Control of the Spatial Double Inverted Pendulum

Jie Ni

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

September 2011

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Engineering.

Abstract

The stabilization of a hip-actuated spatial double inverted pendulum can be considered as a problem of postural control of a humanoid robot. Based on an existing model of this underactuated mechanical system with four degrees of freedom, the ultimate objective is to design a suitable controller to achieve *global stabilization* around the unstable upright equilibrium position. This thesis presents a number of control algorithms and simulation results that provide either *local stabilization* or *semi-global swing-up*.

For the effort of *local stabilization* in the vicinity of the upright equilibrium position, both an LQR controller and three types of linearization-based sliding mode control algorithms are presented. The region of convergence of the LQR controller is investigated. System performance and robustness against disturbances are compared for all controllers.

In order to realize *semi-global swing-up*, two types of nonlinear sliding mode control approaches are explored for the swing up of the system in an attempt to bring the system into the region of convergence of the local linear controllers. The *hybrid approach* is proposed to switch from the swing-up controller to a local linear controller under certain conditions in the vicinity of the upright equilibrium to complete the stabilization effort. However, despite extensive tuning of the controllers, it has not been possible to achieve *global stabilization* with such an approach. Further investigation is needed in order to resolve this issue.

The main contribution of this thesis is a successful extension of existing 2-dimensional sliding mode control algorithms into 3-D for the control of the spatial double inverted pendulum. The linearization-based sliding mode controllers serve as alternatives to LQR for local stabilization. The nonlinear sliding mode controllers are able bring the system from a configuration far from the upright equilibrium to the vicinity of the unstable upright equilibrium in *semi-global swing-up*.

Résumé

La stabilisation d'un double pendule spatiale inversé actionné à la hanche peut-être considérée comme un problème de contrôle de la posture d'un robot humanoïde. Basé sur un modèle existant de ce système mécanique sous-actionné avec quatre degrés de liberté, l'ultime objectif est de concevoir un régulateur approprié pour obtenir une *stabilisation globale* autour de l'instable position d'équilibre debout. Cette thèse présente un certain nombre d'algorithmes de contrôle et les résultats de simulation qui permettent une *stabilisation locale* ou *semi-globale pivoter-vers-le-haut*.

Pour l'effort de *stabilisation locale* dans le voisinage de la position d'équilibre en position verticale, à la fois un contrôleur LQR et trois types de linéarisation basée sur des algorithmes de contrôle de mode glissant sont présentés. La région de la convergence du contrôleur LQR est étudiée. La performance et la robustesse du système sont comparées pour tous les contrôleurs.

Afin de réaliser la strategie *semi-globale pivoter-vers-le-haut*, deux types d'approches de commande non linéaire de mode glissant sont explorés pour le balancement du système dans un essai pour amener le système dans la région de convergence locale des contrôleurs linéaires. *L'approche hybride* est proposée pour passer du contrôleur pour pivoter-vers-le-haut à un contrôleur linéaire local sous certaines conditions dans le voisinage de l'équilibre en position verticale afin de compléter l'effort de stabilisation. Toutefois, malgré des ajustements des contrôleurs, il n'a pas été possible de parvenir à une *stabilisation globale* avec une telle approche. Une enquête plus profonde est nécessaire pour résoudre ce problème.

La contribution principale de cette thèse est la réussite une d'extension d'algorithmes de commande de 2-dimensions de mode glissant qui existent pour le cas de 3-D pour le contrôle du double pendule inversé spatial. Les contrôleurs de mode glissant basés sur un modèle du système linéarisé servent comme alternatives au contrôleur LQR pour la stabilisation locale. Les contrôleurs de mode glissant non-linéaires sont capables, à partir d'une configuration loin de l'équilibre de mettre le système dans la proximité de l'équilibre debout vertical utilisant le principe *semi-global pivoter-vers-le-haut*.

Acknowledgements

First, I would like to extend my sincere thanks to my supervisor Professor Hannah Michalska and Dr. Fernando Castaños Luna. They have both guided me and provided me with thoughtful suggestions through my research. I am grateful for their encouragement and support.

I also would like to express my gratitude to Arash Mohtat for the help on building model of the spatial double inverted pendulum system in SimMechanics.

In addition, I would like to thank my good friend Carl Mueller-Roemer for the assistance in translating the abstract into French.

Finally, I am indebted to my parents for their love, support and encouragement throughout my study.

Contents

List of Figures

List of Tables

List of Acronyms

COM	Center of Mass
DOF	Degrees of Freedom
FOHSMC	First-Order Hierarchical Sliding Mode Controller
IC	Initial Condition
LQR	Linear-Quadratic Regulator
OR	Optimal Ratio
OS	Overshoot
ROC	Region of Convergence
SDIP	Spatial Double Inverted Pendulum
SM	Sliding Mode
SMC	Sliding Mode Control
SOOSMC	Second-Order Optimal Sliding Mode Controller
SOSM	Second-Order Sliding Mode
STA	Super-Twisting Algorithm
VGSTA	Variable Gain Super-Twisting Algorithm
w.r.t.	with respect to

Chapter 1

Background

Humanoid robots are autonomous robots that resemble humans in terms of appearances and behaviours. They are typically designed to perform various tasks to make our lives easier and so are becoming increasingly important for applications. As we gain more knowledge in the area of human cognition, we are better equipped to build more advanced humanoid robots that could perform more complicated tasks.

Robot control is the backbone of robotics. While the control of robots have fascinated control engineers for several decades, control of humanoid robots is playing an increasingly important role in the development of robotics.

The subject of postural stability has been extensively studied in humans, yet the stabilization of the standing posture has not received much attention in robotics literature. The stabilization of inverted pendulum systems is a classical problem in the control field that resembles postural stabilization of humans. However, most of the existing control techniques are applied to 2-D systems, which cannot represent the 3-D movements of humans and animals. (See Fig. 1.1)

In addition, the balance control in the absence of actuation at the ankle has been poorly studied. Finding effective control strategies that are applicable to systems actuated only at the hip joint would contribute to the art of humanoid robot design and control. One possible application of this work can be the stabilization of a walking robot standing on a single leg.

Fig. 1.1 Birds, viverridea, and people, among several other species, master standing behaviours on a small footprint.

One way to model such type of system is to construct a spatial double inverted pendulum (SDIP), which is the 3-dimensional version of an acrobot [1]. In case of a negligible small footprint, we can limit the actuation torques to act at the hip joints while the ankle joint is fixed to the floor and remains un-actuated. We aim to find a suitable method to stabilize and balance such a system in this thesis.

Chapter 2

Literature Review

2.1 Underactuated Mechanical systems

A mechanical system is underactuated if it has more degrees of freedom (DOF) than the number of actuators. Some of the most popular underactuated mechanical systems that exist in control literature for the purpose of stabilization include *Inverted Pendulum*, *Pendubot*, and *Acrobot*. All of them operate in a plane, and a general description of these systems can be found in [1].

The SDIP system falls into the category of the nonlinear underactuated mechanical systems. Stabilization of underactuated systems, forced by fewer actuators than DOF, presents a challenging problem. In this chapter, stabilizing control strategies for a number of similar systems are introduced and compared, both in the plane and in the space.

2.2 Review of Sliding Mode Control

Sliding Mode Control (SMC), also known as Variable Structure Control (VSC), is the emphasis of this thesis, thus its concepts should be reviewed here. It is a relatively new control method that applies the control action primarily as a discontinuous state function to drive the system to the so-called "sliding surface", usually described by the equation $s = 0$. SMC can be used to control both linear and nonlinear systems [2]. Also, SMC can be in the form of discontinuous control law with signum functions of the sliding variable s, continuous functions of s, or a linear combination of both.
Recently invented, Higher-Order Sliding Modes (HOSM) generalize the SM idea, acting

on higher order time derivatives of the sliding variable s instead of just influencing its first derivative as it happens in standard SMC to provide higher accuracy in realization, see [3]. Standing out from a number of algorithms that exist in HOSM are the twisting algorithm and super-twisting algorithm (STA). Both are second order sliding mode (SOSM) control techniques that bring the system to the desired sliding manifold in finite time and ensure that $s = \dot{s} = 0$. The twisting algorithm is used mostly when the system is of relative degree 2 w.r.t. the sliding variable s, see [3], and the control signal is a function of both s and $\dot{s}$ while the advantage of STA over the standard twisting algorithm is twofold:

- STA does not need any information on the time derivative $\dot{s}$ of the sliding variable s.
- The control signal is a continuous function of the sliding variable s. As a result, the chattering effect of the controller output is reduced significantly.

2.3 Linearization-based Stabilization Strategies

By way of a literature survey of existing methods to stabilize the pendulum systems in the vicinity of the unstable upright equilibrium, it has been found that the LQR with state feedback is the most common approach as it has been used to control a number of underactuated inverted pendulum systems including an inverted pendulum on a circular base [4], a trip inverted pendulum [5], an acrobot [1], and the SDIP system [6]. The design of an LQR controller is simple, yet, since the design is based on the linearized state space model around the unstable upright equilibrium point, it can only stabilize the systems when initial states are relatively close to the upright equilibrium.

In addition, SMC has been used to provide local stabilization of underactuated systems such as a Cart-Pendulum system [7] and the rotary Furuta's pendulum system [8]. The former relies on nonlinear state space system transformation and utilizes SOSM with a nonlinear sliding surface while the latter relies on a linearized state space model and the design of variable gain STA (VGSTA) to achieve system stabilization with uncertainties and disturbances. However, those techniques have been studied in application to planar robots only.

2.4 Swing-up Control Strategies in 2-D

In the robotics literature, swing-up and stabilization of double inverted pendulum systems, such as an acrobot or a pendubot, have received great attention in the past 15 years. An acrobot is hip-actuated, and can be viewed as the 2-D version of the SDIP system. One of the earliest work on the swing up control of the acrobot comes from Spong [9], [10]. In [9], he proposed a method based on partial feedback linearization [11] for swing-up and LQR for the stabilization phase. In swing-up phase, a well-designed reference function associating the angular velocity of the unactuated joint with the angular position of the actuated one is used to properly command the swing up process; in [10], the swing up effort relies on an energy-based algorithm.

More recently, better methods have been proposed for the swing up of the acrobot such as the output zeroing controller proposed by T. Yonemura et al. [12] and a new partial linearization controller proposed by T. Henmin et al. [13]. The former takes an output function defined by the angular momentum around the unactuated joint at the foot and controls it to zero in order to perform the swing up; the latter extends the idea by Spong [9] and modifies the reference function to allow simultaneous swing up of both links. However, neither approach was able to completely stabilize the system without switching to a linear controller under certain conditions after the acrobot is swung up. Alternatively, a tracking controller can be implemented to allow the acrobot to track certain oscillatory trajectories, [14], in the vicinity of the upright equilibrium position.

SMC has also been used to control a class of underacutated mechanical systems, especially for the swing-up and stabilization of pendulum systems. F. Mnif [15] designed nonlinear first order SM controller for the swing up of the whirling pendulum based on partial feedback linearization, in which a sliding surface similar to the reference function defined in [9] was used. S.A. Puga et al. [16] designed a hybrid SOSM tracking controller for an acrobot, also based on partial feedback linearization, but used the unactuatd angle at the ankle as the sliding variable, thus by first using the twisting algorithm to swing up the lower link, the upper link will rotate around by almost a full circle before the tracking controller takes over to track the desired trajectories described in [14].

Another common first order SM approach for the swing up and the stabilization of under-

actuated pendulum systems is to decouple the system expressions into the actuated and unactuated subsystems, where a suitable pair of first level sliding surfaces in two different variables, s_1 and s_2, is constructed and a second level sliding surface S is constructed as a linear combination of s_1 and s_2. The control law acts directly on S to guarantee that the second-level sliding surface can be reached in an asymptotically stable manner, but the SM parameters need to change according to different conditions in order to guarantee the asymptotic stability of s_1 and s_2. Existing literature gives such a SM control approach different names, such as the Decoupling SM Control of a pendulum-cart system by M.S. Park et al. [17], Double Layer SM Control for an overhead crane system by W. Wang et al. [18], Hierarchical SM control to swing up a pendubot by D. Qian et al. [19]. According to the literature search, this approach has only been analyzed and tested in single-input systems with underactuation degree of one, *operating in a plane.* Such an approach could be easily extended to work in our SDIP system.

If the control input remains scalar and the underactuation degree is more than two, a more powerful SM algorithm is needed. Based on the concept of hierarchical structure of the sliding surfaces, the so called "Incremental SM controller" was proposed by Y. Hao et al. [20] and it offers $2n-1$ sliding surface layers for $2n$ state variables. In short, the i^{th} sliding surface depends on the $(i-1)^{th}$ sliding surface. The controller is designed to guarantee the rapid convergence to zero for all sliding surfaces and it has been tested in a double-inverted pendulum and cart system, which has 3 subsystems with underactuation degree of 2.

2.5 Swing-up Control Strategies in 3-D

Many of the aforementioned approaches have been successfully applied to simple planar systems, but have not been extended to 3-D, which is the type of system the SDIP belongs to. Miyashita et al., [21], considered a method based on the idea of output zeroing controller [12] for an acrobot to control a 3-D system called AcroBOX, which can be viewed as a 3-D version of an acrobot, except that it has 5-DOF with 3-DOF at the ankle, which is unactuated. For our SDIP system, Xinjilefu et al. achieved global stabilization by using a stochastic programming algorithm to achieve on-line minimization of the system Lagrangian, [22], as well as a hybrid approach with energy and passivity based control, [6], for the swing-up phase combined with LQR for complete stabilization.
In order to implement the SMC for the SDIP system, the existing control methods, such

as the VGSTA described in [8] and Hierarchical SM control described in [19] can be easily extended into 3-D by the vector control approach described in [2], using diagonal gain matrices as parameters of the control system since the sliding surfaces and the torque are 2-dimensional vectors.

Chapter 3

The Objectives of the Thesis

The aim of this thesis is the study and the control of an underactuated mechanical system in 3-D: a spatial double inverted pendulum (SDIP) with 4 DOF actuated only at the hip. Two major linearization-based approaches to stabilization around the unstable upright equilibrium position are examined:

- The LQR control
- The quasi-linear SMC methods

The ROC of the system using both algorithms is approximated by way of simulations. In addition, two nonlinear SM methods are explored towards complementing the swing-up of the system in a semi-global attempt to bring the system towards the unstable upright equilibrium.

Chapter 4

Dynamic Models

Robot dynamics is concerned with the relationship between the actuation forces and the motion of bodies in a mechanical system. It is important in mechanical design, control, and computer simulation. The robot dynamics are mainly classified into the following two problems:

- *Forward dynamics*: given the applied joint forces/torques, determine the joint accelerations.
- *Inverse dynamics*: given the joint accelerations, work out applied join forces/torques.

A dynamic model links the forces/torques with accelerations of a rigid body mechanical system and incorporates the architecture and inertia parameters of the system.

In this chapter, we will show how the equations of motion of a robotic system can be derived using Euler-Lagrange Equation, based on the results from Xinjilefu [23] using forward dynamic calculations. The system model is then presented in detail for the later purposes of controller design and simulation.

4.1 Joint Space Formulation using Euler-Lagrange Equation

In this section, we show that the equations of motion of a robotic system can be derived using Euler-Lagrange equation.

The motion of a robotic system can be described by a set of differential equations in the joint space, referred to as the joint space formulation:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau, \tag{4.1}$$

By convention, the symbol q denotes the vector of joint position, it specifies the coordinates of a point in the system's configuration space. The elements of q are referred to as the *generalized coordinates*. In a similar fashion, we define $\dot{q}$ as the generalized velocity, $\ddot{q}$ as the generalized acceleration. $M(q)$ is a symmetric positive definite matrix called *generalized inertia matrix*; $C(q,\dot{q})$ is a square matrix where $C(q,\dot{q})\dot{q}$ is the vector of Coriolis and centrifugal terms; $G(q)$ is the vector of gravity terms and τ is the vector of the generalized forces. More terms can be added to Eq. (4.1) to account for various dynamic effects, such as frictions, external forces, etc.. In this thesis, Eq. (4.1) suffices for controller algorithm development and these dynamic effects are neglected from the model.

For complex mechanical systems, it is either very difficult to derive the closed form for the terms in Eq. (4.1), or there is no need to calculate them. However, knowing their relationship to the energy of the system could help developing control algorithms, and it is a standard procedure in classical mechanics to derive the equations of motion from Lagrangian of the mechanical system. The Lagrangian is defined as:

$$L(q,\dot{q}) = K(q,\dot{q}) - V(q) \tag{4.2}$$

where $L(q,\dot{q})$ is the Lagrangian function, $K(q,\dot{q})$ and $V(q)$ are the kinetic and potential energies of the system, respectively. The kinetic energy is related to the generalized inertia matrix by

$$K(q,\dot{q}) = \frac{1}{2}\dot{q}^{\top}M(q)\dot{q} \tag{4.3}$$

Conversely, the generalized inertia matrix can be calculated given the expression of the system kinetic energy by

$$M_{i,j}(q) = \frac{\partial^2 K(q,\dot{q})}{\partial\dot{q}_i\partial\dot{q}_j} \tag{4.4}$$

The dynamic equations of motion can then be developed using Euler-Lagrange equation

for the system:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau \tag{4.5}$$

If Eq. (4.5) is written in the scalar form, then we have for $k = 1, \cdots, n$

$$\sum_{j=1}^{n} M_{kj}(q)\ddot{q}_j + \sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij}^{k}(q)\dot{q}_i\dot{q}_j + G_k(q) = e_k^{\top}\tau \tag{4.6}$$

where e_k is the k-th standard basis vector in $\mathbb{R}^n$, and where the gravity terms and the Christoffel symbols of the first type [24] are given by

$$G_k(q) = \frac{\partial}{\partial q_k}V(q) \tag{4.7}$$

$$C_{ij}^{k}(q) = \frac{1}{2}\left[\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ki}(q)}{\partial q_j} - \frac{\partial M_{jk}(q)}{\partial q_i}\right] \tag{4.8}$$

Eq. (4.6) is an alternative expression of Eq. (4.1) in scalar form. The gravity terms in Eq. (4.1) is related to the potential energy of the system by

$$G(q) = \frac{\partial V(q)}{\partial q} = [G_1(q), ..., G_n(q)]^{\top} \tag{4.9}$$

The elements of $C(q, \dot{q})$ in Eq. (4.1) can be defined using the Christoffel symbols $C_{ij}^{k}(q)$ as

$$C_{i,j}(q, \dot{q}) = \sum_{k=1}^{n} C_{ij}^{k}(q)\dot{q}_k \tag{4.10}$$

4.2 System Model

It has been explained by Xinjilefu [23] that as far as the problem of postural control is concerned, the simplest multi-body system that can capture the essential kinematic and dynamic features of standing in the upright position is the SDIP system that can be viewed as an open kinematic chain. Referring to Fig. 4.1, the model has two main rigid cylindrical links of lengths l_i, and radii r_i, with masses m_i, $i = 1, 2$, representing the legs and upper body, respectively. Additional point masses $\bar{m}_1$ and $\bar{m}_2$ are attributed to the hip and to the head, where we consider the *hip* belonging to the lower link and the *head* belonging

to the upper link. The motion of the links is restrained by two universal joints: at the *hip* and at the *ankle*. It is assumed that the balancing act is implemented by two actuating torques at the *hip* while the *ankle* is unactuated.

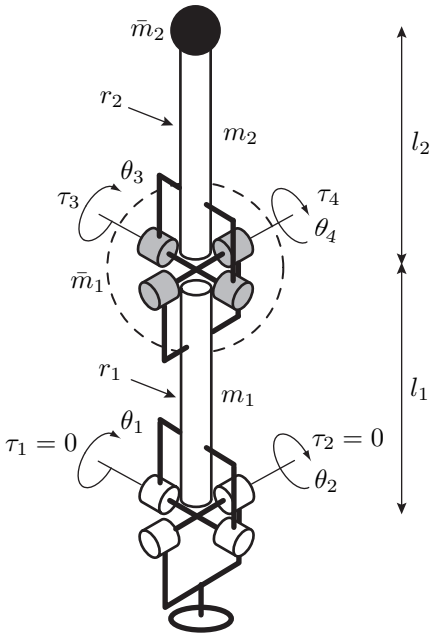

Fig. 4.1 Model of the SDIP with 4-DOF with underactuation degree of 2

In this thesis, we base our analysis on the inertial frame of reference as the Cartesian X-Y-Z coordinates system with origin at the *foot*. We consider the axis of rotation of θ_1 as the X-axis, that of θ_2 as the Z-axis, and Y being the vertical axis perpendicular to the ground (X-Z plane). The model is easily seen to have the following inertia parameters for each link:

- Mass: $m_i + \bar{m}_i$
- COM relative to the bottom of the link in the upright equilibrium:

$$C_i = \begin{bmatrix} 0 \\ \frac{1}{2} l_i \dfrac{m_i + 2\bar{m}_i}{m_i + \bar{m}_i} \\ 0 \end{bmatrix}$$

- Centroidal Mass Moment of Inertia:

$$H_i = \begin{bmatrix} \frac{1}{12}m_i(3r_i^2 + l_i^2) + \frac{1}{4}\frac{m_i\bar{m}_i}{m_i+\bar{m}_i}l_i^2 & 0 & 0 \\ 0 & \frac{1}{2}m_i r_i^2 & 0 \\ 0 & 0 & \frac{1}{12}m_i(3r_i^2 + l_i^2) + \frac{1}{4}\frac{m_i\bar{m}_i}{m_i+\bar{m}_i}l_i^2 \end{bmatrix}$$

for $i = 1, 2$.

Another important property of this model, yet hidden from the previous descriptions, is that the order of joint rotations matters when visualizing the actual configuration given the values of joint angles: the rotation of *even* joint angles θ_2 and θ_4 are effectuated before θ_1 and θ_3, respectively, and the rotations do not commute. For example, a seemingly symmetric configuration of $q = [45°, -45°, 0°, 0°]^\top$ would look different when viewed from different perspectives. When viewed from the Y-Z plane, the lower link makes an angle of 45° w.r.t. the ground (X-Z plane). However, when viewed from the X-Y plane, it makes less than 40° w.r.t. the ground, as observed using SimMechanics, which is the system modelling tool discussed in the next chapter.

The *forward dynamic* calculations were performed by Xinjilefu [23] to derive the system equations. Let $\sin q_i = s_i$ and $\cos q_i = c_i$ for $i = 1, \cdots, 4$, then the symmetric, positive definite generalized inertia matrix $M(q)$ is obtained to have the following entries

$$\begin{aligned}
M_{1,1} &= \tfrac{1}{2}m_1{r_1}^2 + \tfrac{1}{2}m_2{r_2}^2 + \left[D - \tfrac{1}{4}m_1{r_1}^2\right]c_2^2 \\
&\quad + A[1 - (c_2s_4 + s_2c_3c_4)^2] + 2Bc_2(c_2c_3c_4 - s_2s_4) && \text{(4.11a)} \\
M_{1,2} &= s_3c_4[Bs_2 + A(c_2s_4 + s_2c_3c_4)] && \text{(4.11b)} \\
M_{1,3} &= c_4[Bc_2c_3 + A(c_2c_4 - s_2c_3s_4)] + \tfrac{1}{2}m_2{r_2}^2c_2 && \text{(4.11c)} \\
M_{1,4} &= s_3[-Bc_2s_4 + (A + \tfrac{1}{2}m_2{r_2}^2)s_2] && \text{(4.11d)} \\
M_{2,2} &= D + \tfrac{1}{4}m_1{r_1}^2 + \tfrac{1}{2}m_2{r_2}^2 + A[1 - s_3^2c_4^2] + 2Bc_3c_4 && \text{(4.11e)} \\
M_{2,3} &= As_3s_4c_4 && \text{(4.11f)} \\
M_{2,4} &= (A + \tfrac{1}{2}m_2{r_2}^2)c_3 + Bc_4 && \text{(4.11g)} \\
M_{3,3} &= Ac_4^2 + \tfrac{1}{2}M_2{r_2}^2 && \text{(4.11h)} \\
M_{3,4} &= 0 && \text{(4.11i)} \\
M_{4,4} &= A + \tfrac{1}{2}m_2{r_2}^2 && \text{(4.11j)}
\end{aligned}$$

where A, B, D are constants,

$$A = \tfrac{1}{3} m_2 {l_2}^2 + \bar{m}_2 {l_2}^2 - \tfrac{1}{4} m_2 {r_2}^2 \tag{4.12a}$$
$$B = (\tfrac{1}{2} m_2 + \bar{m}_2) l_1 l_2 \tag{4.12b}$$
$$D = (\bar{m}_1 + \bar{m}_2 + \tfrac{1}{3} m_1 + m_2) {l_1}^2 \tag{4.12c}$$

The potential energy is

$$V = E c_1 c_2 + F(c_1 c_2 c_3 c_4 - s_1 s_3 c_4 - c_1 s_2 s_4) \tag{4.13}$$

where E, F are constants,

$$E = (\tfrac{1}{2} m_1 + \bar{m}_1 + m_2 + \bar{m}_2) g\, l_1 \tag{4.14a}$$
$$F = (\tfrac{1}{2} m_2 + \bar{m}_2) g\, l_2 \tag{4.14b}$$

with g denoting the acceleration of gravity. The vector $G(q)$ representing the gravity terms. Here we use Eq. (4.7) instead, since the potential energy of the system is available to us.

$$G_1 = - E s_1 c_2 + F(-s_1 c_2 c_3 c_4 - c_1 s_3 c_4 + s_1 s_2 s_4) \tag{4.15a}$$
$$G_2 = - E c_1 s_2 + F(-c_1 s_2 c_3 c_4 - c_1 c_2 s_4) \tag{4.15b}$$
$$G_3 = F(-c_1 c_2 s_3 c_4 - s_1 c_3 c_4) \tag{4.15c}$$
$$G_4 = F(-c_1 c_2 c_3 s_4 + s_1 s_3 s_4 - c_1 s_2 c_4) \tag{4.15d}$$

The system is governed by

$$M(q)\ddot{q} + \dot{q}^\top Q(q)\dot{q} + G(q) = \tau \tag{4.16}$$

where $\tau \triangleq [0, 0, \tau_3, \tau_4]^\top \in \mathbb{R}^4$, Q is a matrix such that $C(q, \dot{q})\dot{q} \triangleq \dot{q}^\top Q(q)\dot{q} \in \mathbb{R}^4$, the later definition being standard in the literature. The terms involving $\dot{q}_i \dot{q}_i$ represent the centrifugal forces and the terms involving $\dot{q}_i \dot{q}_j, i \neq j$, stand for Coriolis forces. By recalling that $M(q)$ is positive definite and hence invertible, we can introduce the Legendre transformation with respect to $\dot{q}$ [25],

$$p = \frac{\partial L}{\partial \dot{q}} = M(q)\dot{q} \tag{4.17}$$

where p is also known as the generalized momentum and L is the Lagrangian function, the model of the system Eq. (4.16) can be re-written in the Legendre normal form,

$$\dot{q} = M^{-1}(q)p \tag{4.18a}$$

$$\dot{p} = -G(q) + p^{\top}\tilde{Q}(q)p + \tau \tag{4.18b}$$

where $\tilde{Q}(q) \triangleq [\left(M^{\top}\right)^{-1}(\partial M/\partial q - Q)M^{-1}](q)$.
The state space system model can be expressed in either of the 2 forms shown below:

System Model I:

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \ddot{q} \\ -M(q)^{-1}G(q) - M(q)^{-1}\dot{q}^{\top}Q(q)\dot{q} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{4\times 2} \\ M(q)^{-1}F(x) \end{bmatrix} \tau_a \tag{4.19}$$

where $C(q,\dot{q})\dot{q} = \dot{q}^{\top}Q(q)\dot{q}$, $F(x) = [e_3, e_4]$, $\tau_a = [\tau_3; \tau_4]$ and $x = [q; \dot{q}]$

System Model II: stacking up q and p into $x \triangleq [q; p]$ allows us see that the Legendre normal form of the model takes the form of a smooth nonlinear system which is affine in the control,

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} M^{-1}(q)p \\ -G(q) + p^{\top}\tilde{Q}(q)p \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{4\times 2} \\ F(x) \end{bmatrix} \tau_a \tag{4.20}$$

where $\tilde{Q}(q) \triangleq [\left(M^{\top}\right)^{-1}(\partial M/\partial q - Q)M^{-1}](q)$, $F(x) = [e_3, e_4]$, $\tau_a = [\tau_3; \tau_4]$ and $x = [q; p]$
In the above system equations, e_k is the k- standard basis vector in $\mathbb{R}^4$ and τ_a is used instead of τ to accurately reflect the actuated control torque being a 2-dimensional vector in this 4-DOF underactuated mechanical system.

Chapter 5

Simulation Environment

With a proper model constructed for the SDIP system, it then can be implemented based on the following approach:

5.1 System Initialization with Matlab®

To prepare for a typical simulation, a Matlab script is run to perform the tasks in the following order:

1. Initialize the physical parameters for the SDIP system.
2. Symbolically derive the system functions such as $M(q)$, $C(q, \dot{q})$ and $G(q)$, etc. and convert them into Matlab functions.
3. For a given controller, initialize constant parameters and perform necessary calculations to generate the Matlab functions used by the controller.

The system functions can also be linked together to form a mathematical model in Simulink representing the SDIP system according to Eq. (4.1), with inputs $u = [\tau_3, \tau_4]^{\mathrm{T}}$ and state outputs $x \triangleq [q; \dot{q}] \in \Re^8$.

5.2 System Modelling with SimMechanics®

SimMechanics is part of the Simscape toolbox in Simulink, used to build mechanical systems. A SimMechanics model differs significantly from other Simulink models in how it

represents a machine. The mathematical model enables Simulink to simulate the machine. By contrast, a SimMechanics model represents the structure of a machine, the geometric and kinematic relationships of its component bodies and converts them into an internal, equivalent mathematical model.
Modelling mechanical systems is much easier with SimMechanics than with other Simulink models. In order to construct the SDIP system using SimMechanics, a subsystem block diagram (see Fig. 5.1) is created using components in SimMechanics, with the knowledge of the structure of the model, COM and the Centroidal Mass Moment of Inertia of each link described in the previous chapter.
A number of simulations have been run to verify the consistency between the Simulink model and the one modelled by SimMechanics. Therefore, we can conclude that SimMechanics model has indeed simulated the true dynamics of the SDIP system.
The advantages of using SimMechanics to model the mechanical system over relying on the explicit mathematical models for the nonlinear system are summarized as follows:

1. The speed of the simulation is improved significantly.

2. The positions, velocities and accelerations of various components of the system can be measured directly using the body or joint sensors in SimMechanics.

3. The joint angle range in SimMechanics is $[-180^\circ\ 180^\circ]$ while mathematical models do not respect such limits. As a result, if an angle increases past $+180^\circ$, it would automatically restart from -180°, making it easier to interpret the results.

4. The dynamics of the double inverted pendulum can be visualized using the 3-D animation viewer in SimMechanics.

As a result, the SimMechanics model is chosen for the simulation purpose.

5.3 Simulation with Simulink® and SimMechanics®

With the state feedback control law designed and implemented onto Simulink diagram, the joint angles, angular velocities and the torques can be directly viewed in real time during simulation. The kinetic energy of each link is:

$$E_{ki} = \frac{1}{2} {w_i}^\top H_i w_i + \frac{1}{2}(m_i + \bar{m}_i)|\vec{v}_i|^2 \tag{5.1}$$

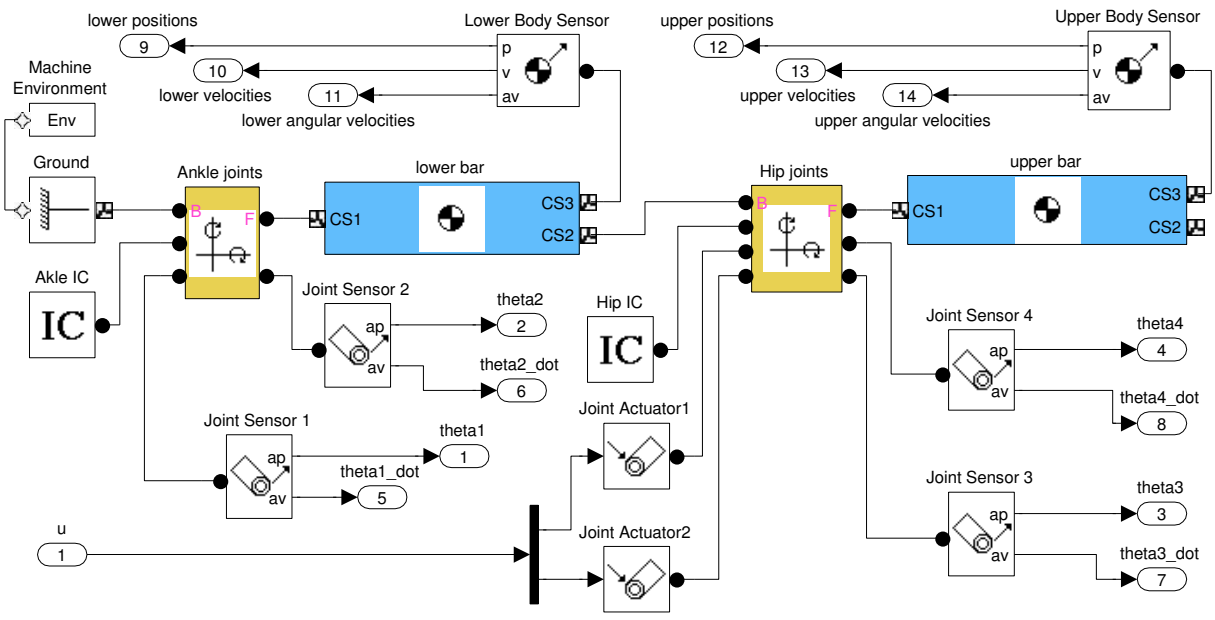

Fig. 5.1 SimMechanics block diagram of the SDIP system

for $i = 1, 2$, where H_i is the Centroidal Mass Moment of Inertia, $\vec{v}_i$ is the velocity and w_i is the angular velocity of the respective link. These velocities are obtained using body sensors attached to both upper and lower bar in the SimMechanics block diagram. Since the system modelled by SimMechanics has been sub-divided into the upper and lower parts, simply summing up the kinetic energies of the two subsystems would result in the total kinetic energy of the system: $E_k = E_{k1} + E_{k2}$.

To determine the potential energy of the system, we use the following equation:

$$E_p = mgh \tag{5.2}$$

where $h = \frac{m_1 y_1 + m_2 y_2}{m_1 + m_2}$ and $y_i, i = 1, 2$ is the y-coordinate of COM of each respective link. We can see that h is essentially the Y-coordinate of the COM of the whole SDIP system. In addition, a small Matlab program was written to calculate the Cartesian coordinates of the COM given the IC, which is used in later analyses.

Chapter 6

Linearization-based Control Algorithms

In this chapter, two linearization-based control algorithms are implemented and tested in an attempt to achieve local asymptotic stabilization close to the unstable upright equilibrium.

6.1 LQR

The initial objective is to design a suitable linear controller for the SDIP system that is able to bring the system from initial angle positions to its upright equilibrium configuration with the largest possible ROC.
The LQR is a feedback controller that provides the solution to the optimal control problem of minimizing the quadratic function representing the cost of a linear dynamic system. It is a standard result that the solution to the optimal control problem can be represented in the form of a linear state feedback law that is guaranteed to produce an asymptotically stable closed-loop system. At its core, the LQR algorithm is an automated way of finding an appropriate state-feedback controller.
The main drawback of an LQR is that since the control law relies on the linearization of the nonlinear system, it has no guarantee to work globally when implemented on the nonlinear system.
In this section, the state feedback LQR based on the linearization of System Model II - Legendre normal form of the model is designed using Eq. (4.20) and tested on the nonlinear system governed by Eq. (4.19).

6.1.1 The Infinite Horizon LQR

For a linear time invariant system described by

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

with a cost function defined as

$$J = \int_0^\infty \left(x^\top Q x + u^\top R u\right) dt \tag{6.1}$$

where Q and R represent weighting matrices and the state feedback control law that minimizes the "cost function" is a linear function of the state given by

$$u = -Kx$$

where K is given by

$$K = R^{-1}B^\top P$$

and P is found by solving the Algebraic Riccati Equation

$$A^\top P + PA - PBR^{-1}B^\top P + Q = 0$$

The solution to the Algebraic Riccati Equation can be found numerically in MATLAB using the command "lqr".

6.1.2 Linearization of the Nonlinear System

The dynamics of the spatial double inverted pendulum is described by Eq. (4.20). We denote $x \triangleq [q; p]$, and the linearized system is computed as follows:

$$\begin{aligned} A &= \left.\frac{\partial f(x)}{\partial x}\right|_{x=0} \\ &= \left.\begin{bmatrix} \mathbf{0}_{4\times4} & M^{-1}(q) \\ -\nabla G(q) & \mathbf{0}_{4\times4} \end{bmatrix}\right|_{q=\mathbf{0},p=\mathbf{0}} \end{aligned} \tag{6.2}$$

where the matrix $\nabla G(q)$ is the gradient of $G(q)$, which is also the Hessian of the potential energy: $[-\nabla G(q)]_{i,j} = -\dfrac{\partial^2 V(q)}{\partial q_i \partial q_j}$.
The matrix B is given by

$$B = \left[\begin{array}{c} \mathbf{0}_{4\times 2} \\ M(q)^{-1}F(q) \end{array} \right]\Bigg|_{q=\mathbf{0}} \tag{6.3}$$

and the control vector is given by

$$u = \tau_a = [\tau_3, \tau_4]^\top \tag{6.4}$$

Using the parameters $m_1 = m_2 = 0$ kg, $\bar{m}_1 = \bar{m}_2 = 1.0$ kg, $l_1 = l_2 = 1.0$ m, $r_1 = r_2 = 0.1$ m, $g = 9.81\ m/s^2$, the corresponding functions $M(q)$ and $G(q)$ are then obtained. For the purpose of model simplification and the speed of simulation, the values of leg and body masses m_1 and m_2 are set to zero since they are much smaller than the hip and head masses $\bar{m}_1$ and $\bar{m}_2$, respectively. The matrix A and B are given by:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -2 \\ 0 & 0 & 0 & 0 & -2 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & 5 \\ 29.43 & 0 & 9.81 & 0 & 0 & 0 & 0 & 0 \\ 0 & 29.43 & 0 & 9.81 & 0 & 0 & 0 & 0 \\ 9.81 & 0 & 9.81 & 0 & 0 & 0 & 0 & 0 \\ 0 & 9.81 & 0 & 9.81 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The Kalman Controllability Test has been performed for the (A,B) pair in Matlab with the command ctrb(A,B). Since the controllability matrix $C = [B\ AB\ A^2B\ ...\ A^7B]$ is of full rank 8, the system is controllable, thus the design of a linear controller is feasible.

6.1.3 The Design of the LQR Weighting Matrices

A suitable linear proportional controller was designed based on the linearized state space model of the system. The LQR design has been widely used in many similar applications, especially on the local stabilization of inverted pendulums. Due to its popularity and superiority over other linear controller design approaches such as pole placement, it has

been chosen to achieve local stabilization for this system. The goal of such design is to find the best Q and R pair, to make the ROC of the controller as applied to the nonlinear system largest possible. To achieve that, the linearized model of the system derived from Eq. (4.20) was designed for first.
A number of simulations based on the linear state space model has been performed in Simulink with a large set of different initial conditions. As expected, none of the simulations failed to stabilize because the model of the "system" is linear and does not represent the true dynamics of the SDIP system. Nevertheless, by choosing a proper positive definite diagonal matrices $Q = diag\,(q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$ and $R = diag(r_1, r_2)$ the following generalized observations have been made:

1. Decreasing the values in the R matrix resulted in an increase of the gain K, stronger control torques, and less overall %OS.

2. Increasing the values in the Q matrix resulted in less overall %OS and longer settling time.

3. The %OS in angular positions could be kept as low as possible by setting the penalty weights for the angular velocities (last 4 diagonal entries of Q, i.e. $q_5, ..., q_8$) much larger than those for the angular positions (first 4 entries of Q).

4. The actuated angles, θ_3 and θ_4, always have much higher overshoot/undershoot in terms of both positions and velocities than the unactuated ones (θ_1 and θ_2). This can be explained by noting that the controller torques are acting directly on the hip, creating much larger ranges of movements than the corresponding ones at the ankle.

The Q and R matrices were selected by way of multiple simulations with the above observations in mind. As a result, the Q matrix has been designed to possess the following characteristics:

1. $q_k = q_{k+1}\,\forall k\,odd$, for the simplicity of the design.

2. $q_{3,4} \gg q_{1,2}$ and $q_{7,8} \gg q_{5,6}$, the entries (penalty weights) controlling the hip need to be much larger than the corresponding entries controlling the ankle joints, for both positions and velocities to reinforce the phenomenon described in observation No.4.

3. $q_{5,6} > q_{1,2}$ and $q_{7,8} \gg q_{3,4}$, to reinforce observation No.3.

6.1.4 LQR Control of the Nonlinear System

Eventually, the linearized model should be replaced by the nonlinear one to simulate the true dynamics of the SDIP system.
The LQR controller has been designed using the cost function from Eq. (6.1). where

$$Q = diag\,(25, 25, 500, 500, 50, 50, 3000, 3000)$$

$$R = diag\,(100, 100)$$

Solving the Algebraic Riccati Equation using MATLAB gives us the following result:

$$u = -Kx \tag{6.5}$$

where

$$K = \begin{bmatrix} -440.5 & 0 & -147.2 & 0 & -43.4 & 0 & 17.9 & 0 \\ 0 & -440.5 & 0 & -147.2 & 0 & -43.4 & 0 & 17.9 \end{bmatrix} \tag{6.6}$$

6.1.5 LQR Design Approach

The whole LQR design procedure can be viewed as consisting of following 3 stages:

1. Completion of the linearization and the computations of state space system matrices.
2. Computation of the optimal gain matrix K provided with Q and R matrices, along with the linearized system matrices generated in the first stage.
3. Repetitive simulations are run to see how the system responds to varying ICs.

With such a design approach, each stage depends on the outputs from the preceding stage, and previous stage(s) do not have to be repeated if the inputs are not changed. (See Fig. 6.1)

6.1.6 Simulation Results

The motion of the SDIP is simulated using the controller K described previously. The simulations are performed in fixed steps of 10^{-3} using ode4 (Runge Kutta).
The results are displayed using two different initial conditions consisting of plots showing

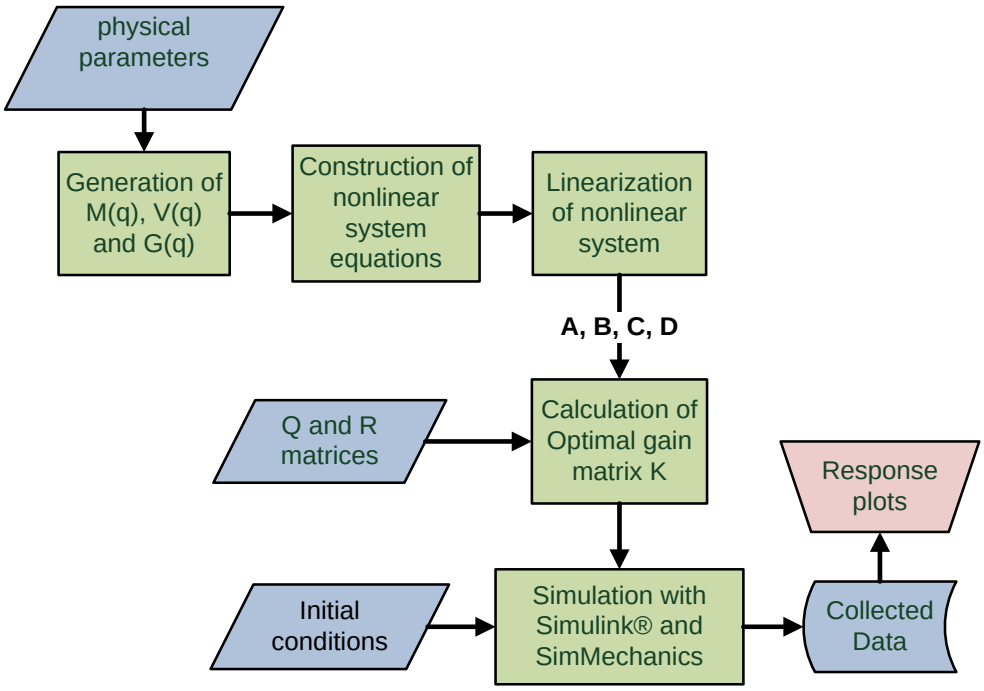

Fig. 6.1 The LQR controller design approach

joint angles θ, their velocities, kinetic energy, potential energy as well as input torques to the system, with simulation horizon of 5 seconds. The initial angular velocities are set to zero in both examples, because it is often the case in reality that a mechanical system start from zero velocity. In addition, the relative elevation of centre of gravity of the system can be directly viewed from the potential energy since they are proportional according to equation $E_p = mgh$.

Example 1 The IC is $q_0 = [0°, 0°, 7°, 0°]^\top$, $\dot{q}_0 = \mathbf{0}$. The system stabilizes to its equilibrium configuration in about 4.5 seconds, as shown in Fig. 6.2 to Fig. 6.5 next.

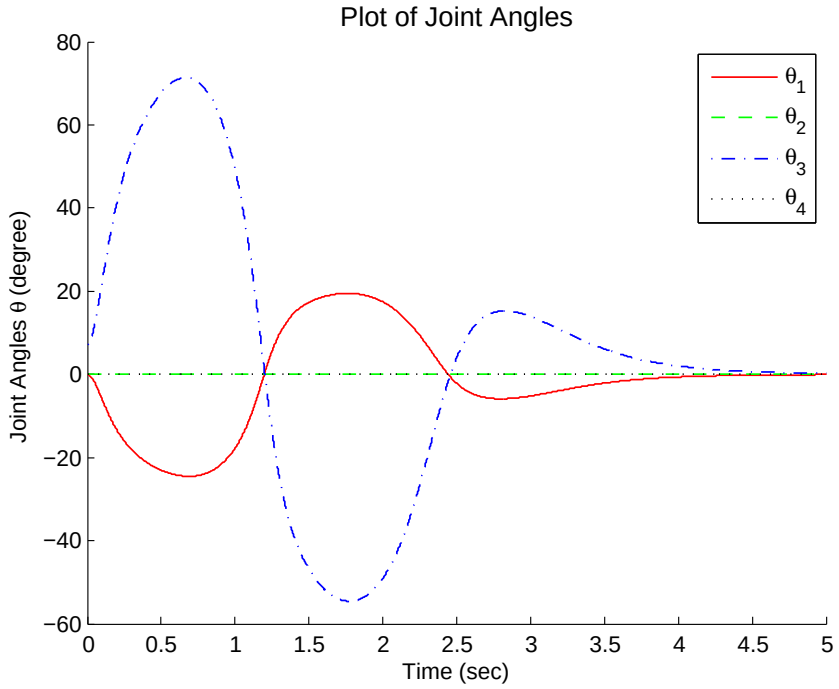

Fig. 6.2 Example 1. Change of the joint angles w.r.t. time

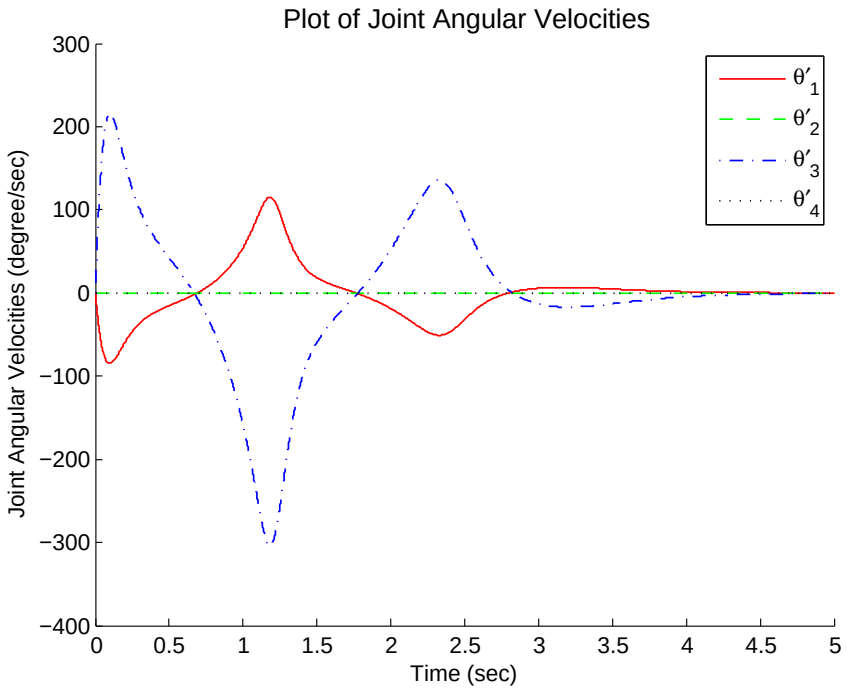

Fig. 6.3 Example 1. Change of joint angle rates w.r.t. time

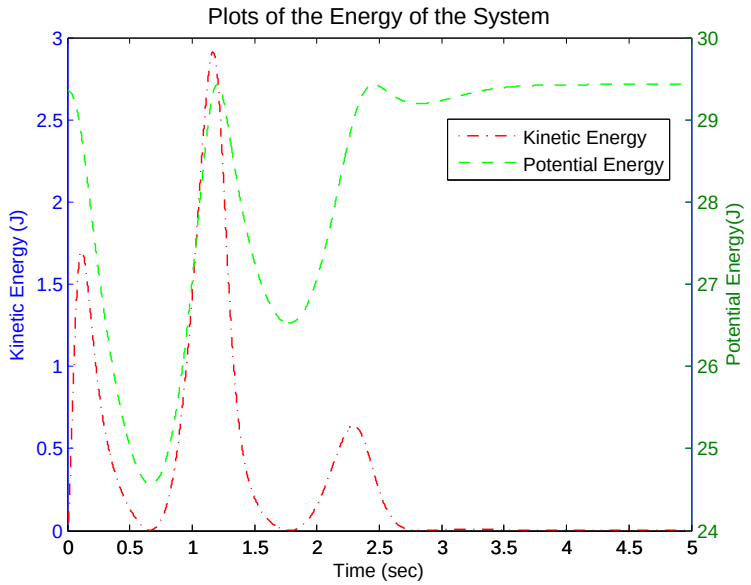

Fig. 6.4 Example 1. Change of kinetic and potential energies w.r.t. time

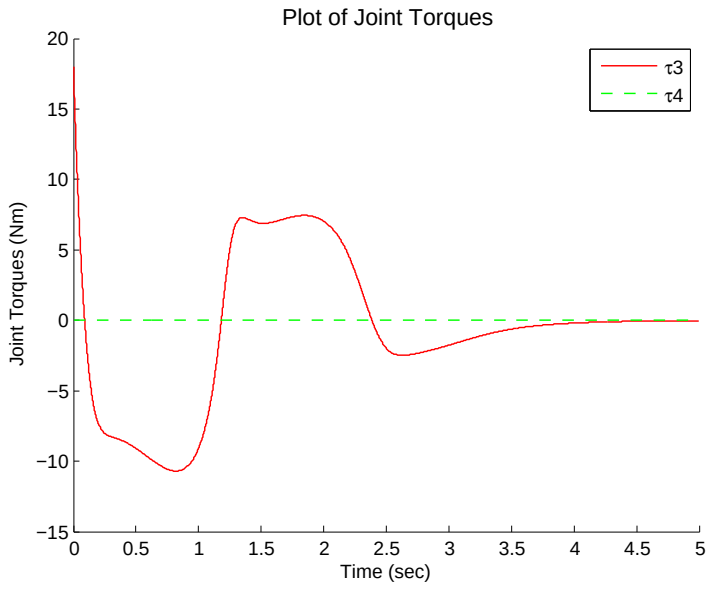

Fig. 6.5 Example 1. Change of Applied Torques from LQR

As we can see from the previous diagrams, the system operates in a 2-D (Y-Z plane) because only one angle: θ_3 is perturbed initially.

Example 2 The IC is $q_0 = [10°, -10°, -25°, 25°]^\top$, $\dot{q}_0 = \mathbf{0}$. The system stabilizes to its equilibrium configuration in about 3 seconds, as shown in Fig. 6.6 to Fig. 6.9 next.

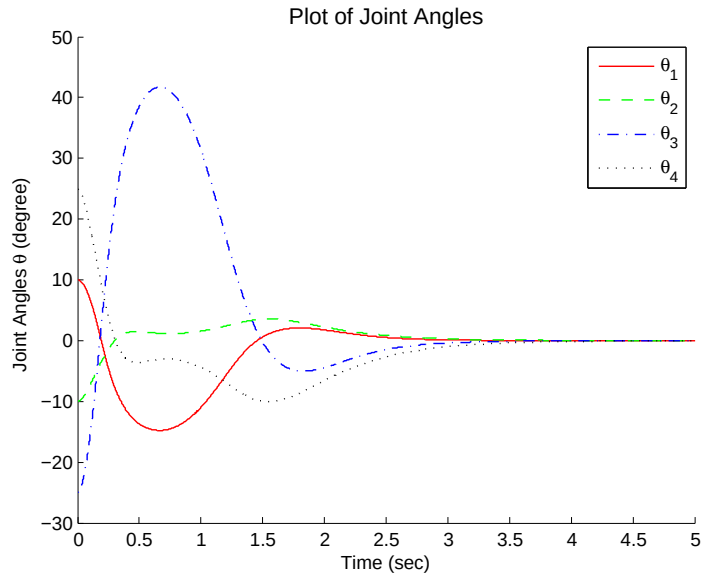

Fig. 6.6 Example 2. Change of the joint angles w.r.t. time

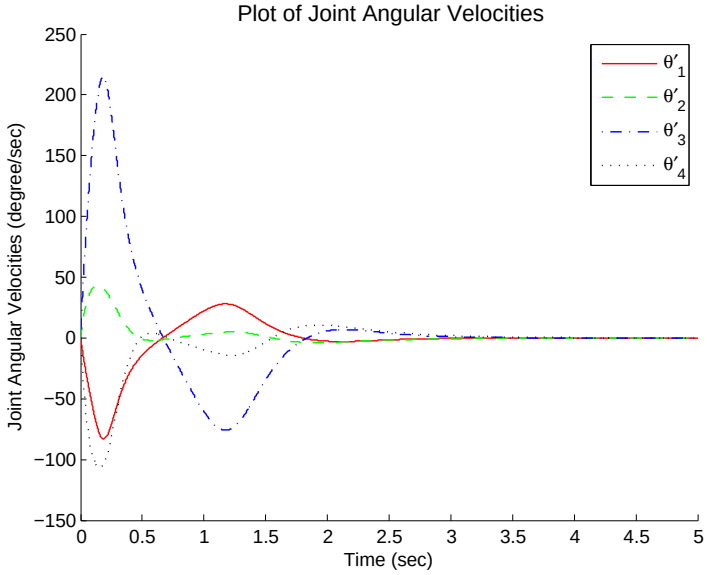

Fig. 6.7 Example 2. Change of joint angle rates w.r.t. time

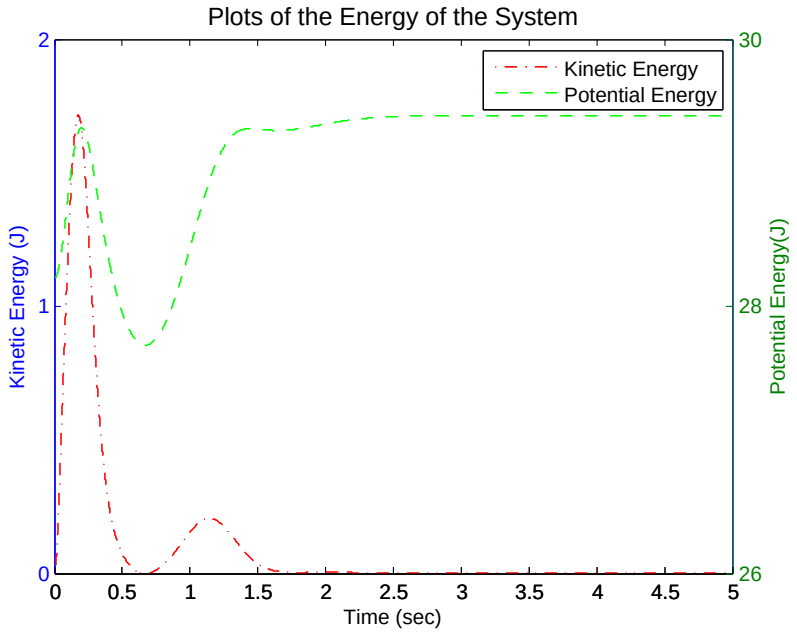

Fig. 6.8 Example 2. Change of kinetic and potential energies w.r.t. time

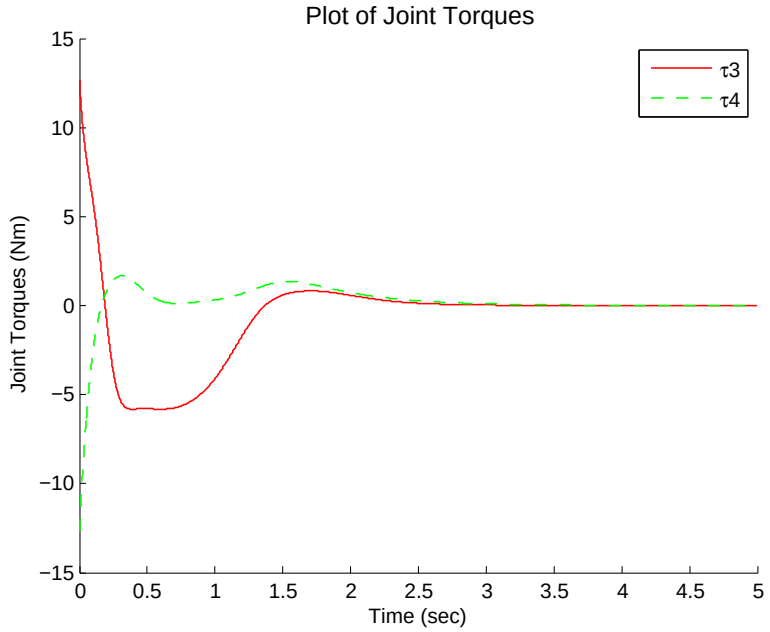

Fig. 6.9 Example 2. Change of Applied Torques from LQR

If Q is kept the same and $R = diag(0.01, 0.01)$, then

$$K_{high} \approx 10^4 \times \begin{bmatrix} -1.62 & 0 & -0.56 & 0 & -0.16 & 0 & 0.06 & 0 \\ 0 & -1.62 & 0 & -0.56 & 0 & -0.16 & 0 & 0.06 \end{bmatrix}$$

As we can see, the corresponding gain is huge and it may not be practically realized in the presence of constraints on the actuation signals. Nevertheless, it has been shown to extend the ROC significantly compared with using $R = diag(100, 100)$. Therefore, it can be concluded that the smaller the R matrix becomes, the harder the controller tries to push the system into equilibrium, resulting in a larger ROC. Of course, that does not mean R matrix should be made arbitrarily small. It has been tested that making its values too small (below 10^{-3}) does not improve the ROC and could even worsen the performance of the controlled system.

After extensive simulations using the gain matrix K_{high}, a selection of results obtained by varying IC (all initial velocities are set to 0) is summarized Table 6.1. The first 4 rows list the individual upper bounds of θ_1 to θ_4 (the largest deviation of a respective angle by setting all other 3 angles to zero). We can see that system behaves in an asymmetric manner, i.e, θ_1 and θ_3 have a larger ROC than θ_2 and θ_4, respectively. Also, It could be inferred that the ROC is somewhat related to R_{com}. Physically this means that it is easier to stabilize the system when the COM is closer to its equilibrium. However, by comparing between the results in the table, we can also see that R_{com} is only a rough indicator on the stabilizability of the system. These results have led us to perform a an extensive suite of simulations to determine ROC in Y-Z plane by varying the θ_1 from $-34°$ to $34°$ and θ_3 from $-97°$ to $97°$, respectively, in steps of $1°$. The results collected in Fig. 6.10 are quite promising, because the ROC is not limited to the IC with COM quite close to the upright equilibrium, but as long as the θ_1 is no larger in magnitude than $\pm 34°$, there is always a corresponding range of initial angle positions θ_3 can belong to in order for the system to be stabilized. For example, the IC of $[30°, 0°, -80°, 0°]$ with a relatively large R_{com} of 33.38 cm is still in the ROC of the LQR controller.

q_0	R_{com} (cm)	in ROC
$[\pm 3.8°, 0°, 0°, 0°]^\top$	9.9466	Yes
$[0°, \pm 2.68°, 0°, 0°]^\top$	7.016	Yes
$[0°, 0°, \pm 10°, 0°]^\top$	8.7156	Yes
$[0°, 0°, 0°, \pm 7°]^\top$	6.105	Yes
$[\pm 4°, 0°, 0°, 0°]^\top$	10.4698	No
$[0°, 0°, \pm 10.5°, 0°]^\top$	9.1502	No
$[0.5°, 0.5°, 2°, 1.8°]^\top$	4.197	Yes
$[10°, -10°, -25°, 25°]^\top$	9.350	Yes
$[10°, -10°, -30°, 30°]^\top$	9.13	No
$[20°, 0°, -50°, 0°]^\top$	15.707	Yes

R_{com} is the COM distance between initial and equilibrium configurations

Table 6.1 Simulaiton results of LQR for different initial conditions

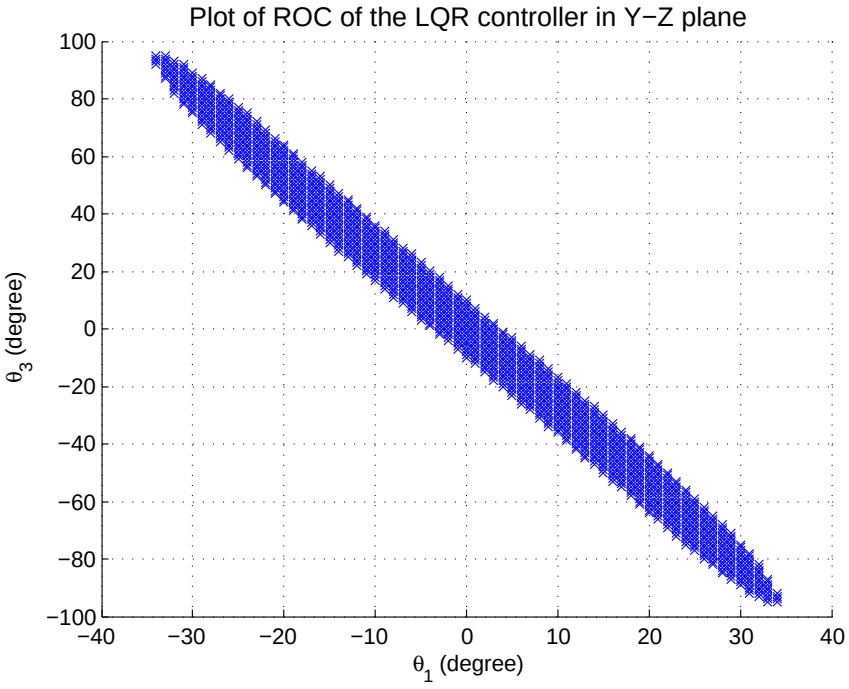

Fig. 6.10 ROC of the LQR operating in Y-Z plane

6.1.7 The Advantage of System Model II

The same LQR design approach has also been followed with System Model I based on the linearization of Eq. (4.19). After extensive simulations by varying ICs, it has been observed that System Model II in Legendre normal form offers a much broader ROC than its untransformed counterpart.
The advantage of the LQR controller based on System Model II over System Model I can be inferred from the obvious fact that the former system has to constantly calculate the M matrix and update the state variable $p = M(q)\dot{q}$ in real time, since p depends non-linearly on both q and $\dot{q}$. As a result, the overall control system has become more sophisticated.

6.1.8 Relationships between physical parameters and ROC

The previous examples all used the same set of physical parameters and the (A,B) pair described earlier. However, one might wonder what would happen to the ROC if those physical parameters are changed. A number of simulations have been performed and the following relationships have been observed:

- If the lengths of the pendulums l_1 and l_2, as well as their masses m_1 and m_2 are kept constant, increasing the ratio between hip mass $\bar{m}_1$ and head mass $\bar{m}_2$ would bring the COM closer to the hip, making R_{com} smaller for all initial configurations, resulting in larger overall ROC. For example, when $\bar{m}_1$ is increased to 2.0 kg while $\bar{m}_2$ remains 1.0 kg, the system with the IC of $[\pm 4°, 0°, 0°, 0°]$ could be stabilized while it was not within the ROC given the original parameters as shown in Table (6.1).

- If the lengths of the pendulums l_1 and l_2, as well as head and hip masses $\bar{m}_1$ and $\bar{m}_2$ are kept constant, increasing the masses of both pendulums by the same amount and keeping $m_1 = m_2$ will enlarge the overall ROC. For example, when $m_1 = m_2 = 1.0$ kg, the system with IC of $[10°, -10°, -30°, 30°]$ could be stabilized while it was not within the ROC given the original parameters when m_1 and m_2 were assumed to be negligible, as shown in Table (6.1). It can be explained by nothing that the COM is brought closer to the hip largely due the increase in the mass of the lower pendulum m_1, while the presence of m_2 makes sure the COM is still above the hip.

- Increasing the length of upper pendulum could enlarge the overall ROC as it would enlarge the torque acting on the hip. For example, if l_2 is increased to 1.5m, the IC of $[\pm 4°, 0°, 0°, 0°]$ could be stabilized.

- The radii of the pendulums r_1 and r_2 have no effects on the ROC of the system.

6.2 Quasi-linear STA Sliding Mode Controller

A novel Lyapunov based design of VGSTA [8] has been recently proposed ensuring global finite time convergence to the desired sliding surface for LTI systems with absolutely continuous matched uncertainties/disturbances bounded together with their gradients by known functions.

6.2.1 Initial System Transformation

The approach is explained below and it applies principally to an LTI system of type:

$$\dot{x} = Ax + B(u + d(x)) \tag{6.7}$$

where $x \in \Re^n$ is a state vector, and $u \in \Re^m$ is a continuous control law, $d(x)$ represents matched perturbation in the system (6.7). Therefore, the system is quasi-linear, as $d(x)$ is the only nonlinearity in Eq. (6.7). The method necessitates the following additional assumptions:
A1: rank$(B) = m$.
A2: the pair (A, B) is controllable,
A3: the function d together with its gradient is bounded by known continuous functions.
Under (A1) and (A2), we can apply a state transformation T to transform the system into the regular form, as follows:[1]

$$\tilde{x} = Tx, \ where\ T = \begin{bmatrix} B^{\perp} \\ B^{+} \end{bmatrix}, \ B^{+} = (B^T B)^{-1} B^T, \ B^{+}B = I_m, \ B^{\perp}B = 0$$

The system in the regular form is hence obtained as:[2]

$$\begin{aligned} \dot{\tilde{x}}_1 &= A_{11}\tilde{x}_1 + A_{12}\tilde{x}_2 \\ \dot{\tilde{x}}_2 &= A_{21}\tilde{x}_1 + A_{22}\tilde{x}_2 + u + \tilde{d}\,(\tilde{x}) \end{aligned} \tag{6.8}$$

[1] B^+ is the Moore-Penrose pseudoinverse of B
[2] Note that A_{ij}'s, $i = 1, 2$ and $j = 1, 2$ are submatrices of TAT^{-1} due to the transformation.

where $x_1 \in \Re^{n-m}$ and $x_2 \in \Re^m$.
Let us enforce the sliding mode on the surface:

$$s = \tilde{x}_2 - K\tilde{x}_1 \tag{6.9}$$

If the pair (A_{11}, A_{12}) is controllable, then the matrix K can be designed using either pole placement or LQR to prescribe the required performance of the reduced-order system on the surface:

$$\dot{\tilde{x}}_1 = (A_{11} + A_{12}K)\tilde{x}_1 \tag{6.10}$$

From (6.9), we get $\tilde{x}_2 = s + K\tilde{x}_1$, so system (6.8) becomes:

$$\begin{aligned} \dot{\tilde{x}}_1 &= (A_{11} + A_{12}K)\tilde{x}_1 + A_{12}s \\ \dot{s} &= A_{21}\tilde{x}_1 + A_{22}\tilde{x}_2 + u - K(A_{11}\tilde{x}_1 + A_{12}\tilde{x}_2) + \tilde{d}(\tilde{x}) \\ &= (A_{21} + A_{22}K - K(A_{11} + A_{12}K))\tilde{x}_1 + (A_{22} - KA_{12})s + u + \tilde{d}(\tilde{x}) \end{aligned} \tag{6.11}$$

Suppose $d(x) = 0$, then we get

$$\dot{s} = (A_{21} + A_{22}K - K(A_{11} + A_{12}K))\tilde{x}_1 + (A_{22} - KA_{12})s + u \tag{6.12}$$

Thus on the surface (6.9), $s = \dot{s} = 0$ and consequently

$$u_{eq} = -(A_{21} + A_{22}K - K(A_{11} + A_{12}K))\tilde{x}_1 \tag{6.13}$$

Let us choose the controller in the following form

$$\begin{aligned} u &= -(A_{21} + A_{22}K - K(A_{11} + A_{12}K))\tilde{x}_1 - (A_{22} - KA_{12})s + v \\ &= u_{eq} - (A_{22} - KA_{12})s + v \end{aligned} \tag{6.14}$$

Then the controlled system (6.11) takes the form

$$\begin{aligned} \dot{\tilde{x}}_1 &= (A_{11} + A_{12}K)\tilde{x}_1 + A_{12}s \\ \dot{s} &= v + \tilde{d}(\tilde{x}) \end{aligned} \tag{6.15}$$

6.2.2 Sliding Mode Control Algorithms

As we can see from (6.15), the system is of relative degree 1 w.r.t. the sliding variable s. According to [3], v could either be designed as a discontinuous control law, or a continuous state function with $\ddot{s}(x)$ being discontinuous. With the latter approach, a SOSM controller can be designed to avoid chattering and make $s = \dot{s} = 0$ in finite time, while the $\ddot{s}$ is discontinuous.
Therefore, we could use the following three SMC algorithms to control the system, with $v \in \Re^m$:

1. First Order SM controller with $M_0 = const. \in \Re$

$$v = -M_0 sign(s) \tag{6.16}$$

2. STA with constant gains k_1 and k_2.
   The components of v_i of the control vector v, for $i = 1, ..., m$ are defined as:

$$v_i = -k_1 |s_i|^{\frac{1}{2}} sign(s_i) - k_2 \int_0^t sign(s_i) dt \tag{6.17}$$

   For simplicity, we use the same set of gains k_1 and k_2 for all components v_i of v.

3. VGSTA with variable gains $k_1(\tilde{x})$ and $k_2(\tilde{x})$. In this approach:

$$v_i = -k_{1i}(\tilde{x})\phi_{1i}(s_i) - \int_0^t k_{2i}(\tilde{x})\phi_{2i}(s_i) dt, \ i = 1, ..., m \tag{6.18}$$

   where

$$\phi_{1i}(s_i) = |s_i|^{\frac{1}{2}} sign(s_i) + k_3 s_i \ k_3 > 0$$
$$\phi_{2i}(s_i) = \tfrac{1}{2} sign(s_i) + \tfrac{3}{2} k_3 |s_i|^{\frac{1}{2}} sign(s_i) + {k_3}^2 s_i$$

For the VGSTA design, we note that when $k_3 = 0$ and k_1 and k_2 are constant, a standard STA is recovered. The gain k_3 is used to deal with perturbations growing linearly in s, and the variable gains $k_1(\tilde{x})$ and $k_2(\tilde{x})$ adaptively adjust gains for the control system, making it possible to keep the system stay on the sliding surface without being affected by perturbations bounded by some known functions.

In order to properly design $k_1(\tilde{x})$ and $k_2(\tilde{x})$, we now note that the perturbation can always be written as:

$$\tilde{d}(\tilde{x}) = \tilde{d}(\tilde{x}_1, \tilde{x}_2) = \tilde{d}(\tilde{x}_1, s + K\tilde{x}_1) \tag{6.19}$$

$$= [\underbrace{\tilde{d}(\tilde{x}_1, s + K\tilde{x}_1) - \tilde{d}(\tilde{x}_1, K\tilde{x}_1)}_{g_1(\tilde{x}_1, s)}] + \underbrace{\tilde{d}(\tilde{x}_1, K\tilde{x}_1)}_{g_2(\tilde{x}_1)} \tag{6.20}$$

For $i = 1, ..., m$, the perturbation $\tilde{d}(\tilde{x})$ is bounded by

$$\begin{aligned} |g_{1i}(\tilde{x}_1, s)| &\leq \rho_{1i}(\tilde{x})\phi_{1i}(s_i) \\ |\tfrac{d}{dt} g_{2i}(\tilde{x}_1)| &\leq \rho_{2i}(\tilde{x})\phi_{2i}(s_i) \end{aligned} \tag{6.21}$$

where $\phi_{1i}(s_i)$ and $\phi_{2i}(s_i)$ are continuous functions.
System (6.15) with VGSTA (6.18) can now be written as

$$\begin{aligned} \dot{\tilde{x}}_1 &= (A_{11} + A_{12}K)\tilde{x}_1 + A_{12}s \\ \dot{s}_i &= -k_{1i}(\tilde{x})\phi_{1i}(s_i) + z_i + g_{1i}(\tilde{x}_1, s) \\ \dot{z}_i &= -k_{2i}(\tilde{x})\phi_{1i}(s_i) + \tfrac{d}{dt} g_{2i}(\tilde{x}_1) \end{aligned} \tag{6.22}$$

According to [8], $s = 0$ can be reached in finite time if the variable gains are selected as

$$\begin{aligned} k_{1i}(\tilde{x}) &= \delta + \tfrac{1}{\beta}\left\{\tfrac{1}{4\epsilon}[2\epsilon\rho_{1i} + \rho_{2i}]^2 + 2\epsilon\rho_{2i} + \epsilon + [2\epsilon + \rho_{1i}](\beta + 4\epsilon^2)\right\} \\ k_{2i}(\tilde{x}) &= \beta + 4\epsilon^2 + 2\epsilon k_{1i}(\tilde{x}) \end{aligned} \tag{6.23}$$

where $\beta > 0$, $\epsilon > 0$, $\delta > 0$ are positive constants. The complete proof is described in [8].

6.2.3 SDIP Model Representation to fit the SM approach

In our nonlinear system in System Model II from Eq. (4.20), since B is a constant, it may be expressed as:

$$\begin{aligned} \dot{x} &= f(x) + Bu, \ or \\ \dot{x} &= Ax + Bu + e\,(x) \end{aligned} \tag{6.24}$$

where $x \in \Re^8$, $u = \tau_a \in \Re^2$, $B = \begin{bmatrix} 0_{6\times 2} \\ I_2 \end{bmatrix}$ and $e(x)$ is the linearization error, therefore

$$e(x) = f(x) - Ax = \begin{bmatrix} M^{-1}(q)\, p \\ -G(q) + p^T \tilde{Q}(q)\, p \end{bmatrix} - Ax \tag{6.25}$$

Since $p = M(q)\,\dot{q}$, we can see that $M^{-1}(q)\, p = \dot{q}$
After the transformation and using $B^+ = [\mathbf{0}_{2\times 6}\ I_2]$ and $B^\perp = [I_6\ \mathbf{0}_{6\times 2}]$, we get

$$\begin{aligned} \dot{x}_1 &= A_{11}x_1 + A_{12}x_2 + e_1(x) \\ \dot{x}_2 &= A_{21}x_1 + A_{22}x_2 + u + e_2(x) \end{aligned} \tag{6.26}$$

so $x = [x_1; x_2]$, $B^+ e(x) = e_2(x)$ and $B^\perp e(x) = e_1(x)$ with $e(x) = [e_1(x); e_2(x)]$.

We can see that such transformation has not altered the states of the system at all. It only served to separate the system into its upper and lower parts consisting of the top 6 rows and bottom 2 rows, respectively, since the system in Legendre normal form can be considered to be already in the regular form after truncating top 6 rows of $e(x)$, i.e, $e_1(x)$. Although $e_1(x)$ has not been taken care of in the above regular form, as the perturbation must be in the same dimension as the torque input, we have at least been able to consider part of the error in our quasi-linear model. We can now treat $e_2(x)$ as $\tilde{d}(\tilde{x})$ in the (6.8). Thus by tracking the error of $e_2(x)$ in real time, we are able to calculate the variable gains of control the system on-line according to procedure described in Eq. (6.21) and Eq. (6.23), and that is the intuitive reasoning why VGSTA is superior to standard STA.

For our system, the sliding surface should be chosen to be of the same dimension as $u \in \Re^2$

$$s = x_2 - Kx_1 \tag{6.27}$$

The total control law and altered system takes the same form as in (6.14) and (6.15), respectively with $\tilde{x}$ and $d(x)$ replaced with x and $e_2(x)$, respectively.

6.2.4 Simulation Results for the disturbance-free system

Using the same (A,B) pair obtained earlier for LQR controller, K was designed using LQR method since (A_{11}, A_{12}) is controllable. With $Q = diag(1, 1, 1, 1, 4, 4)$ and $R = diag(5, 5)$, resulting in the gain matrix:

$$K = \begin{bmatrix} 31.95 & 0 & 11.12 & 0 & 3.21 & 0 \\ 0 & 31.95 & 0 & 11.12 & 0 & 3.21 \end{bmatrix} \tag{6.28}$$

Example 3 In this example, we use the First Order SM controller, with $M_0 = 10$. The IC is $q_0 = [10°, -10°, -25°, 25°]^\top$, $\dot{q}_0 = \mathbf{0}$.

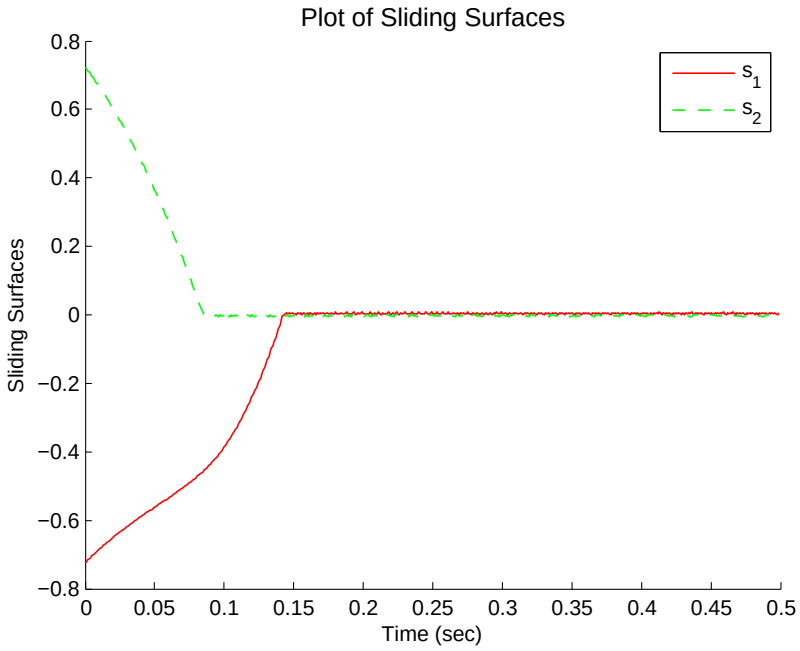

Fig. 6.11 Example 3. Change of the sliding surfaces w.r.t. time

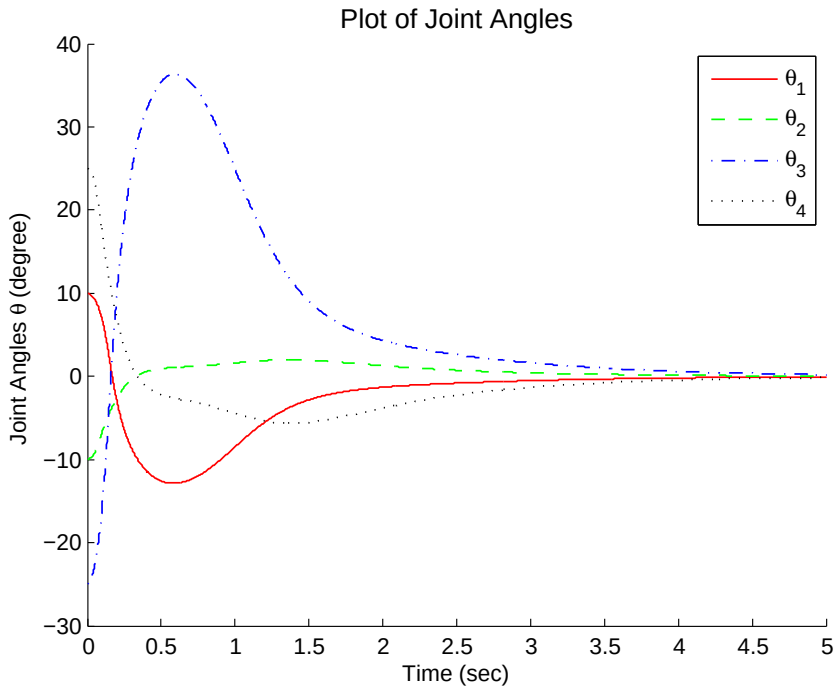

Fig. 6.12 Example 3. Change of the joint angles w.r.t. time

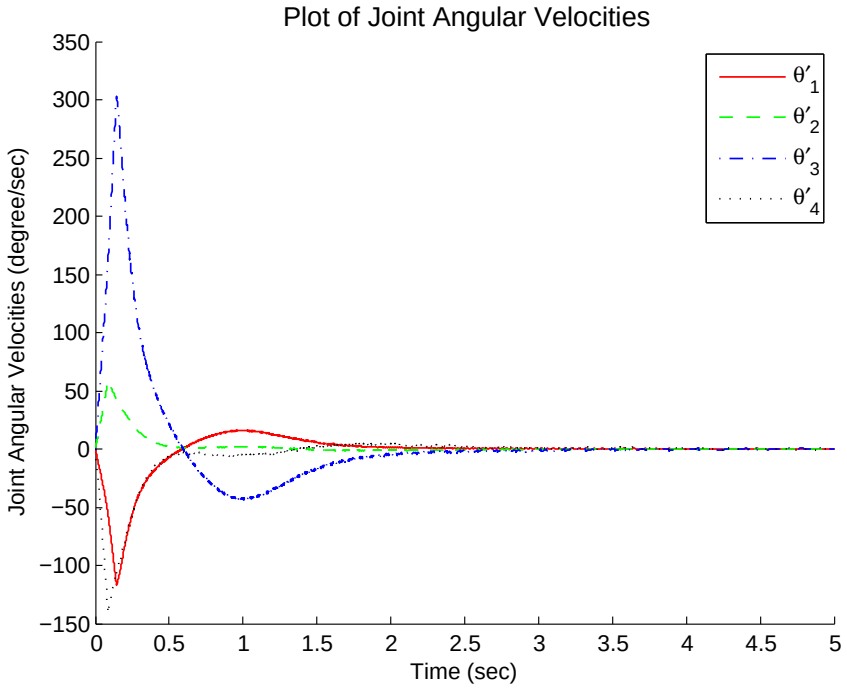

Fig. 6.13 Example 3. Change of joint angle rates w.r.t. time

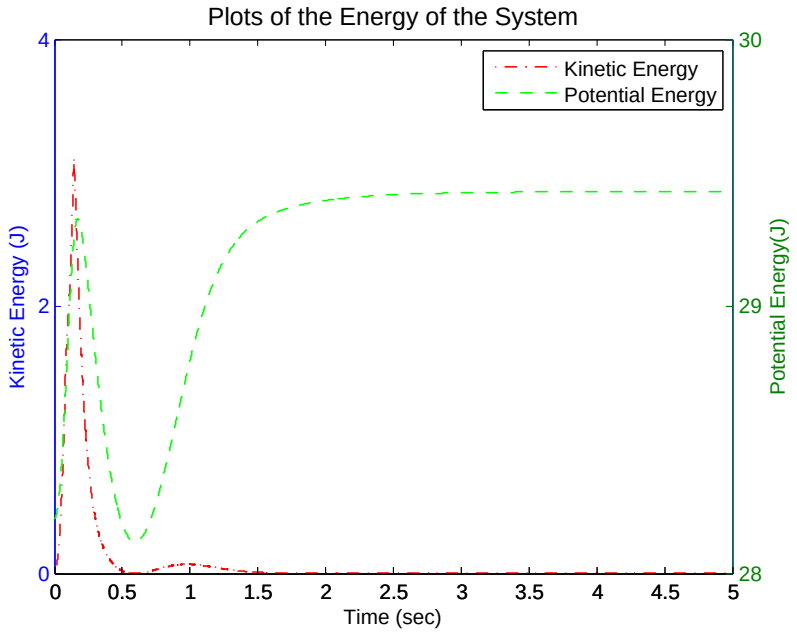

Fig. 6.14 Example 3. Change of kinetic and potential energies w.r.t. time

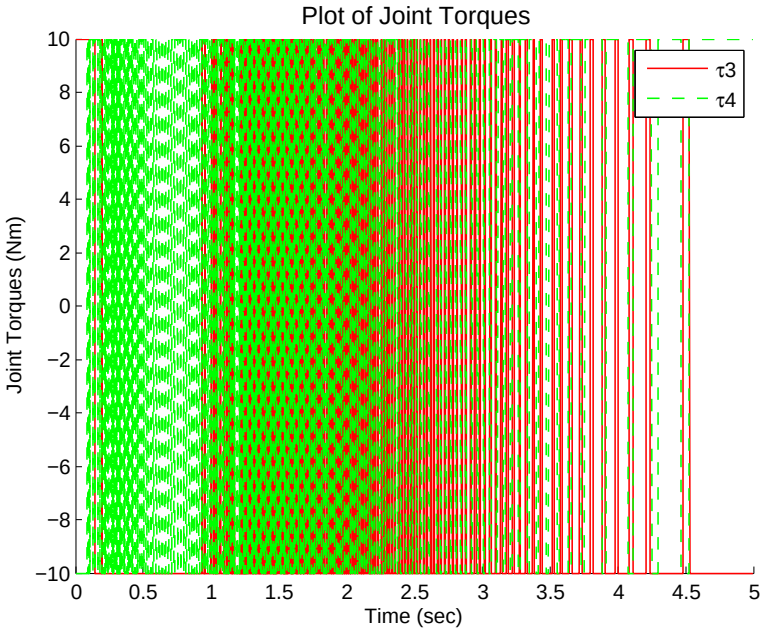

Fig. 6.15 Example 3. Change of Applied Torques

Example 4 In this example, we use the standard STA, with $k_1 = 20$, $k_2 = 15$. The IC is $q_0 = [10°, -10°, -25°, 25°]^\top$, $\dot{q}_0 = \mathbf{0}$.

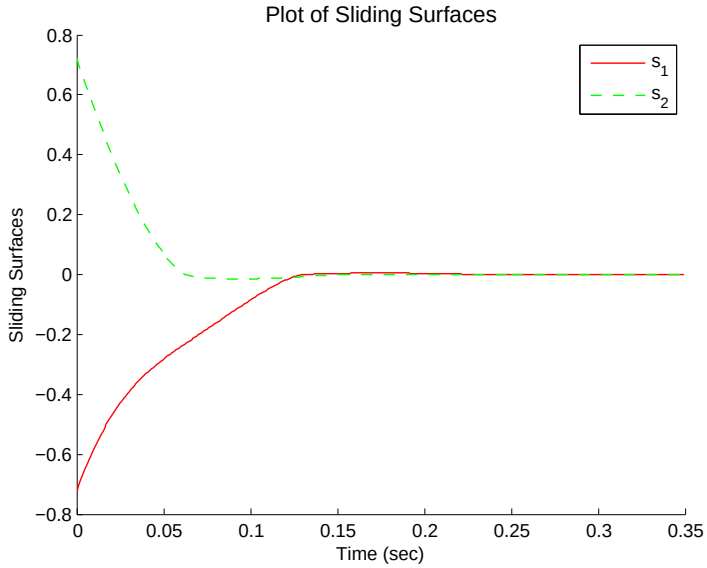

Fig. 6.16 Example 4. Change of the sliding surfaces w.r.t. time

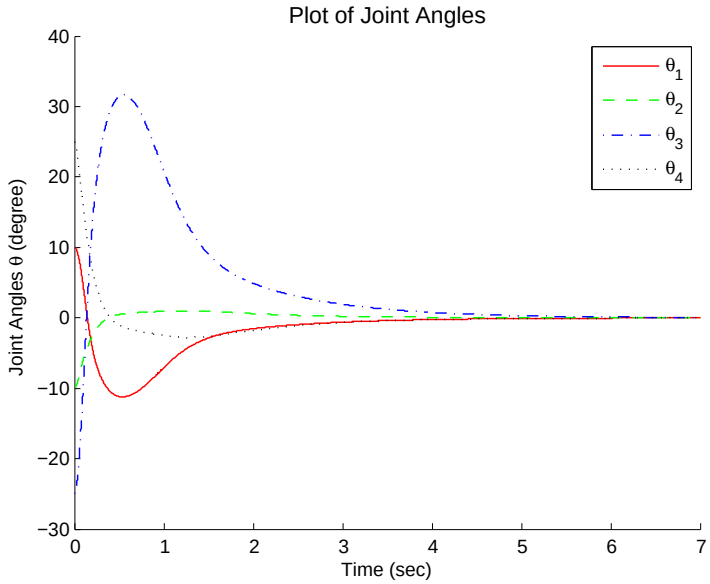

Fig. 6.17 Example 4. Change of the joint angles w.r.t. time

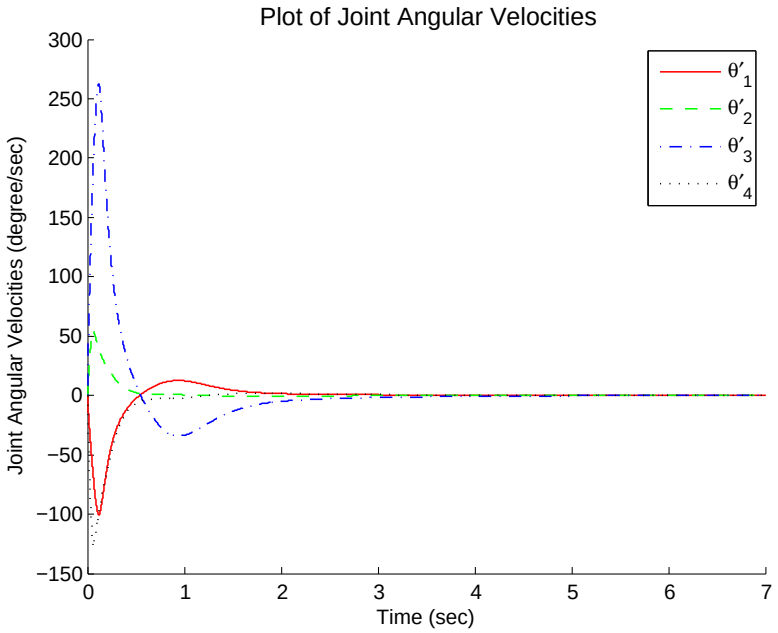

Fig. 6.18 Example 4. Change of joint angle rates w.r.t. time

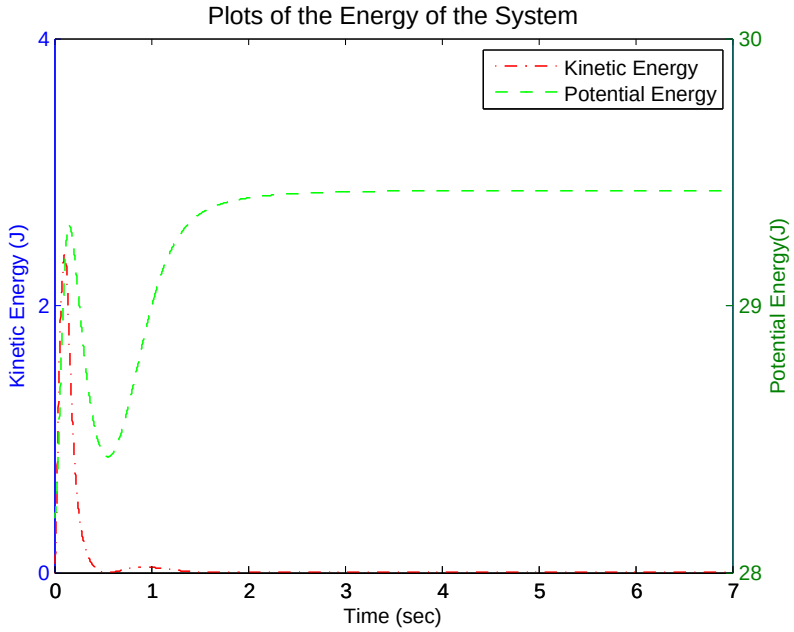

Fig. 6.19 Example 4. Change of kinetic and potential energies w.r.t. time

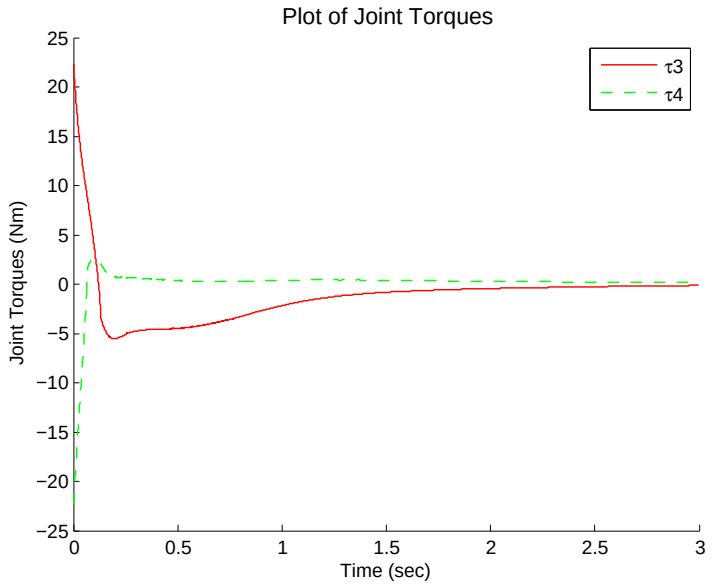

Fig. 6.20 Example 4. Change of Applied Torques

Example 5 In this example, we use VGSTA, with $k_3 = 20$, $\delta = 0$, $\beta = 10$ and $\epsilon = 0.5$. The IC is again $q_0 = [10°, -10°, -25°, 25°]^\top$, $\dot{q}_0 = \mathbf{0}$.

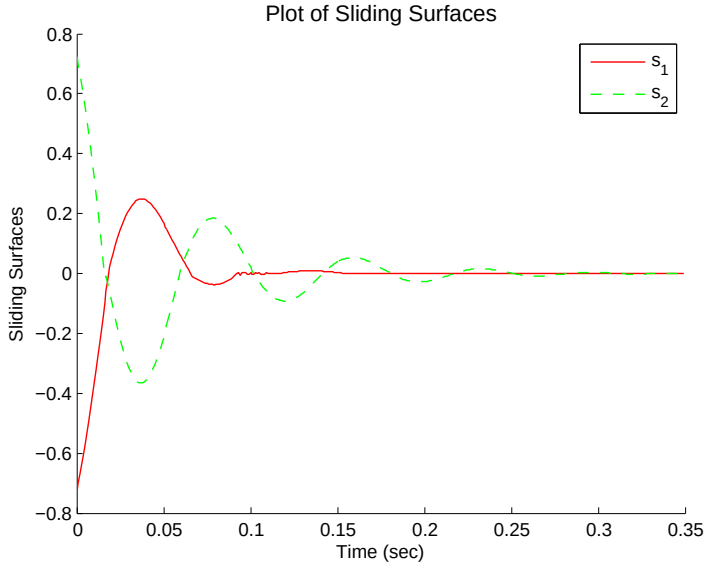

Fig. 6.21 Example 5. Change of the sliding surfaces w.r.t. time

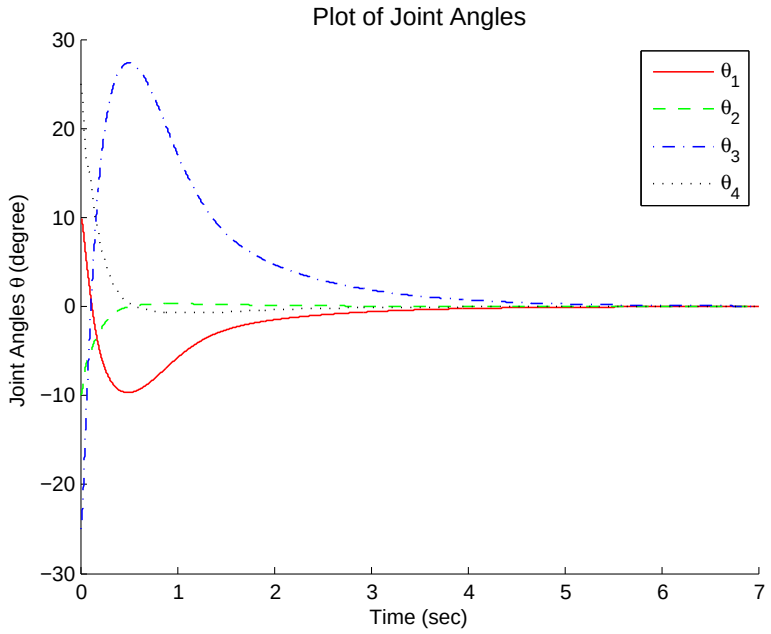

Fig. 6.22 Example 5. Change of the joint angles w.r.t. time

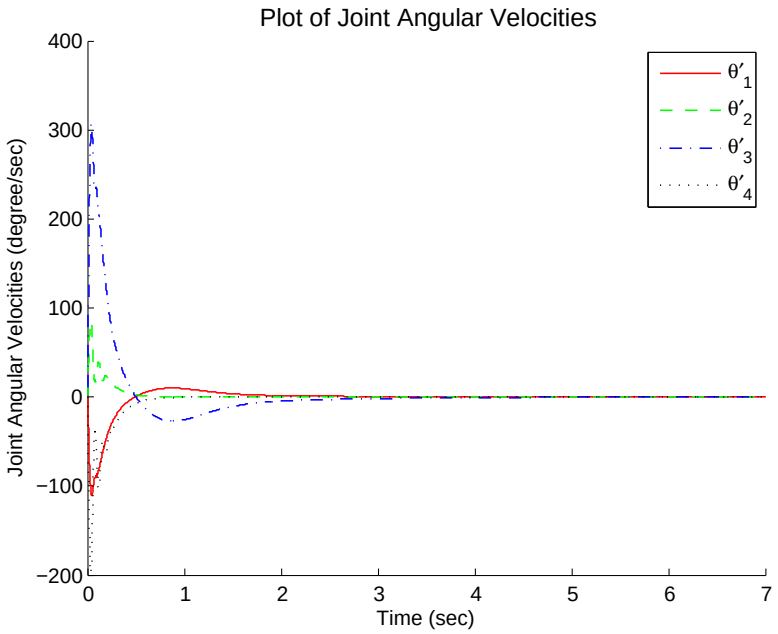

Fig. 6.23 Example 5. Change of joint angle rates w.r.t. time

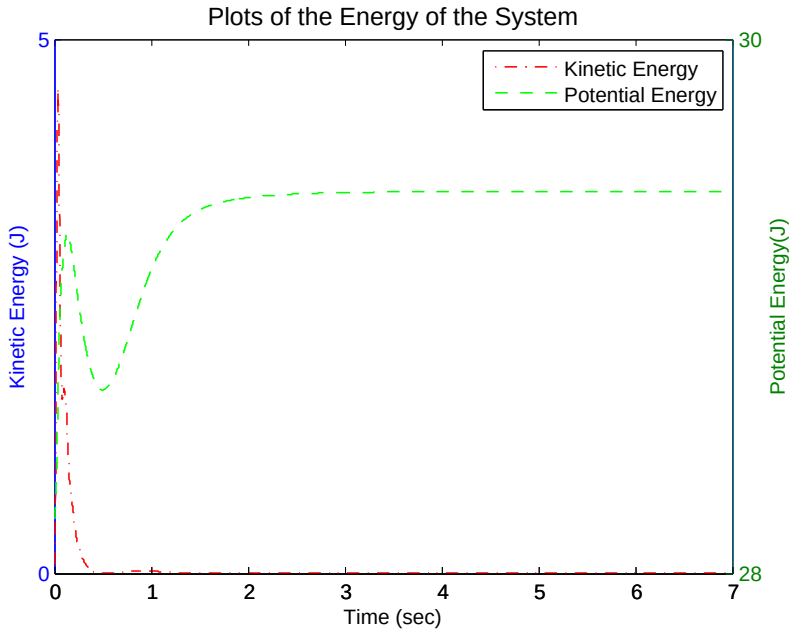

Fig. 6.24 Example 5. Change of kinetic and potential energies w.r.t. time

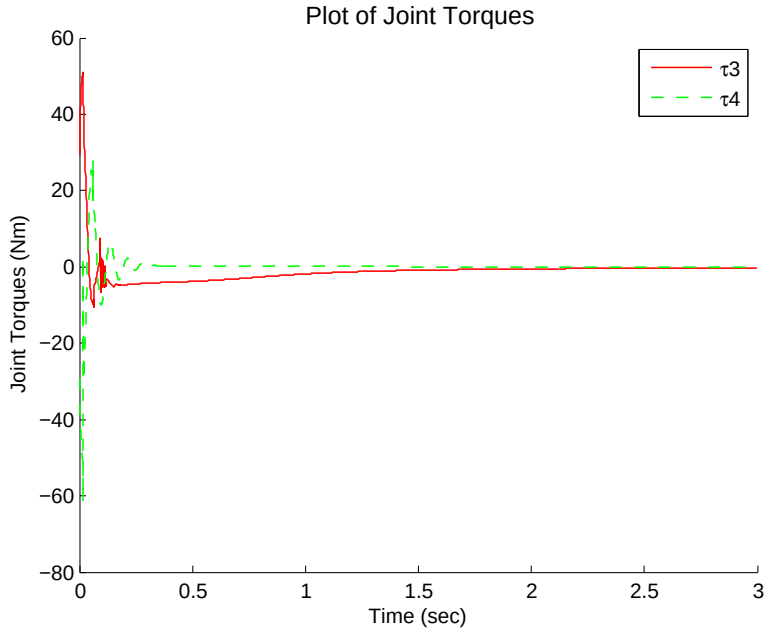

Fig. 6.25 Example 5. Change of Applied Torques

Example 6 In this example, we again use VGSTA, with $k_3 = 20$, $\delta = 0$, $\beta = 10$ and $\epsilon = 0.5$. The IC is now $q_0 = [4°, 0°, 0°, 0°]^\top$, $\dot{q}_0 = \mathbf{0}$.

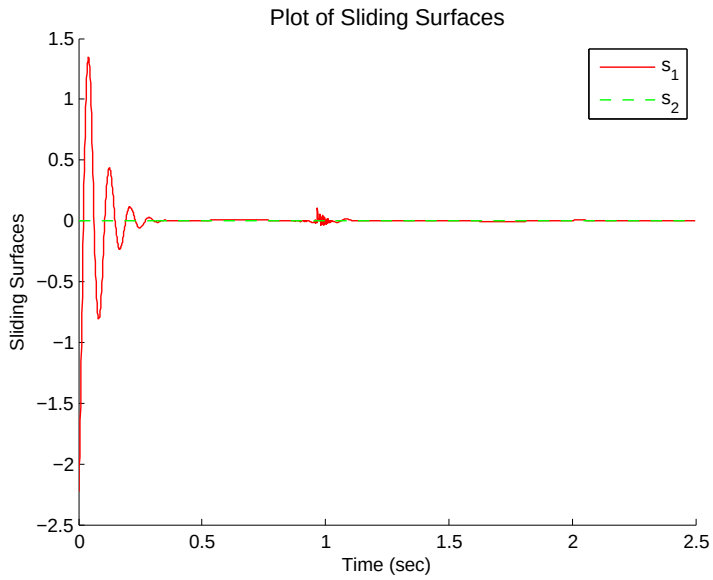

Fig. 6.26 Example 6. Change of the sliding surfaces w.r.t. time

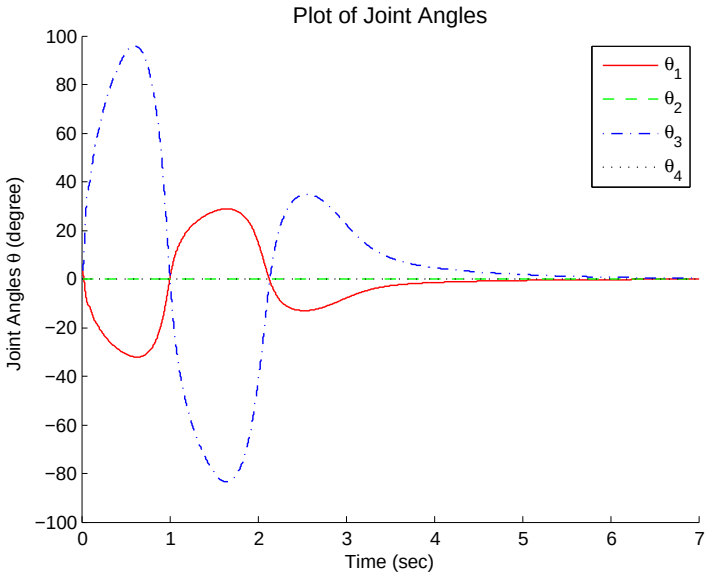

Fig. 6.27 Example 6. Change of the joint angles w.r.t. time

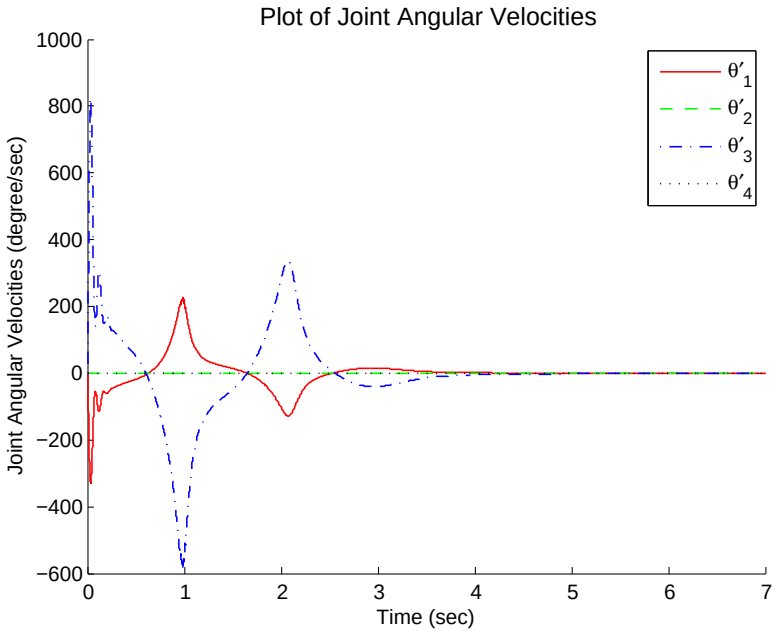

Fig. 6.28 Example 6. Change of joint angle rates w.r.t. time

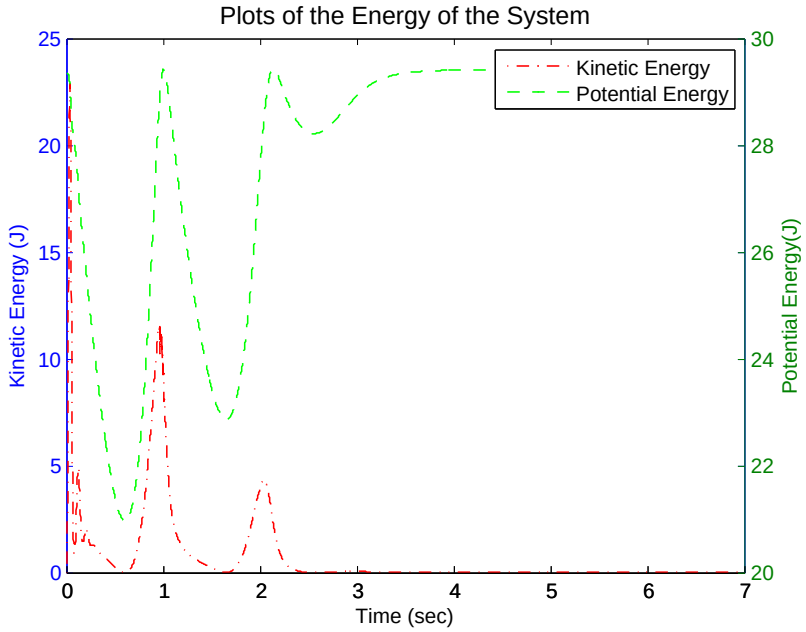

Fig. 6.29 Example 6. Change of kinetic and potential energies w.r.t. time

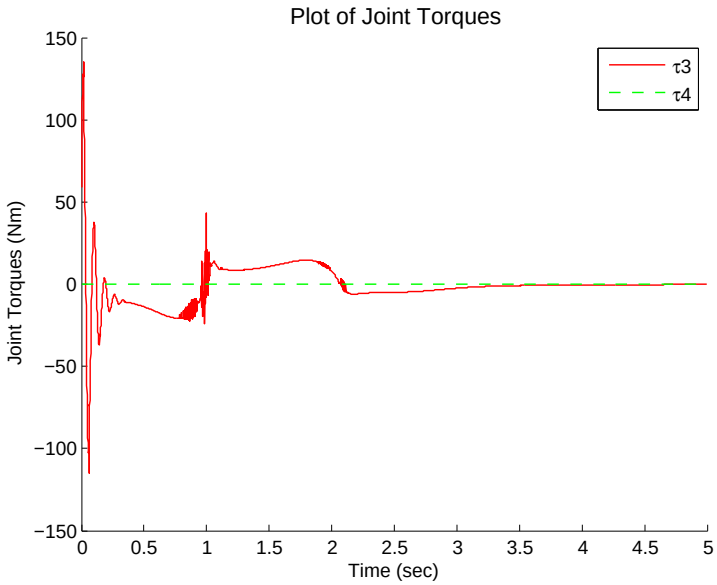

Fig. 6.30 Example 6. Change of Applied Torques

VGSTA offers a feasible alternative to LQR. As evidenced through simulations, IC of in Example 6 can be controlled properly only with VGSTA, as all the other methods including LQR could not render the system stable. However, the cost of the VGSTA controller is much higher than other controllers as intensive real-time calculation is required.
By varying the gain parameters for VGSTA, it has been observed that:

- Increasing k_3 could help deal with more perturbations but increase the overall torque
- Decreasing K would slightly reduce the torque, but lengthens settling time and somewhat reduce the ROC slightly.
- A good balance between beta and epsilon should be achieved: according to the expressions, decreasing ϵ would decrease overall gain; increasing β would increase the number of oscillations, making k_2 bigger, yet decreasing k_1.

6.2.5 Simulation Results for the system with disturbance inputs

Robustness Test In Examples 7 through 10, we apply a periodic disturbance pulse of torque amplitude of 5Nm, width of 0.4 second with a period of 1s added to both torques, with the IC $q_0 = [10°, -10°, -25°, 25°]^\top$, $\dot{q}_0 = \mathbf{0}$ for LQR, SM, standard STA and VGSTA,

respectively.

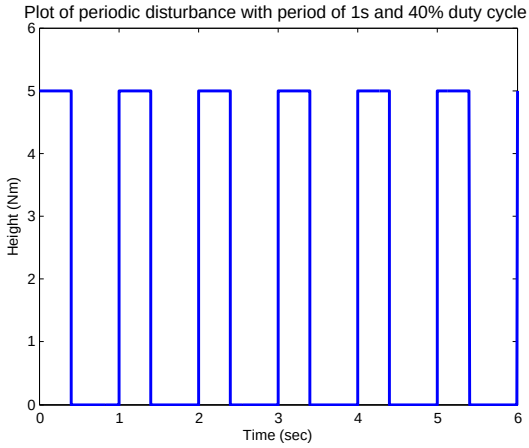

Fig. 6.31 Periodic Pulse Disturbance

Example 7 LQR controller using the same gain as in Example 1: the LQR controller failed to stabilize after a few seconds.

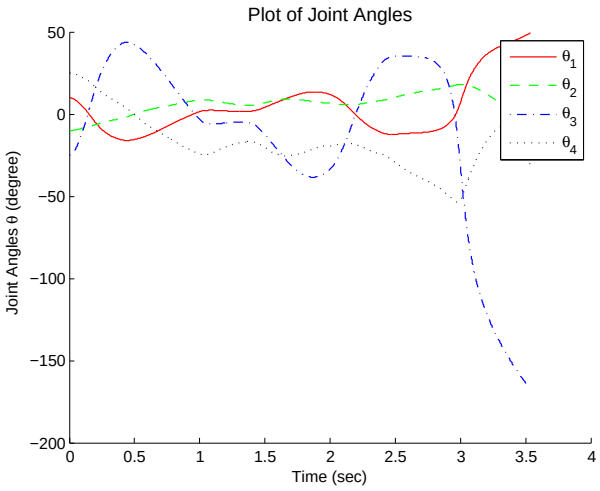

Fig. 6.32 Example 7. Change of the joint angles w.r.t. time

Example 8 First Order SM controller using the same gain as in Example 3: the controllers handles the disturbance well with fast switching inputs.

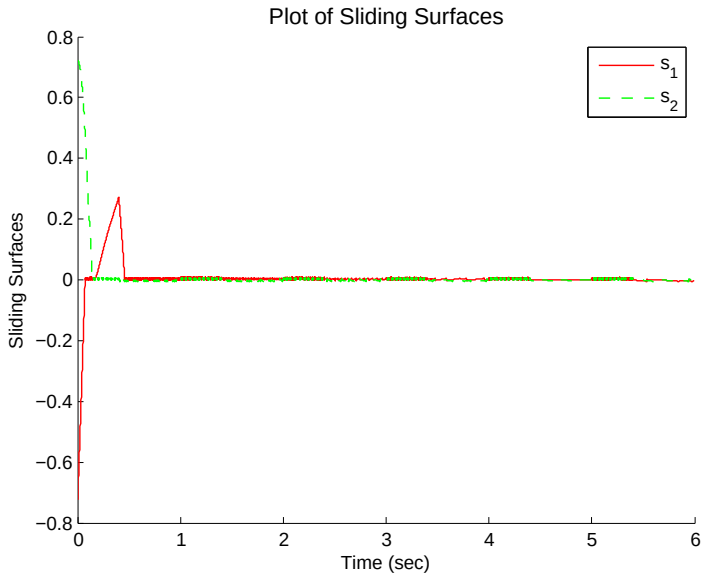

Fig. 6.33 Example 8. Change of the sliding surfaces w.r.t. time

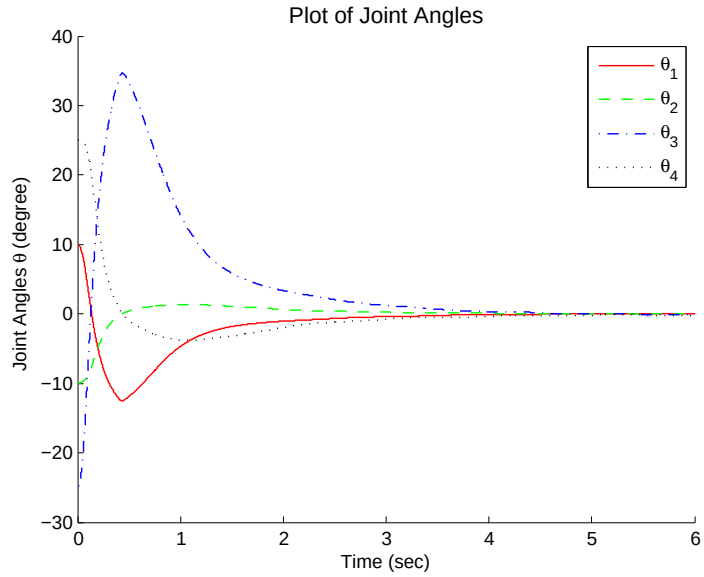

Fig. 6.34 Example 8. Change of the joint angles w.r.t. time

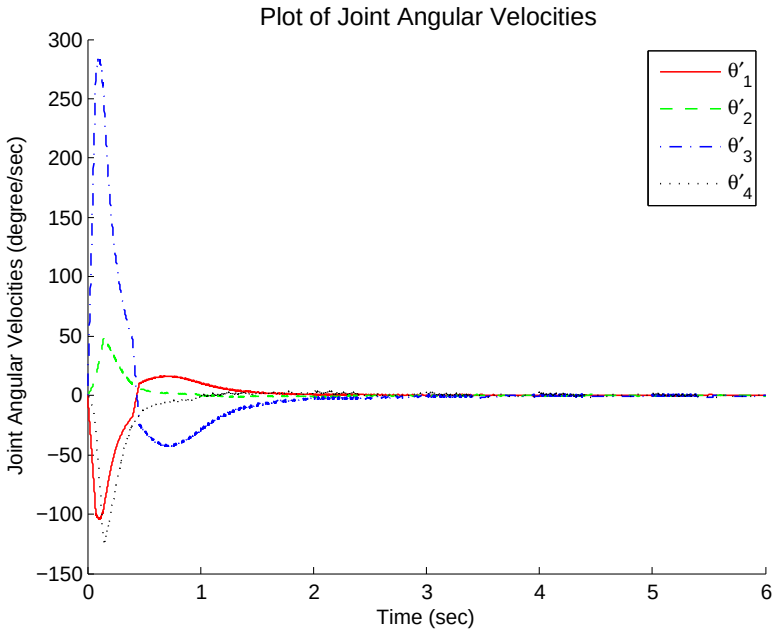

Fig. 6.35 Example 8. Change of joint angle rates w.r.t. time

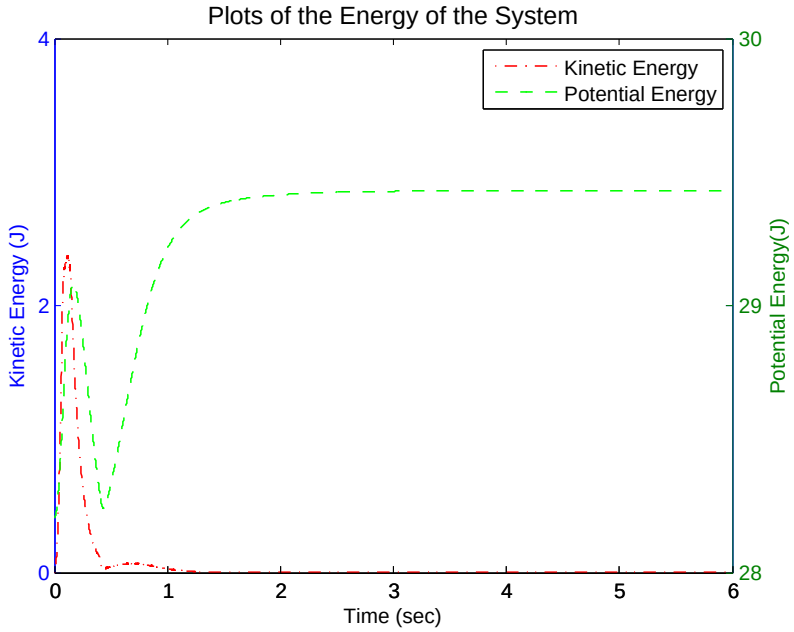

Fig. 6.36 Example 8. Change of kinetic and potential energies w.r.t. time

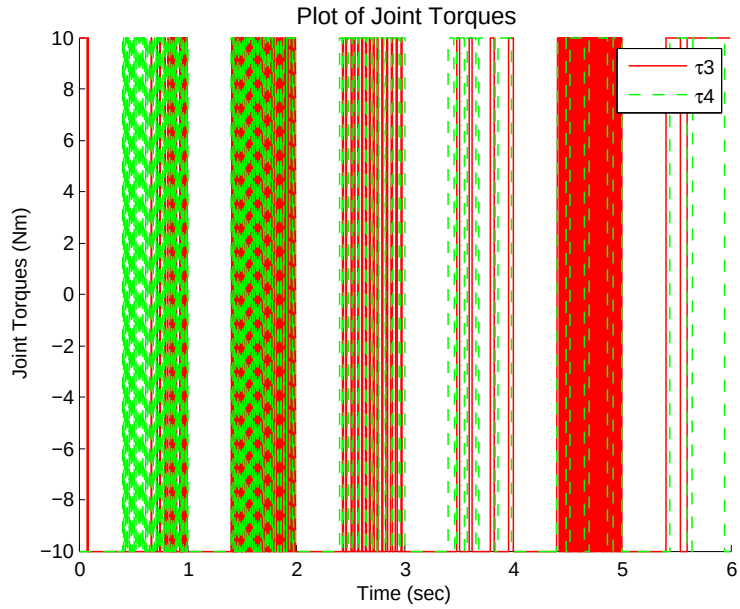

Fig. 6.37 Example 8. Change of Applied Torques

Example 9 STA controller using the same gain as in Example 4.

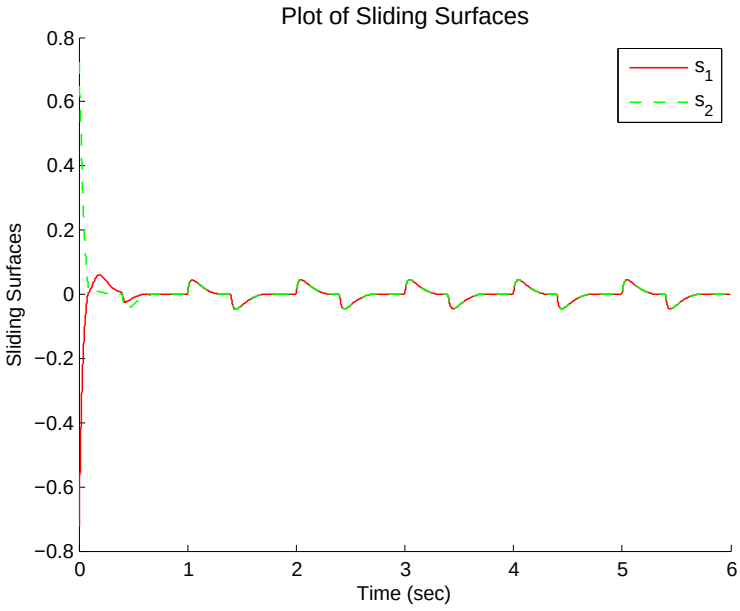

Fig. 6.38 Example 9. Change of the sliding surfaces w.r.t. time

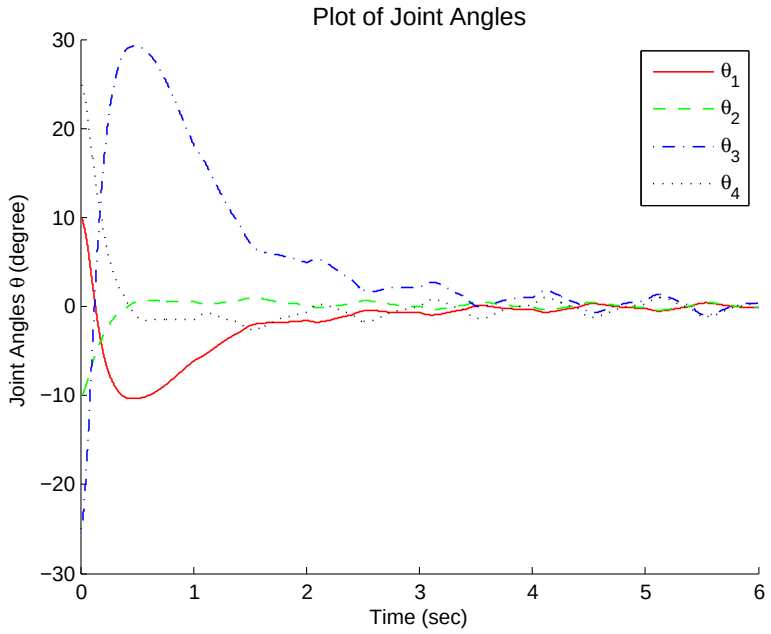

Fig. 6.39 Example 9. Change of the joint angles w.r.t. time

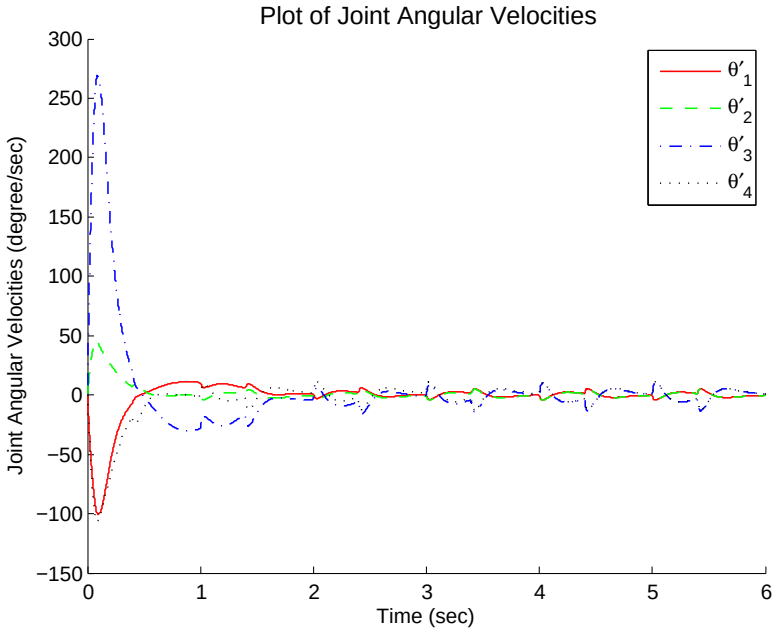

Fig. 6.40 Example 9. Change of joint angle rates w.r.t. time

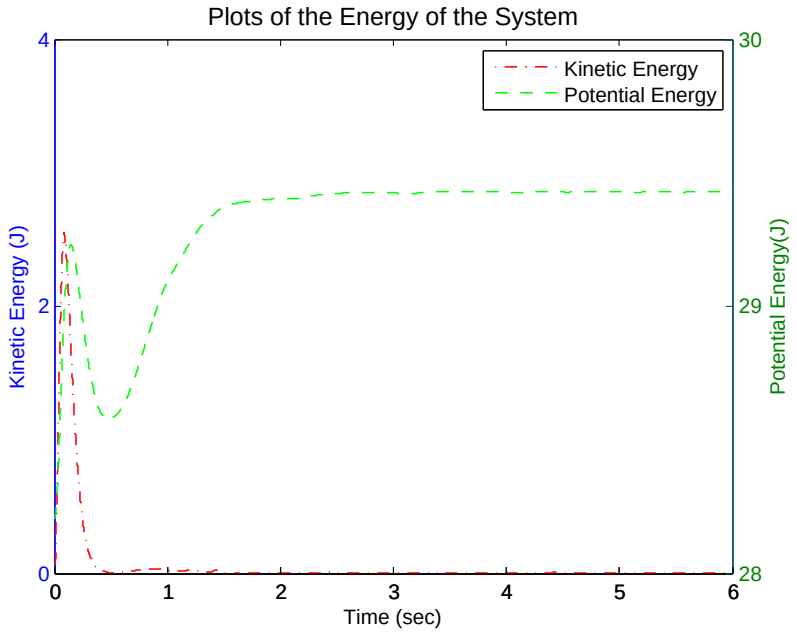

Fig. 6.41 Example 10. Change of kinetic and potential energies w.r.t. time

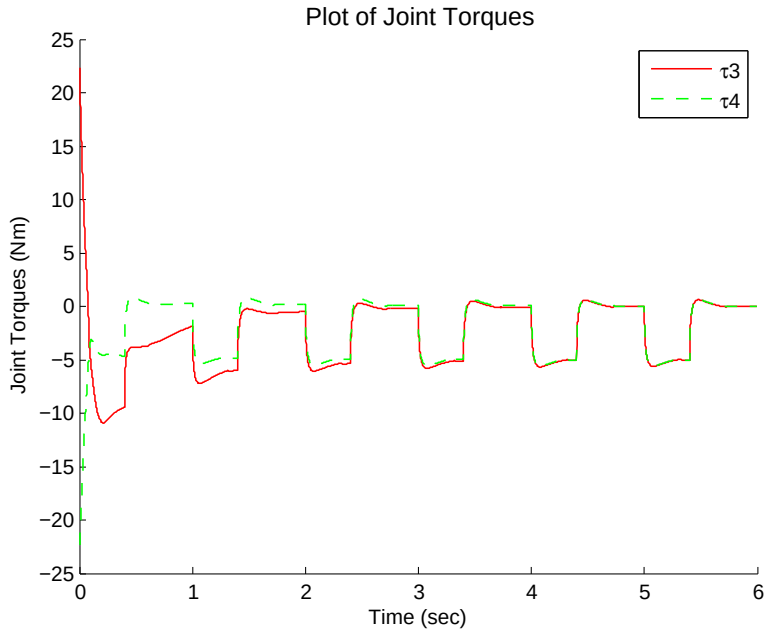

Fig. 6.42 Example 10. Change of Applied Torques

Example 10 VGSTA controller using the same gain as in Example 5.

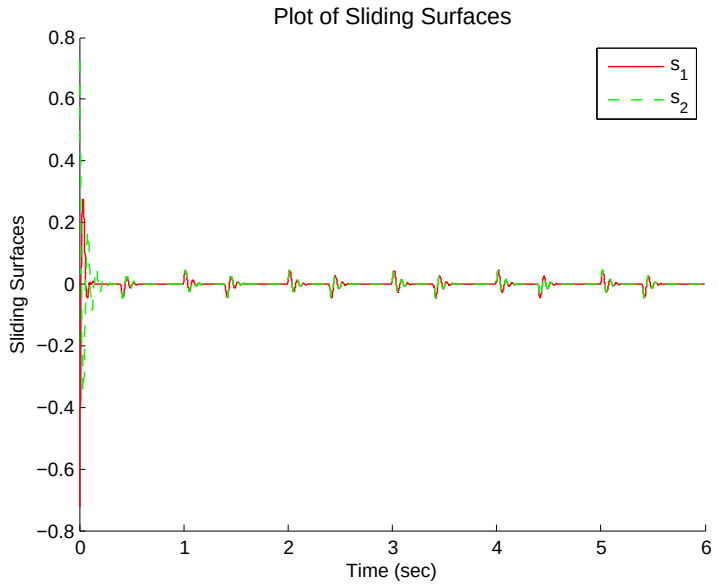

Fig. 6.43 Example 10. Change of the sliding surfaces w.r.t. time

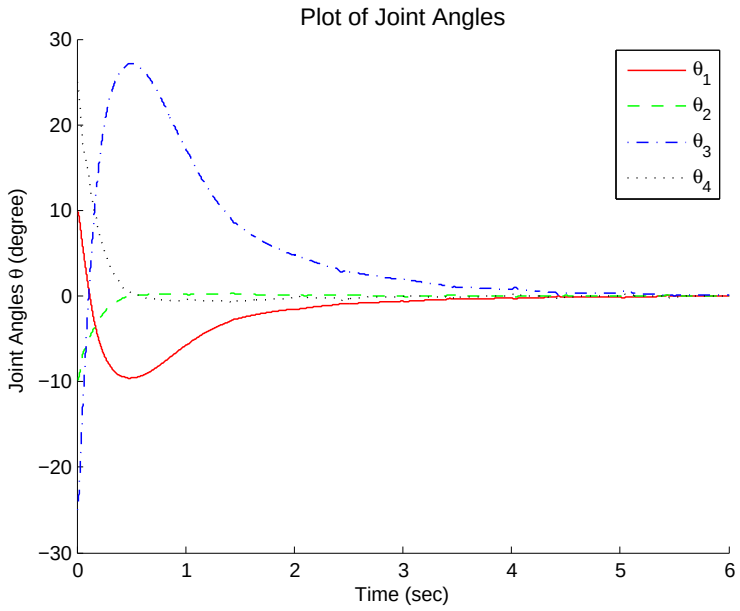

Fig. 6.44 Example 10. Change of the joint angles w.r.t. time

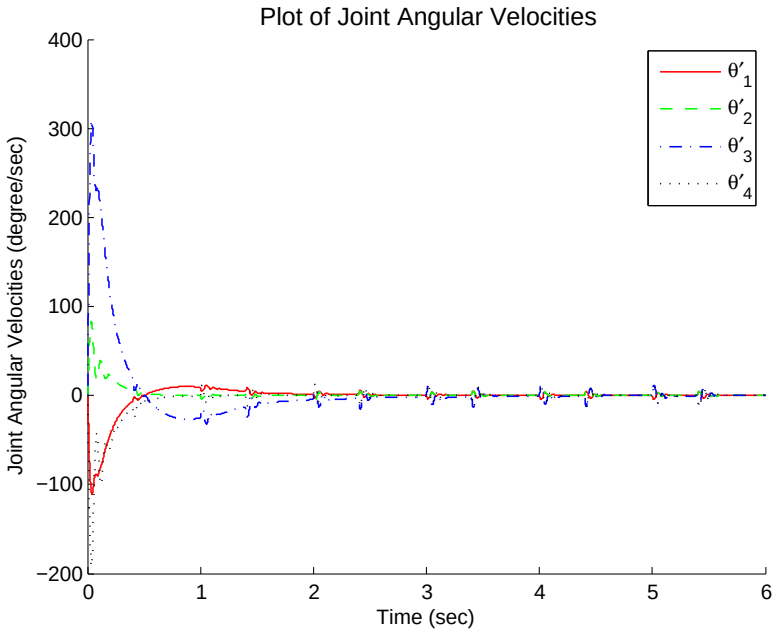

Fig. 6.45 Example 10. Change of joint angle rates w.r.t. time

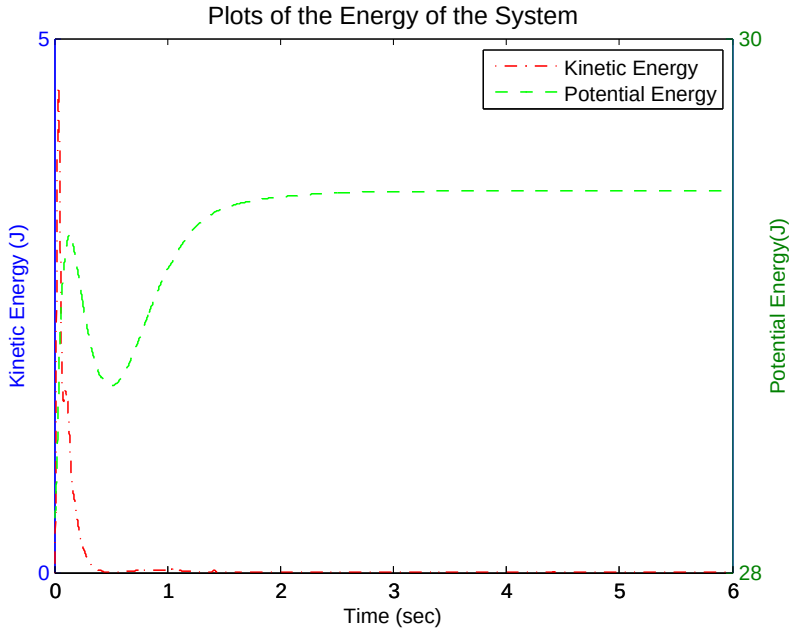

Fig. 6.46 Example 10. Change of kinetic and potential energies w.r.t. time

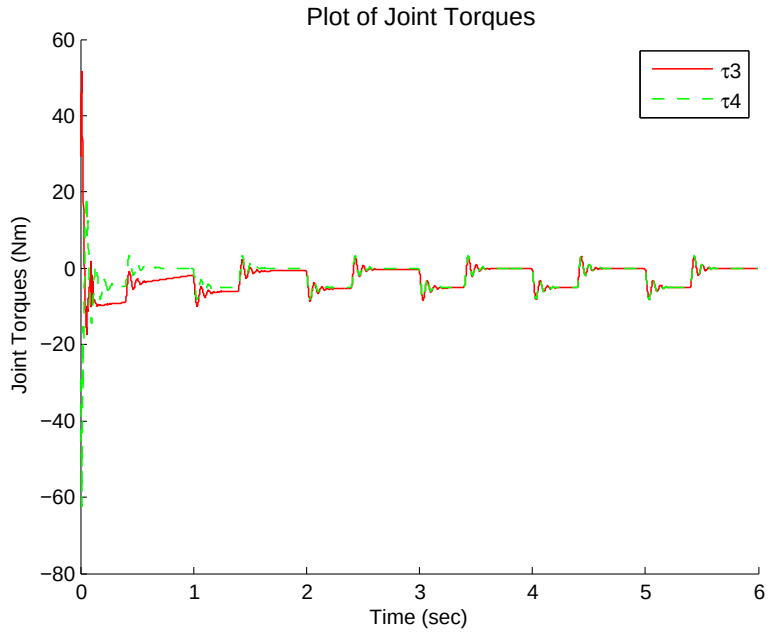

Fig. 6.47 Example 10. Change of Applied Torques

6.3 Discussions on Linearization-based Control Algorithms

In this chapter we presented two major approaches to stabilize the SDIP system when it starts off at a position relatively close to the unstable upright equilibrium: the LQR and quasi-linear SMC. The LQR controller relies on constant gains derived from solving the Algebraic Riccati Equation, and it has been shown that higher gains would enlarge the ROC but in the real physical systems, torque limits must be imposed and a good balance between ROC and controller gain should be achieved. The SMC methods constantly track the sliding surfaces, drive the system into the sliding manifolds to ensure that asymptotic stabilizations can be achieved.

The SM controller using VGSTA has been tested to extend the ROC slightly by successfully stabilizing the system with IC of $q_0 = [4^\circ, 0^\circ, 0^\circ, 0^\circ]^\top$, $\dot{q}_0 = \mathbf{0}$, as shown in Example 6. However, as verified by a number of simulations, the overall ROC based on these sliding mode methods have not been significantly improved over the LQR because the sliding surfaces are designed based on the assumption that the system is linear, but when the system moves

further away from equilibrium position, the nonlinearities of the system will invalidate the quasi-linear SMC approaches.

One major advantage of the SM controller is its robustness against disturbances applied at the inputs to the system, as demonstrated in the simulations, largely because the system is designed to be more sophisticated than the LQR controller, which is simply a constant proportional gain matrix K. In addition, the comparisons among the three sliding mode methods are discussed below:

- The first order constant gain SM controller is the simplest SM controller of the three, with fast switching inputs that drive the sliding surfaces to zero very fast. However, the controller exhibits chattering effects, yet has offered strong robustness in the presence of periodic disturbance inputs. (See Fig. 6.34 and Fig. 6.35)
- The standard STA controller is able to successfully stabilize the system in the presence of the periodic disturbances. More importantly, it produces continuous control inputs to the systems so chattering effects are avoided. According to the simulation plots, in the presence of disturbances, the joint angles and velocities do not follow the trajectories that are as smooth as the other two SM controllers. (See Fig. 6.39 and Fig. 6.40)
- The VGSTA controller combines the advantages of both of the previous controllers by adaptively changing the gains to smoothly achieve asymptotic stabilizations with continuous inputs with insensitivity to disturbances. (See Fig. 6.44 and Fig. 6.47) However, due to its sophistication in implementation, it has been shown to be more computationally expensive and runs more slowly in simulation as compared to the previous two methods.

Both the LQR and quasi-linear SMC approaches are able to balance the SDIP with small initial angle perturbations from upright equilibrium configuration. It is clear that the ROC is unsatisfactorily small for standing postural control purposes. Therefore, it is necessary to explore the nonlinear approaches in order to significantly extend the ROC. Nonlinear SMC methods are explored in the next chapter.

Chapter 7

Nonlinear Sliding Mode Control Approaches

In this chapter, two types of nonlinear controllers are studied and implemented in an attempt to enlarge the ROC and provide *semi-global swing-up* of the spatial double inverted pendulum towards its equilibrium configuration.

7.1 The Hybrid Approach

The two main hybrid approaches for achieving a satisfactory behaviour of the SDIP system can be summarized as follows:

- *Global stabilization* (switching from swing-up controller to LQR or quasi-linear STA controller as soon as the system reaches the ROC of the linear controller.)
- *Tracking of small periodic trajectories* (switching from swing-up to tracking controller using SMC when the system reaches the vicinity of the upright equilibrium)

Spong [9], and Xinjilefu [6] have used the former approach to tackle the stabilization problem by first using a nonlinear algorithm for the swing-up phase and then switching to an LQR controller for the balance phase after the system reaches the vicinity of the equilibrium configuration. Puga et al. [16] have used the latter approach for the swing-up and tracking of an acrobot.

It is obvious that achieving *global stabilization* is ideal if the controller can be designed properly. The latter approach does not put the system at rest at the unstable equilibrium and can be achieved for the SDIP system only if the zero-dynamics of the system can be properly identified in 3-D. However, it is extremely difficult to obtain due to the enormous size of the expressions that govern the dynamics of the system. As a result, our focus is to design a nonlinear SM controller which is able to push the system into the ROC of the linear controller in order to achieve *global stabilization*, rather than relying on the stabilization of the zero-dynamics.

Xinjilefu [6] has stated the following switching condition from the nonlinear swing-up controller to the LQR controller: *if* $V_{max} + L(q, \dot{q}) < 3\%V_{max}$*, switch to* LQR, where V_{max} is the maximum potential energy of the system achieved only at the unstable equilibrium position, and $L(q, \dot{q})$ is the Lagrangian of the system. This is a good indicator of how stable the system becomes because the lower its value becomes, the closer it is to the upright equilibrium position and the less velocity it has.

7.2 Nonlinear First-Order Sliding Mode Control

A first-order sliding mode controller is a SM controller that drives the motion of system states onto the sliding surface $s = 0$ and keep the system on the sliding manifold. In this section, we design and simulate a *First-Order Hierarchical Sliding Mode Controller* (FOHSMC).

7.2.1 Hierarchical SM Controller Design

This section is devoted to the implementation of the hierarchical sliding mode control approach [19] that has been designed and implemented for 2-D systems such as a pendubot with underactuation degree of 1 into this 3-D system with underactuation degree of 2. First, the lower half of System Model I from Eq. (4.19) can be viewed as:

$$\ddot{q} = f(q, \dot{q}) + B(q, \dot{q})\tau_a = -M(q)^{-1}[G(q) - \dot{q}^T Q(q)\dot{q}] + M(q)^{-1}F(q)\tau_a \tag{7.1}$$

where the matrices and vectors can be partitioned as:

$$\ddot{q} = \begin{bmatrix} \ddot{q}_u \\ \ddot{q}_a \end{bmatrix} \qquad f(q,\dot{q}) = \begin{bmatrix} f_u(q,\dot{q}) \\ f_a(q,\dot{q}) \end{bmatrix} \qquad B(q,\dot{q}) = \begin{bmatrix} B_u(q,\dot{q}) \\ B_a(q,\dot{q}) \end{bmatrix}$$

Thus the system model can be re-written into the following form:

$$\ddot{q}_u = f_u(q,\dot{q}) + B_u(q,\dot{q})\tau_a \tag{7.2a}$$
$$\ddot{q}_a = f_a(q,\dot{q}) + B_a(q,\dot{q})\tau_a \tag{7.2b}$$

where $q_u = [q_1\ q_2]^\top$ and $q_a = [q_3\ q_4]^\top$ are the unactuated and actuated DoFs, respectively. We can now construct a suitable pair of bottom-level sliding surfaces:

$$s_1 = C_1 q_a + \dot{q}_a \tag{7.3a}$$
$$s_2 = C_2 q_u + \dot{q}_u \tag{7.3b}$$

where $C_1 = diag(c_{11}, c_{12})$ and $C_2 = diag(c_{21}, c_{22})$ are positive definite diagonal parameter matrices. Then using the equivalent control method, by setting $\dot{s}_1 = \dot{s}_2 = 0$, the equivalent control law of the subsystems can be obtained as:

$$\tau_{eq1} = -B_a(q,\dot{q})^{-1}(f_a(q,\dot{q}) + C_1\dot{q}_a) \tag{7.4a}$$
$$\tau_{eq2} = -B_u(q,\dot{q})^{-1}(f_u(q,\dot{q}) + C_2\dot{q}_u) \tag{7.4b}$$

For our SDIP system, it is difficult to control 4 state outputs with 2 actuators. Therefore, to ensure that each subsystem follows its own sliding surface, the total control law should include the equivalent control law from each subsystem. The total control law can be defined as:

$$\tau_a = \tau_{eq1} + \tau_{eq2} + \tau_{sw} \tag{7.5}$$

where τ_{sw} is the switching part of the sliding mode controller. The top-level sliding surface can be constructed as follows:

$$S = Ps_1 + Qs_2 \tag{7.6}$$

where $P = diag(p_1, p_2)$ and $Q = diag(q_1, q_2)$ are sliding mode parameter matrices. The objective is to design a swing-up control law that drives the system into the desired sliding surfaces. The switching control law that guarantees reachability of the top-level sliding surface can be designed by first defining the Lyapunov energy function as follows:

$$V(t) = \frac{1}{2} S^{\mathrm{T}} S \tag{7.7}$$

A Lyapunov analysis can be performed by differentiating V w.r.t. time t:

$$\begin{aligned}
\dot{V} &= S^{\mathrm{T}} \dot{S} = S^{\mathrm{T}} (P\dot{s}_1 + Q\dot{s}_2) \\
&= S^{\mathrm{T}} [P(C_1 \dot{q}_a + \ddot{q}_a) + Q(C_2 \dot{q}_u + \ddot{q}_u)] \\
&= S^{\mathrm{T}} [P(C_1 \dot{q}_a + f_a + B_a(\tau_{eq1} + \tau_{eq2} + \tau_{sw})) \\
&\quad + Q(C_2 \dot{q}_u + f_u + B_u(\tau_{eq1} + \tau_{eq2} + \tau_{sw}))] \\
&= S^{\mathrm{T}} [PB_a(\tau_{eq2} + \tau_{sw}) + QB_u(\tau_{eq1} + \tau_{sw})] \\
&= S^{\mathrm{T}} [(QB_u \tau_{eq1} + PB_a \tau_{eq2}) + (PB_a + QB_u)\tau_{sw}]
\end{aligned} \tag{7.8}$$

Let $(QB_u\tau_{eq1} + PB_a\tau_{eq2}) + (PB_a + QB_u)\tau_{sw} = -K_1 S - K_2 \mathrm{sgn}(S)$ where K_1 and K_2 are positive definite diagonal constant matrices, then

$$\tau_{sw} = -(PB_a + QB_u)^{-1} [(QB_u\tau_{eq1} + PB_a\tau_{eq2}) + K_1 S + K_2 \mathrm{sgn}(S)]$$

As a result, the total control law τ_a of the control system is given by:

$$\begin{aligned}
\tau_a &= \tau_{eq1} + \tau_{eq2} + \tau_{sw} \\
&= \tau_{eq1} + \tau_{eq2} - (PB_a + QB_u)^{-1} [(QB_u\tau_{eq1} + PB_a\tau_{eq2}) + K_1 S + K_2 \mathrm{sgn}(S)] \\
&= (PB_a + QB_u)^{-1} (PB_a\tau_{eq1} + QB_u\tau_{eq2} - K_1 S - K_2 \mathrm{sgn}(S))
\end{aligned} \tag{7.9}$$

Then (7.8) becomes:

$$\begin{aligned}
\dot{V} &= S^{\mathrm{T}} [-K_1 S - K_2 \mathrm{sgn}(S)] \\
&= -K_1 \left\| S \right\|^2 - K_2 \left\| S \right\|_1 < 0
\end{aligned} \tag{7.10}$$

Therefore, the top-level sliding surface is stable. However, the first layer sliding surfaces are not guaranteed to follow their own surfaces. The conditions for first-layer stabilities

have been proposed and analyzed in [19], where the weighting parameters are varied with varying conditions, but are difficult to be transplanted into our controller. As a result, our best hope is that by choosing a suitable set weighting matrices, the kinetic energy could be minimized at the vicinity of the unstable equilibrium, i.e, V_{max} + Lagrangian is small enough so the linear control can take over.

The diagonal weighting matrices P, Q, C_1 and C_2 are designed and tuned with the following considerations in mind:

- The weighting matrices P and Q dictate the relative importance of each bottom-level sliding surface to the top-level sliding surfaces of actuated and unactuated joint angles, respectively.
- C_1 and C_2 dictate the relative relative rates of convergence to and after reaching the bottom-level sliding surfaces of actuated and the actuated joint angles.
- The gains K_1 and K_2, should be chosen to match the speed of convergence of the top-level sliding surface.

7.2.2 Simulation Results

In the following two examples, FOHSMC is used to perform the swing-up of the system. After some tunings, the gain parameters are designed to be: $K_1 = diag(8, 8)$, $K_2 = diag(0.3, 0.3)$, $C_1 = diag(5, 5)$ and $C_2 = diag(4, 4)$. The plot of V_{max} + Lagrangian will be included in all of the simulation results that follow, where V_{max} is 29.43 J with the physical parameters used in previously in section (6.1).

Example 11 FOHSMC with $P = diag(7, 5), Q = diag(3, 2)$ and $q_0 = [10°, -10°, -30°, 30°]^\top$, $\dot{q}_0 = \mathbf{0}$, outside the ROC of the linear controller.

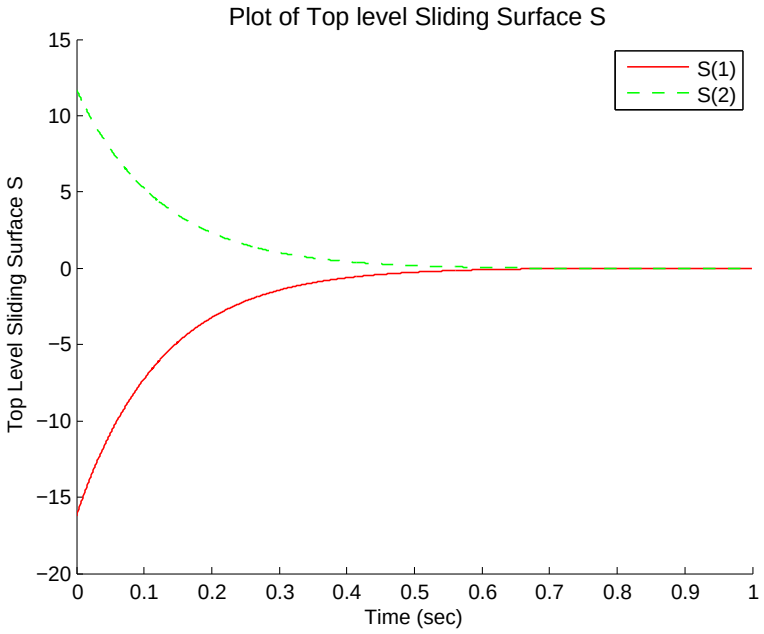

Fig. 7.1 Example 11. Change of S w.r.t. time

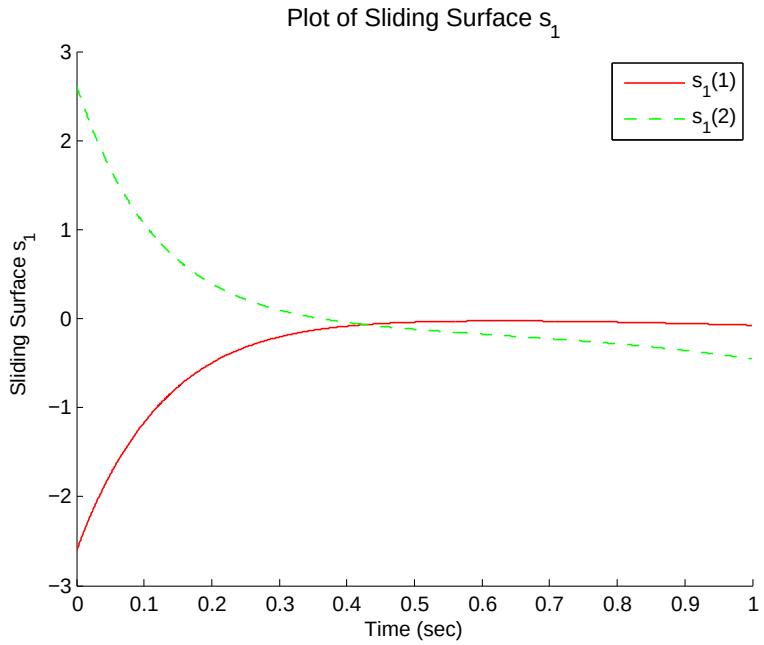

Fig. 7.2 Example 11. Change of s_1 w.r.t. time

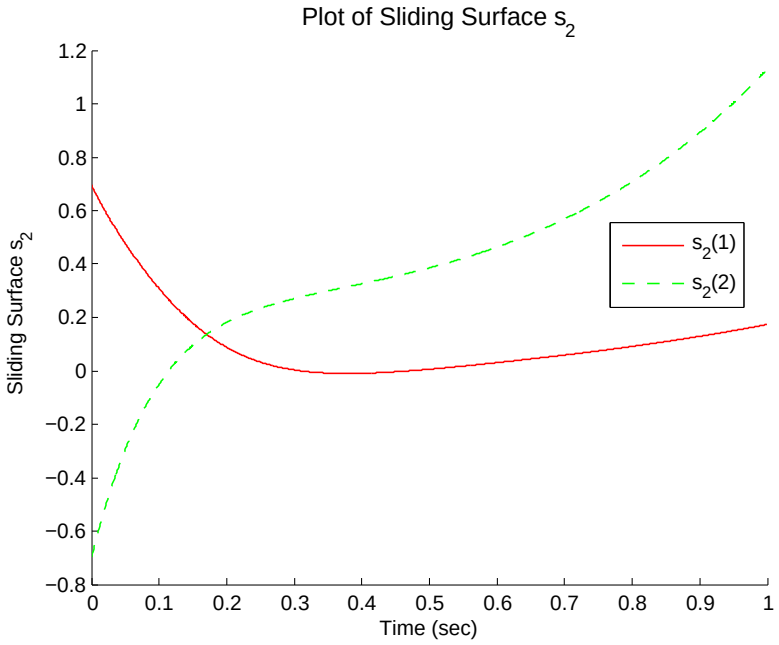

Fig. 7.3 Example 11. Change of s_2 w.r.t. time

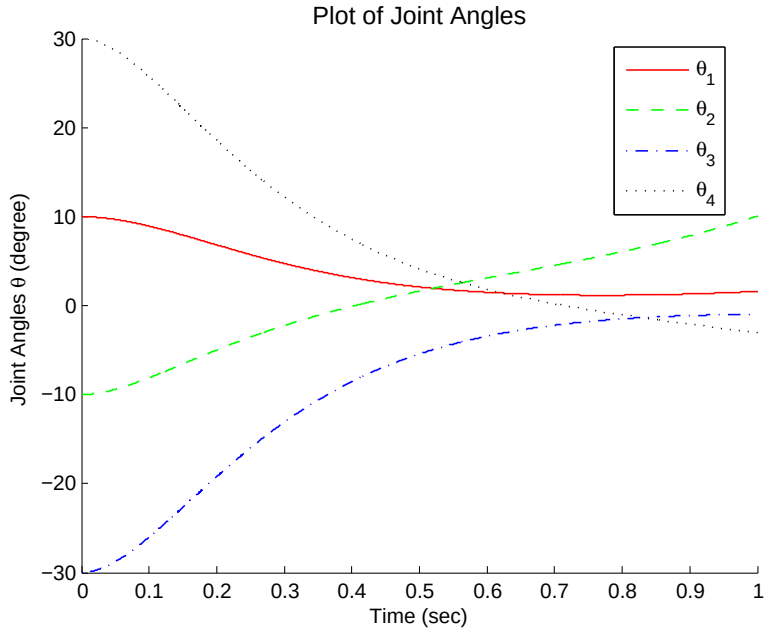

Fig. 7.4 Example 11. Change of the joint angles w.r.t. time

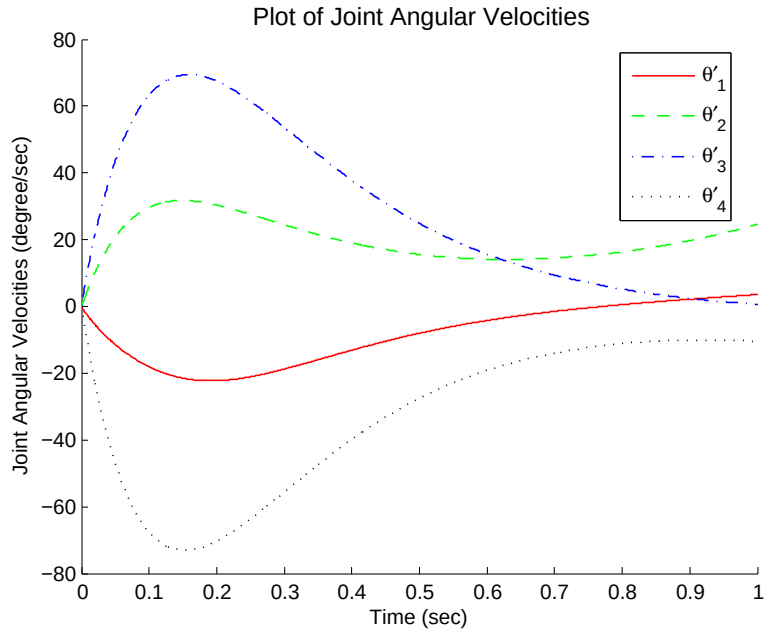

Fig. 7.5 Example 11. Change of joint angle rates w.r.t. time

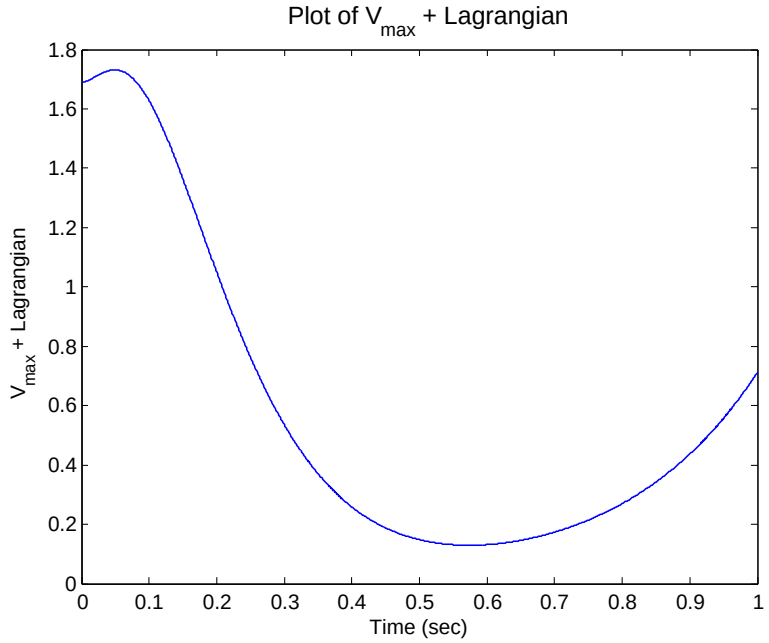

Fig. 7.6 Example 11. Change of V_{max} + Lagrangian w.r.t. time

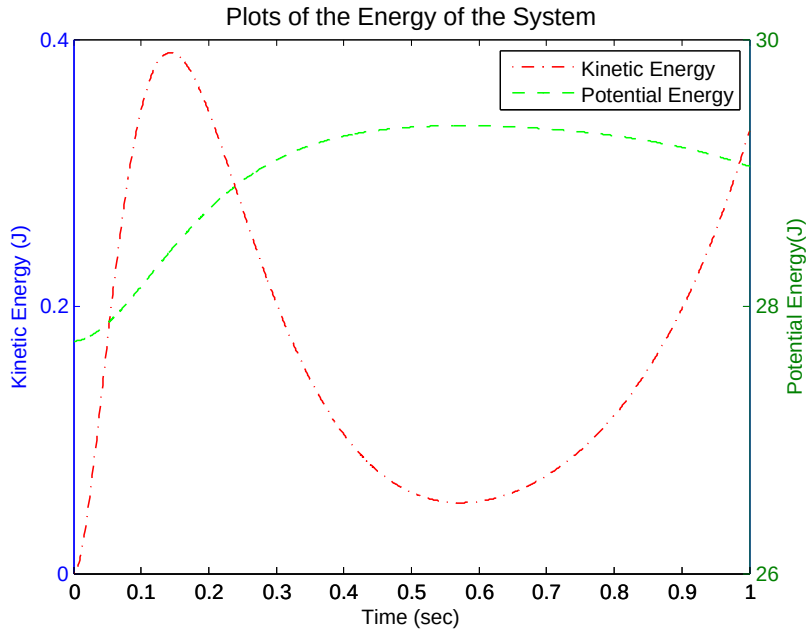

Fig. 7.7 Example 11. Change of kinetic and potential energies w.r.t. time

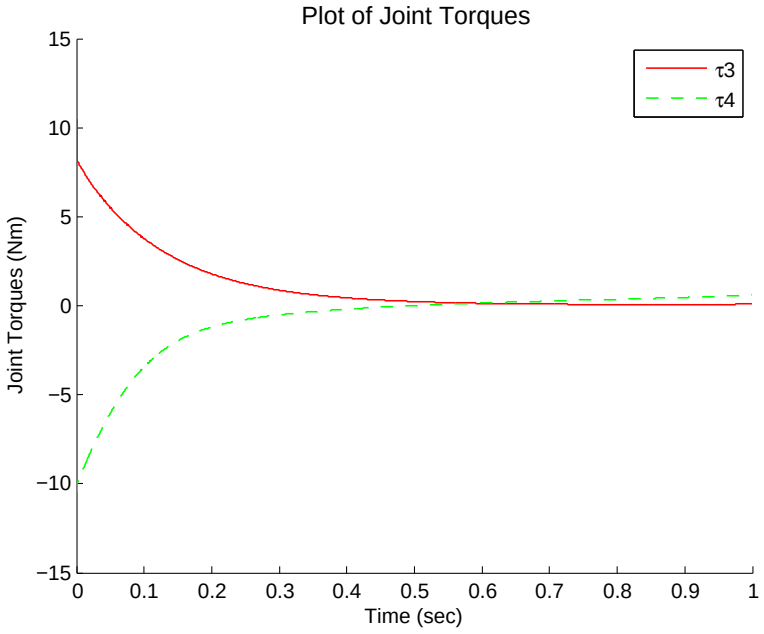

Fig. 7.8 Example 11. Change of Applied Torques

Example 12 FOHSMC with $P = diag(8, 7), Q = diag(1, 0.5)$ and $q_0 = [30°, 20°, -135°, -25°]^\top$, $\dot{q}_0 = \mathbf{0}$, quite far from the unstable equilibrium.

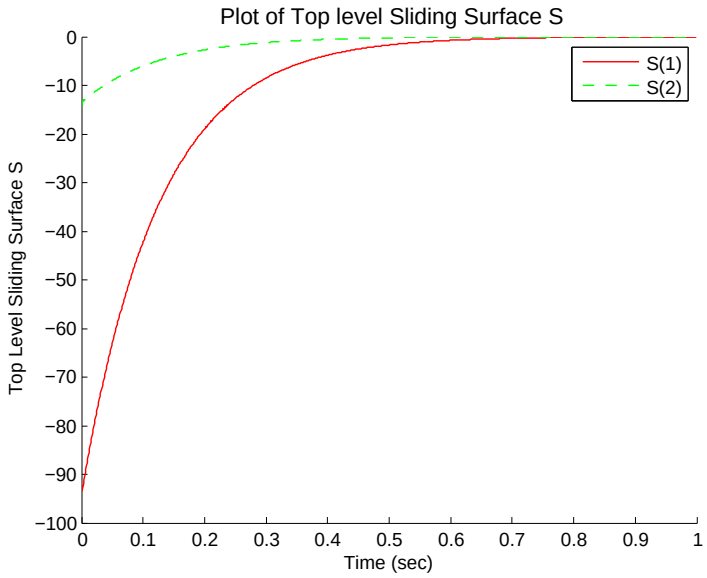

Fig. 7.9 Example 12. Change of S w.r.t. time

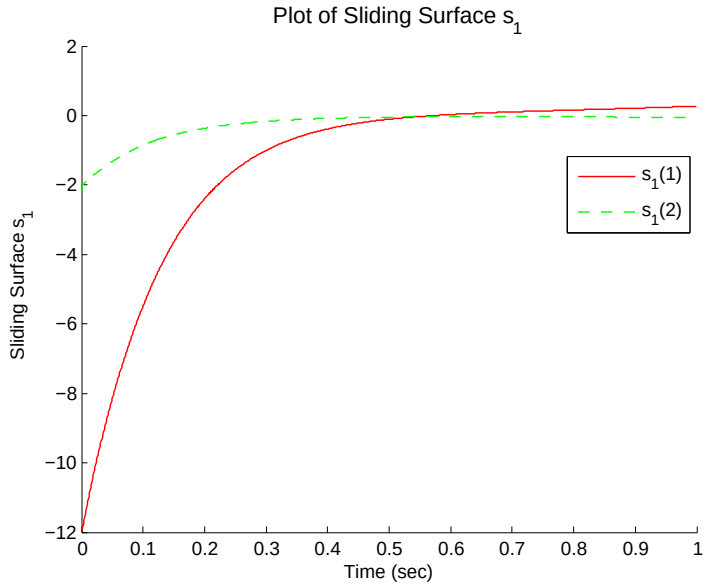

Fig. 7.10 Example 12. Change of the s_1 w.r.t. time

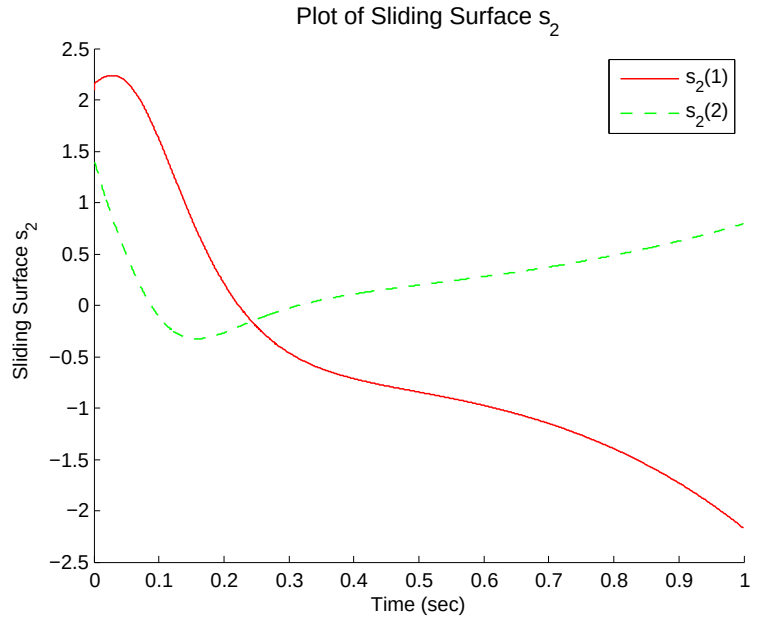

Fig. 7.11 Example 12. Change of the s_2 w.r.t. time

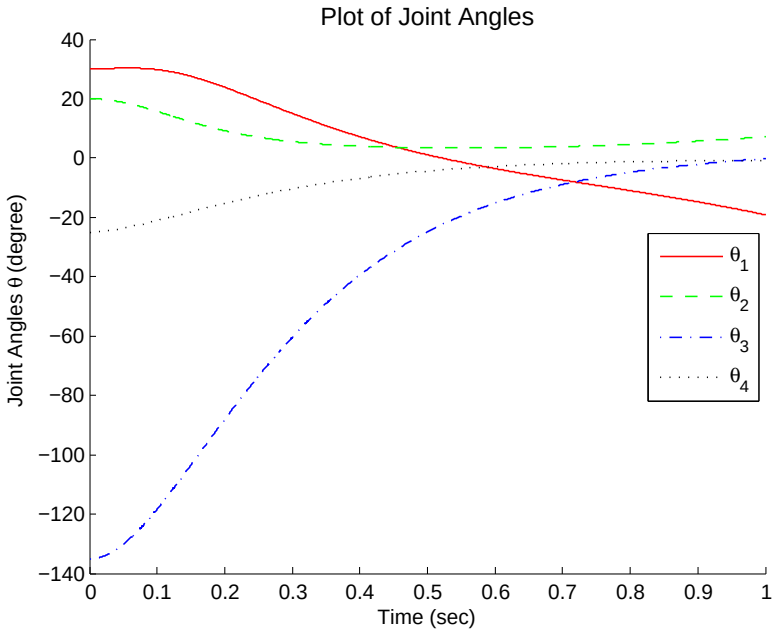

Fig. 7.12 Example 12. Change of the joint angles w.r.t. time

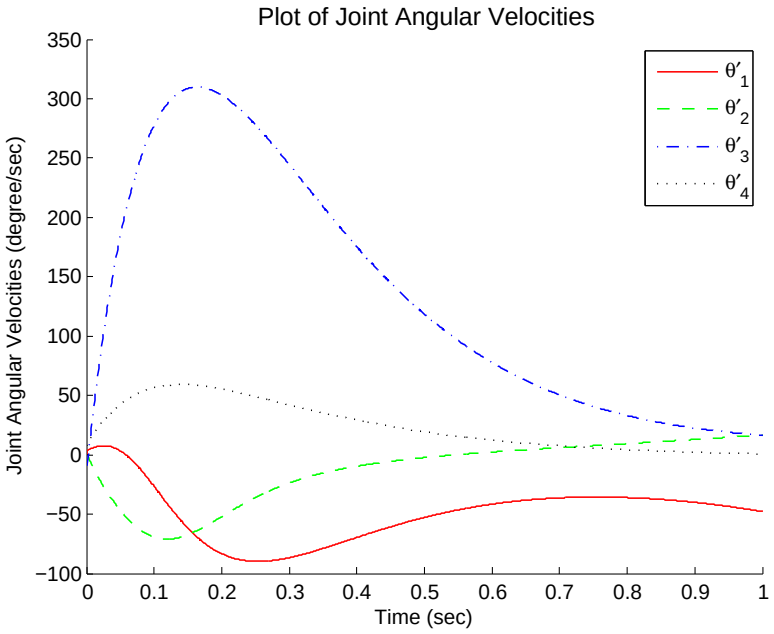

Fig. 7.13 Example 12. Change of joint angle rates w.r.t. time

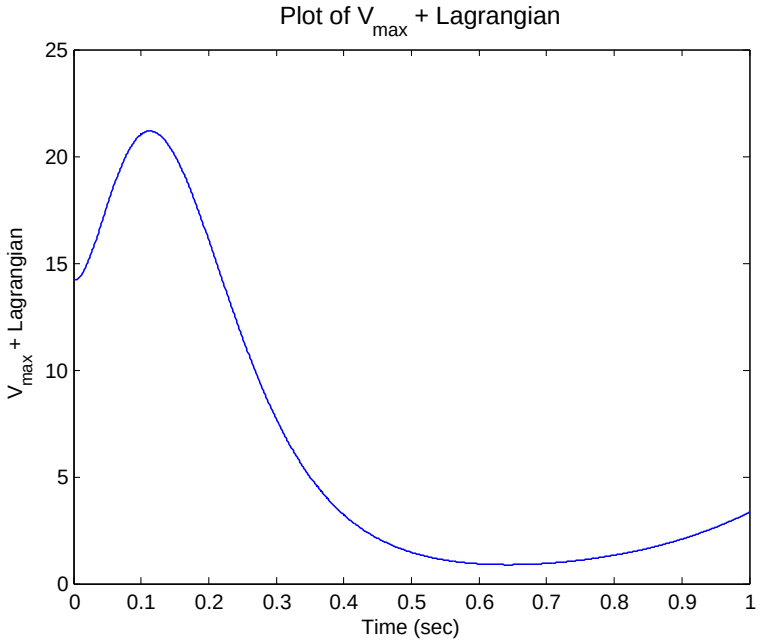

Fig. 7.14 Example 12. Change of V_{max} + Lagrangian w.r.t. time

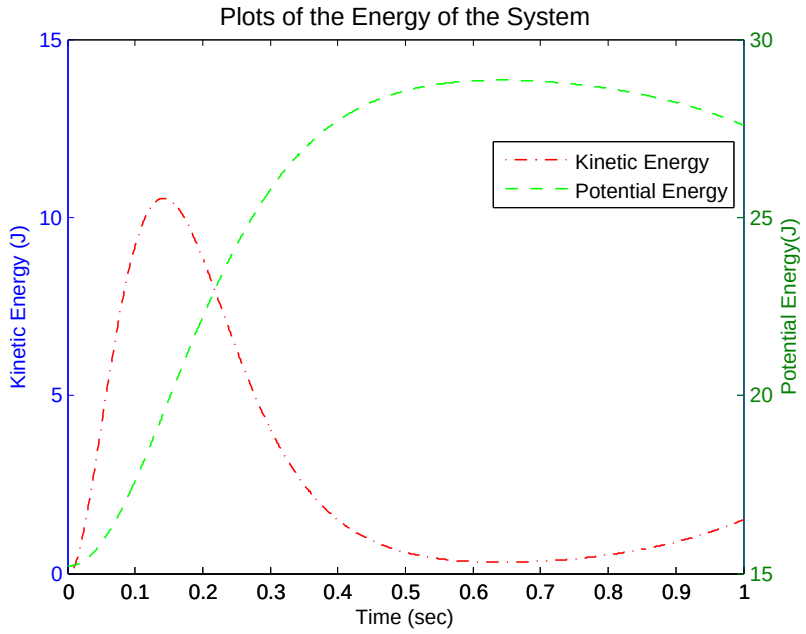

Fig. 7.15 Example 12. Change of kinetic and potential energies w.r.t. time

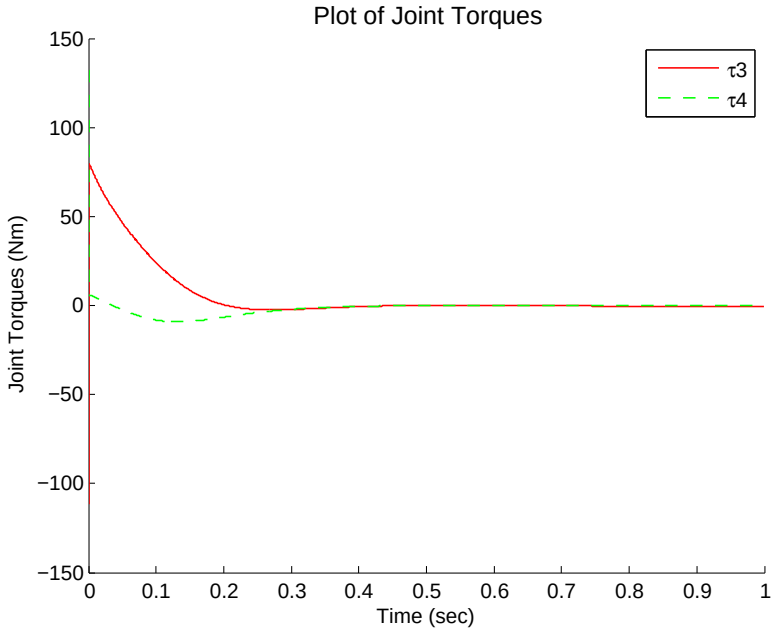

Fig. 7.16 Example 12. Change of Applied Torques

7.2.3 Discussions

In this section we have extended the existing hierarchical first order sliding mode control approach to 3-D in order to work for our SDIP system. According to simulation results, the swing-up controller is able to bring the system to a region quite close to the upright equilibrium configuration. However, when switching condition described by Xinijlefu [6] is satisfied at $t \in [0.1,\ 1]s$ in Fig. (7.6) and $t \in [0.5,\ 0.8]s$ in Fig. (7.14), neither the LQR controller nor the VGSTA controller used in the previous chapter has been able to take over the nonlinear controller during the above time interval to stabilize the system successfully. This has led us to believe that aforementioned switching condition is not sufficient to guarantee that the system has indeed arrived inside the ROC of the linear controller, as previously thought by Xinijlefu.

Also, since the sliding surfaces are decoupled into two actuated and unactuated angular components, the strategy of simply using linear combination of these two sliding surfaces to combine into the top-level sliding surface has not yet worked well for the system using the *hybrid approach* since all angles are intercorrelated in certain ways. Nevertheless, it has great potential for further improvements. Such investigation is beyond the scope of this thesis. As expected, according to the simulation results, the top-level sliding surface can be brought to zero easily and in achieving so, the system can be always brought to the vicinity of the upright equilibrium by a reasonable choice of P and Q weighting matrix pairs.

7.3 Nonlinear Second-Order Sliding Mode Control

With second order SM control, the derivative of the sliding surface can be brought to zero in addition to the sliding surface itself in finite time. As a result, it offers clear advantages in both robustness and high accuracy of resulting motions [16]. In this section, we design and simulate a *Second-Order Optimal Sliding Mode Controller* (SOOSMC).

7.3.1 Partial Feedback Linearization

Due to the positive definiteness the generalized inertia matrix, the entire class of underactuated mechanical systems possesses the so-called collocated partial feedback linearization property [11].

Collocated linearization refers to control that linearizes the equations associated with the actuated DOF, whereas non-collocated partial feedback linearization refers to control that linearizes the equations associated with the unactuated DOF, which is only possible under a special assumption of the generalized inertia matrix. We consider both the collocated linearization and the non-collocated linearization in this thesis. Referring to Eq. (4.1), the Euler-Lagrangian equations of the underactuated spatial double inverted pendulum can be written as:

$$M_{11}(q)\ddot{q}_u + M_{12}(q)\ddot{q}_a + C_u(q,\dot{q}) + G_u(q) = \mathbf{0} \tag{7.11a}$$

$$M_{21}(q)\ddot{q}_u + M_{22}(q)\ddot{q}_a + C_a(q,\dot{q}) + G_a(q) = \tau_a \tag{7.11b}$$

where the matrices and vectors are partitioned as

$$M(q) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \qquad C(q,\dot{q})\dot{q} = \begin{bmatrix} C_u(q,\dot{q}) \\ C_a(q,\dot{q}) \end{bmatrix}$$

$$G(q) = \begin{bmatrix} G_u(q) \\ G_a(q) \end{bmatrix}, \qquad q_u = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \quad q_a = \begin{bmatrix} q_3 \\ q_4 \end{bmatrix}$$

Collocated Partial Feedback Linearization

In Eq. (7.11a), the 2×2 block matrix $M_{11}(q)$ is invertible, as $M(q)$ is positive definite, then $\ddot{q}_u$ can be solved as

$$\ddot{q}_u = -M_{11}^{-1}M_{12}\ddot{q}_a - M_{11}^{-1}C_u(q,\dot{q}) - M_{11}^{-1}G_u(q) \tag{7.12}$$

When substituting Eq. (7.12) into Eq. (7.11b) we obtain

$$(M_{22} - M_{21}M_{11}^{-1}M_{12})\ddot{q}_a + (C_a - M_{21}M_{11}^{-1}C_u) + (G_a - M_{21}M_{11}^{-1}G_u) = \tau_a \tag{7.13}$$

Let us redefine the following:

$$\overline{M}_a = M_{22} - M_{21}M_{11}^{-1}M_{12}$$
$$\overline{C}_a = C_a - M_{21}M_{11}^{-1}C_u$$
$$\overline{G}_a = G_a - M_{21}M_{11}^{-1}G_u$$

then $\overline{M}_a$ is positive definite since $M(q)$ is positive definite. Eq. (7.13) can be written as

$$\overline{M}_a\ddot{q}_a + \overline{C}_a + \overline{G}_a = \tau_a \tag{7.14}$$

If we choose a new control input v such that

$$\tau_a = \overline{M}_a v + \overline{C}_a + \overline{G}_a \tag{7.15}$$

then Eq. (7.14) is partially feedback linearized to

$$\ddot{q}_a = v \tag{7.16}$$

together with Eq. (7.11a), the complete system can be described by

$$\begin{aligned} M_{11}(q)\ddot{q}_u + C_u(q,\dot{q}) + G_u(q) = &- M_{12}(q)v \\ \ddot{q}_a = &\ v \end{aligned} \tag{7.17}$$

Non-Collocated Partial Feedback Linearization

Similarly, in Eq. (7.11a), the 2×2 block matrix $M_{12}(q)$ seems to be invertible.[1] Whenever it is invertible, $\ddot{q}_a$ can be solved as:

$$\ddot{q}_a = -M_{12}^{-1}M_{11}\ddot{q}_u - M_{12}^{-1}C_u(q,\dot{q}) - M_{12}^{-1}G_u(q) \tag{7.18}$$

After substituting Eq. (7.18) into Eq. (7.11b) we eventually obtain:

$$\overline{M}_u\ddot{q}_u + \overline{C}_u + \overline{G}_u = \tau_a \tag{7.19}$$

where

$$\begin{aligned} \overline{M}_u &= M_{21} - M_{22}M_{12}^{-1}M_{11} \\ \overline{C}_u &= C_a - M_{22}M_{12}^{-1}C_u \\ \overline{G}_u &= G_a - M_{22}M_{12}^{-1}G_u \end{aligned}$$

[1] with the exception of a set of states most probably of measure zero in $\Re^4$, as the singularity of $M_{12}(q)$ has not been encountered in simulations.

If we choose a new control input u such that

$$\tau_a = \overline{M}_u u + \overline{C}_u + \overline{G}_u \tag{7.20}$$

then Eq. (7.19) is partially feedback linearized to

$$\ddot{q}_u = u \tag{7.21}$$

together with Eq. (7.11a), the complete system can be described by

$$\begin{aligned} M_{12}(q)\ddot{q}_a + C_u(q,\dot{q}) + G_u(q) &= -M_{11}(q)u \\ \ddot{q}_u &= u \end{aligned} \tag{7.22}$$

7.3.2 Controller Design with Twisting Algorithm

Suppose we want to control only the unactuated joint angles q_u, then let us define the sliding surface:

$$s = q_u \tag{7.23}$$

The system (7.19) has a relative degree of two w.r.t. the sliding surface s, so the 2-sliding point set is defined as:

$$s = \dot{s} = 0 \tag{7.24}$$

the controller u can be designed using the so-called twisting algorithm to force the trajectories of the system to the 2-sliding manifold in finite time:

$$\begin{aligned} u &= -K_1 s - K_2 \dot{s} - K_3 \mathrm{sgn}(s) - K_4 \mathrm{sgn}(\dot{s}) \\ &= -K_1 q_u - K_2 \dot{q}_u - K_3 \mathrm{sgn}(q_u) - K_4 \mathrm{sgn}(\dot{q}_u) \end{aligned} \tag{7.25}$$

where $K_i, i = 1, ..., 4$ are positive definite diagonal matrices. Similarly, the controller

$$v = -K_1 q_a - K_2 \dot{q}_a - K_3 \mathrm{sgn}(q_a) - K_4 \mathrm{sgn}(\dot{q}_a) \tag{7.26}$$

will drive the actuated joint angles to zero in finite time.

7.3.3 Optimal Controller Design

Our purpose is to combine the controllers u and v in some optimal way in an attempt to stabilize all angles, from Eq. (7.15) and Eq. (7.20), we obtain:

$$\overline{M}_u^{-1}\tau_a - \overline{M}_u^{-1}(\overline{C}_u + \overline{G}_u) + u = 0 \tag{7.27a}$$
$$\overline{M}_a^{-1}\tau_a - \overline{M}_a^{-1}(\overline{C}_a + \overline{G}_a) + v = 0 \tag{7.27b}$$

stacking up the above two equations assuming the same controller is used, we obtain:

$$\begin{bmatrix} \overline{M}_u^{-1} \\ \overline{M}_a^{-1} \end{bmatrix} \tau_a - \begin{bmatrix} \overline{M}_u^{-1}(\overline{C}_u + \overline{G}_u) + u \\ \overline{M}_a^{-1}(\overline{C}_a + \overline{G}_a) + v \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

As we can see, it is impossible to find a controller that can simultaneously achieve objectives of controlling both q_u and q_a unless $[\overline{M}_u^{-1}(\overline{C}_u + \overline{G}_u) + u; \overline{M}_a^{-1}(\overline{C}_a + \overline{G}_a) + v]$ is in the range of $[\overline{M}_u^{-1}; \overline{M}_a^{-1}]$. Since the 4×2 matrix $[\overline{M}_u^{-1}; \overline{M}_a^{-1}]$ does not have an inverse, direct computation of τ_a is not feasible. However, we can construct an optimal control problem by first introducing an OR α, which is a relative weight of controlling q_a vs. q_u, now rewrite the augmented system as:

$$\begin{bmatrix} \overline{M}_u^{-1} \\ \alpha\overline{M}_a^{-1} \end{bmatrix} \tau_a - \begin{bmatrix} \overline{M}_u^{-1}(\overline{C}_u + \overline{G}_u) + u \\ \alpha(\overline{M}_a^{-1}(\overline{C}_a + \overline{G}_a) + v) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$
$$\triangleq B\tau_a - c = \mathbf{0} \tag{7.28}$$

with $B \triangleq [\overline{M}_u^{-1}; \alpha\overline{M}_a^{-1}]$, $c \triangleq [\overline{M}_u^{-1}(\overline{C}_u + \overline{G}_u) + u; \alpha(\overline{M}_a^{-1}(\overline{C}_a + \overline{G}_a) + v)]$
Now the problem is transformed into finding τ_a such that $\|c - B\tau_a\|$ is minimized, therefore:

$$\tau_{modified} = B^+ c \tag{7.29}$$

where B^+ is the Moore-Penrose pseudoinverse of B.

7.3.4 Simulation Results

In the following examples, the SOSM controller with twisting algorithm is used to perform the swing-up of the system. After some tunings, the gain parameters chosen are: $K_1 = diag(20, 20), K_2 = diag(7, 7), K_3 = diag(0.7, 0.7)$ *and* $K_4 = diag(0.2, 0.2)$.

Example 13 To show the effectiveness of the twisting algorithm, we let $q_0 = [5°, 0°, -3°, 0°]^\top$, $\dot{q}_0 = \mathbf{0}$, and only uses the unactuated angular positions as the sliding surface.

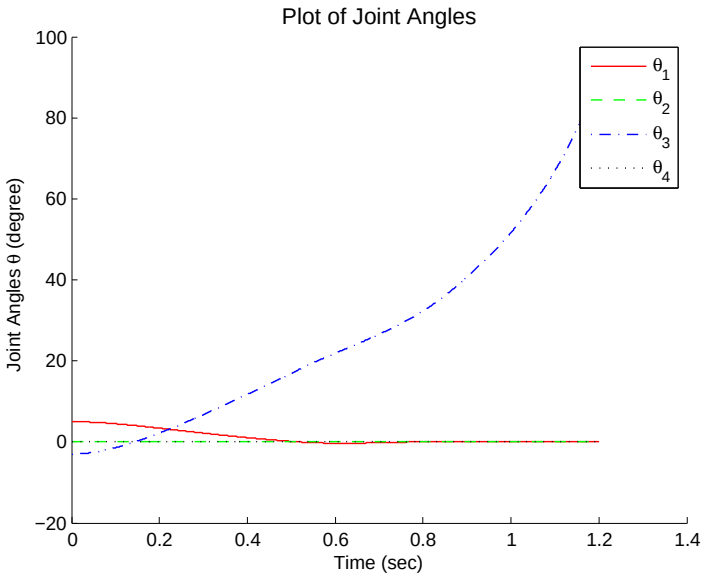

Fig. 7.17 Example 13. Change of the joint angles w.r.t. time

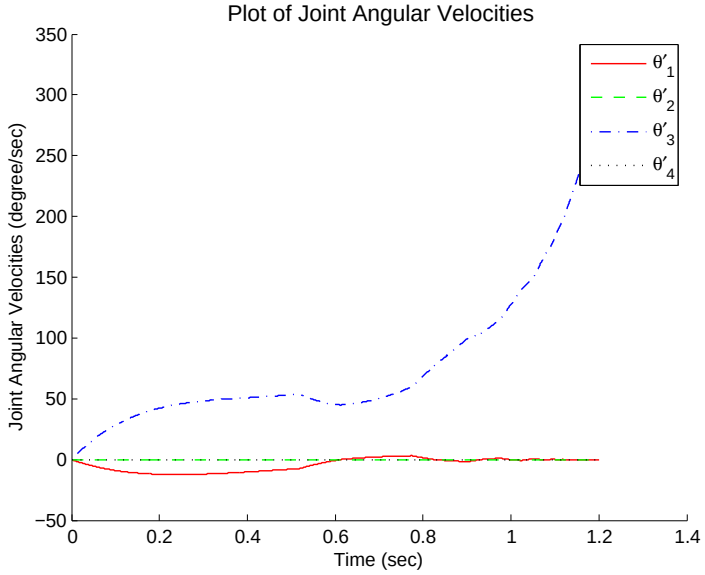

Fig. 7.18 Example 13. Change of joint angle rates w.r.t. time

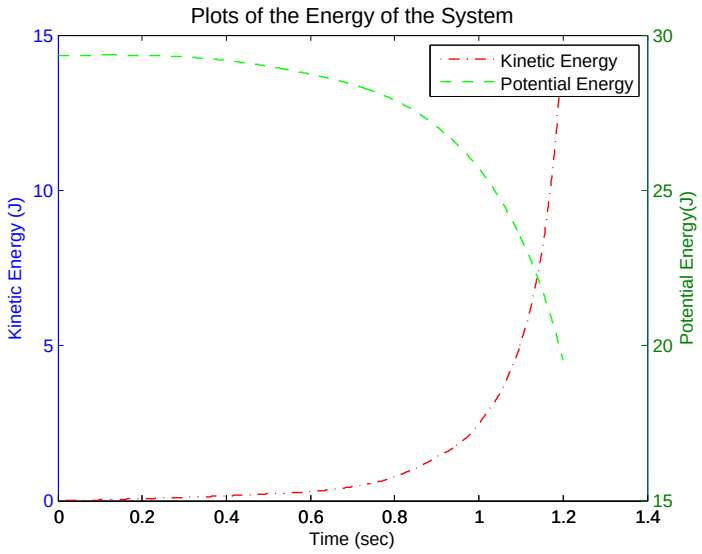

Fig. 7.19 Example 13. Change of kinetic and potential energies w.r.t. time

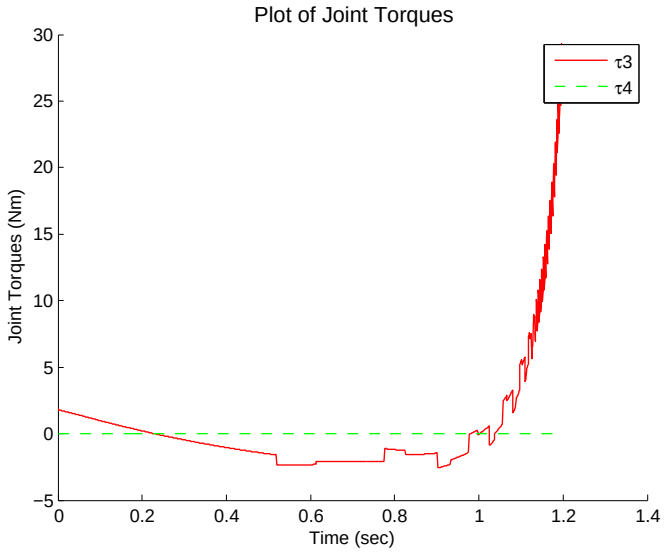

Fig. 7.20 Example 13. Change of Applied Torques

As expected, the controller is able to drive both the position and velocity of the unactuated angle θ_1 to zero in less than 1 second while the actuated angle θ_3 was not taken care of and moved away from the vicinity of equilibrium position with an accelerated rate.

Example 14 To see how different the system behaves by applying the SOOSMC, we use $OR = 0.3$ and the same IC as in Example 13: $q_0 = [5°, 0°, -3°, 0°]^\top$, $\dot{q}_0 = \mathbf{0}$.

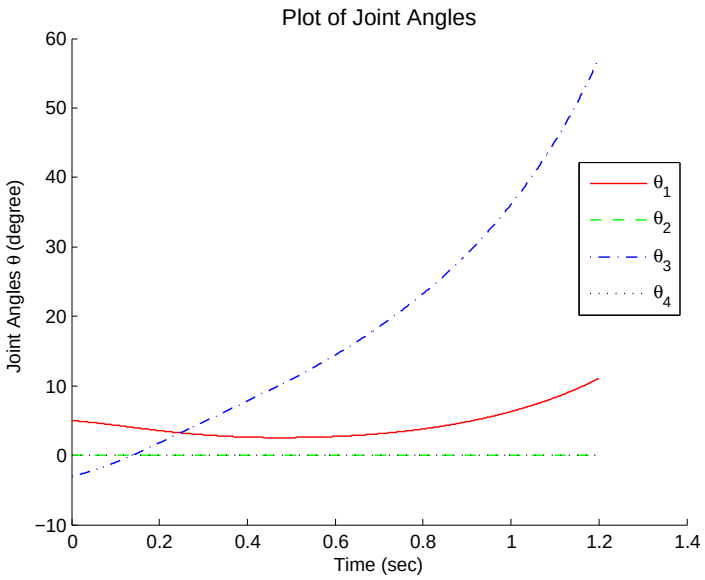

Fig. 7.21 Example 14. Change of the joint angles w.r.t. time

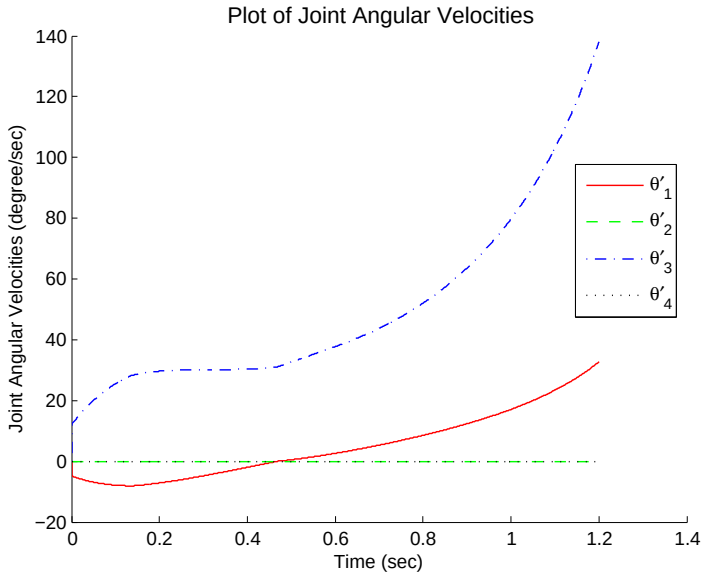

Fig. 7.22 Example 14. Change of joint angle rates w.r.t. time

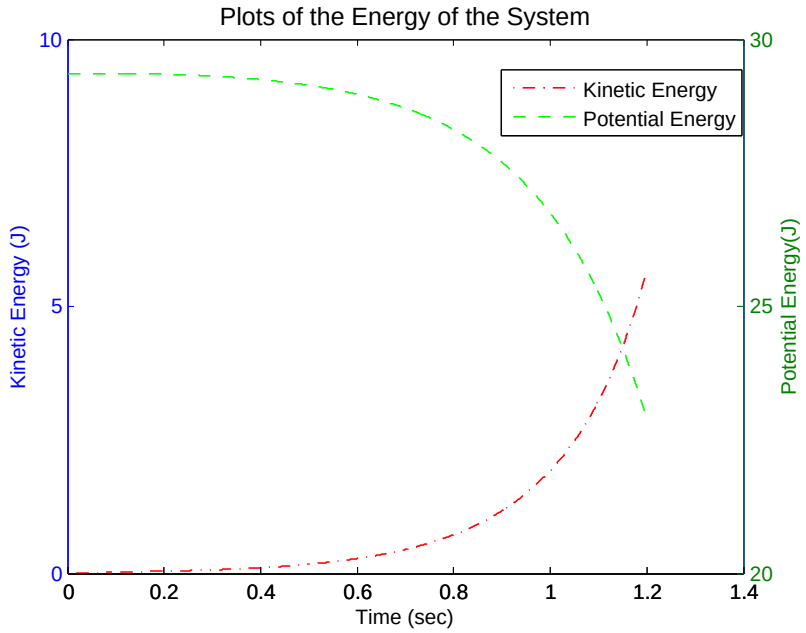

Fig. 7.23 Example 14. Change of kinetic and potential energies w.r.t. time

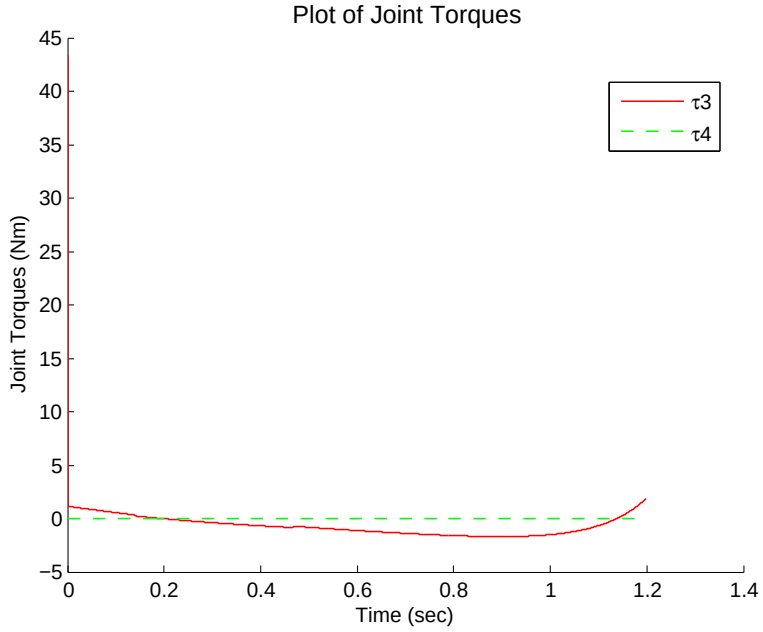

Fig. 7.24 Example 14. Change of Applied Torques

Although the controller with OR of two could not simultaneously take care of the stabilization of both angles, the control effort has been divided so both angles are under some control, since actuated angle θ_3 has only gone up to 50° by 1.2 seconds compared to 80° in Example 13, see Fig (7.17) vs Fig (7.21). It is also noted that with the optimal control approach, the control torque and the total kinetic energy did not increase as much as those in the earlier example during the first 1.2 seconds, see Fig (7.19) vs Fig (7.23) as well as well as Fig (7.20) vs Fig (7.24).

Although the two angles do not seem to cooperate well with each other in this example, we have been able to see what the optimal control approach can offer. Its benefits would not be realized unless the IC moves further away from the upright equilibrium position, as the two examples that follow will show.

Example 15 SOOSMC with $OR = 2$ with $q_0 = [10°, -10°, -30°, 30°]^\top$, $\dot{q}_0 = \mathbf{0}$, the same IC as in Example 11 with FOHSMC.

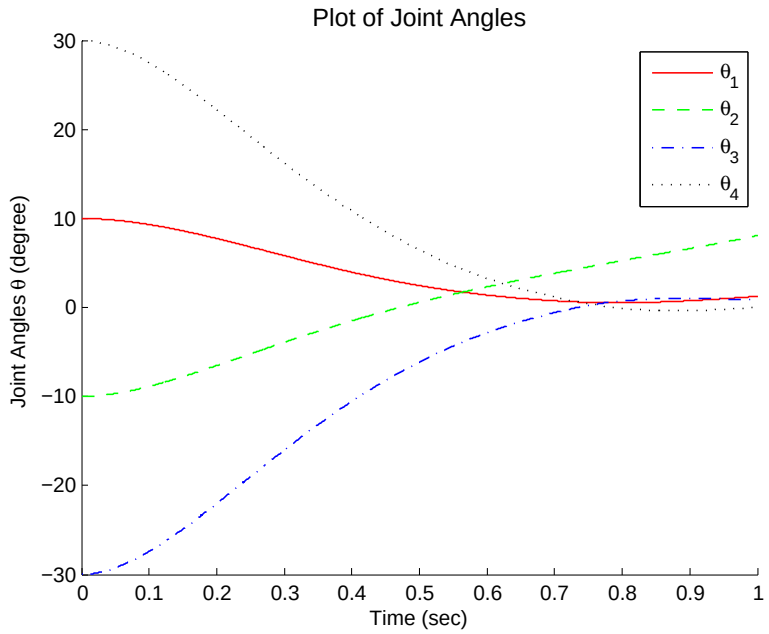

Fig. 7.25 Example 15. Change of the joint angles w.r.t. time

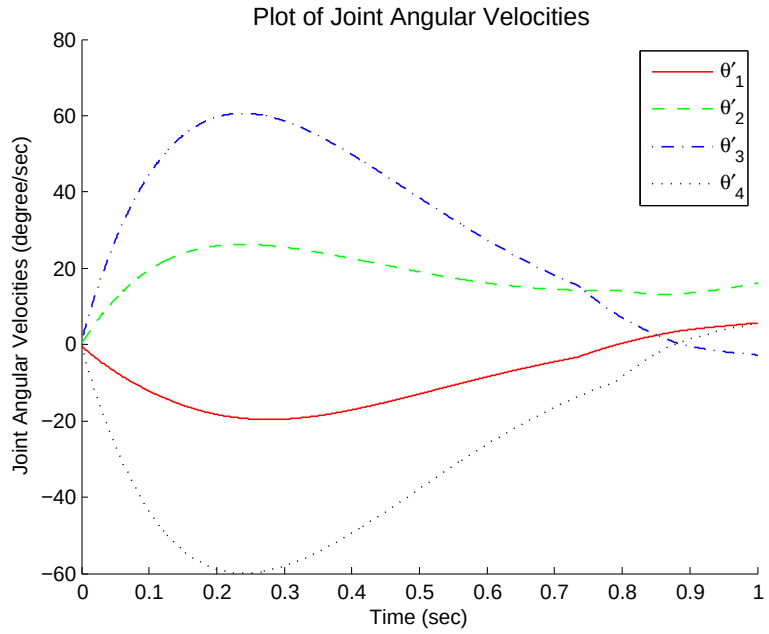

Fig. 7.26 Example 15. Change of joint angle rates w.r.t. time

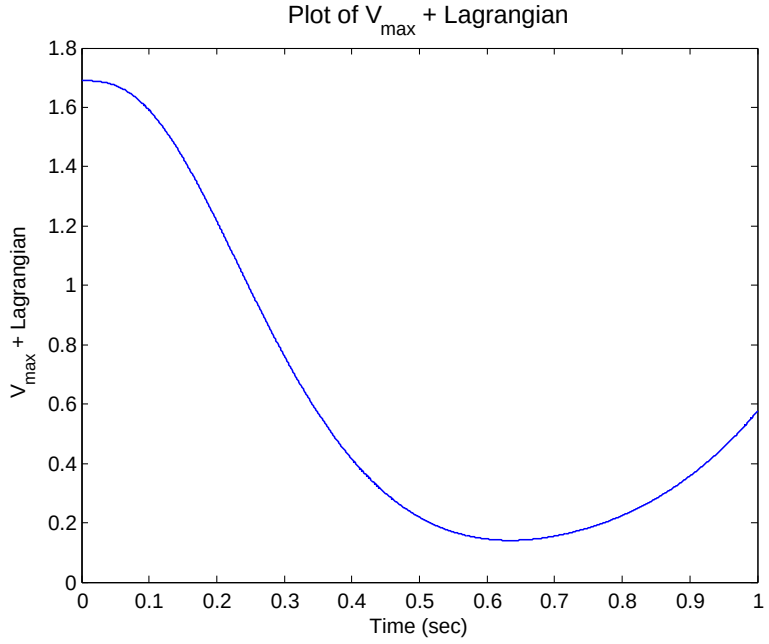

Fig. 7.27 Example 15. Change of V_{max} + Lagrangian w.r.t. time

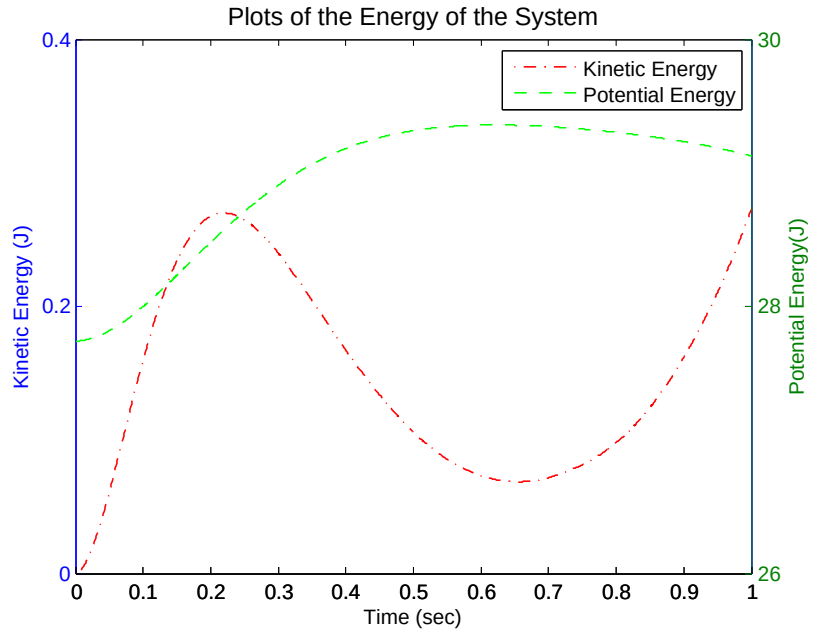

Fig. 7.28 Example 15. Change of kinetic and potential energies w.r.t. time

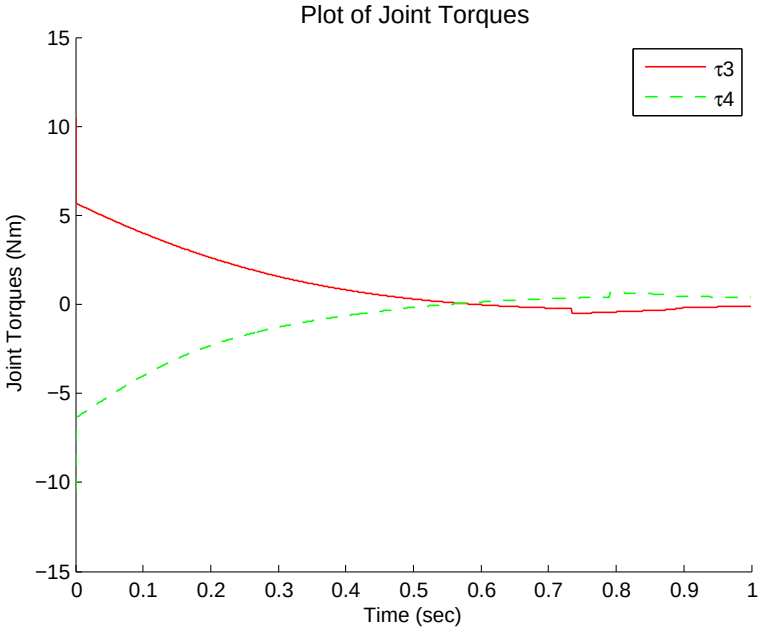

Fig. 7.29 Example 15. Change of Applied Torques

Example 16 SOOSMC with $OR = 0.8$ and the same IC as in Example 12: $q_0 = [30°, 20°, -135°, -25°]^\top$, $\dot{q}_0 = \mathbf{0}$.

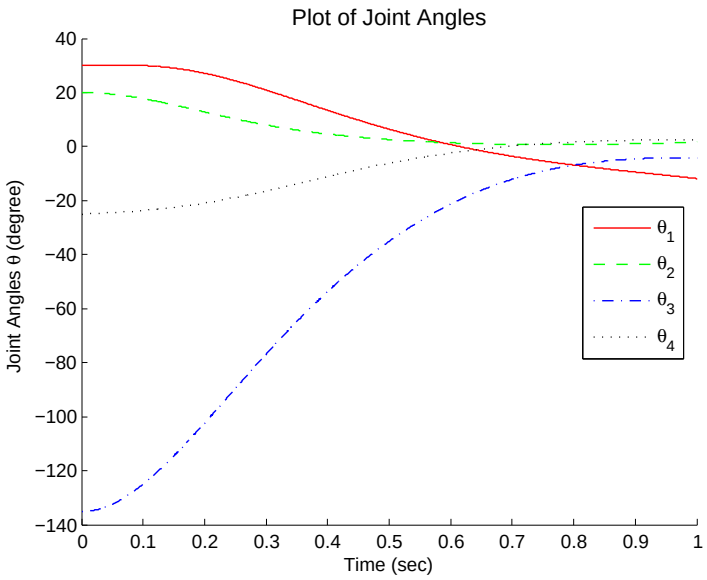

Fig. 7.30 Example 16. Change of the joint angles w.r.t. time

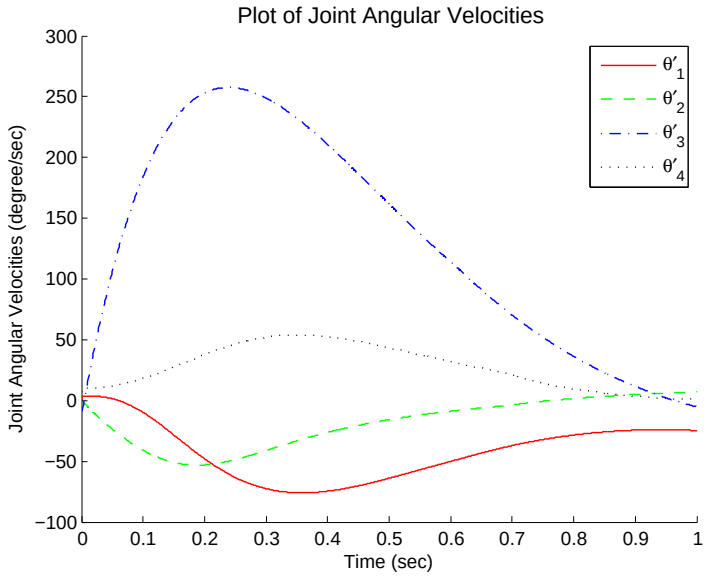

Fig. 7.31 Example 16. Change of joint angle rates w.r.t. time

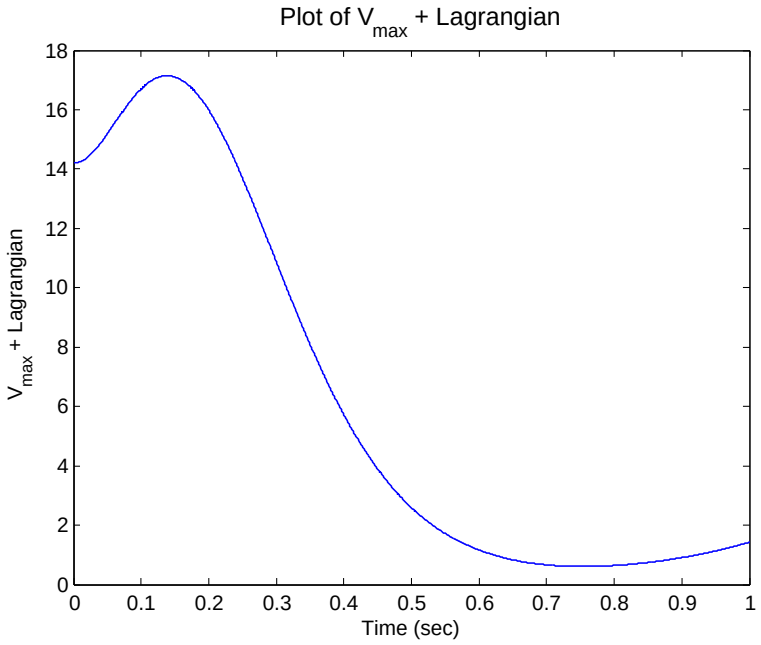

Fig. 7.32 Example 16. Change of V_{max} + Lagrangian w.r.t. time

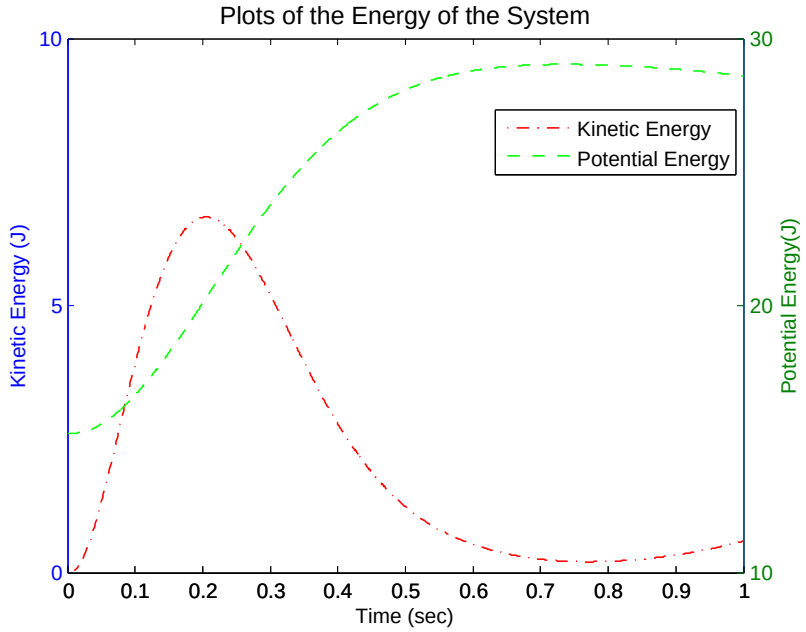

Fig. 7.33 Example 16. Change of kinetic and potential energies w.r.t. time

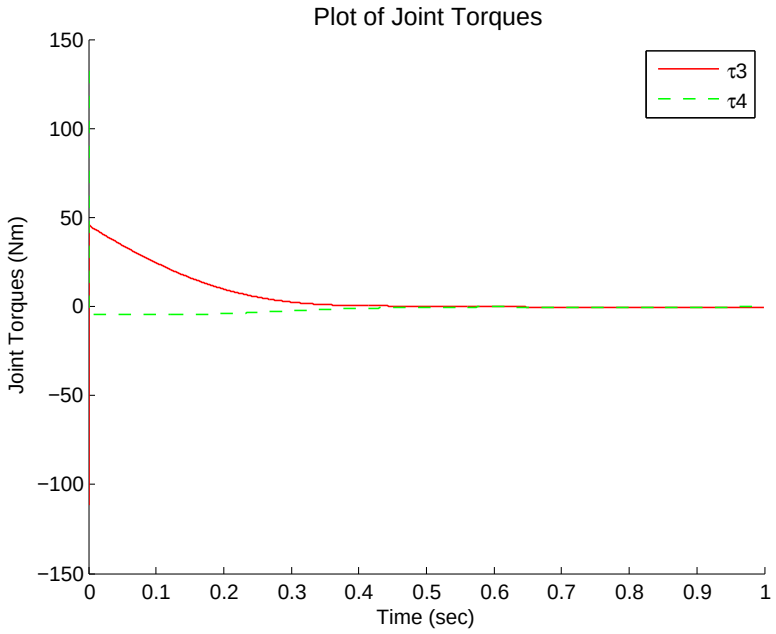

Fig. 7.34 Example 16. Change of Applied Torques

7.3.5 Discussions

With $q_0 = [10°, -10°, -30°, 30°]^\top$ and $q_0 = [30°, 20°, -135°, -25°]^\top$, both the FOHSMC and the SOOSMC produced similar results in terms of the swing-up behaviours and and it has been easily noted that the switching condition based on the value of $V_{max} + Lagrangian$ had been met for both scenarios. It has also been observed that the SOOSMC is able to perform slightly better than the FOHSMC along the transient trajectories. (See Fig. (7.4) vs Fig. (7.25) as well as Fig. (7.12) vs Fig. (7.30)). Yet, after extensive simulations by varying the ICs, neither controller has been able to successfully switch to the linear controllers (either LQR or SM) in order to render the system asymptotically stable.

In all, both the FOHSMC and SOOSMC are able to bring the system from a configuration far from the ROC of the linear controller to a region quite close to the unstable upright equilibrium. However, it has not been possible to determine a sufficient condition to switch to the linear controller in order to render the system stable. In addition, one major limitation of the designed nonlinear SM controllers is that the controller parameters must be properly tuned for the given IC in order to swing up the system in the desired

manner. Therefore, we conclude that in order for the *hybrid approach* to succeed with the current design of the SM swing-up controller, detailed investigation on the *sufficient* switching condition is needed and further computationally expensive tunings for the swing-up controller would also be necessary for this purpose.

Chapter 8

Conclusions and Future Research Directions

In this thesis, the SDIP system is considered to be a model that is well fit to study the phenomenon of the standing postural control because it is simple, yet captures the key kinematic and dynamics features of a standing animal or machine. The LQR controller worked well when the system starts off at a position relatively close to the unstable upright equilibrium. However, this "simple" system is found to be difficult to control globally, not only due to the spatial nature of the system but also because this 4-DOF system is under-actuated with control deficiency of two.
In summary, the main achievements in this thesis include:

1. A proper design of the LQR controller for the local stabilization of the SDIP system and an exploration of its ROC.

2. A successful implementation of the STA controllers as superior alternatives to LQR.

3. Successful extensions of existing 2-D versions of nonlinear SM control methods to 3-D to perform swing-up of the system.

While we are encouraged by the above accomplishments, the objective of global stabilization requires more work. The problem remains to be solved in the future, and the following possible approaches could be envisaged:

- Redefining the nonlinear SM surface using momentum control w.r.t. the foot [21] for the purpose of swing-up.

- Developing an adaptive control algorithm to tune the weighting matrices in an on-line fashion for the FOHSMC or SOOSMC.
- Discovery of a *sufficient* switching condition from the swing-up controller to the LQR.

After a successful controller design, it must be adapted to work on the real physical plant. Currently, the motor in the already built prototype could only support up to 0.5Nm. As a result, it is preferable for the theoretically designed algorithm to minimize the gains of the controller. In addition, an observer can be designed to avoid state feedback in order to reduce the number of sensors required.

References

[1] G. Sood, "Simulation and control of a hip actuated robotic model for the study of human standing posture," Master's thesis, Department of Electrical and Computer Engineering, McGill University, 2008. 2, 3, 4

[2] V. Utkin, J. Guldner, and J. Shi, *Sliding Mode Control in Electro-Mechanical Systems, Second Edition*. CRC Press, 2009. 3, 7

[3] W. Perruquetti and J.-P. Barbot, *Sliding Mode Control in Engineering*. CRC Press, 2002. 4, 34

[4] L. Wiklendt and S. K. Chalup, "Balance control of a simulated inverted pendulum on a circular base," in *Australasian Conference on Robotics and Automation (ACRA)*, 2009. 4

[5] E. Lee and J. Perkins, "Comparison of techniques for stabilization of a triple inverted pendulum." 4

[6] Xinjilefu, V. Hayward, and H. Michalska, "Hybrid stabilizing control for the spatial double inverted pendulum," in *"International Symposium on Brain, Body and Machine", Springers series in Advances in Intelligent and Soft Computing, Angeles, J., Boulet, B., Clark, J., Kovecses, J. and Siddiqi, K. (Eds.), in press*, 2010. 4, 6, 58, 59, 71

[7] S. Riachy, Y. Orlov, T. Floquet, R. Santiesteban, and J.-P. Richard, "Second-order sliding mode control of underactuated mechanical systems i: Local stabilization with application to an inverted pendulum," *International Journal of Robust and Nonlinear Control*, vol. 18, pp. 529–543, 2008. 4

[8] A. Davila, J. Moreno, and L. Fridman, "Variable gains super-twisting algorithm: A lyapunov based design," in *2010 American Control Conference (ACC)*, July 2010, pp. 968 –973. 4, 7, 32, 35

[9] M. Spong, "Swing up control of the acrobot," in *Proceedings of 1994 IEEE International Conference on Robotics and Automation*, May 1994, pp. 2356 –2361 vol.3. 5, 58

[10] ——, "The swing up control problem for the acrobot," *Control Systems Magazine, IEEE*, vol. 15, no. 1, pp. 49 –55, Feb 1995. 5

[11] ——, "The control of underactuated mechanical systems," in *Plenary lecture at the First International Conference on Mechatronics, Mexico City*, 1994. 5, 71

[12] T. Yonemura and M. Yamakita, "Swing up control of acrobot based on switched output functions," in *SICE 2004 Annual Conference*, vol. 3, Aug. 2004, pp. 1909 – 1914. 5, 6

[13] T. Henmi, M. Deng, and A. Inoue, "Swing-up control of the acrobot using a new partial linearization controller based on the lyapunov theorem," in *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC '06)*, 2006, pp. 60 –65. 5

[14] M. Berkemeier and R. Fearing, "Tracking fast inverted trajectories of the underactuated acrobot," vol. 15, no. 4, Aug. 1999, pp. 740 –750. 5

[15] F. Mnif, "Vss control for a class of underactuated systems," *International Journal of Computational Cognition*, vol. 3, p. 2, June 2005. 5

[16] S. Puga and L. Aguilar, "Hybrid second-order sliding-mode tracking control for acrobot," in *International Conference on Industrial Electronics and Control Applications (ICIECA 2005)*, 2005, p. 6. 5, 58, 71

[17] M.-S. Park, D. Chwa, and S.-K. Hong, "Decoupling control of a class of underactuated mechanical systems based on sliding mode control," in *SICE-ICASE, 2006. International Joint Conference*, Oct. 2006, pp. 806 –810. 6

[18] W. Wang, J. Yi, D. Zhao, and X. Liu, "Double layer sliding mode control for second-order underactuated mechanical systems," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, Aug. 2005, pp. 295 – 300. 6

[19] D. Qian, J. Yi, and D. Zhao, "Hierarchical sliding mode control to swing up a pendubot," in *American Control Conference, 2007.*, July 2007, pp. 5254 –5259. 6, 7, 59, 62

[20] Y. Hao, J. Yi, D. Zhao, and D. Qian, "Design of a new incremental sliding mode controller," in *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, June 2008, pp. 3407 –3412. 6

[21] N. Miyashita, M. Kishikawa, and M. Yamakita, "3d motion control of 2 links (5 d.o.f.) underactuated manipulator named acrobox," in *American Control Conference, 2006*, June 2006, p. 6. 6, 87

[22] X. Xinjilefu, V. Hayward, and H. Michalska, "Stabilization of the spatial double inverted pendulum using stochastic programming seen as a model of standing postural control," in *Proc. 9th RAS International Conference on Humanoid Robots (Humanoids09)*, Dec. 2009, pp. 367 –372. 6

[23] X. Xinjilefu, "Stabilization of the spatial double inverted pendulum," Master's thesis, Department of Electrical and Computer Engineering, McGill University, 2010. 9, 11, 13

[24] R. M. Murray and S. S. Sastry, "Nonholonomic motion planning: steering using sinusoids," *IEEE Transactions on Automatic Control*, vol. 38, no. 5, pp. 700–716, 1993. 11

[25] J. Marsden and T. Ratiu, *Introduction to mechanics and symmetry: a basic exposition of classical mechanical systems.* Springer Verlag, 1999. 14