Stabilization of the Acrobot via Multiple sliding surface Control

Nadeem Qaiser[1], Naeem Iqbal[2], Naeem Qaiser[3]
[1,2]Dept. of Electrical Engineering, [3]Dept of Computer and Information Sciences
[1,2,3]Pakistan Institute of Engineering and Applied Sciences
Nilore Islamabad
Pakistan

Abstract: - This paper considers the stabilization problem of the Acrobot, a widely studied benchmark nonlinear under-actuated mechanical system. For such systems the design of control Law becomes a challenging task owing to complex internal dynamics and lack of feedback linearizibility. These result in need of closed form solutions for highly nonlinear equations or hybrid/switching controllers. A novel nonlinear controller design, using recently introduced Multiple Sliding Surface Control technique is presented as the solution. Proposed controller doesn't require analytical calculation of certain derivatives requiring closed form solutions of highly nonlinear equations. The proposed design procedure is shown to be simpler and more intuitive than existing designs. Advantages over conventional Energy Shaping and Backstepping controllers are analyzed theoretically and verified using numerical simulations.

Key-Words: - Acrobot, Under Actuated Mechanical Systems, Multiple Sliding Surface Control, backstepping, singular perturbation model

1 Introduction

Acrobot is a planar robot that mimics the human acrobat who hangs from a bar and tries to swing up to a perfectly balanced upside-down position with his/her hands still on the bar.

The Acrobot, having a rich research past, first introduced and studied by Murray and Hauser [1], is a benchmark nonlinear under actuated mechanical system. These are mechanical control systems with fewer actuators (i.e. controls) than configuration variables or degrees of freedom. The Acrobot has been a test bed mainly for Energy Shaping and Damping Injection based approaches. Normally a supervisory hybrid/switching control strategy is applied to asymptotic stabilization of the system. First a controller (usually nonlinear) swings up the arm. Then, a balancing controller, obtained by Jacobian linearization or (local) exact feedback linearization stabilizes it around its upright position. Because of the large range of the motion the swing-up problem is highly nonlinear in nature, attracting attention of many control designers. Several solutions have been proposed ranging from Pseudo linearization techniques to use of fuzzy controllers and neural networks. We are more interested in classical nonlinear techniques only.

M. W. Spong [2] proposes two distinct design algorithms for swing-up control. One design exploits unstable zero dynamics of the system for swing up while energy-pumping scheme is employed in the other. Global stabilization of Acrobot has also been shown using Integrator Backstepping procedure (IBS) by Reza [3].

Based on results obtained by Sontag and Sussman [4], IBS is a powerful step-by-step design tool. However it suffers not only the problem of "explosion of terms" but also requires certain system functions to be C^n [5]. The control law obtained through a cumbersome design procedure is usually very complicated. Multiple Sliding Surfaces (MSS) control [5], a procedure similar to integrator backstepping, avoids this phenomenon. Although it falls short of integrator backstepping in terms of theoretical rigor, as the need for analytical differentiation is pushed to a numerical one but still is very practical due to its numerical nature easily implement-able by modern digital computers. It is simple, more intuitive and applies to a more general class of systems as the requirement on nonlinear function is to be C^1 only.

Concept of Dynamic Surface Control (DSC), a dynamic extension to MSS, introduced by Swaroop et al. [6] resolves these issues by using low pass filters but it augments system dynamics and some time is not practical due to unfeasible filter component values.

Multiple Sliding Surface control technique is used to stabilize Acrobot demonstrating the design method simplicity that also results in a less complicated control law. The design uses MSS technique to track the required stabilization function for the unstable zero dynamics. The design procedure and control law is simpler than IBS design and doesn't require a supervisory controller like the one by Spong et al. [2].

The paper starts formally with Section 2, containing the dynamic model for the Acrobot. Here necessary coordinate transformations are also given, as the dynamic model is not in a control design amenable form. Controller design strategy and procedure appear in section 3. Section 4 presents simulation results comparing controller performance to existing designs followed by brief concluding remarks in Section 5.

2 Dynamical Model

The Acrobot as illustrated in Figure 1 is an underactuated two-link planar robot with only one actuator at elbow. The controller task is to stabilize the Acrobot to any of its arbitrary equilibrium points particularly the upright equilibrium position.

Dynamic model of Acrobot can be obtained easily by using Euler Lagrange method [7] for a two link planar robot with the input torque absent for Link 1. The Lagrangian is given as

$$L(q,\dot{q}) = \frac{1}{2}\dot{q}^T M\dot{q} - V(q) \tag{1}$$

where $(q,\dot{q})$ are generalized coordinates and quasi velocities, M the inertia matrix and $V(q_1)$ is the potential energy function given as

$$V(q_1,q_2) = Ag\cos(q_1) + B\cos g(q_1+q_2)$$
$$A = m_1 l_1 + m_2 L_1$$
$$B = m_2 l_2$$

Using configuration variables as shown in Figure 1 and $\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = Q(q)u$ yields the equations of motion

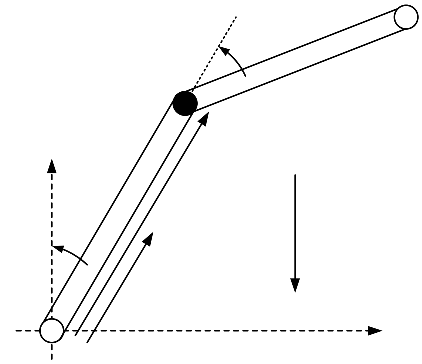

Figure 1 The Acrobot

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} h_1+\phi_1 \\ h_1+\phi_1 \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix} \tag{2}$$

The element of Inertia matrix are given by

$$m_{11}(q_2) = a + b\cos(q_2)$$
$$m_{12}(q_2) = c + (b/2)\cos(q_2)$$
$$m_{22}(q_2) = c$$

with

$$a = m_1 l_1 + m_2(L_1^2 + l_2^2) + I_1 + I_2$$
$$b = 2m_1 L_1 l_2$$
$$c = m_2 l_2^2 + I_2$$

There is no need for the definition of h_i and ϕ_i in design procedure; interested readers may find the same in [2].

Lagrangian model of Acrobot possesses certain very interesting properties, for instance note that the inertial matrix M depends only on q_2 in contrast to potential energy function. Thus Acrobot does posses kinetic symmetry with respect to q_1 in spite of not possessing symmetry in classical sense

The state space model of Acrobot is unsuitable for direct application of MSS for not being in strict feedback form [5]. Feedback Linearization [8] employs a change of control and co-ordinate transformation, which leaves the system dynamics linear or at least partially linear, more amenable to control. It has been shown in [1] that Acrobot cannot be fully linearized with static state feedback and coordinate transformations. Spong [2] proposes separate linearization schemes for each link, leading to two different control schemes. We prefer the coordinate transformation proposed by Reza [9]. As based upon theses transformation a globally asymptotically stabilizing controller [3] for the reduced dynamics already exists greatly reducing the design labor.

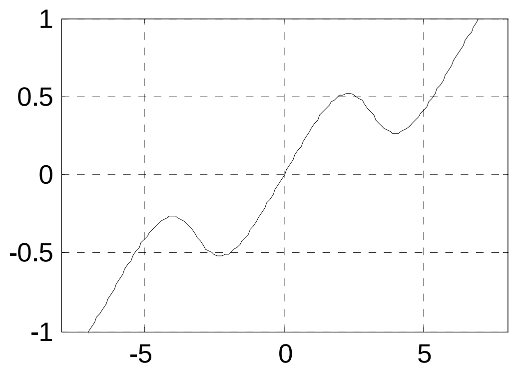

Figure 2 Plot q_2 vs function $\gamma(q_2)$.

Based on Theorem 1[9] the following change of coordinates

$$\begin{aligned} q_r &= q_1 + \gamma(q_2) \\ p_r &= m_{11}(q_2)\,p_1 + m_{12}(q_2)\,p_2 \end{aligned} \tag{3}$$

where

$$\gamma(q_2) = \int_0^{q_2} \frac{m_{12}(s)}{m_{11}(s)}\, ds$$

$\gamma(q_2)$ is a nonlinear function and is plotted in figure 2. This transforms the Acrobot dynamics to a cascade nonlinear system in strict feedback form.

$$\begin{aligned} \dot{q}_r &= p_r / m_{11}(q_2) \\ \dot{p}_r &= g_r(q_r, q_2) \end{aligned} \tag{4}$$

$$\begin{aligned} \dot{q}_2 &= p_2 \\ \dot{p}_2 &= u \end{aligned} \tag{5}$$

with

$$\gamma(q_2) = \frac{q_2}{2} + \left(\frac{2c-a}{\sqrt{a^2-b^2}} \right) \arctan\left(\sqrt{\frac{a-b}{a+b}} \tan\left(\frac{q_2}{2} \right) \right) \tag{6}$$

and

$$\begin{aligned} g_r(q_r, q_2) &= Ag \sin(q_r - \gamma(q_2)) \\ &\quad + B \sin(q_r - \gamma(q_2) + (\frac{q_2}{2})) \end{aligned} \tag{7}$$

Remark 1: System after coordinate transformation is a cascaded interconnection of a nonlinear, Core or Reduced subsystem and a linear double integrator subsystem.

Remark 2: Spong et al. [2] using a standard method from [8] perform partial feed feedback linearization of the system. Afterwards two schemes are proposed but both are for swing up action only. For stabilization switching to a locally linear controller is needed. Reza [3] proposes a standard IBS based design that avoid switching to a second controller but when we revisit the design for the core system we notice it requires solutions of certain very difficult nonlinear analytical equations. Besides it entails propagation of derivatives causing explosion of terms resulting in a very complicated control law through a very cumbersome design procedure. Our design approach *doesn't require such solutions in closed forms* besides employing a simple design algorithm.

3 Controller design

DSC technique is not applicable in usual fashion as the core system is non-affine in control. Thus first assuming q_2 as the virtual input a stabilization function is found for the core subsystem. Afterwards MSS technique is used to design a u forcing q_2 to track the required stabilization function, ultimately stabilizing the total system. MSS is chosen as it has not only nice trajectory tracking feature with arbitrarily small bounded error but it also doesn't exhibits the phenomenon of explosion of terms associated with IBS.

3.1 Core Subsystem Controller Design

The equations of the core system are highly nonlinear and the virtual input appears in a very complicated manner. A static feedback in the explicit form

$$v = \alpha(q_r) \pm a \arctan(c_1 q_r + c_2 p_r) \quad a, c_1, c_2 > 0 \tag{8}$$

exists which globally asymptotically stabilizes the core system where $\alpha(q_r)$ is a smooth function that satisfies

$$g_r(q_r, \alpha(q_r)) = 0 \tag{9}$$

The sign $\pm$ in feedback depends on whether $\nabla_v g_r(q_r, \alpha(q_r)) < 0$ or > 0 . For necessary assumptions and stability proof refer to Appendix A [3].

Remark 3: Reza [3] recommends standard backstepping to complete the design. Theoretically the procedure yields a static feedback that guarantees stability of total system. However practically backstepping requires availability of (8) in closed form for construction of Lyapunov functions for subsequent design that in turn demand explicit solution of (9) in closed form. It is very hard to find such solutions even by symbolic computing engines. Implicit solutions have been found requiring inversions by exotic methods like use of neural networks, splines, and look up tables or other curve fitting approaches providing numerical solutions useless for backstepping. Our scheme doesn't require this solution necessarily to be available in closed form as demonstrated in the following design procedure.

3.2 Outer Subsystem Controller Design

To stabilize (4) q_2 is required to follow the trajectory given as

$$q_{2d} = v(q_r, p_r) = \alpha(q_r) \pm a \arctan(c_1 q_r + c_2 p_r) \qquad (10)$$

Applying MSS technique we design the desired control law that forces the linear system to generate the required stabilization function. It's trivial to verify that required assumptions are satisfied by (5) regarding the system and by (10) regarding the trajectory as

- System is in strict feedback form
- f is C^1 for (4)
- q_{2d} is sufficiently smooth and bounded with bounded derivatives up to the second order in a limited workspace as $\alpha(q_r)$ is smooth and $\arctan(c_1 q_r + c_2 p_r)$ is a sigmoidal function.

Design procedure:

Instead of a closed form solution for (9) our design allows us to use any exotic technique like splines, lookup tables etc. However we prefer a numerical solution for the isolated root $\alpha(q_r)$ for precision reasons. This is done online numerically which modern computers can do easily.

Let the error in generation of stabilization function (10) by q_2 be S_1

$$S_1 := q_2 - q_{2d} \qquad (11)$$

$$\dot{S}_1 = \dot{q}_2 - \dot{q}_{2d} = p_2 - \dot{q}_{2d} \qquad (12)$$

Assuming p_2 as next virtual input p_3 is chosen to drive S_1 to zero.

$$p_{2d} = -K_1 S_1 + \dot{q}_{2d} \quad (K_1 > 0) \qquad (13)$$

Define the second surface as

$$S_2 := p_2 - p_{2d} \qquad (14)$$

$$\dot{S}_2 = \dot{p}_2 - \dot{p}_{2d} = u - \dot{p}_{2d} \qquad (15)$$

Control input u is designed to derive S_2 to zero

$$u = -K_2 S_2 + \dot{p}_{2d} \qquad (K_2 > 0) \qquad (16)$$

This completes the design. The error dynamics can be written as

$$\begin{bmatrix} \dot{S}_1 \\ \dot{S}_2 \end{bmatrix} = \begin{bmatrix} -K_1 & 0 \\ 0 & -K_2 \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} \qquad (17)$$

The Matrix for error system is trivially Hurwitz showing the surface dynamics globally asymptotically stable. As the dynamics system is linear GAS implies global exponential stability, Khalil [10].

It is interesting to see that the control law or its derivative is never required in their analytical forms during design. Notice that the direct calculation of $\dot{q}_{2d}(t)$ and $\dot{p}_{2d}(t)$ required at this step by the conventional backstepping design procedure requires its availability in analytical form, calculation of which has already been seen as a very difficult task. Even if it is available it leads to complexity due to "explosion of terms". Motivated by MSS technique this problem is dealt by numerical differentiation, i.e.

$$\dot{q}_{2d}(n) \approx \frac{q_{2d}(n) - q_{2d}(n-1)}{\Delta T}$$

$$\dot{p}_{2d}(n) \approx \frac{p_{2d}(n) - p_{2d}(n-1)}{\Delta T}$$

The upper bound of error for this calculation is

$$O(h^2) \le M_3 \frac{h^2}{6}$$

where

$$M_3 = \max_{a \le x \le b} \left| f^{(3)}(x) \right|$$

With modern high speed digital electronics the processing speed can be set very high easily as compared to the slowly evolving dynamics of the mechanical system. Thus by keeping $\Delta T := h$ sufficiently small the error can practically is made very close to zero.

This solves the issue of finding closed form solution for (9) which is otherwise available easily using numerical techniques. A comparison to existing designs in [1], [2] and [3] reveals the ease of design and simplicity of obtained control law.

The stability of the system follows from the stability theory of cascaded systems. This requires Globally Lipschitz nature of core system and GAS of the core system [3] that avoids peaking phenomenon, and the exponential stability of the error surface dynamics that are robust to errors from numerical derivative calculation, Khalil [10].

4 Simulation Results

System stability and controller performance were studied numerically using simulation software. For an objective performance comparison we use almost the same system parameters as [2] namely: $m_1 = m_2 = 1$ $l_1, l_2 = 1$ and $I_1, I_2 = 1/3$.

As obvious from design K_i can be set moderately high for faster convergence rates. Conventional MSS techniques recommends $K_2 > K_1$ but as seen in section 3 it's not necessary. Time step constant ΔT set bounds for numerical error and hence must be set as low as possible. However actuator saturation must be kept in mind as smaller values increase control effort peaks and make the control signal noisy. Average control effort and rise time can be tuned by adjusting c_1 and transients damping can be enhanced using c_2.

Keeping in mind the above discussion following controller parameters were used for simulations.

a=1, $c_1 = 3$, $c_2 = 1$, $K_1 = 5$, $K_2 = 10$ and ΔT=0.07

In simulations the algorithm employed to solve (9) uses a combination of bisection, secant, and inverse quadratic interpolation methods for fast convergence, [11]. The function $\alpha(q_r)$ can be approximated with a straight line, as shown in Figure 3. This approximation is used to calculate a suitable guess for the algorithm to start with. The nonlinear function $\gamma(q_2)$ is constructed piecewise.

As depicted in Figure 4 and Figure 5 the nonlinear controller aggressively stabilizes Acrobot from both sets of initial conditions to its upright unstable equilibrium point.

Control effort has high initial peaks, Figure 6. These are due to the convergence of the error system and the swing up phase when energy is being pumped into the system. As depicted in Figure 6 right from start S_2 remains well within bounds whereas S_1 converges exponentially to a bounded value. However stability guaranty still comes from the state boundedness property of the driven system.

Simulation results are satisfactory keeping in mind especially the design ease and control law simplicity.

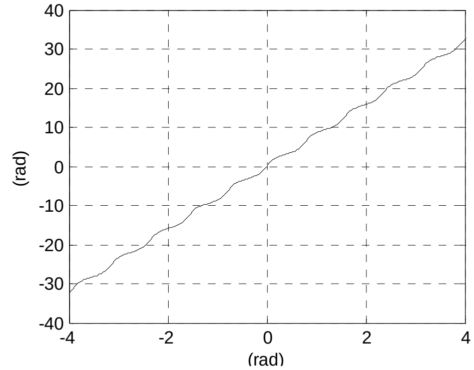

Figure 3 Plot of q_r vs function $\alpha(q_r)$.

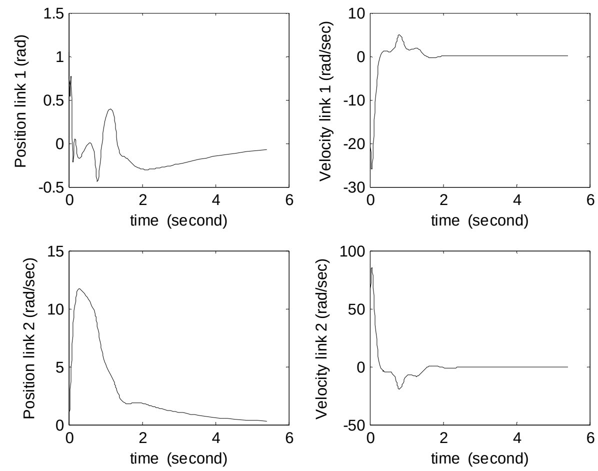

Figure 4 Trajectory (q_1, p_1, q_2, p_2) for Acrobot with initial conditions $(\pi/3, 0, 0)$.

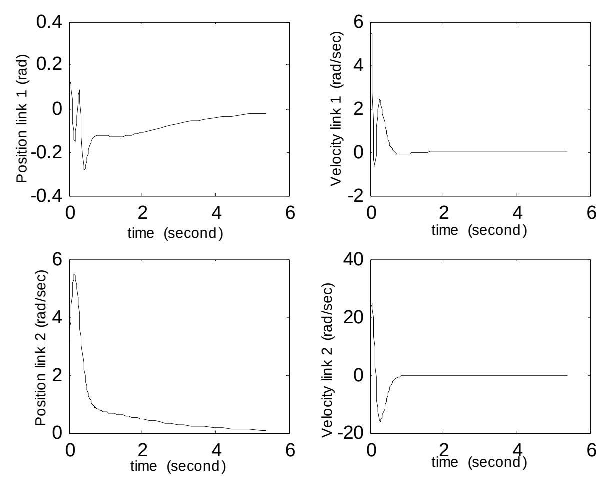

Figure 5 Trajectory (q_1, p_1, q_2, p_2) for Acrobot with initial conditions $(0, \pi, 0, 0)$.

5 Conclusions

The MSS technique has been applied to design a new controller for the non-linear ACROBOT system. The model is brought to strict feedback form before applying the MSS.

Some critical issues concerning the stability and performance of the system are studied numerically. Design simplicity is demonstrated and controller performance is compared to existing designs using both theoretical and simulation studies. In conclusion, the proposed controller has a relatively simple design procedure avoiding the need for closed form solutions of complex nonlinear equations. It tackles the issue of *explosion of terms* and also avoids use of low pass filters. The structure is also simpler requiring no supervisory switching controller after the swing up phase. Further work includes generalization of the scheme to cover the whole subclass.

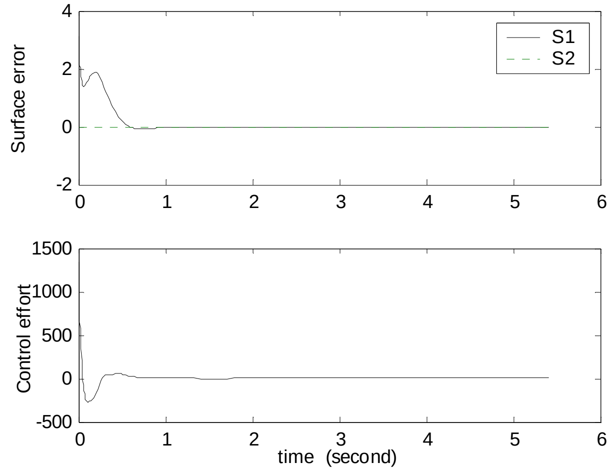

Figure 6 Error Surfaces and control effort , initial conditions $(0,\pi,0,0)$ **.**

References:

[1] R. M. Murray and J. Hauser, “A case study in approximate linearization: The Acrobot Example, ", Proceedings American Control Conferrence 1990

[2] Spong, M.W, “The swing up control problem for the Acrobot”, Control Systems Magazine, IEEE Volume 15, Issue 1, Feb. 1995 Pages: 49 – 55

[3] R.Olfati-Saber, “Control of underactuated mechanical systems with two degrees of freedom and symmetry,” in *Proc. Amer. Control Conf.*, vol. 6, Chicago, IL, June 2000, pp. 4092–4096

[4] E. D. Sontag and H. J. Sussman, “Further Comments on the stability of the Angular Velocity of a Rigid Body”, *Systems and Control Letters*. Vol. 12, pp. 213-217

[5] J. K. Hedrick and Y. Y. Yip, “Multiple sliding surface control,: theory and application,” *Trans. ASME, J. Dyn. Syst. Meas. Control*, vol. 122, pp. 586–593, 2000

[6] D. Swaroop, J. K. Hedrick, P. P. Yip, and J. C. Gerdes, “Dynamic surface control for a class of nonlinear systems,” *IEEE Trans. Automat. Contr.*,vol. 45, pp. 1893–1899, Oct. 2000

[7] Herbert Goldstein, *Classical Mechanics*, Second Eddition, Addison Wesley, USA, 1981.

[8] Alberto Isidori. *Nonlinear control systems*, Springer-Verlag, Berlin; New York, 2nd edition, ch. 4, 1989

[9] Olfati-Saber, "Normal Forms for Underactuated Mechanical Systems with Symmetry," *IEEE Trans. on Automatic Control*, vol. 47, pp. 305-308, Feb. 2002

[10] H. K. Khalil. *Nonlinear systems*, Prentice-Hall, USA 3rd edition, pp. 165, pp. 423-435, pp. 589, 2000

[11] Forsythe, G. E., M. A. Malcolm, and C. B. Moler, *Computer Methods for Mathematical Computations*, Prentice-Hall, 1976