This problem set is due *by 11:59pm* on **Thursday, April 16**.

Problem 1 (The Spring Loaded Inverted Pendulum) *Download the file* `slip.m` *from the course website. This file contains the basic matlab code needed for the simu lation of the passive SLIP model.*

a) **Return Map Stability** *The initial conditions given in the simulation are in the basin of attraction of the stable limit cycle, but are not on the limit cycle itself. Determine the fixed point of the return map in simulation. Now, numerically evaluate the derivatives of the return map around the stable fixed point. What are the non-trivial eigenvalues of this derivative matrix? Submit your results.*

Extra Credit: *Compare the fixed point and derivatives of the return map to their analytical values using the small touch down angle approximation of the return map? An analytical expression for the return map is given in [1].*

b) **Feedback Control** *Using the provided matlab skeleton, implement a feedback control system that regulates the apex to a desired height $y_{apex} = 1.5m$ (as shown in lecture) with a constant energy of 30J. Your control input is the leg angle right before touchdown. Submit your code and a plot of your results.*

Problem 2 (Motion Planning with RRTs) *Find a motion plan using Rapidly-exploring Random Trees to find a solution to the swing-up task for the torque-limited simple pendulum:*

$$ml^2\ddot{\theta} + b\dot{\theta} + mgl\cos(\theta) = u.$$

Use the following system parameters: $m = 1, l = 1, b = 0.1, g = 9.8, u \in [-3, 3]$. For your start and goal, use: $\mathbf{x}_{start} = [-\frac{\pi}{2}, 0]^T, \mathbf{x}_{goal} = [\frac{\pi}{2}, 0]^T$. Submit your code, and a plot of the motion plan trajectory in the phase plane superimposed *over your RRT.*

Hint 1: *Be careful to get the wrap-around effects correct ($\theta = -\pi$ vs. $\theta = \pi$).*

Hint 2: *Remember that, in order to enforce the dynamics, after you pick a point at random in $\mathbf{x}$, you'll want to select an action $\mathbf{u}$ that moves you towards that sample from the "nearest" point on your tree.*

Use a Euclidean distance metric and if you wish, you may consider using any of the tricks described in [2] for speeding up your algorithm.

Extra Credit: Motion Planning Metrics The performance of the randomized search algorithms depends heavily on the distance metric. For extra credit, find a metric which improves the performance of the search for this problem (note that you will have to run both searches multiple times and average the performance to demonstrate an improvement). Comment on whether your proposed metric is a true distance metric (satisfying non-negativity, identity of indiscernibles, symmetry and the triangle inequality), and whether it is specific to the pendulum or might have general applicability to underactuated systems.

References

[1] Hartmut Geyer. *Simple models of legged locomotion based on compliant limb behavior*. PhD thesis, University of Jena, 2005.

[2] S. LaValle and J. Kuffner. Rapidly-exploring random trees: Progress and prospects. In *Proceedings of the Workshop on the Algorithmic Foundations of Robotics*, 2000.

6.832 Underactuated Robotics
Spring 2009