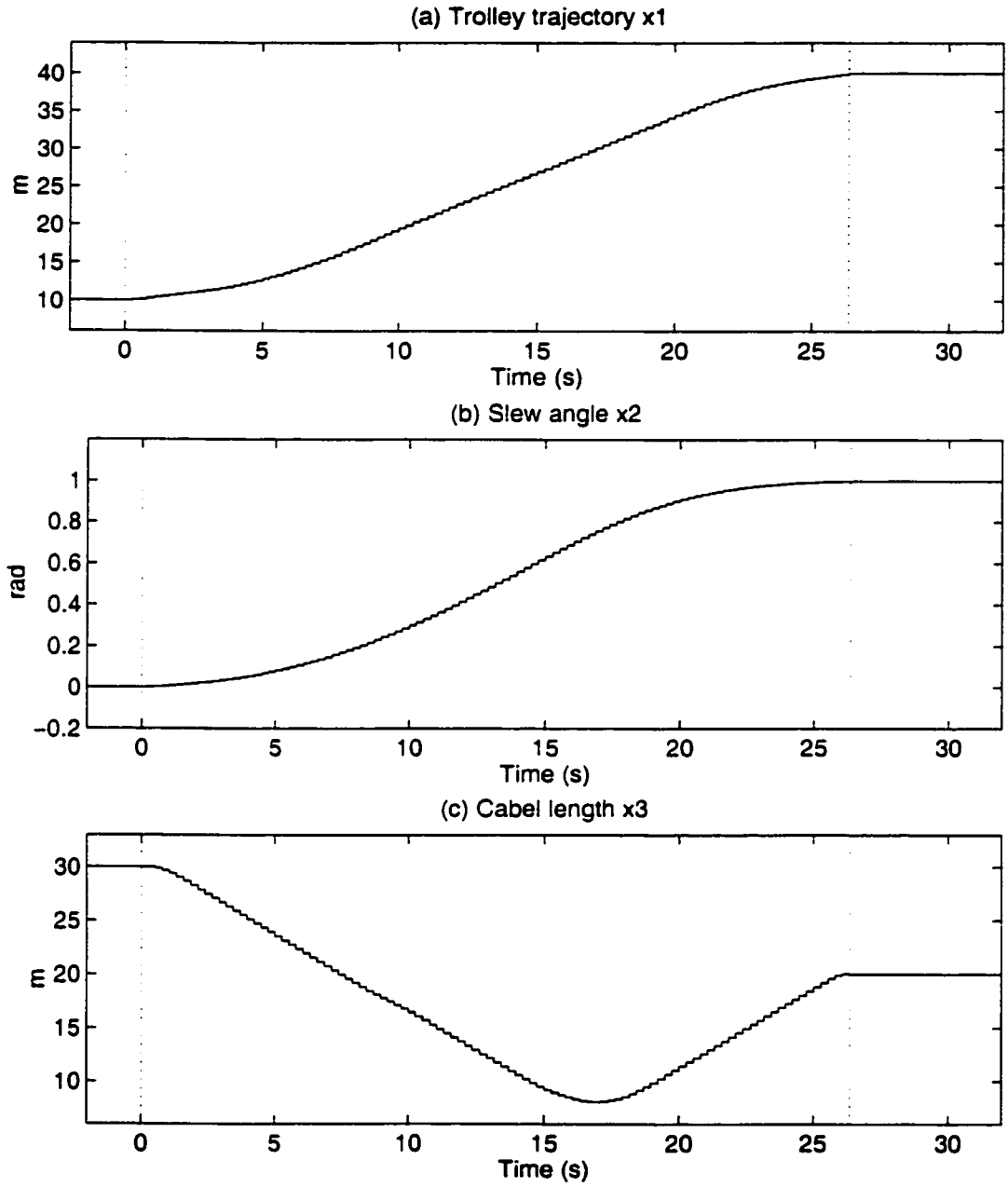


Figure 4.14: Rigid-body positions of the optimal 3D motion

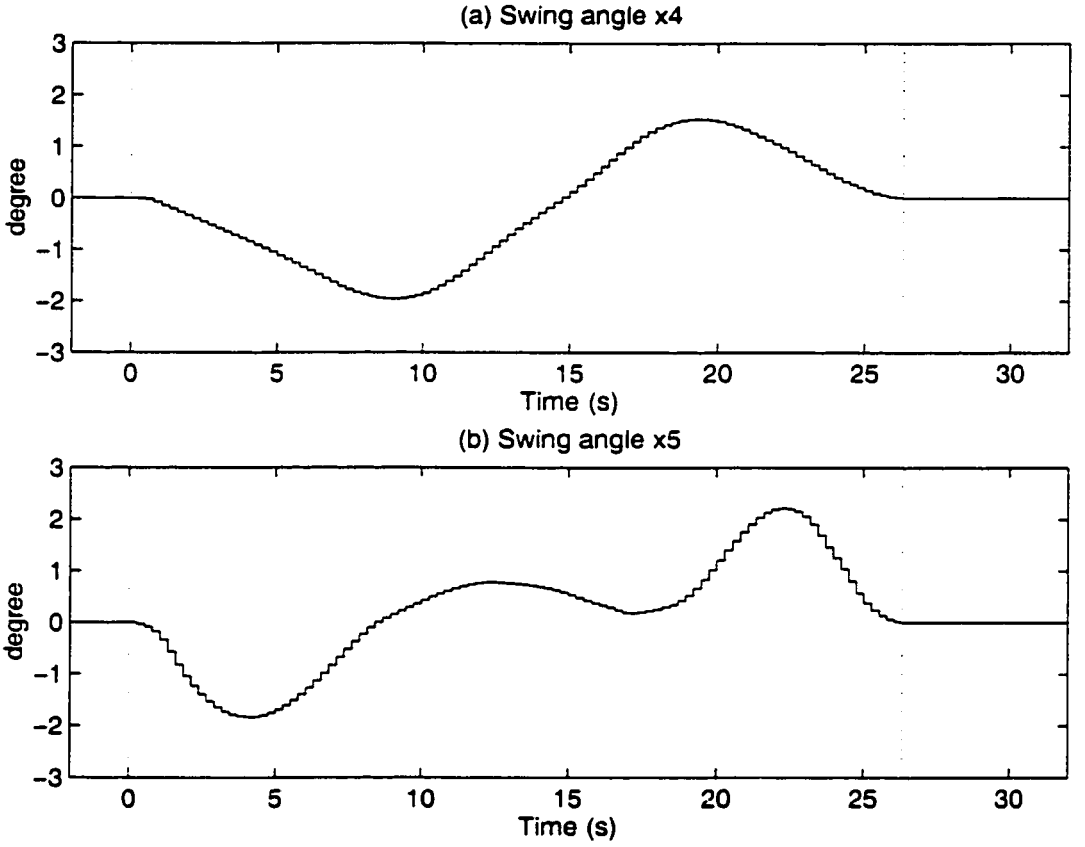


Figure 4.15: Load swings of the optimal 3D motion

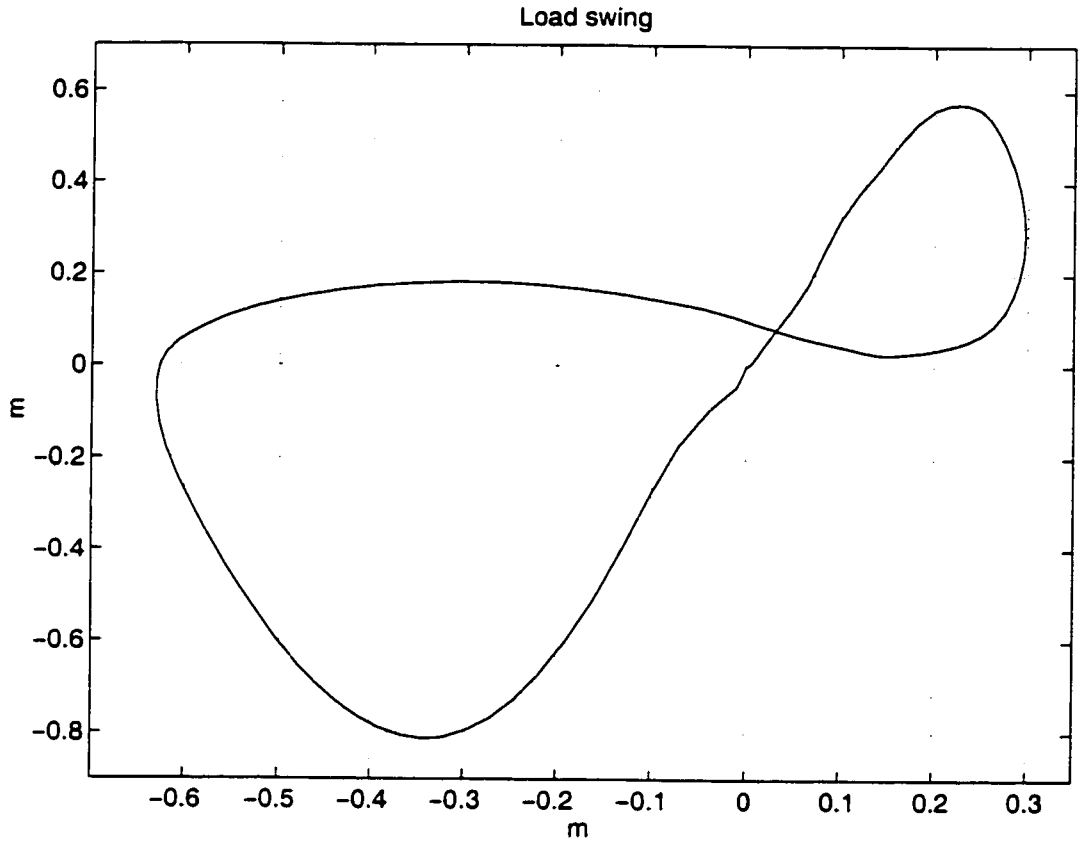


Figure 4.16: Horizontal load displacement relative to the trolley during the optimal 3D motion

optimal solution. The computed optimal solution, nevertheless, reasonably meets the optimization criteria defined in Section 3.1.

4.6 Summary

Two different cases of crane motion were studied in this Chapter. First a minimum-time load transfer was considered. The load was subject to 30 m displacement between two given motionless positions. The case was made simple by keeping the hoist cable constant with no slewing motion. The numerical solution obtained by applying the optimization algorithm of Chapter 3 closely approximates an available analytical solution. Then, a fast optimal trajectory with small load swing for the same crane motion was computed. The travel time was considerably reduced as compared with a non-optimal transport, representative of a manually controlled tower crane. In addition, the optimal load trajectory appeared to be much smoother than the non-optimal load trajectory.

The most general case of crane motions, i.e. simultaneous trolley travel, boom slew and load hoist was studied next. Analytical solution was unavailable, but the optimization algorithm was used to compute a solution that minimizes a functional of load swing and travel time. The result was compared with a typical load transport of a conventional tower crane. Similar to the basic motion, the simulation results of the optimal solution offered significant time saving with much smoother load trajectory.

Chapter 5

Conclusions

The research in this thesis investigated the application of optimal trajectories for tower cranes. The main goal is to improve the performance of the existing manually controlled tower cranes. This work started with the development of a simplified dynamical model for tower crane motions. Assuming direct control on the speeds of crane motors, a set of five second order nonlinear differential equations was derived. Using the motor accelerations as inputs, these equations were converted into a state-space form with 10 states and 3 controls. Considering normal operation, the state-space model was then linearized in load swing angles.

A cost functional was defined to ensure fast load handling with small load swing during a desired load displacement. The crane optimization required minimization of the given cost functional subject to a set of certain operational and safety constraints. Once the crane optimization was formulated as a special case of the Lagrange optimization, the necessary optimality conditions were derived.

Motivated by the Han-Powell method in finite optimization, an upper bound approximation to the second order method was then proposed to solve a continuous-

time Lagrange optimization with fixed end time and inequality constraints on controls and terminal states. This was further extended to allow free end time and path (trajectory) constraints. The resulting iterative algorithm solves a direction finding subproblem (**DFS**) at each iteration to make progress towards an optimal solution. This was done by coverting **DFS** into a standard linear quadratic programming using parameterized controls and states. It was shown that the solution of **DFS** is descent under certain conditions.

The developed optimization algorithm was applied to compute optimal load transports for two different cases. For the basic motion, the numerical optimal solution closely matched the available analytical solution. For the general motion, however, no analytical solution was available. Both optimal motions were also compared with non-optimal load displacements representatives of a manually controlled tower crane. The comparisons illustrated significant improvement, in terms of transport time and load swing, in the performance of optimal motions.

5.1 Future research

This research has opened more doors than it has closed. Recommendations for some future work are outlined below.

Modeling

- The existing crane model can be improved by removing the assumptions made in page 23 especially . Friction, as an important part of every highly-geared mechanisms, must be considered in an enhanced model of the actual crane. In addition, the hoist cable may flex under certain conditions, especially when

it is long and light loads are moved.

- Wind, as an important source of external interference in the crane operations, should be considered in an improved crane model. This may lead to a stochastic model of the crane rather than the existing deterministic model.

- Having a crane prototype is a real advantage. The simulation results can be verified on a small-scaled prototype before implementing on an actual crane.

Control

- The optimization algorithm used in this research may converge to a poor local optimum solution because of an inadequate initial value. A more efficient algorithm that is globally convergent is then required.

- The robustness analysis of the open-loop solutions obtained by the optimization algorithm remains to be done. In particular, changes in the optimal solutions due to parameter variations in the cost function can be studied.

- Crane automation cannot be accomplished without a closed-loop control design. Once the open-loop optimal solution of a desired load displacement is computed, a feedback controller must be employed to preserve near-optimal performance. Such a controller must be robust to disturbances and model imperfection.

Appendix A

Maple code for the load swing equations

The following shows how Maple was used to derive equations (2.12).

```
> with(linalg);
Warning, new definition for norm
Warning, new definition for trace
```

[*BlockDiagonal, GramSchmidt, JordanBlock, LUdecomp, QRdecomp, Wronskian, addcol, addrow, adj, adjoint, angle, augment, backsub, band, basis, bezout, blockmatrix, charmat, charpoly, cholesky, col, coldim, colspace, colspan, companion, concat, cond, copyinto, crossprod, curl, definite, delcols, delrows, det, diag, diverge, dotprod, eigenvals, eigenvalues, eigenvectors, eigenvects, entermatrix, equal, exponential, extend, ffgausselim, fibonacci, forwardsub, frobenius, gausselim, gaussjord, geneqns, genmatrix, grad, hadamard, hermite, hessian, hilbert, htranspose, ihermite, indexfunc, innerprod, intbasis, inverse, ismith, issimilar, iszero, jacobian, jordan, kernel, laplacian, leastsqrs, linsolve, matadd, matrix, minor, minpoly, mulcol, mulrow, multiply, norm, normalize, nullspace, orthog, permanent, pivot, potential, randmatrix, randvector, rank, ratform, row, rowdim, rowspace, rowspan, rref, scalarmul, singularvals, smith, stackmatrix, submatrix, subvector, sumbasis, swapcol, swaprow, sylvester, toeplitz, trace, transpose, vandermonde, vecpotent, vectdim, vector, wronskian*]

```
> d:=vector([r(t)*cos(theta(t))*sin(phi(t)),r(t)*sin(theta(t)),-r(t)
  *cos(theta(t))*cos(phi(t))]);
```

$$d := [r(t)\cos(\theta(t))\sin(\phi(t)),\ r(t)\sin(\theta(t)),\ -r(t)\cos(\theta(t))\cos(\phi(t))]$$

```
> omega:=vector([0,0,diff(psi(t),t)]);
```

$$\omega := \left[0, 0, \frac{\partial}{\partial t}\psi(t)\right]$$

```
> Vo:=vector([diff(R(t),t),R(t)*diff(psi(t),t),0]);
```

$$Vo := \left[\frac{\partial}{\partial t}R(t),\ R(t)\left(\frac{\partial}{\partial t}\psi(t)\right),\ 0\right]$$

```
> V:=evalm(Vo+vector([diff(d[1],t),diff(d[2],t),diff(d[3],t)])+cross
  prod(omega,d));
```

$$V := \left[\left(\frac{\partial}{\partial t}R(t)\right) + \left(\frac{\partial}{\partial t}r(t)\right)\cos(\theta(t))\sin(\phi(t)) - r(t)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right)\sin(\phi(t))\right.$$
$$+ r(t)\cos(\theta(t))\cos(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right) - \left(\frac{\partial}{\partial t}\psi(t)\right)r(t)\sin(\theta(t)),\ R(t)\left(\frac{\partial}{\partial t}\psi(t)\right)$$
$$+ \left(\frac{\partial}{\partial t}r(t)\right)\sin(\theta(t)) + r(t)\cos(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right) + \left(\frac{\partial}{\partial t}\psi(t)\right)r(t)\cos(\theta(t))\sin(\phi(t)),$$
$$-\left(\frac{\partial}{\partial t}r(t)\right)\cos(\theta(t))\cos(\phi(t)) + r(t)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right)\cos(\phi(t))$$
$$\left. + r(t)\cos(\theta(t))\sin(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right)\right]$$

```
> kinetic:=(1/2)*(m*innerprod(V,V)+M*innerprod(Vo,Vo)+Ib*diff(psi(t)
  ,t)^2);
```

$$
\begin{aligned}
kinetic := \frac{1}{2} m \Bigg(& r(t)^2 \sin(\theta(t))^2 \left(\frac{\partial}{\partial t}\theta(t)\right)^2 \sin(\phi(t))^2 + r(t)^2 \cos(\theta(t))^2 \cos(\phi(t))^2 \left(\frac{\partial}{\partial t}\phi(t)\right)^2 \\
& + 2\left(\frac{\partial}{\partial t}R(t)\right)\left(\frac{\partial}{\partial t}r(t)\right)\cos(\theta(t))\sin(\phi(t)) - 2\left(\frac{\partial}{\partial t}R(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right) r(t)\sin(\theta(t)) \\
& + 2R(t)\left(\frac{\partial}{\partial t}\psi(t)\right)\left(\frac{\partial}{\partial t}r(t)\right)\sin(\theta(t)) + \left(\frac{\partial}{\partial t}\psi(t)\right)^2 r(t)^2 \cos(\theta(t))^2 \sin(\phi(t))^2 \\
& + r(t)^2 \sin(\theta(t))^2 \left(\frac{\partial}{\partial t}\theta(t)\right)^2 \cos(\phi(t))^2 + r(t)^2 \cos(\theta(t))^2 \sin(\phi(t))^2 \left(\frac{\partial}{\partial t}\phi(t)\right)^2 \\
& + R(t)^2 \left(\frac{\partial}{\partial t}\psi(t)\right)^2 + \left(\frac{\partial}{\partial t}r(t)\right)^2 \sin(\theta(t))^2 + \left(\frac{\partial}{\partial t}r(t)\right)^2 \cos(\theta(t))^2 \sin(\phi(t))^2 \\
& + \left(\frac{\partial}{\partial t}\psi(t)\right)^2 r(t)^2 \sin(\theta(t))^2 + r(t)^2 \cos(\theta(t))^2 \left(\frac{\partial}{\partial t}\theta(t)\right)^2 + \left(\frac{\partial}{\partial t}r(t)\right)^2 \cos(\theta(t))^2 \cos(\phi(t))^2 \\
& + \left(\frac{\partial}{\partial t}R(t)\right)^2 + 2\, r(t)^2 \sin(\theta(t))^2 \left(\frac{\partial}{\partial t}\theta(t)\right)\sin(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right) \\
& + 2R(t)\left(\frac{\partial}{\partial t}\psi(t)\right) r(t)\cos(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right) \\
& - 2\left(\frac{\partial}{\partial t}r(t)\right)\cos(\theta(t))\sin(\phi(t))^2 r(t)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right) \\
& - 2\, r(t)^2 \cos(\theta(t))\cos(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right)\sin(\theta(t)) \\
& - 2\left(\frac{\partial}{\partial t}R(t)\right) r(t)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right)\sin(\phi(t)) + 2\left(\frac{\partial}{\partial t}R(t)\right) r(t)\cos(\theta(t))\cos(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right) \\
& + 2R(t)\left(\frac{\partial}{\partial t}\psi(t)\right)^2 r(t)\cos(\theta(t))\sin(\phi(t)) + 2\left(\frac{\partial}{\partial t}r(t)\right)\sin(\theta(t))\, r(t)\cos(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right) \\
& + 2\, r(t)^2 \cos(\theta(t))^2 \left(\frac{\partial}{\partial t}\theta(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right)\sin(\phi(t)) \\
& - 2\left(\frac{\partial}{\partial t}r(t)\right)\cos(\theta(t))\cos(\phi(t))^2 r(t)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right)\Bigg) \\
& + \frac{1}{2} M \left(\left(\frac{\partial}{\partial t}R(t)\right)^2 + R(t)^2 \left(\frac{\partial}{\partial t}\psi(t)\right)^2\right) + \frac{1}{2} Ib \left(\frac{\partial}{\partial t}\psi(t)\right)^2
\end{aligned}
$$

```
> potential:=-m*g*r(t)*cos(theta(t))*cos(phi(t));
```

$$potential := -m\, g\, r(t)\cos(\theta(t))\cos(\phi(t))$$

```
> L:=simplify(kinetic-potential);
```

$$
\begin{aligned}
L := {} & \frac{1}{2} m\, \mathrm{R}(t)^2 \left(\frac{\partial}{\partial t}\psi(t)\right)^2 + \frac{1}{2} M\, \mathrm{R}(t)^2 \left(\frac{\partial}{\partial t}\psi(t)\right)^2 + \frac{1}{2} m\, \mathrm{r}(t)^2 \left(\frac{\partial}{\partial t}\theta(t)\right)^2 + \frac{1}{2} m \left(\frac{\partial}{\partial t}\psi(t)\right)^2 \mathrm{r}(t)^2 \\
& + \frac{1}{2} m\, \mathrm{r}(t)^2 \cos(\theta(t))^2 \left(\frac{\partial}{\partial t}\phi(t)\right)^2 + m \left(\frac{\partial}{\partial t}\mathrm{R}(t)\right)\left(\frac{\partial}{\partial t}\mathrm{r}(t)\right)\cos(\theta(t))\sin(\phi(t)) \\
& + m \left(\frac{\partial}{\partial t}\mathrm{R}(t)\right)\mathrm{r}(t)\cos(\theta(t))\cos(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right) + m\, \mathrm{R}(t)\left(\frac{\partial}{\partial t}\psi(t)\right)^2 \mathrm{r}(t)\cos(\theta(t))\sin(\phi(t)) \\
& - m \left(\frac{\partial}{\partial t}\mathrm{R}(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right)\mathrm{r}(t)\sin(\theta(t)) + m\, \mathrm{R}(t)\left(\frac{\partial}{\partial t}\psi(t)\right)\mathrm{r}(t)\cos(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right) \\
& - m\, \mathrm{r}(t)^2 \cos(\theta(t))\cos(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right)\sin(\theta(t)) \\
& - m \left(\frac{\partial}{\partial t}\mathrm{R}(t)\right)\mathrm{r}(t)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right)\sin(\phi(t)) + m\, \mathrm{R}(t)\left(\frac{\partial}{\partial t}\psi(t)\right)\left(\frac{\partial}{\partial t}\mathrm{r}(t)\right)\sin(\theta(t)) \\
& + \frac{1}{2} m \left(\frac{\partial}{\partial t}\mathrm{R}(t)\right)^2 - \frac{1}{2} m \left(\frac{\partial}{\partial t}\psi(t)\right)^2 \mathrm{r}(t)^2 \cos(\theta(t))^2 \cos(\phi(t))^2 \\
& + m\, \mathrm{r}(t)^2 \left(\frac{\partial}{\partial t}\theta(t)\right)\sin(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right) + \frac{1}{2} Ib \left(\frac{\partial}{\partial t}\psi(t)\right)^2 + \frac{1}{2} M \left(\frac{\partial}{\partial t}\mathrm{R}(t)\right)^2 \\
& + m\, g\, \mathrm{r}(t)\cos(\theta(t))\cos(\phi(t)) + \frac{1}{2} m \left(\frac{\partial}{\partial t}\mathrm{r}(t)\right)^2
\end{aligned}
$$

```
> L1:=simplify(simplify(diff(subs(x=diff(theta(t),t),diff(subs(diff(
  theta(t),t)=x,L),x)),t))-simplify(subs({x=diff(theta(t),t),y=theta
  (t)},diff(subs(theta(t)=y,subs(diff(theta(t),t)=x,L)),y))));
```

$$
\begin{aligned}
L1 := {} & m\, \mathrm{r}(t)^2 \left(\frac{\partial^2}{\partial t^2}\theta(t)\right) + 2\, m\, \mathrm{r}(t)\left(\frac{\partial}{\partial t}\theta(t)\right)\left(\frac{\partial}{\partial t}\mathrm{r}(t)\right) + m\, \mathrm{r}(t)^2 \sin(\phi(t))\left(\frac{\partial^2}{\partial t^2}\psi(t)\right) \\
& + 2\, m \left(\frac{\partial}{\partial t}\mathrm{R}(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right)\mathrm{r}(t)\cos(\theta(t)) + m\, \mathrm{R}(t)\left(\frac{\partial^2}{\partial t^2}\psi(t)\right)\mathrm{r}(t)\cos(\theta(t)) \\
& - m \left(\frac{\partial^2}{\partial t^2}\mathrm{R}(t)\right)\mathrm{r}(t)\sin(\theta(t))\sin(\phi(t)) + 2\, m\, \mathrm{r}(t)\sin(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right)\left(\frac{\partial}{\partial t}\mathrm{r}(t)\right) \\
& + m\, \mathrm{r}(t)^2 \cos(\theta(t))\left(\frac{\partial}{\partial t}\phi(t)\right)^2 \sin(\theta(t)) + m\, \mathrm{R}(t)\left(\frac{\partial}{\partial t}\psi(t)\right)^2 \mathrm{r}(t)\sin(\theta(t))\sin(\phi(t)) \\
& + 2\, m\, \mathrm{r}(t)^2 \cos(\theta(t))^2 \cos(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right) \\
& - m \left(\frac{\partial}{\partial t}\psi(t)\right)^2 \mathrm{r}(t)^2 \cos(\theta(t))\cos(\phi(t))^2 \sin(\theta(t)) + m\, g\, \mathrm{r}(t)\sin(\theta(t))\cos(\phi(t))
\end{aligned}
$$

```
> L2:=simplify(simplify(diff(subs(x=diff(phi(t),t),diff(subs(diff(ph
  i(t),t)=x,L),x)),t))-simplify(subs({x=diff(phi(t),t),y=phi(t)},dif
  f(subs(phi(t)=y,subs(diff(phi(t),t)=x,L)),y))));
```

$$
\begin{aligned}
L2 := {} & 2\,m\,r(t)\cos(\theta(t))^2\left(\frac{\partial}{\partial t}\phi(t)\right)\left(\frac{\partial}{\partial t}r(t)\right) - 2\,m\,r(t)^2\cos(\theta(t))\left(\frac{\partial}{\partial t}\phi(t)\right)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right) \\
& + m\,r(t)^2\cos(\theta(t))^2\left(\frac{\partial^2}{\partial t^2}\phi(t)\right) + m\left(\frac{\partial^2}{\partial t^2}R(t)\right)r(t)\cos(\theta(t))\cos(\phi(t)) \\
& - 2\,m\,r(t)\cos(\theta(t))\cos(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right)\sin(\theta(t))\left(\frac{\partial}{\partial t}r(t)\right) \\
& - 2\,m\,r(t)^2\cos(\theta(t))^2\cos(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right)\left(\frac{\partial}{\partial t}\theta(t)\right) \\
& - m\,r(t)^2\cos(\theta(t))\cos(\phi(t))\left(\frac{\partial^2}{\partial t^2}\psi(t)\right)\sin(\theta(t)) - m\,R(t)\left(\frac{\partial}{\partial t}\psi(t)\right)^2 r(t)\cos(\theta(t))\cos(\phi(t)) \\
& - m\left(\frac{\partial}{\partial t}\psi(t)\right)^2 r(t)^2\cos(\theta(t))^2\cos(\phi(t))\sin(\phi(t)) + m\,g\,r(t)\cos(\theta(t))\sin(\phi(t))
\end{aligned}
$$

```
> simplify(solve({L1,L2},{diff(theta(t),t,t),diff(phi(t),t,t)}));
```

$$
\begin{aligned}
\{ & \frac{\partial^2}{\partial t^2}\phi(t) = \left(-2\cos(\theta(t))\left(\frac{\partial}{\partial t}\phi(t)\right)\left(\frac{\partial}{\partial t}r(t)\right) + \left(\frac{\partial}{\partial t}\psi(t)\right)^2 r(t)\cos(\theta(t))\cos(\phi(t))\sin(\phi(t))\right. \\
& + 2\,r(t)\cos(\theta(t))\cos(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right)\left(\frac{\partial}{\partial t}\theta(t)\right) - g\sin(\phi(t)) \\
& + 2\,r(t)\left(\frac{\partial}{\partial t}\phi(t)\right)\sin(\theta(t))\left(\frac{\partial}{\partial t}\theta(t)\right) - \left(\frac{\partial^2}{\partial t^2}R(t)\right)\cos(\phi(t)) \\
& + 2\cos(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right)\sin(\theta(t))\left(\frac{\partial}{\partial t}r(t)\right) + r(t)\cos(\phi(t))\left(\frac{\partial^2}{\partial t^2}\psi(t)\right)\sin(\theta(t)) \\
& \left. + R(t)\left(\frac{\partial}{\partial t}\psi(t)\right)^2\cos(\phi(t))\right)/(\cos(\theta(t))\,r(t)),\ \frac{\partial^2}{\partial t^2}\theta(t) = -\left(2\left(\frac{\partial}{\partial t}\theta(t)\right)\left(\frac{\partial}{\partial t}r(t)\right)\right. \\
& + r(t)\sin(\phi(t))\left(\frac{\partial^2}{\partial t^2}\psi(t)\right) + 2\left(\frac{\partial}{\partial t}R(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right)\cos(\theta(t)) + R(t)\left(\frac{\partial^2}{\partial t^2}\psi(t)\right)\cos(\theta(t)) \\
& - \left(\frac{\partial^2}{\partial t^2}R(t)\right)\sin(\theta(t))\sin(\phi(t)) + 2\sin(\phi(t))\left(\frac{\partial}{\partial t}\psi(t)\right)\left(\frac{\partial}{\partial t}r(t)\right) \\
& \left. + r(t)\cos(\theta(t))\left(\frac{\partial}{\partial t}\phi(t)\right)^2\sin(\theta(t)) + R(t)\left(\frac{\partial}{\partial t}\psi(t)\right)^2\sin(\theta(t))\sin(\phi(t))\right)
\end{aligned}
$$

$$+ 2\,r(t)\cos(\theta(t))^2\cos(\phi(t))\left(\frac{\partial}{\partial t}\phi(t)\right)\left(\frac{\partial}{\partial t}\psi(t)\right)$$
$$-\left(\frac{\partial}{\partial t}\psi(t)\right)^2 r(t)\cos(\theta(t))\cos(\phi(t))^2\sin(\theta(t)) + g\sin(\theta(t))\cos(\phi(t))\Big) / r(t)\}$$

References

[1] A.R. Golafshani and J.D. Aplevich. Computation of time-optimal trajectories for tower cranes. *The 4th IEEE Conference on Control Applications*, pages 1134–1139, September 1995.

[2] Ali R. Golafshani. *Modeling and time optimal control design of a tower crane.* Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada, 1994.

[3] *Making The Right Move: Series Profile Tower Cranes WOLFF.* MAN GHH Logistics GmbH, P.O.Box 2640, D-74016 Heibronn, Hans-Riesser-Strasse 7, D-74076 Heibronn, Germany.

[4] *Design progress: HC cranes.* Liebherr-Werk Biberach GmbH, P.O.Box 1663, D-7950 Biberach/Rib 1, Germany.

[5] W. B. Ledbetter R. L. Peurifoy and C. J. Schexnayder. *Construction Planning, Equipment, and Methods.* McGraw-Hill, New York, fifth edition, 1996.

[6] D.E. Dickie. *Crane Handbook.* Construction Safety Association of Ontario, 74 Victoria St. Toronto, Ontario, M5C 2A5, Canada, first edition, October 1975.

[7] David A. Day and Neal B.H. Benjamin. *Construction Equipment Guide.* John Wiley & Sons Inc., New York, second edition, 1991.

[8] *TOPKIT MD and Maxi TOPKIT series: high performance cranes for the most complex sites.* Potain, 18, rue de Charbonnires, BP 173 - F 69132 Ecully Cedex, France.

[9] Towers take control. *Cranes Today*, pages 25–31, Feb. 1997.

[10] Yehiel Rosenfeld and Aviad Shapira. Automation of existing tower cranes: economic and technological feasibility. *Automation in Construction*, 7(4):285–298. May 1998.

[11] Yehiel Rosenfeld. Automation of existing cranes: from concept to prototype. *Automation in Construction*. 4(2):125–138. June 1995.

[12] H. Butler. G. Honderd. and J. V. Amerongen. Model reference adaptive control of a gantry crane scale model. *IEEE Control Systems Magazine*. 11(1):57–62. Jan. 1991.

[13] J.W. Auernig and H. Troger. Time optimal control of overhead cranes with hoisting of the load. *Automatica, the Journal of IFAC*, 23(4):437–447, July 1987.

[14] Kamal A.F. Moustafa and T.G. Abou-El-Yazid. Load sway control of overhead cranes with load hoisting via stability analysis. *JSME International Journal*, 39(1):34–40, March 1996.

[15] Kamal A.F. Moustafa. Feedback control of overhead cranes swing with variable rope length. *Proceedings of the 1994 American Control Conference*, 1:691–695, June 1994. Part 1 (of 3).

[16] Y. Sakawa. Optimal control of overhead traveling crane. *International Journal of Control*, pages 2085–2105, Nov. 1997.

[17] Y. Sakawa and K. Sato. Modeling and control of a flexible rotary crane. *International Journal of Control*, 48(5):2085–2105, Nov. 1988.

[18] Y. Sakawa and A. Nakazumi. Modeling and control of a rotary crane. *Journal of Dynamic Systems, Measurement, and Control*, 107:200–206, September 1985.

[19] K. Hara. T. Yamamoto. A. Kobayashi, and M. Okamoto. Jib crane control to suppress load swing. *International Journal of Systems Science*. 20(5):715–731. May 1989.

[20] Crane automation on the way. *World Construction*. page 48. November 1986.

[21] *The Litronic crane control system.* Liebherr-Werk Biberach GmbH. P.O.Box 1663. D-7950 Biberach/Rib 1. Germany.

[22] Bauma 98 exhibition. The world fair for construction machinery, building material machinery. construction vehicles and equipment. *The New Munich Trade Fair Centre*. 30 March - 5 April 1998.

[23] Donald T. Greenwood. *Principles of Dynamics.* Prentice-Hall, Englewood Cliffs, New Jersey, second edition, 1988.

[24] William L. Brogan. *Modern Control Theory.* Prentice-Hall, Englewood Cliffs, New Jersey, third edition, 1991.

[25] Chi-Tsong Chen. *Linear System Theory and Design.* Saundres College Publishing, 1984.

[26] Lamberto Cesari. *Optimization-Theory and Applications.* Springer-Verlag, New York, 1983.

[27] Frank M. Callier and Charles A. Desoer. *Linear System Theory.* Springer-Verlag, New York, 1991.

[28] M. Vidyasagar. *Nonlinear System Analysis.* Prentice-Hall, Englewood Cliffs, New Jersey. first edition. 1978.

[29] W.A. Gruver and E. Sachs. *Algorithmic Methods in Optimal Control.* Pitman Advanced Publishing Program. Boston. 1981.

[30] S.K. Mitter L.S. Lasdon and A.V. Waren. The conjugate gradient method for optimal control problems. *IEEE Transactions on Automatic Control.* 12:132–138. 1967.

[31] C.T. Kelley and E.W. Sachs. Quasi-Newton methods and unconstrained optimal control problems. *SIAM Journal of Control and Optimization.* 25(6):1503–1516, 1987.

[32] J.H. Chou and I.R. Horng. Application of Chebyshev polynomials to the optimal control of time-varying linear systems. *International Journal of Control.* 41(1):135–144. 1985.

[33] J. Vlassenbroeck. A Chebyshev polynomial method for optimal control with state constraints. *Automatica*, 24(4):499–506, 1988.

[34] M. Razzaghi. Optimal control of time-varying linear systems via Fourier series. *Journal of Optimization Theory and Applications*, 65(2):375–384, 1990.

[35] T. Endow. Optimal control via Fourier series of operational matrix of integration. *IEEE Transactions on Automatic Control*, 34(7):770–773, 1989.

[36] M. Razzaghi. Solution of linear two-point boundary value problems via Taylor series. *Journal of Franklin Institute*, 326:511–521, 1989.

[37] C.P. Neuman and A. Sen. A suboptimal control algorithm for constrained problems using cubic splines. *Automatica*, 19:601–613, 1973.

[38] C.J. Goh and K.L. Teo. Control parametrization: a unified approach to optimal control problems with general constraints. *Automatica*, 24(1):3–18, 1988.

[39] K.L. Teo and L.S. Jennings. Nonlinear optimal control problems with continuous state inequality constraints. *Journal of Optimization Theory and Applications*, 63(1):1–22, 1989.

[40] M. Fukushima and Y. Yamamoto. A second order algorithm for continuous time nonlinear optimal control problems. *IEEE Transactions on Automatic Control*, 31:673–676, 1986.

[41] S.P. Han. Superlinearly convergent variable metric algorithm for general nonlinear programming problems. *Mathematical Programming*, 11:263–282, 1976.

[42] S.P. Han. A globally convergent method for nonlinear programming. *Journal of Optimization Theory and Applications*, 22(3):297–309, July 1977.

[43] M.J.D. Powell. Algorithms for nonlinear constraints that use Lagrangian functions. *Mathematical Programming*, 14:224–248, 1978.

[44] Michael Athans and Peter L. Falb. *Optimal Control: An Introduction to the Theory and Its Applications*. McGraw-Hill, New York, 1966.

[45] Francis B. Hildebrand. *Advanced Calculus for Applications*. Prentice-Hall, Englewood Cliffs, New Jersey, 2nd edition, 1976.

[46] Arthur E. Bryson and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation, and Control.* Hemisphere Publication Corporation, New York, 1975.

[47] Philip E. Gill, Walter Murray, and Margaret H. Wright. *Practical Optimization.* Academic Press, New York, 1981.

[48] D. Q. Mayne and E. Polak. First-order strong variation algorithms for optimal control. *Journal of Optimization Theory and Applications*, 16(3/4):277–301, 1975.

[49] E.B. Lee and L. Markus. *Foundations of Optimal Control Theory.* John Wiley & Sons, New York, 1967.

[50] Tom M. Apostol. *Calculus.* volume I. Blaisdell Publishing, Waltham, Massachusetts, second edition, 1967.

[51] David G. Luenberger. *Linear and Nonlinear Programming.* Addison-Wesley, second edition, 1984.

[52] Brian D.O. Anderson and John B. Moore. *Optimal Control: Linear Quadratic Methods.* Prentice-Hall, Englewood Cliffs, New Jersey, 1990.

[53] Carl De Boor. *A practical guide to splines.* Springer-Verlag, New York, 1978.

[54] E04NAF. NAG fortran library routine for solving quadratic programming or linear programming problems.

[55] Andrew Grace. *Optimization Toolbox User's Guide.* MathWorks, South Natick, MA, November 1990.

[56] Robert J. Vanderbei. *LOQO User's Manual - Version 4.05.* Princeton University, School of Engineering and Applied Science, Department of Operations Research and Financial Engineering, Princeton, NJ 08544, July 1999.

[57] A.R. Golafshani and J.D. Aplevich. Time-optimal control of basic motion of tower cranes. *The IASTED Conference on Intelligent Systems and Control,* October 1999.

[58] W.H. Press. B.P. Flannery. S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press. Cambridge, 1992.

[59] Kiwi-Newton Construction Ltd. *Industrial/Comercial General Contractor.* Waterloo. Ontario, 1999.