Actuating a Simple 3D Passive Dynamic Walker

Russ Tedrake
Computer Science and
Artificial Intelligence Lab
Massachusetts Institute of Technology
Cambridge, MA 02139
Email: russt@ai.mit.edu

Teresa W. Zhang, Ming-fai Fong
Department of
Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139
Email: {resa, flamingo}@mit.edu

H. Sebastian Seung
Howard Hughes Medical Institute
Brain & Cognitive Sciences
Massachusetts Institute of Technology
Cambridge, MA 02139
Email: seung@mit.edu

***Abstract*— The passive dynamic walker described in this paper is a robot with a minimal number of degrees of freedom which is still capable of stable 3D dynamic walking. First, we present the reduced-order dynamic models used to tune the characteristics of the robot's passive gait. Our sagittal plane model is closely related to the compass gait model, but the steady state trajectory passively converges from a much larger range of initial conditions. We then experimentally quantify the stability of the mechanical device. Finally, we present an actuated version of the robot and some preliminary active control strategies. The control problem for the actuated version of the robot is interesting because although it is theoretically challenging (4 degrees of under-actuation), the mechanical design of the robot made it relatively easy to create controllers which allowed the robot to walk stably on flat terrain and even up a small slope.**

I. Introduction

In the late 1980's, Tad McGeer [1] introduced a class of walking robots, known as *passive dynamic walkers*, which walk stably down a small decline without the use of any motors. The most impressive passive dynamic walker [2] has knees and arms, and walks with a surprisingly anthropomorphic gait. These machines provide an elegant illustration of how proper machine design can generate stable and potentially very energy efficient walking. These ideas, however, are only beginning to have an impact on the way fully actuated bipedal robots are designed and controlled (i.e., [3], [4]).

To bridge the gap between passive and active walkers, a number of researchers have investigated the problem of adding a small number of actuators to an otherwise passive device ([5], [6], [7]). There are two major advantages to this approach. First, actuating a few degrees of freedom on an otherwise passive walker is a way to capitalize on the energy efficiency of passive walking and the robustness of actively controlled systems. Second, by allowing the dynamics of the system to solve a large portion of the control problem, it may be possible to simplify the control problem that is solved by the actuators.

The goal of this paper is to describe the mechanical design of our simple 3D passive dynamic walker, and the design of some preliminary active control strategies that have been applied to a partially-actuated version of that robot. These simple controllers were created as a baseline with which to compare the learned controllers that are the main focus of this research. They allow the robot to walk on flat terrain, and even up a small slope.

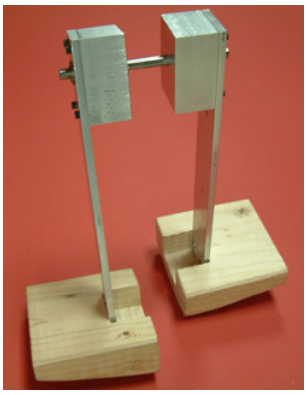

Fig. 1. The Passive Walker

II. Passive Dynamic Walker

The passive dynamic walker shown in Figure 1 represents the simplest machine that we could build which captures the essence of stable dynamic walking in three dimensions. It has only a single passive pin joint at the hip. When placed at the top of a small ramp and given a small push sideways, the walker will rock onto a single stance leg, allowing the opposite leg to leave the ground and swing forward down the ramp. Upon landing, the robot rocks onto the opposite foot, and the cycle continues. A video montage been included in the proceedings.

The energetics of this passive walker are common to all passive walkers: the energy lost due to friction and collisions when the swing leg returns to the ground are balanced by the gradual conversion of potential energy into kinetic energy as the walker moves down the slope. A particular challenge with this walker is the design of its large curved feet. Using only reduced planar models of its dynamics, we are able to design feet which tune the step frequency and step length of the robot to produce an elegant and robust gait.

This walker is morphologically equivalent to the Tinkertoy walker [8], except that on our robot the center of the radius of curvature of the feet is higher than the center of mass. Because of this, standing is a statically stable configuration.

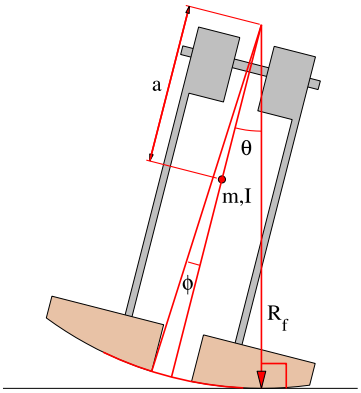

Fig. 2. Frontal Plane Model

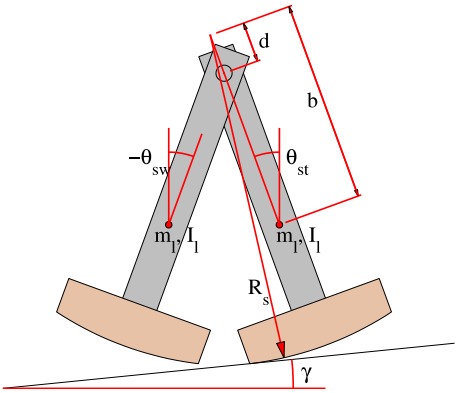

Fig. 3. Sagittal Plane Model

A. Frontal Plane Model

To model the dynamics in the frontal plane, we assume that the robot is always in contact with the ground at exactly one point and that the foot rolls without slipping. The equations of motion, in terms of body angle θ, for this planar model are given in three parts using the form

$$H(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + G(\theta) = 0.$$

When $|\theta| > \phi$, the ground contact point is in the curved portion one of the feet (the boundary condition on the outside of the foot is not modeled), and the dynamics are:

$$\begin{aligned} H(\theta) &= I + ma^2 + mR_f^2 - 2mR_f a\cos\theta, \\ C(\theta,\dot{\theta}) &= mR_f a\dot{\theta}\sin\theta, \\ G(\theta) &= mga\sin\theta. \end{aligned}$$

When $|\theta| \leq \phi$, the ground contact is along the inside edge of the foot. In this case, the dynamics are:

$$\begin{aligned} H(\theta) &= I + ma^2 + mR_f^2 - 2mR_f a\cos(\theta-\alpha), \\ C(\theta,\dot{\theta}) &= 0, \\ G(\theta) &= mg(a\sin\theta - R_f\sin\alpha). \end{aligned}$$

where $\alpha = \theta - \phi$ if $\theta > 0$, otherwise $\alpha = \theta + \phi$.

Finally, the collision of the swing leg with the ground is modeled as an inelastic (angular momentum conserving) impulse,

$$\dot{\theta}^+ = \dot{\theta}^- \cos\left[2\tan^{-1}\left(\frac{R_f\sin\phi}{R_f\cos\phi - a}\right)\right],$$

which occurs when $\theta = 0$.

A simulation of these dynamics produces a damped oscillation that will eventually result in the robot standing in place (energy lost on impact is not restored). Our primary concern for this model is the *frequency* of that oscillation. For a given mass and moment of inertia, we can change the frequency by changing R_f. The actuated version of the robot, presented in section III, carries its mass very differently than the purely passive version of the robot, due to the added mass motors and sensors. By simulating this model, we were able to find very different radii for the feet in the frontal plane that allow different versions of the robot to both oscillate back and forth with the desired step frequency of approximately 1.4 Hz. The newest version of our actuated walker is considerably heavier because it carries the computer and batteries on board, so we reduced the desired step frequency to 0.8 Hz for this robot.

B. Sagittal Plane Model

In the sagittal plane, the dynamics are a slightly modified version of the well-studied compass gait [9]. The parameters of this model can be found in Figure 3, and the dynamics are given by:

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{0},$$

where $\mathbf{q} = [\theta_{st}, \theta_{sw}]^T$, and:

$$\begin{aligned} H_{11} =& I_l + m_l b^2 + m_l d^2 + 2m_l R_s^2 - 2m_l R_s(b+d)\cos(\theta_{st}-\gamma) \\ H_{12} =& H_{21} = m_l(b-d)[d\cos(\theta_{st}-\theta_{sw}) - R_s\cos(\theta_{sw}-\gamma)] \\ H_{22} =& I_l + m_l(b-d)^2, \\ C_{11} =& m_l R_s(b+d)\sin(\theta_{st}-\gamma)\dot{\theta}_{st} + \frac{1}{2}m_l d(b-d)\sin(\theta_{st}-\theta_{sw})\dot{\theta}_{sw} \\ C_{12} =& m_l(b-d)[d\sin(\theta_{st}-\theta_{sw})(\dot{\theta}_{sw} - \frac{1}{2}\dot{\theta}_{st}) + R_s\sin(\theta_{sw}-\gamma)\dot{\theta}_{sw}] \\ C_{21} =& m_l(b-d)[d\sin(\theta_{st}-\theta_{sw})(\dot{\theta}_{st} - \frac{1}{2}\dot{\theta}_{sw}) - \frac{1}{2}R_s\sin(\theta_{sw}-\gamma)\dot{\theta}_{sw}] \\ C_{22} =& \frac{1}{2}m_l(b-d)[d\sin(\theta_{st}-\gamma) + R_s\sin(\theta_{sw}-\gamma)]\dot{\theta}_{st} \\ G_1 =& m_l g(b+d)\sin\theta_{st} - 2m_l g R_s\sin\gamma, \\ G_2 =& m_l g(b-d)\sin\theta_{sw}. \end{aligned}$$

The abbreviation st is shorthand for the stance leg and sw for the swing leg.

As with the compass gait model, we assume the swing foot can swing through the ground. Only the transfer of support as the swing leg becomes stance leg is modeled as a collision. Because our model has large curved feet rather than point feet, some portion of the swing foot remains below the ground for the portion of the swing phase after the swing leg moves past the stance leg. Therefore, we are not able to model the time of collision as the time of the second impact. Instead, using the output of the frontal plane model, we estimate that a collision occurs once every half period of the roll oscillations.

At this moment, the collision is again modeled as an angular momentum conserving impulse:

$$\mathbf{\Omega}^+(\mathbf{q})\dot{\mathbf{q}}^+ = \mathbf{\Omega}^-(\mathbf{q})\dot{\mathbf{q}}^-,$$

where

$$\begin{aligned}
\Omega^-_{11} =& 2bd\cos(\theta_{sw}-\theta_{st}) - (b+d)R_s\cos(\theta_{sw}-\gamma) \\
& - 2bR_s\cos(\theta_{st}-\gamma) + 2R_s^2 + b^2 - bd \\
\Omega^-_{12} =& \Omega^-_{21} = (b-d)(b - R_s\cos(\theta_{sw}-\gamma)) \\
\Omega^-_{22} =& 0 \\
\Omega^+_{11} =& (b-d)[d\cos(\theta_{st}-\theta_{sw}) - R_s\cos(\theta_{st}-\gamma) + (b-d)] \\
\Omega^+_{12} =& -R_s(b-d)\cos(\theta_{st}-\gamma) - R_s(b+2d)\cos(\theta_{sw}-\gamma) \\
& + d^2 + 2R_s^2 + R_s b\cos(\theta_{sw}+\gamma) - b^2\cos(2\theta_{sw}) \\
& + d(b-d)\cos(\theta_{st}-\theta_{sw}) \\
\Omega^+_{21} =& (b-d)^2 \\
\Omega^+_{22} =& (b-d)(d\cos(\theta_{st}-\theta_{sw}) - R_s\cos(\theta_{st}-\gamma)
\end{aligned}$$

This simulation generates stable trajectories (see Figure 4) that are extremely similar to those generated by the compass gait (compare with [9], [4]), except that they are much more stable. Our dynamics do not model the edges of the feet, so our simulation actually models a passive walker shaped like two halves of an ellipsoid. Nevertheless, we have not been able to find any initial conditions from which the system does not return to a stable gait. Figure 4 was generated using the initial conditions with all variables set to zero and a slope of 0.027 radians, which is approximately the starting configuration that we use on our passive walker.

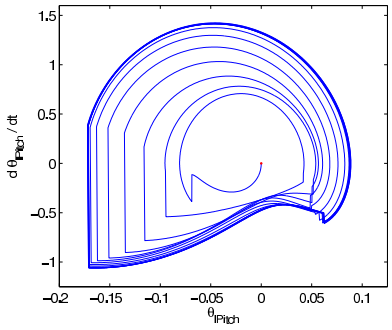

Fig. 4. Limit Cycles in the Sagittal Plane Model. θ_{lPitch} is the pitch angle of the left leg, which is recovered from θ_{st} and θ_{sw} in the simulation with some simple book-keeping.

The step length and the forward velocity of the steady state gait can be tuned by adjusting the radius of curvature, R_s. Smaller radii cause the robot to fall forward more quickly. For the slope of 0.027 radians, a simulation of our model predicts that our robot will take steps of 2.56 inches (6.50 cm) and walk with an average forward velocity of 8.81 inches/second (22.38 cm/s).

Simulations of the frontal and sagittal plane models, complete with the parameter values used to generate these plots, are available on the web [10]. A simulation of the full 3D dynamics will be available there soon.

C. Experiments

The planar models are tools for designing the curvature of the feet to approximately tune the step frequency and step length of our robot. The coupling between these models is more complicated, and we are currently studying them in a simulation of the full 3D dynamics. The most important characteristic of this coupling is that energy from the sagittal plane stabilizes oscillations in the frontal plane. Consequently, we should expect to observe smaller steps than that predicted by the sagittal plane model.

Using the curvature of the feet in the frontal (R_f) and sagittal (R_s) planes determined from the planar models, we machined experimental feet on our CNC milling machine. The surface of the feet are given by:

$$z = \sqrt{R_f^2 - x^2} - R_f + \sqrt{R_s^2 - y^2} - R_s.$$

Using these feet, our robot produces stable periodic trajectories when placed on a small decline. Figure 5 demonstrates this stability with a sample trajectory of the machine walking down a slope of 0.027 radians. θ_{roll} is the roll angle of the robot body, in radians, which was simply called θ in the frontal plane model. This data was actually recorded from the actuated version of the robot with its ankle joints mechanically locked in place.

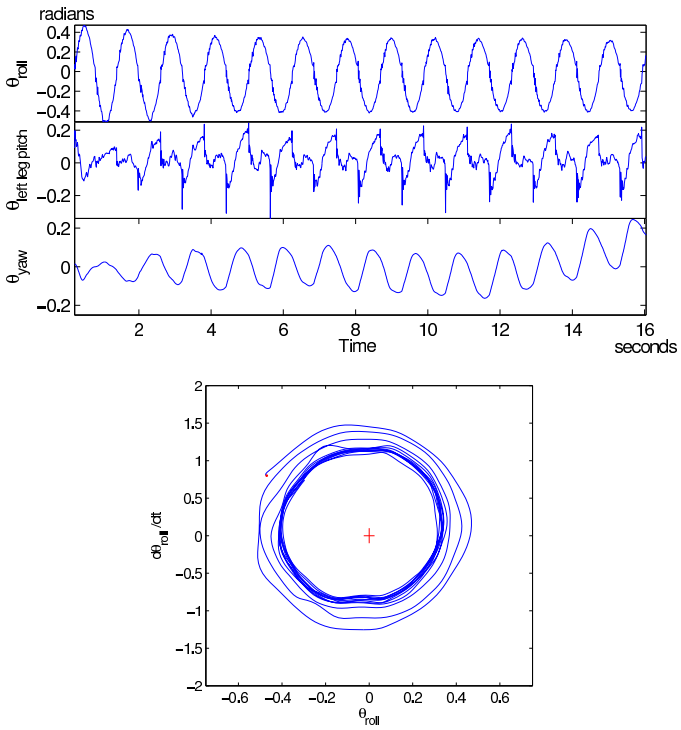

Fig. 5. Passive Walking Experiments. The top figure plots the raw (unfiltered) yaw, pitch, and roll sensors for walking down a slope of 0.027 radians. The bottom figure is the state space plot of the resulting limit cycle in the roll axis, low-pass filtered at with a cut-off at 8 Hz.

The limit cycle displayed in Figure 5 is fairly typical. Notice that the initial conditions for the robot are slightly outside the steady state trajectory, but that trajectories converge to a

very reproducible cycle in roll and pitch. The robot has an uncompensated moment about the yaw axis - it will twist and turn whenever the point contact of the foot slips on the ground. This can be seen in the wandering trace of the yaw variable.

Upon close inspection, we determined that the majority of the noise visible in our unfiltered data is actually due to a mechanical vibration of the leg at approximately 10 Hz. For this reason, we have decided to low-pass filter our limit cycle plots at 8 Hz with a 4th-order Butterworth filter (sampling rate is 100 Hz).

D. Local Stability Analysis

The local stability of the passive walkers is traditionally quantified by examining the eigenvalues of the linearized step-to-step return map [1], taken around a point in the period either immediately preceding or immediately following the collision. While we are currently designing a foot contact switch that will not interfere with the curved feet of the robot, the return map for this analysis is evaluated through the hyperplane $\theta_{roll} = 0$, when $\dot{\theta}_{roll} > 0$. The point in the cycle when the robot passes through the vertical position in the frontal plane is the expected point of impact.

The state of our passive robot is described by 4 variables (θ_{yaw}, θ_{lPitch}, θ_{rPitch}, θ_{roll}) and their derivatives, therefore the return map has dimension 7. $lPitch$ is short for left leg pitch and $rPitch$ is for the right leg pitch. To evaluate the eigenvalues of the return map experimentally on our robot, we ran the robot from a large number of initial conditions and created the vectors $\mathbf{x}_j^i$, 7×1 vectors which represent the state of the system on the ith crossing of the jth trial. For each trial we estimated $\mathbf{x}_j^*$, the equilibrium of the return map. Finally, we performed a least squares fit of the matrix $\mathbf{A}$ to satisfy the relation

$$(\mathbf{x}_j^{i+1} - \mathbf{x}_j^*) = \mathbf{A}(\mathbf{x}_j^i - \mathbf{x}_j^*).$$

This was accomplished by accumulating the data from all trials into matrices

$$\mathbf{X} = [\mathbf{x}_1^1 - \mathbf{x}_1^*, \mathbf{x}_1^2 - \mathbf{x}_1^*, \ldots, \mathbf{x}_2^1 - \mathbf{x}_2^*, \ldots]$$
$$\mathbf{Y} = [\mathbf{x}_1^2 - \mathbf{x}_1^*, \mathbf{x}_1^2 - \mathbf{x}_1^*, \ldots, \mathbf{x}_2^2 - \mathbf{x}_2^*, \ldots]$$

and computing

$$\mathbf{A} = \mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}.$$

After 63 trials with the robot walking down a ramp of 0.027 radians, our linear approximation of the return map had the following eigenvalues: 0.88, 0.87, 0.75, 0.70, $0.34 \pm 0.11i$. 61 trials with on a slope of 0.035 radians produces very similar eigenvalues (0.88, 0.70, $0.43 \pm 0.01i$, $0.36 \pm 0.08i$, 0.21). We have also studied the associated eigenvectors, but find them difficult to interpret since they are sensitive to the units and scale of our data. The largest eigenvalue of 0.88 indicates that this system is locally stable.

The distribution of equilibrium trajectories was unimodal and narrow for both slopes (examined separately). We believe that most of the variance in the distribution of equilibrium trajectories can be accounted for by sensor noise, small disturbances, and changes in the effective ramp angle when the robot yaws to one side or the other.

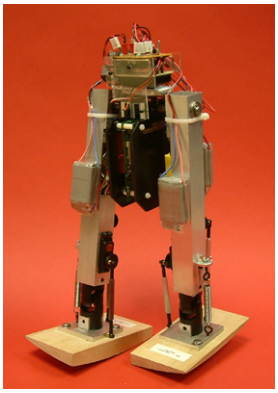

Fig. 6. The Toddler Robot

E. Domain of Attraction

In practice, the robot can be initialized from a large range of initial conditions and can recover from relatively large perturbations. Because the return map has some eigenvalues close to 1, this recovery takes many steps. Walking trials are nearly always started with the robot tilted sideways to a non-zero θ_{roll} position but with θ_{yaw}, θ_{lPitch}, and θ_{rPitch} close to zero. It is not necessary to give the robot any initial forward velocity.

When started in this configuration, one of three things happen. If $|\theta_{roll}|$ is too small, approximately less than 1.25ϕ, then the robot will converge to a stable fixed point at $\theta_{roll} = \dot{\theta}_{roll} = 0$. If $2\phi < |\theta_{roll}| < \psi$, where ψ is the angle at which the center of mass of the robot is directly above the outside edge of the foot, then the robot returns to a stable gait. For larger $|\theta_{roll}|$, the robot falls over sideways. On our robot, $\phi = 0.03$ radians and $\psi = 0.49$ radians, which makes for a very large basin of attraction in this dimension. Compare this with the predictions of the compass gait model, which must be initialized much closer to the steady state trajectory in order to produce stable walking.

III. Toddler - the Actuated Version

In order for the robot to walk on the flat, it must actively restore the energy lost during impact. One candidate active control strategy would be to apply torques at the existing hip joint, but it may be difficult to actuate the hip without disrupting the basic passive gait. On the robot shown in Figure 6, the hip joint is passive, but we have added two active joints (pitch and roll) at each ankle. We call this robot "Toddler" because the word is normally used to describe a child during the time that they are learning to walk, and this robot is primarily designed to investigate learning algorithms for dynamic walking. The name is also appropriate because the robot literally toddles back and forth when it walks.

Toddler's four active degrees of freedom are actuated by servo motors through mechanical linkages. They are configured so that when the motors are commanded to hold their

zero position, the robot simulates the passive walker. The robot is also equipped with a two axis gyroscopic tilt sensor (measuring θ_{lPitch} and θ_{roll}), three rate gyros (measuring $\dot{\theta}_{yaw}$, $\dot{\theta}_{rPitch}$ and $\dot{\theta}_{roll}$), and two potentiometers at the hips to measure the relative hip angles. Notice that this machine has one more degree of freedom than it's passive counterpart, which we call θ_{bPitch} or body pitch. The body has a center of mass below the hip, so it hangs passively, and primarily contains an embedded PC/104 stack with a 700 MHz processor which runs the control algorithms. Power is supplied by lithium polymer battery packs.

By assuming that the robot is always in contact with the ground, we can describe the generalized state of this robot with 5 variables (θ_{yaw},θ_{lPitch},θ_{bPitch}, θ_{rPitch}, θ_{roll}) plus their derivatives. Because the robot only has four actuators, it is clearly an under-actuated system. The challenge is to produce a control strategy for the ankle actuators which (directly or indirectly) controls all 5 degrees of freedom.

To solve this problem, we first focus our attention on stabilizing the oscillation in the frontal plane. The frontal plane model is a simplification of the dynamics of the robot on a ramp, but it is also a reasonable representation of the robot's dynamics when it is placed on a flat surface. The frontal plane model for the actuated version can be written as:

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{u},$$

where $\mathbf{q} = [\theta, \theta_{la}, \theta_{ra}]^T$ and $\mathbf{u} = [0, u_{la}, u_{ra}]^T$. The abbreviations la and ra are short for left and right ankle, respectively. The impact model can be written as

$$\mathbf{\Omega}^+(\mathbf{q})\dot{\mathbf{q}}^+ = \mathbf{\Omega}^-(\mathbf{q})\dot{\mathbf{q}}^-.$$

At each collision with the ground, the kinetic energy of the system, T, changes by:

$$\Delta T = \frac{1}{2}\dot{\mathbf{q}}^T \left[\mathbf{\Omega}(\mathbf{q})^T \mathbf{H}(\mathbf{q})\mathbf{\Omega}(\mathbf{q}) - \mathbf{H}(\mathbf{q})\right] \dot{\mathbf{q}},$$

where $\mathbf{\Omega} = (\mathbf{\Omega}^+)^{-1}\mathbf{\Omega}^-$. In order to stabilize the oscillations, the control torques, $\mathbf{u}$, must restore the energy lost from these collisions. In the following sections, we will discuss two simple actuation strategies that have already been applied successfully to the robot.

A. Feed Forward Ankle Trajectories

The first actuation strategy that we experimented with use the idea of coupled oscillators. The mechanical system in the frontal plane is a damped oscillator. If we produce a second oscillator in the control system which could entrain the dynamics of the mechanical oscillator, then the result would be stable walking. For simplicity, we started this experiment by considering only one-way coupling: a feed-forward control system which forces the mechanical oscillator.

The ankle servos are PD controllers which follow a reference trajectory. The oscillator is simply:

$$\begin{aligned}\theta^d_{ra} &= \alpha \sin(2\pi\omega t)\\ \theta^d_{la} &= -\theta^d_{ra}\end{aligned}$$

Whether or not the robot's dynamics could entrain to the dynamics of the controller depended on the relationship between the controllers frequency, ω, and the passive step frequency of the robot. Surprisingly, the best entrainment occurred when the controller was a little slower than the passive step frequency ($\omega = 0.55$ for our robot stepping at 0.8 Hz). We believe that the success of this simple controller can be attributed to the mechanical design of the robot.

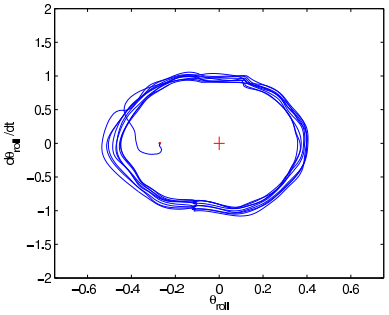

Fig. 7. Example Feed forward Limit Cycle

Local stability analysis of the limit cycles generated by the feed forward controller suggests that this controller is more stable than the passive system. The eigenvalues of the return map evaluated from 89 trials on flat terrain were 0.80, 0.60, $0.49 \pm 0.04i$, 0.36, 0.25, $0.20 \pm 0.01i$, 0.01. Practically, this controller converges from a large region of state space, but is very sensitive to disturbances in phase. The robot must be initialized in phase with the controller, and relatively small perturbations can knock it out of phase.

B. Feedback Ankle Trajectories

A more direct approach to stabilizing the roll oscillations is to build a controller which, on every cycle, injects exactly the amount of energy into the system that was lost during that cycle. Even simpler is the idea implemented by Marc Raibert's height controller for hopping robots ([11], [12]): if we inject a roughly constant amount of energy into the system during every cycle, then system will settle into a stable oscillation with an amplitude that is monotonically related to the energy injected.

Our heuristic for injecting a roughly constant amount of energy on each cycle is implemented using a state machine with only two states. The right ankle is given by:

$$\theta^d_{ra} = \begin{cases} 0.08 \text{ rad} & \text{if } \theta_{roll} > 0.1 \text{ rad and } \dot{\theta}_{roll} > 0.5 \text{ rad/s} \\ 0 \text{ rad} & \text{otherwise,} \end{cases}$$

and the left ankle controller is symmetric. With this state machine, as the robot rolls from an upright position onto one foot, it crosses a threshold position and velocity at which the stance ankle is servo-ed by a small fixed amount, causing the robot to accelerate further in the direction that it was moving. As the robot is moving back toward the upright position, the ankle is servo-ed back to the original zero position, which further accelerates the center of mass toward

the upright position. Both ankles are at the zero position at $\theta_{roll} = 0$ in order to minimize the energy lost by the collision with the ground. The desired angle in the non-zero state is monotonically related to the resulting amplitude of oscillation.

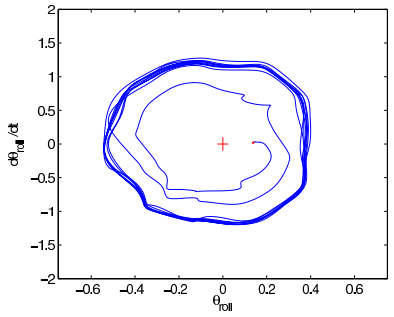

Fig. 8. Example Feedback Limit Cycle. This trajectory demonstrates a gradual convergence from small initial conditions.

The local stability analysis of this controller reveals that this controller converges more quickly than both the feed-forward controller and the purely passive device. After 58 trials on flat terrain, our linear return map analysis produced the eigenvalues: 0.78, $0.69 \pm 0.03i$, 0.44, $0.36 \pm 0.04i$, $0.13 \pm 0.06i$, 0.13. Practically, this controller is very stable. It is able to recover from very large disturbances, and from very small initial conditions.

C. Velocity Control

We control the dynamics of the sagittal plane independently using a simple velocity control algorithm. The robot walks in place when the center of mass of the entire robot is directly above the point of contact between the foot and the ground. When the center of mass is out in front of the ground contact point, the robot will lean forward. As soon as one leg leaves the ground, the passive joint at the hip allows it to swing forward, and the robot begins walking. This happens naturally when the robot is on an incline. The farther the center of mass is from the ground contact point, the faster the robot will move in that direction. On Toddler, the operator specifies the desired forward speed by joystick, and the corresponding placement of the center of mass is controlled by actuating the ankle pitch actuators. The heuristic makes it easy for Toddler to walk on flat terrain, and even up small inclines. For large changes in slope, the gains of the roll stabilizing controllers must also adapt.

The direction of the robot can also be controlled, to a limited degree, by differentially actuating the right and left ankles (either pitch or roll). Currently, the yawing of the robot due to momentum of the swing leg and slipping on the stance leg limit the controllability of the robot's heading. Future versions of the robot will have arms to compensate for this yaw, and should be able to turn more accurately.

IV. Conclusions and Future Work

The passive dynamic walker presented in this paper has only a few degrees of freedom, but it is capable of stable 3D dynamic walking. The dynamics are simple enough that they can be fully modeled and understood. We have presented the preliminary modeling work, which considered the frontal and sagittal planes individually, and are currently studying the three dimensional dynamics to better understand the coupling terms.

The control problem for the Toddler robot is interesting because although it is theoretically challenging (4 degrees of under-actuation), the mechanical design of the robot made it relatively easy to create controllers which allowed the robot to walk stably on flat terrain and even up a small slope. This feature of the robot makes it an excellent platform for studying machine learning control strategies, which is the true focus of our project. The simple controllers presented here are being used as baselines with which we can compare the performance of our learned controllers. In the future, we would also like to implement more elegant model-based, under-actuated control strategies that might allow for improved active control of step length and step frequency, and possibly allow the robot to walk over rough or intermittent terrain.

Acknowledgments

The authors would like to thank Derrick Tan for helping to make the robot wireless. This work was supported by the David and Lucille Packard Foundation (contract 99-1471), the National Science Foundation (grant CCR-0122419), Howard Hughes Medical Institute, and the MIT Undergraduate Research Opportunities Program.

References

[1] T. McGeer, "Passive dynamic walking," *International Journal of Robotics Research*, vol. 9, no. 2, pp. 62–82, April 1990.
[2] S. H. Collins, M. Wisse, and A. Ruina, "A three-dimensional passive-dynamic walking robot with two legs and knees," *International Journal of Robotics Research*, vol. 20, no. 7, pp. 607–615, July 2001.
[3] J. Pratt, "Exploiting inherent robustness and natural dynamics in the control of bipedal walking robots," Ph.D. dissertation, 2000.
[4] M. W. Spong and G. Bhatia, "Further results on control of the compass gait biped." International Conference on Intelligent Robots and Systems, 2003, pp. 1933–1938.
[5] J. Camp, "Powered "passive" dynamic walking," Master's thesis, 1997.
[6] R. Q. van der Linde, "Active leg compliance for passive walking." IEEE International Conference on Robotics and Automation (ICRA), 1998, pp. 2339–2345.
[7] A. D. Kuo, "Energetics of actively powered locomotion using the si mplest walking model," *Journal of Biomechanical Engineering*, vol. 124, pp. 113–120, 2002.
[8] M. J. Coleman and A. Ruina, "An uncontrolled toy that can walk but cannot stand sti ll," *Physical Review Letters*, vol. 80, no. 16, pp. 3658 – 3661, April 1998.
[9] A. Goswami, B. Espiau, and A. Keramane, "Limit cycles and their stability in a passive bipedal gait." IEEE Conference on Robotics and Automation, 1996, pp. 246 – 251.
[10] R. Tedrake, "http://hebb.mit.edu/people/russt/."
[11] M. H. Raibert, *Legged Robots That Balance*. The MIT Press, 1986.
[12] M. Buehler and D. E. Koditschek, "Analysis of a simplified hopping robot." IEEE Conference on Robotics and Automation, 1988, pp. 817–819.