Generalizable Framework for Designing and Optimizing Dynamic, Robust, and Adaptive Bipedal Locomotion

Patrick MacAlpine
Department of Computer Science, The University of Texas at Austin
1616 Guadalupe St Suite 2.408 Austin Texas 78701 USA
patmac@cs.utexas.edu

ABSTRACT

Before robots can become a viable technology for assisting people with everyday tasks, they must be able to adapt to the ever-changing conditions found in real, human-centered environments, designed for a person's ability to walk on two legs. This dictates the need for humanoid robots that can exhibit robust, bipedal locomotion. This thesis explores different methods for developing and optimizing walks for humanoid robots.

1. INTRODUCTION

Most robots today use wheels for locomotion, as wheels are both energy efficient and easy to control. While this is fine for flat, smooth surfaces, wheels are less practical for environments containing rough terrain and obstacles. Humans, on the other hand, use bipedal locomotion to get around. Bipedal locomotion, defined as walking upright on two legs, allows humans to easily navigate obstacles such as stairs and uneven ground. Dynamic stabilization of bipedal robots is a very challenging problem for which no general algorithm exists. Getting a robot to walk on two legs is difficult due to the required balance when only one leg is on the ground.

The objective of this thesis is to develop a generalizable framework for creating optimal walks for bipedal robots that are able to cope with different environments (dynamic), are stable in the presence of external forces (robust) and can adjust to mechanical variances (adaptive). This framework, when completed, will provide a strong foundation for implementing bipedal walks as well as serve as a basis for further research in robot mobility.

2. ROBOT SOCCER

Robot Soccer [3] has served as an excellent research domain for autonomous agents and multi-agent systems over the past decade and a half. It is a great platform for testing learning scenarios in which multiple skills, decisions, and controls have to be learned by a single agent, and agents themselves have to cooperate or compete. Robot soccer has spread over several popular platforms, each having its own advantages. For example, the real robot competitions, including the humanoid robot league, have typically emphasized low-level robot control challenges. On the other hand, the RoboCup 2D simulation platform [4] has emphasized high-level team strategy challenges. For this research, I focus on the RoboCup 3D simulation platform, which has recently been gaining increased popularity. This platform integrates both these low-level and high-level challenges under one umbrella.

In the 3D simulation league [4], teams of nine simulated humanoids play in a simulation environment with realistic physics, state-noise, multidimensional actions and real-time control. One advantage of the 3D simulation domain over real robots is avoiding the high cost of errors, and the relatively slow feedback loop, that happens when testing new skills in the real world. Due to the complexity of the environment, parts of the agent are hard to design by hand. For instance, it is a significant challenge to design a walk that is both fast and stable. The 3D simulation platform allows for designing and investigating general methodologies for skill and strategy acquisition in a complex, challenging domain, using machine learning.

The RoboCup 3D simulation environment is based on SimSpark[5], a generic physical multiagent system simulator. SimSpark uses the Open Dynamics Engine[2] (ODE) library for its realistic simulation of rigid body dynamics with collision detection and friction. ODE also provides support for the modeling of advanced motorized hinge joints used in the humanoid agents.

The robot agents in the simulation are homogeneous and are modeled after the Aldebaran Nao robot [1], which has a height of about 57 cm, and a mass of 4.5 kg. The agents interact with the simulator by sending torque commands and receiving perceptual information. Each robot has 22 degrees of freedom: six in each leg, four in each arm, and two in the neck. In order to monitor and control its hinge joints, an agent is equipped with joint perceptors and effectors. Joint perceptors provide the agent with noise-free angular measurements every simulation cycle (20 ms), while joint effectors allow the agent to specify the torque and direction in which to move a joint. Although there is no intentional noise in actuation, there is slight actuation noise that results from approximations in the physics engine and the need to constrain computations to be performed in real-time. Agents are also outfitted with noisy accelerometer and gyroscope perceptors, as well as a force resistance perceptors on the sole of each foot.

3. COMPLETED WORK

In recent AAMAS papers [10, 8] I have explored learning walks with different types of walk engines and learning algorithms in the RoboCup 3D simulation environment. For the 2010 RoboCup 3D simulation competition we used machine learning techniques to optimize a series of fixed key frame poses for the agent to cycle through in order to walk and turn in different directions. After comparing machine learning algorithms, including hill climbing and genetic algorithms, we found that the CMA-ES algorithm [7] produced the best results. As fully described in [10], our team's walk was not omnidirectional and could only move in set directions such as forward, sideways, and backwards. The walk was not able to keep up with the top agents in the competition and we ended up finishing just outside the top eight.

For the 2011 RoboCup 3D simulation competition, after deciding that an omnidirectional walk gave us the best chance for quickly moving and turning, we chose to use a double linear inverted pendulum model omnidirectional walk engine based on the research performed by Graf et al. [6]. The main advantage of an omnidirectional walk is that it allows the robot to request continuous velocities in the forward, side, and turn directions, permitting it to approach its destination more quickly. In addition, the robustness of this engine allowed the robots to quickly change directions, adapting to the changing situations encountered during soccer games. We used CMA-ES to optimize multiple parameter sets for the walk engine specific to different tasks, such as moving to a target position on the field or dribbling a soccer ball, as described in [8]. This omnidirectional walk engine, and associated machine learned parameter sets, was the key factor in us going undefeated and winning the 2011 competition.

As an extension to walking, we also developed a kick engine to allow our humanoid soccer playing agents to quickly move the soccer ball [8]. This system uses inverse kinematics to control the kicking foot of an agent such that it follows an appropriate trajectory through the ball. This trajectory is defined by set waypoints, ascertained through machine learning, relative to the ball along a cubic Hermite spline.

4. RELATED WORK

Current state-of-the-art research often incorporates a concept known as Zero-Moment-Point (ZMP) [11] to help stabilize robots. ZMP is a calculation of the point at which all inertial forces acting on a robot sum to zero. More intuitively, this is the center of pressure exerted by a robot's single, planted foot on the ground. If this calculated point travels outside the surface dimensions of the robot's foot, then the robot may become stable and fall. While ZMP is useful in maintaining stability, walks generated using ZMP tend to be energy inefficient and unnatural looking. Also if a robot enters a state in which the constraints for ZMP stability are exceeded, ZMP does not provide a way to recover balance.

Recently a biologically inspired approach known as Central Pattern Generators (CPGs) was developed to generate bipedal walks [9]. CPGs, consisting of groups of neurons, exist in the spinal cords of vertebrates and produce rhythmic patterns believed to control stepping. In bipedal robots these CPGs, implemented as artificial neural networks, have successfully been used for generating and controlling walks that adapt to perturbations, thus allowing for robots to regain their balance from instable positions. Additionally the CPG based walks appear more biologically natural and efficient than those of ZMP. One drawback of CPGs, however, is that the relationship between their mathematical constructs and resulting robotic dynamics are hard to discern. This makes crafting CPGs for specific tasks much more difficult than ZMP approaches.

5. FUTURE RESEARCH

My ongoing research agenda includes experimenting with further learning algorithms and types of bipedal walk engines. Additionally I intend to explore the construction of optimization tasks for learning walks. This includes modeling walking trajectories within an optimization task after those observed in human infants learning to walk, and also modifying these tasks during the course of learning so as to emphasize the parts for which an agent is currently struggling.

In order to create a generalizable framework for designing bipedal locomotion I expect to test out the walk learning infrastructure I develop on different types of robots. This will provide an analysis of how robust the learning methods are to heterogeneous robot models. Ultimately, as a validation of my framework, I hope to apply what we can learn in simulation to the actual Nao robots which we use to compete in the Standard Platform league of RoboCup.

6. REFERENCES

[1] Aldebaran Humanoid Robot Nao. `http://www.aldebaran-robotics.com/eng/`.

[2] Open Dynamics Engine. `http://www.ode.org/`.

[3] RoboCup. `http://www.robocup.org/`.

[4] RoboCup 3D simulation league. `http://en.wikipedia.org/wiki/RoboCup_3D_Soccer_Simulation_League`.

[5] SimSpark. `http://simspark.sourceforge.net/`.

[6] C. Graf, A. Härtl, T. Röfer, and T. Laue. A robust closed-loop gait for the standard platform league humanoid. In *Proc. of the 4th Workshop on Humanoid Soccer Robots in conjunction with the 2009 IEEE-RAS Int. Conf. on Humanoid Robots*, pages 30 – 37, Paris, France, 2009.

[7] N. Hansen. *The CMA Evolution Strategy: A Tutorial*, January 2009. `http://www.lri.fr/~hansen/cmatutorial.pdf`.

[8] P. MacAlpine, D. Urieli, S. Barrett, S. Kalyanakrishnan, F. Barrera, A. Lopez-Mobilia, N. Stiurca, V. Vu, and P. Stone. UT Austin Villa 2011: A champion agent in the RoboCup 3D soccer simulation competition. In *Proc. of 11th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, 2012. To appear.

[9] L. Righetti and A. Ijspeert. Programmable central pattern generators: an application to biped locomotion control. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 1585 –1590, may 2006.

[10] D. Urieli, P. MacAlpine, S. Kalyanakrishnan, Y. Bentor, and P. Stone. On optimizing interdependent skills: A case study in simulated 3d humanoid robot soccer. In K. Tumer, P. Yolum, L. Sonenberg, and P. Stone, editors, *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, volume 2, pages 769–776. IFAAMAS, May 2011.

[11] M. Vukobratović and B. Borovac. Zero-moment point - thirty five years of its life. *International Journal of Humanoid Robotics*, 1(1):157–173, 2005.