ROBUST EXECUTION OF AGGRESSIVE MANEUVERS FOR PLANETARY ROBOTICS

Timothy C. Bretl*, Stephen Rock†
Aerospace Robotics Lab, Department of Aeronautics and Astronautics
Stanford University, Stanford, CA 94305

* PhD Candidate. Member AIAA.
† Associate Professor. Associate Fellow AIAA.

ABSTRACT

This paper considers the control of planetary rovers that use mobility strategies incorporating *aggressive maneuvers* such as jumping or hopping. Aggressive maneuvers are difficult to execute since they involve very precise discrete changes in the set of continuous dynamics governing the system. Robustness to noise in the time of initiation of these changes is an appropriate design criterion for control synthesis, which for a linear system can be formulated as a quasiconcave optimization problem and efficiently solved. This paper extends this synthesis method to a representative nonlinear system, the "climbing robot," and demonstrates the effectiveness of the resulting controller.

1 INTRODUCTION

One way of increasing the potential level of success of a mission involving planetary robotics is to enhance the mobility of the robots involved. A more mobile robot is able to explore more varied terrain and to explore basic terrain more efficiently and robustly. There is always a fundamental limit in the level of mobility of a robot given its hardware design. The extent to which this limit is reached is dependent on the range of motions that can be handled by on-board motion planning and control algorithms.

One class of motions that have proved difficult to plan and to execute is that of *aggressive maneuvers* such as non-periodic jumping or hopping. A general aggressive maneuver is defined in this paper as a trajectory beginning and ending with a discrete change in the set of continuous dynamics governing a robotic system, such that the intermediate set of dynamics has low controllability. By enabling aggressive maneuvers, mobility can be increased without increasing size, mass, or physical complexity, an important consideration for planetary exploration applications.

The idea of using sophisticated planning and control software to enable the design of robots with minimal hardware has been explored previously.[1] However, previous solutions to the problem of planning and executing any incorporated aggressive maneuvers have been application-specific. This problem can be framed more generally in the language of hybrid systems theory, by modeling the complete system as a collection of continuous models with rules for discrete transitions between them. If this hybrid model has one of several very specific forms, it can be analyzed as in Heemels et al.[2] Specific applications involving this type of model such as periodic legged locomotion have been addressed.[3] However, tractable methods for planning and control synthesis for general hybrid models do not exist.

Randomized motion planning, a technique capable of online generation of feasible planned trajectories even in complex configuration spaces, has also been applied to problems of this type.[4,5,6] However, feasible trajectories for aggressive maneuvers that are obtained from a randomized planner are not guaranteed to be robust.

This paper shows how to modify feasible trajectories for aggressive maneuvers in order to guarantee a specified level of robustness.

Section 2 motivates the use of aggressive maneuvers and presents the "climbing robot" as a representative system to be examined. Section 3 describes the challenges involved in executing a single aggressive maneuver with this system, in particular motivating the consideration of robustness to time-of-transition as an appropriate control design criterion. Section 4 formulates the corresponding synthesis problem for a general linear system, and shows how this formulation can be extended to a nonlinear system. Section 5 applies the extended nonlinear control synthesis to the climbing robot system and shows its effectiveness. Finally, Section 6 presents possibilities for future work.

2 MOTIVATION

Planetary mobility strategies based on jumping or hopping robots can be much more efficient than those based on wheeled or rocket-powered robots.[7] Several systems based on this approach are already under development. The increased mobility of these systems also allows the exploration of more varied terrain and provides an additional layer of mission robustness. If the robots are unable to move to a desired location because normal mobility strategies are insufficient or have failed, the more aggressive jumping or hopping strategies can be used. However, more sophisticated control algorithms are required before the mobility strategies of these systems can be considered reliable.

In order to begin developing these control algorithms, this paper examines a minimal robotic system that in fact *depends* on jumping in order to navigate through its environment. This system is the simple "climbing robot," shown in Figure 1. This robot moves in a vertical plane and consists of a single rigid bar, the endpoints of which can attach to, detach from, and exert a torque on pegs scattered throughout its environment.

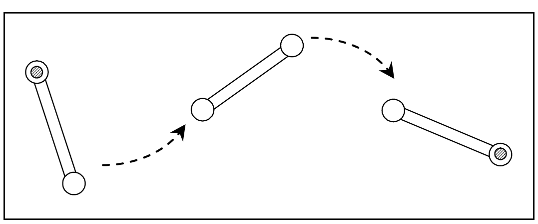

Fig. 1. A simple climbing robot, shown performing an aggressive maneuver. This maneuver is a jump between two pegs, consisting of a detach action, free flight, and an attach action.

Although this system and its environment are highly idealized, the planning and control techniques developed in this paper can be applied directly to more complicated real designs. As will be described in Section 3, as long as the contact dynamics operate on a small enough time scale that they can be modeled as discrete, then the control synthesis method presented can be applied to situations with three-dimensional motion and high-DOF manipulation.

3 CONTROL DESIGN CRITERION

This section frames the problem of executing a single aggressive maneuver, or a single jump of the climbing robot. It characterizes the preimage of a jump endpoint, discusses the selection of a nominal transition point, and motivates the consideration of robustness to time-of-transition as an appropriate control design criterion.

3.1 PROBLEM SETUP

Assume, as shown in Figure 2, that the robot, which consists of a massless link of length L between two equal masses m, is attempting to jump from Peg A to Peg B. To simplify analysis in this particular example, it will detach End 1 from Peg A and, after a period of free flight, attach End 2 to Peg B. In general, it could attach either end to Peg B.

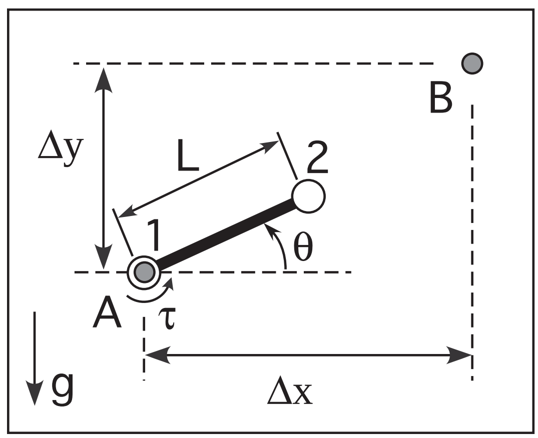

Fig. 2. Problem setup for the climbing robot attempting to jump between Pegs A and B.

The robot's trajectory is not controllable after it detaches from the first peg, so the success of a jump depends only on the state $(\theta_0, \dot{\theta}_0)$ of the robot at the time it detaches. Given these initial conditions and the location (x_A, y_A) of Peg A, the position $(x(t), y(t))$ of End 2 of the robot for any future time t is expressed as follows:

$$x(t) = x_A + \tfrac{L}{2}\cos(\theta_0) + + \tfrac{L}{2}(\cos(\theta_0 + \dot{\theta}_0 t) - \dot{\theta}_0 t \sin(\theta_0)) \quad (1)$$

$$y(t) = y_A + \tfrac{L}{2}\sin(\theta_0) - \tfrac{gt^2}{2} + + \tfrac{L}{2}(\sin(\theta_0 + \dot{\theta}_0 t) - \dot{\theta}_0 t \cos(\theta_0)) \quad (2)$$

Assume that End 2 of the robot can attach to Peg B if it is within some radius ε of the peg. For a jump to be successful, End 2 must be able to attach to Peg B at some future time t_1. For the peg geometry $(\Delta x, \Delta y)$ shown in Figure 2, this relationship is expressed as follows:

$$\left\| \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} - \begin{bmatrix} x(t_1) - x_A \\ y(t_1) - y_A \end{bmatrix} \right\|_2 \leq \varepsilon \quad (3)$$

3.2 Goal Preimage

The set of solutions to Equations 1-3 forms a region in the space of initial conditions. In the absence of noise, this region is the *preimage* of the jump goal $(\Delta x, \Delta y)$, since the robot can detach from Peg A at a state corresponding to any point in this region to safely arrive at Peg B.[8]

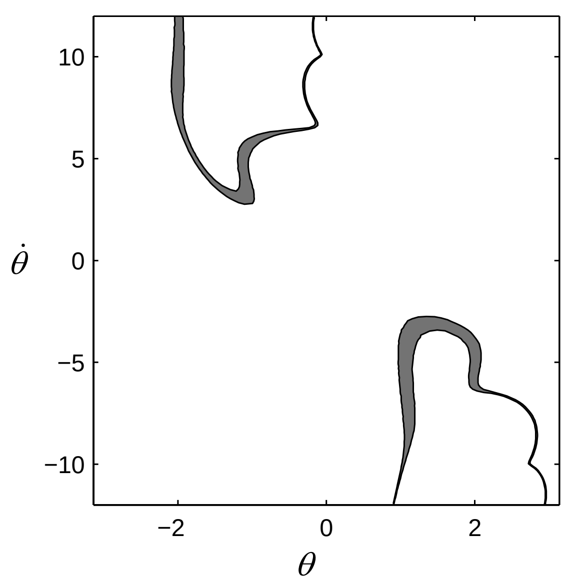

Fig. 3. Goal preimage in the range $\dot{\theta}_0 \in [-12, 12]$ for a single jump of the climbing robot. The region is the shaded area only.

Figure 3 shows a portion of this preimage for the climbing robot for the peg geometry $(\Delta x, \Delta y) = (2, -2)$ and $L = 1$. The width of the region is determined by the variable ε. The set of solutions for $\varepsilon = 0$, resulting in free-flight trajectories that exactly place End 2 on Peg B, is a curve running through the region.

If some control was possible during free-flight, the preimage would be expanded to include those initial states resulting in jump trajectories that can be modified to place End 2 on Peg B. This expansion would be quantified by modifying Equations 1-3, from which the preimage is calculated.

3.3 Nominal Transition Point

Any planning technique can be used to select a nominal transition point, or a nominal set of initial conditions inside the goal preimage at which to detach from Peg A. Typically there will be considerations such as obstacle avoidance or global path optimization that result in the selection of a particular detach state. However, it is possible that the selection is arbitrary, because characterizing the entire goal preimage is computationally intensive.

In particular, the goal preimage is very small relative to the space of all possible initial conditions, and presents a bottleneck for any planning algorithm. The complete configuration space of the climbing robot can be described as *non-expansive* around any single jump.[4] This characteristic is typical of the configuration spaces of robotic systems using aggressive maneuvers.

In this paper, it is assumed that a continuous trajectory to a nominal transition point is given. The problem, discussed in the next section, is to locally modify this trajectory to achieve a specified level of robustness. The actual transition point taken along the modified trajectory will be close enough to the nominal point to satisfy the higher-level considerations mentioned above.

3.4 Robustness Criterion

Several types of uncertainty arise in the execution of an aggressive maneuver. First, the uncontrollable trajectory between discrete transitions, or the free-flight of the climbing robot between detach and attach points, is subject to disturbances. Second, there is input noise and measurement noise before the first discrete transition, or before the detach point.

Each type of uncertainty alters the goal preimage. For example, assume that the accumulated error from free-flight disturbances can be bounded such that Equations 1 and 2 are known to be accurate to within a radius δ. Then a new radius of attachment can be defined as $\varepsilon' = \varepsilon - \delta$, shrinking the preimage to a set of initial conditions that are still guaranteed to result in trajectories that place End 2 within ε of Peg B. Measurement noise before the first discrete transition shrinks the preimage in a similar way, assuming that this noise can be bounded. *Continuous* input noise, or noise in the torque input while still attached to the first peg, has the same effect.

However, there is also *discrete* input noise, or uncertainty in the exact time at which the robot detaches from Peg A. This time is uncertain because the first transition is modeled as a discrete event but typically has associated dynamics which are difficult to quantify. In addition, if the system controller is implemented digitally, the sample rate of the controller constrains the exact time at which the discrete transition can be initiated. For example, if the actuator that initiates the detach action for the climbing robot operates at 5 Hz, the detach time will vary over a range of ± 0.1 seconds.

The effect of this type of uncertainty on the goal preimage can not be computed independently of the continuous trajectory of the robot before the detach action.

Assume that the contact dynamics are such that the detach action can be initiated independently of the continuous torque inputs. Also assume that the detach action has a sufficiently short time scale such that the assumption that it is a discrete transition is valid. Finally, assume that the robot can actuate normally until the time at which it detaches. In other words, between the time at which the robot commands a detach action and the time at which the detach action actually occurs, the effect of the continuous torque input does not change.

Then for a jump to be considered robust, the continuous trajectory through the nominal detach state must remain within the goal preimage for at least as long as the range over which the time-of-transition is expected to vary. In other words, instead of modifying the goal preimage, the continuous trajectory must be modified in order to make the system robust to this uncertainty.

For example, consider the nominal transition point $(\theta_0, \dot{\theta}_0) = (1.3, -3.2)$ in the goal preimage shown in Figure 3. A feasible continuous trajectory through this point, shown in Figure 4, was generated using a randomized planner of the type described by Sanchez and Latombe.[6] If this trajectory were tracked exactly and if the detach action occurred at exactly the right time, the resulting jump would be successful. However, the trajectory only remains within the goal preimage for 0.1 seconds, so if the time-of-transition is expected to vary over a range of larger than ± 0.05 seconds, then the plan is not robust.

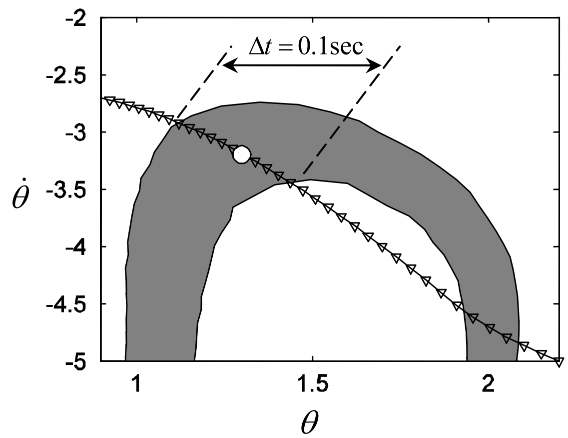

Fig. 4. Arbitrary trajectory through a desired transition point in a portion of the goal preimage for a single jump of the climbing robot. The trajectory remains in the region for only 0.1 seconds.

4 CONTROL SYNTHESIS

This section formulates the synthesis problem for a general linear system as a quasiconcave optimization problem, which can be solved efficiently. It then extends this formulation to apply to nonlinear systems.

In both cases it will be assumed that the goal preimage is convex. This assumption does not hold for the region shown in Figure 3 and does not hold in general either for linear or nonlinear systems. However, this assumption holds within some local neighborhood of a nominal transition point, which is all that needs to be considered for local modification of the continuous trajectory. In addition, this assumption allows the use of the highly tractable synthesis method to be presented in this section, the solution to which provides a good sub-optimal policy. This solution could be used as a starting point for optimization within the full nonconvex region. Therefore, a convex subset of the true goal preimage will be used for synthesis.

4.1 Linear Formulation

Assume that the time of initiation of the first discrete transition for a single aggressive maneuver is expected to vary over a time interval T. Then the control synthesis problem is given as follows:

Problem 1. Generate a trajectory that starts and remains in the goal preimage corresponding to a single aggressive maneuver for a continuous time interval of at least length T.

Assume that the set of continuous dynamics before the discrete transition is given by

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{4}$$

Also assume that the goal preimage is convex and can be expressed by the set of linear inequalities

$$\mathbf{F}x(t) \le \mathbf{g} \tag{5}$$

Then Problem 1 can be expressed more precisely as follows:

Problem 2. (Existence)

$$\begin{aligned} \text{Find} &: \mathbf{x}_0, \mathbf{u} \\ \text{Subject To} &: \mathbf{F}\mathbf{x}(t) \le \mathbf{g}, 0 \le t \le T \\ & \quad u_{\min} \le u(t) \le u_{\max}, 0 \le t \le T \end{aligned}$$

The initial condition $\mathbf{x}_0$ is a reachable point in the goal preimage. After the solution to this problem is found, an additional modification to the nominally planned continuous trajectory must be generated to reach $\mathbf{x}_0$.

Problem 2 is non-trivial when the system is not stabilizable about any point in the convex constraint region, which is often the case.

In general, the solution to Problem 2 is not unique. The problem statement can be extended to include multiple objectives. For example, let $(\mathbf{X}, \mathbf{U})_T$ be the set of all solutions $(\mathbf{x}_0, \mathbf{u})$ to Problem 2 for a given value of T. Then the solution to Problem 2 of minimum input norm for a given value of T can be found by solving

Problem 3. (Minimum Input Norm)

$$\begin{aligned} \text{Find} &: \arg\min_{\mathbf{x}_0, \mathbf{u}} \|\mathbf{u}\|_2 \\ \text{Subject To} &: (\mathbf{x}_0, \mathbf{u}) \in (\mathbf{X}, \mathbf{U})_T \end{aligned}$$

In addition, let $\Psi(T): T \to (\mathbf{x}_0, \mathbf{u})$ be the solution to Problem 3, and define the *minimum-escape-time* function as

$$f(\mathbf{x}_0, \mathbf{u}) = \inf\{T : \Psi(T) \in (\mathbf{X}, \mathbf{U})_T\} \tag{6}$$

Then the solution to Problem 3 for the maximum possible value of T, the "most robust" solution of minimum input norm, can be found by solving

Problem 4. (Most Robust)

$$\text{Find} : \arg\max_{\mathbf{x}_0, \mathbf{u}} f(\mathbf{x}_0, \mathbf{u})$$

The most robust solution can tolerate the highest possible level of uncertainty in time-of-transition.

The solution to Problem 2 that lies within the goal preimage of minimum possible size can be found in a similar way. This solution is called the "most precise" solution, since it results in an actual transition point that is closest to the nominal transition point for a given level of uncertainty. Assume without loss of generality that the set of linear equalities given in Equation 5 are centered about the origin. Then the most precise solution is found by solving

Problem 5. (Most Precise)

$$\begin{aligned} \text{Find} &: \arg\min_{\mathbf{x}_0, \mathbf{u}} \gamma \\ \text{Subject To} &: \mathbf{F}\mathbf{x}(t) \le \gamma\mathbf{g}, 0 \le t \le T \\ & \quad u_{\min} \le u(t) \le u_{\max}, 0 \le t \le T \end{aligned}$$

4.2 Linear Solution

The state of the system at any time step t is

$$\mathbf{x}(t) = \mathbf{A}^t\mathbf{x}_0 + \sum_{k=0}^{t-1} \mathbf{A}^{(t-1)-k}\mathbf{B}u(k) \tag{7}$$

So for a fixed value of T, the constraints on the domain of Problem 2 are linear in the variables $\mathbf{x}_0$

and $\mathbf{u}$. The additional objective function of Problem 5 is also a linear function of these variables and an additional size parameter γ. Therefore, both problems are convex and can be solved with a standard linear program solver. The objective function of Problem 3 is a quadratic function of $\mathbf{u}$ while the constraints are identical to those of Problem 2, so Problem 3 is also convex and can be solved as a quadratic program.

Further, sublevel sets of $-f(\mathbf{x}_0,\mathbf{u})$ are convex, so $f(\mathbf{x}_0,\mathbf{u})$ is quasiconcave. Thus Problem 4 is a quasiconcave maximization problem. Quasiconcave optimization problems can be solved efficiently using bisection around a convex subproblem.[9,10]

4.3 Extension to Nonlinear Systems

Although the control synthesis method presented in Sections 4.1 and 4.2 assumes linear dynamics, it extends naturally to nonlinear dynamics.

It has been assumed that the goal preimage has been truncated to a local convex subset. Therefore, it can be expected that a linearization about an arbitrary point in the convex subset will be very accurate throughout that subset. As a result, the solution to the control synthesis problem applied to the linearization will be a very good approximation of the optimal solution of the nonlinear synthesis problem within the convex subset.

This method will *not* necessarily provide a good approximation of the optimal solution of the synthesis problem within the full non-convex goal preimage.

5 Application

This section applies the extended nonlinear control synthesis method presented in this paper to the climbing robot system. The resulting trajectories are presented in comparison to the arbitrary trajectory shown in Figure 4.

5.1 Linearization

The nonlinear dynamics of the climbing robot system are as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\theta\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\dot{\theta}\\ -(g/L)\cos\theta\end{bmatrix} + \begin{bmatrix}0\\ (1/mL^2)\tau\end{bmatrix} \tag{8}$$

For an operating point $(\theta_0,\dot{\theta}_0)$, which can be chosen to be the nominally desired discrete-transition point, the linearized dynamics can be expressed in terms of perturbations $\hat{\theta}$ and $\dot{\hat{\theta}}$ as

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\hat{\theta}\\ \dot{\hat{\theta}}\end{bmatrix} = \mathbf{A}\begin{bmatrix}\hat{\theta}\\ \dot{\hat{\theta}}\end{bmatrix} + \mathbf{B}_1\tau + \mathbf{B}_2 \tag{9}$$

where

$$\mathbf{A} = \begin{bmatrix}0 & 1\\ (g/L)\sin\theta_0 & 0\end{bmatrix} \tag{10}$$

$$\mathbf{B}_1 = \begin{bmatrix}0\\ (1/mL^2)\end{bmatrix} \tag{11}$$

$$\mathbf{B}_2 = \begin{bmatrix}\dot{\theta}_0\\ -(g/L)\cos\theta_0\end{bmatrix} \tag{12}$$

5.2 Results

The nominally desired transition point for each set of results is $(\theta_0,\dot{\theta}_0) = (1.3,-3.2)$, the same that was used in Section 3.4.

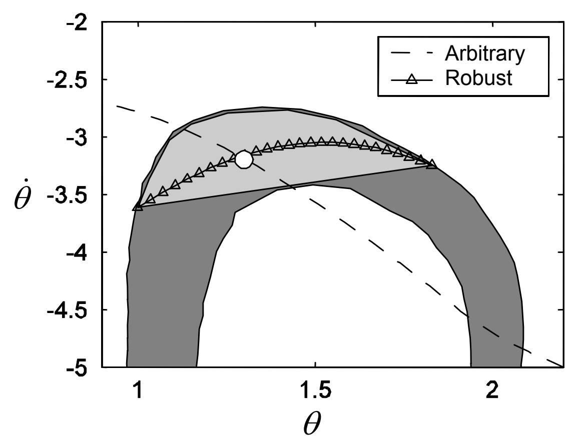

Fig. 5. Most-robust trajectory through a convex subset of the goal preimage.

Figure 5 shows the *most robust* solution of the control synthesis problem. This solution is the trajectory that remains in the convex subset of the goal preimage as long as possible, in this case for 0.26 seconds compared to 0.1 seconds for the arbitrary trajectory. So the system is now more than twice as robust to noise in time-of-transition.

The optimal trajectory appears to be curving to follow the arc of the goal preimage, although it begins and terminates in corners of the convex subset in order to maximize the time in the region. This result indicates both that better performance would be obtained by a synthesis method that considered the full non-convex transition region and that a good heuristic for solving the non-convex problem might be to try to follow a parameterized curve through the region.

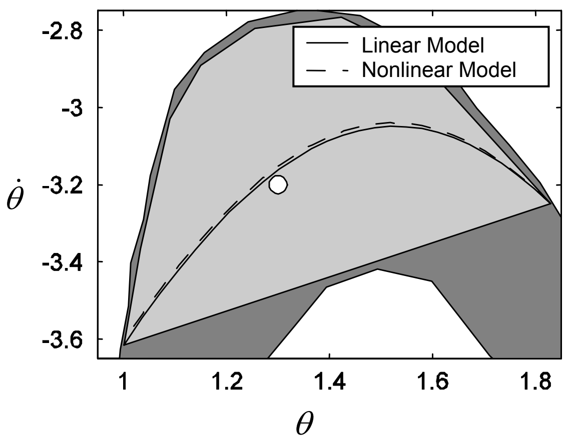

Fig. 6. Comparison between linear and nonlinear models of the most-robust trajectory.

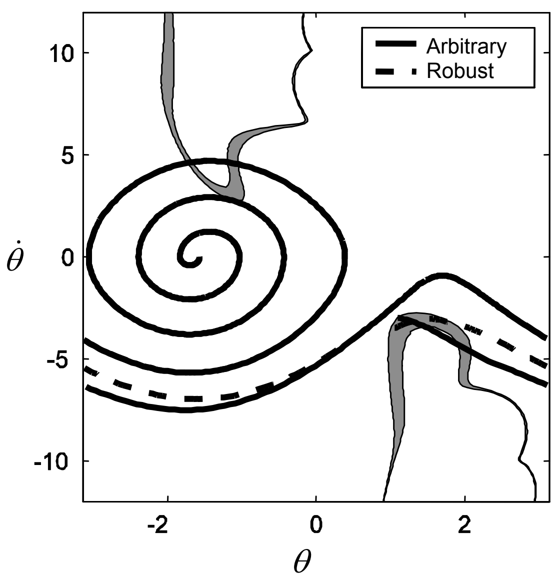

Fig. 7. Comparison between full arbitrary and most-robust trajectories.

Figure 6 shows a comparison between the trajectories obtained from integrating the linearized model and the true nonlinear model. As expected, the trajectories are very close, supporting the claim that the linear control synthesis method presented in Section 4 can be applied to nonlinear systems to generate a good approximation to the true optimal control policy.

Figure 7 shows the full continuous trajectories associated with the arbitrary and most robust solutions, starting from $(\theta, \dot{\theta}) = (-1.57, 0)$. As the two trajectories reach the nominal transition point, the most robust trajectory deviates locally from the arbitrary trajectory in order to reach the initial point of the solution shown in Figure 5.

Figure 8 shows the *most precise* solution of the control synthesis problem at a required level of robustness $T = 0.1$ seconds. This solution is the trajectory that remains for this length of time within the smallest possible convex region surrounding the desired transition point. Figure 8 indicates that, if the expected level of noise is known a priori, the controller can achieve much better performance.

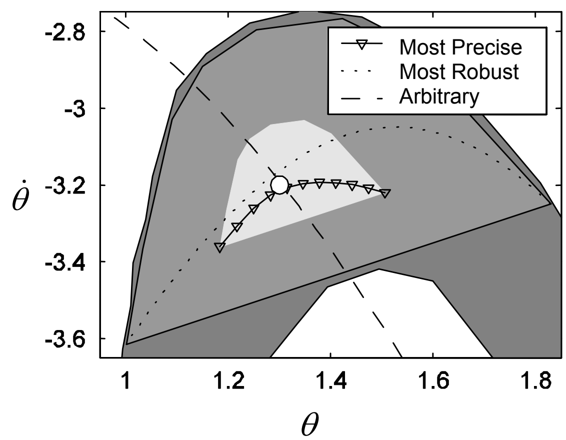

Fig. 8. Most-precise trajectory through a convex subset of the goal preimage.

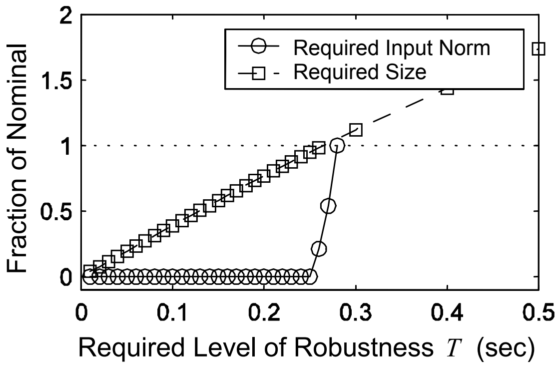

Fig. 9. Tradeoff between the required level of robustness and the achievable objectives.

Finally, Figure 9 shows the tradeoff between the required level of robustness and the minimum required input norm or minimum required goal preimage size. These curves demonstrate that the consideration of noise in time-of-transition affects the design of a robust controller, a primary argument of this paper.

6 Conclusion

This paper motivated the use of *aggressive maneuvers* to increase the level of mobility of a planetary robotic system toward its fundamental limit for a given hardware design. It demonstrated that robustness to noise in the time of the discrete transition that begins an aggressive maneuver is an appropriate design criterion for control synthesis, and presented a synthesis formulation for linear systems that could be efficiently solved. It extended the synthesis to apply to nonlinear systems, and applied the resulting algorithm to a specific robotic system, the "climbing robot."

There is high potential for future work. For example, the synthesis method presented in this paper could be extended to consider non-convex goal preimages, as mentioned in Section 5. Also, the control synthesis method could be integrated with a global planner.

References

[1] Lynch, K.M. and M.T. Mason (1997). Dynamic manipulation with a one joint robot. *Proc. IEEE International Conference on Robotics and Automation* pp. 359-366.

[2] Heemels, W.P.M.H., B. De Schutter and A. Bemporad (2001). Equivalence of hybrid dynamical models. *Automatica* **37**(7), 1085-1091.

[3] Goodwine, B. and J. Burdick (1997). Gait controllability for legged robots. *Proc. IEEE International Conference on Robotics and Automation*.

[4] Hsu, D., J.C. Latombe and R. Motwani (1999). Path planning in expansive configuration spaces. *International Journal of Computational Geometry and Applications* **9(4-5)**, 495-512.

[5] Kindel, R., D. Hsu, J.C. Latombe and S. Rock (2000). Kinodynamic motion planning amidst moving obstacles. *Proc. IEEE International Conference on Robotics and Automation*.

[6] Sanchez, G. and J.C. Latombe (2001). A single-query bi-directional probabilistic roadmap planner with lazy collision checking. *International Symposium on Robotics Research*.

[7] Fiorini, P., S. Hayati, M. Heverly and J. Gensler (1999). A hopping robot for planetary exploration. *Proc. IEEE Aerospace Conference*.

[8] Lozano-Perez, T., M. Mason, and R.H. Taylor (1984). Automatic Synthesis of Fine-Motion Strategies for Robots. *International Journal of Robotics Research*, **3**(1).

[9] Boyd, S. and C. Barratt (1991). *Linear Controller Design – Limits of Performance.* Prentice-Hall.

[10] Boyd, S., L. El Ghaoui, E. Feron and V. Balakrishnan (1994). *Linear Matrix Inequalities in System and Control Theory*. SIAM.