Standing Balance Control Using a Trajectory Library

[1]Chenggang Liu and [2]Christopher G. Atkeson
[1]Shanghai Jiao Tong University, [2]Carnegie Mellon University;
email: cgliu2008@gmail.com web: http://www.andrew.cmu.edu/user/cgliu

INTRODUCTION

Standing balance control is an important control problem for humanoid robots. In [1], it is shown that a single optimization criterion can be used to generate multiple balance recovery strategies. We employ a library of optimal trajectories and the neighboring optimal control method to compute local approximations to the optimal control. We take advantage of a parametric nonlinear optimization method, SNOPT, to generate initial trajectories and then use Differential Dynamic Programming (DDP) to refine them. A library generation method is proposed, which keeps the trajectory library to a reasonable size. We compare the proposed controller with an LQR based gain scheduled controller with the same optimization criterion. Simulation results demonstrate the performance of the proposed method.

METHODS

We use a parametric nonlinear optimization method, SNOPT [2], based on Sequential Quadratic Programming (SQP) to generate initial trajectories. We found that SNOPT is generally more robust in terms of finding a solution trajectory than DDP, but the solutions are not necessarily physically correct or smooth, and do not have any feedback controller.

Differential Dynamic Programming (DDP) is a second order gradient local trajectory optimization method for optimal control of nonlinear systems [3]. It optimizes each point of the trajectory, producing a more physically accurate and smoother trajectory and a set of feedback gains at each time step. Its convergence is rapid when the initial trajectory is good.

We consider three types of initial conditions. For constant pushes, the push size and the push location are not zero, but the state is zero. For instantaneous pushes, the initial joint velocities are not zero. For recovery after constant pushes are removed, the initial joint angles are not zero. For each type of push, initial conditions are generated on a uniform grid. Trajectories are optimized for each initial condition.

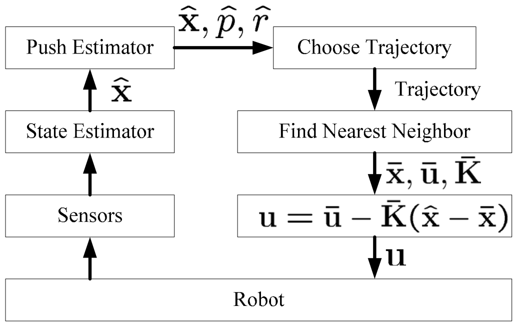

Figure 1: Standing balance controller architecture.

The final controller is shown in Figure 1. In each time step, the current state estimate, $\hat{\mathbf{x}}$, the push size estimate, $\hat{p}$, and the push location estimate, $\hat{r}$, are created. One optimal trajectory is chosen from the library. Finally, the closest state to $\hat{\mathbf{x}}$ on the optimal trajectory, $\bar{\mathbf{x}}$, along with the corresponding control, $\bar{\mathbf{u}}$, and the feedback gain matrix, $\bar{\mathbf{K}}$, are used to compute the control, $\mathbf{u}$.

RESULTS AND DISCUSSION

As an example, we show the response to a constant forward push at the head of 40 Newtons as shown in Figure 2. Its performance is also compared with a gain scheduled controller based on Linear Quadratic Regulators, which falls down for constant forward pushes at the head larger then 36 Newtons. In contrast, the controller proposed here is able to handle a constant forward push of 55 Newtons. Additional simulation results further demonstrate the performance of the proposed method.

REFERENCES

1. Christopher Atkeson and Benjamin Stephens, Multiple Balance Strategies from One Optimization Criterion, *The IEEE-RAS 2007 International Conference on Humanoid Robots,* Pittsburgh, PA, November 2007.
2. P. E. Gill, W. Murray, and M. A. Saunders, SNOPT: An SQP algorithm for large-scale constrained optimization, *SIAM Journal on Optimization*, **12**, 979-1006, 1997
3. D. Jacobson and D. Mayne, *Differential Dynamic Programming,* Elsevier, New York, NY, 1970

Figure 2: The robot under the constant forward push at the head of 40 Newtons. The frames are taken in intervals of 0.3 seconds.