Planning for Bipedal Locomotion using a Library of Primitives and Trajectory Optimization

Siyuan Feng and Christopher G. Atkeson
The Robotics Institute, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213
{*sfeng, cga*}@*cs.cmu.edu*

1 Motivation

Motion planing is crucial for humanoid robots to walk outside labs and be useful in daily life. But it is computationally intractable to generate a full body motion plan in real time due to high dimensionality. However this kind of planning is usually unnecessary for simple locomotion tasks (i.e. walking without much perturbation). Such tasks are ideal for carefully designed primitives because of their simplicity and efficiency. On the other hand, trajectory based optimal controllers are much more expensive to generate, but capable of more difficult tasks. We are proposing to combine the advantages of both in a way to best utilize limited online computation and still be able to traverse relatively rough terrain dynamically.

2 Related work

In the graphics community, Yin et al. [6] demonstrated simple state machine based controllers that are cable of a variety of simple locomotion behaviors. Parameters for these controllers are also optimized in [5] to handle external perturbation and various terrain. These simple controllers are shown to be robust against external perturbation to some extent. Coros et al. presented a planning framework for the stepping stone problem in a reduced state and action space [2]. In [1], an online optimized preview plan is generated for a reduced model, which is then mapped back to the full robot state. Although this approach demonstrated promising capabilities, it still does not run in real time.

3 Proposed approach

Our approach gives step to step plans using a library of primitives. This requires a mapping between robot states on consecutive steps given an action and certain terrain. For each primitive a_i, we will learn the approximate step to step dynamics under various initial conditions across different terrains using Gaussian Process Regression similar to [4]. GPR naturally provide us with an estimation of the next state together with its uncertainty. Since we will train GPR with successful (s, a, s') tuples, it also serves as an approximation for the basin of attraction given certain primitive and terrain. During execution, we will continuously add new training examples to adapt the learned mapping to the current terrain.

A preview planer is used on top of these learned dynamics to select a sequence of optimal primitives to maximize stability and minimize state uncertainty. If the primitives are insufficient for the task (e.g. a large step down), we will then switch to a trajectory based approach. Similar to [3], we will formulate initial trajectory generation as a constrained optimization problem, and apply off-the-shelf optimizers to solve it. We then use Differential Dynamic Programming to locally optimize along the initial trajectory and compute local linear polices.

4 Open questions

We are heavily relying on the robustness of the underlying primitives and computed policies for execution and transition between actions. Since a trajectory based policy can be quite sensitive to changes in initial condition, we may need to use a small set of trajectories for successful transition from the previous action. The proposed approach encapsulates model uncertainty, external perturbation and actuation uncertainty all in the learned step to step dynamics. Learning this mapping can be very hard, or the estimation step can be too slow for the preview controller.

References

[1] S. Coros, P. Beaudoin, and M. van de Panne. Robust task-based control policies for physics-based characters. *ACM Transactions on Graphics*, 28(5):1, December 2009.

[2] S. Coros, P. Beaudoin, K. K. Yin, and M. van de Pann. Synthesis of constrained walking skills. *ACM Transactions on Graphics*, 27(5):1, December 2008.

[3] C. Liu and C. G. Atkeson. Standing balance control using a trajectory library. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3031–3036. Ieee, October 2009.

[4] J. Morimoto and C. G. Atkeson. Nonparametric representation of an approximated Poincaré map for learning biped locomotion. *Autonomous Robots*, 27(2):131–144, September 2009.

[5] J.M. Wang and D.J. Fleet. Optimizing walking controllers for uncertain inputs and environments. *ACM Transactions on Graphics*, 2010.

[6] K.K. Yin, K. Loken, and M. van de Panne. Simbi-

con: Simple biped locomotion control. *ACM Transactions on Graphics (TOG)*, 26(3):105, 2007.