Proceedings of the Thirty-Sixth Annual Allerton Conference on Communication, Control, and Computing, Monticello, Illinois, September 1998

Exploiting Natural Dynamics in the Control of a Planar Bipedal Walking Robot

Jerry E. Pratt and Gill A. Pratt
MIT Leg Laboratory
545 Technology Square
Cambridge, MA 02139
www.leglab.ai.mit.edu

Abstract

Natural dynamics can be exploited in the control of bipedal walking robots: the swing leg can swing freely once started; a kneecap can be used to prevent the leg from inverting; and a compliant ankle can be used to naturally transfer the center of pressure along the foot and help in toe off. Each of these mechanisms helps make control easier to achieve and results in motion that is smooth and natural looking.

We describe a simple control algorithm using these natural mechanisms which requires very little computation. The necessary sensing consists of joint angles and velocities, body pitch and angular velocity, and ground reaction forces.

Using this simple algorithm, we have controlled a seven link planar bipedal robot, called Spring Flamingo, to walk. Video, photographs, and more information on Spring Flamingo can be found at http://www.leglab.ai.mit.edu

1 Introduction

A powerful practice in machine design and control is to design mechanisms which have natural dynamics that make control simpler and more efficient. For example, planes have wings so that they naturally fly stably, requiring only a simple power source and simple control. Early locomotives used fly-ball governors, a mechanical feedback device, to help maintain constant speed. Satellites and rifle bullets spin to help maintain orientation.

Nature has evolved creatures that exploit the natural dynamics of similar mechanisms. Birds have wind-swept wings. Fish are neutrally buoyant. Running animals have springy legs. And walking animals have kneecaps, compliant ankles, and legs which can swing passively.

Many researchers [5, 10, 4, 1, 3], starting with Mc Geer and his passive dynamic walker, have exploited natural dynamics in order to make walking machines which are fully passive. These devices rely completely on their dynamics, and interaction with gravity, in order to walk.

Passive walkers have limitations, of course, such as limited capabilities and the need to walk down a slope. Powered robots [6, 16, 15, 14, 2, 8, 9, 11, 18, 19] can avoid these limitations. However, the control of powered bipedal robots has often been very complicated and the resultant motion often looks unnatural and is inefficient. Many of the controllers for powered robots are model based, requiring an accurate model of the dynamics of the

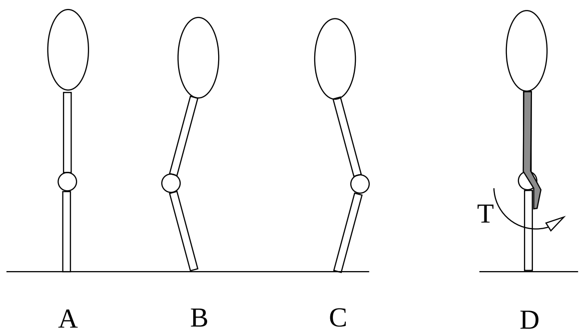

Figure 1: Diagram illustrating kneecap advantages. Without a kneecap, a biped with a straight support leg is in an unstable buckling configuration (A). Feedback control will result in chatter between knee inflections B and C due to delay, etc. With a kneecap (D), a constant torque with no feedback is enough to stablize the system against buckling.

robot in order to work. Several of the robots use trajectory tracking approaches which require pre-specified trajectories of either the body or the joints themselves.

In this paper, we describe an approach to powered bipedal walking which exploits the natural dynamics of the robot and requires only simple control. We exploit three different natural mechanisms. We use a knee cap to prevent the leg from inverting, which makes control of height easy. We use a compliant ankle limit so that the center of pressure on the foot travels forward with the center of mass of the body. And we exploit the natural swing dynamics of the leg to make swing control simpler and natural looking. This approach was used in both simulation and on a physical robot as described below. The robots walk fairly naturally and efficiently.

2 Natural Dynamic Mechanisms

2.1 Knee Cap

Walking with straight support legs is more efficient than with bent legs since energy requirements in muscles and motors are proportional to the torque at the joint, even if there is no velocity. However, since the leg must support the weight of the body, a straight leg poses an interesting challenge. Figure 1 illustrates the issue. When the body is directly over the foot (A), no torque is required at the knee. However, this is an unstable latch configuration. If the knee moves slightly either way, the leg will buckle (B or C). It is challenging to control this situation. Due to controller non-idealities (bandwidth limitation, delays, etc.) a straight knee controller will typically exhibit chatter between configurations B and C.

Adding a knee cap (D) can greatly simplify the control and make the resultant motion smoother and more efficient. A very simple control technique to keep the leg straight is to apply a constant torque so that the knee pushes against the stop. Of course other techniques can be used. Also, if the line of force on the body passes in front of the kneecap, the knee will be locked against the kneecap without any actuator torque.

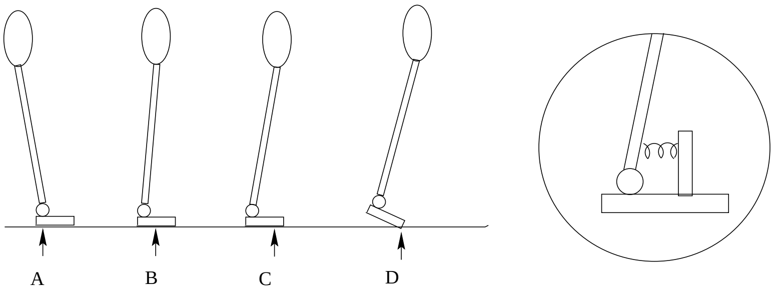

Figure 2: Diagram illustrating compliant ankle. In normal walking, the center of pressure on the foot travels forward as the center of mass travels forward (A-D). A compliant ankle (insert) can naturally achieve this effect. However, energy injection at toe off requires actuation.

2.2 Compliant Ankle

Feet and ankles provide many benefits to bipedal walking. They reduce velocity fluctuations since the center of pressure on the foot can travel forward, staying below the center of mass of the body. They also help to control speed and to inject energy at the end of the stride through toe off.

The torque at the ankle can be controlled actively. However, torque requirements can be quite high, since the foot provides a significant lever arm when the center of pressure is near the toe. A compliant ankle provides most of the benefits of a foot and ankle but without the torque requirements. An actuator can then be used in addition to the passive ankle for fine control and energy injection at toe off. Figure 2 illustrates the situation. In configuration A, the center of mass is behind the foot and there is zero ankle torque. In configurations B and C, the center of mass is travelling forward. The ankle torque increases, thereby moving the center of pressure of the foot forward from the heel to the toe. In configuration D, the robot goes into toe off, releasing the energy stored in configurations B and C and perhaps injecting some more, through active torques, to maintain walking. The inset shows a simple spring configuration which can give the ankle the desired compliance.

Choosing an adequate spring torque versus displacement curve is important in achieving the desired behavior. In the simulation discussed below we use a quadratic spring ($\tau = k(\theta - \theta_0)^2$) and tune the stiffness parameter. For the robot we use a rubber stop and adjust the position of the stop for best results.

2.3 Passive Swing Leg

In our previous algorithm for Spring Flamingo [15], as with most powered bipedal walkers, we used control techniques similar to those used for robotic arms to control the swing leg along a trajectory to a desired landing position. However, with a suitable leg, the natural swing dynamics are such that once the swing starts, the leg will continue without any intervention, as illustrated in Figure 3. Gravity alone can be used to initiate swing, as in the case of the passive dynamic walkers. Hip torque can be added in order to make the leg swing faster as in animals.

We use the passive swing properties of the leg in the control of our simulation and robot. In both cases, the hip is servoed forward to a desired angle and the knee is allowed to swing freely. At the end of the swing, moderate damping is added to the knee

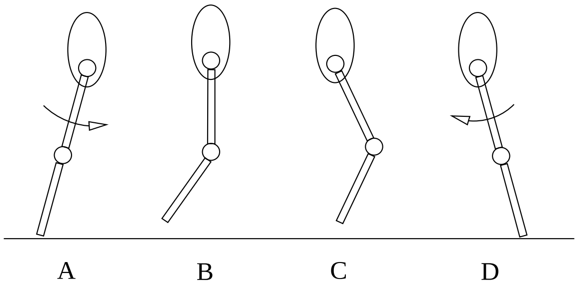

Figure 3: Diagram illustrating passive swing. Swing is initiated (A) through a forward torque on the hip, supplied either by hip actuators or gravity. The leg can swing passively (B - C) until swing is stopped (D) through a backward torque on the hip, again supplied either by hip actuators or gravity.

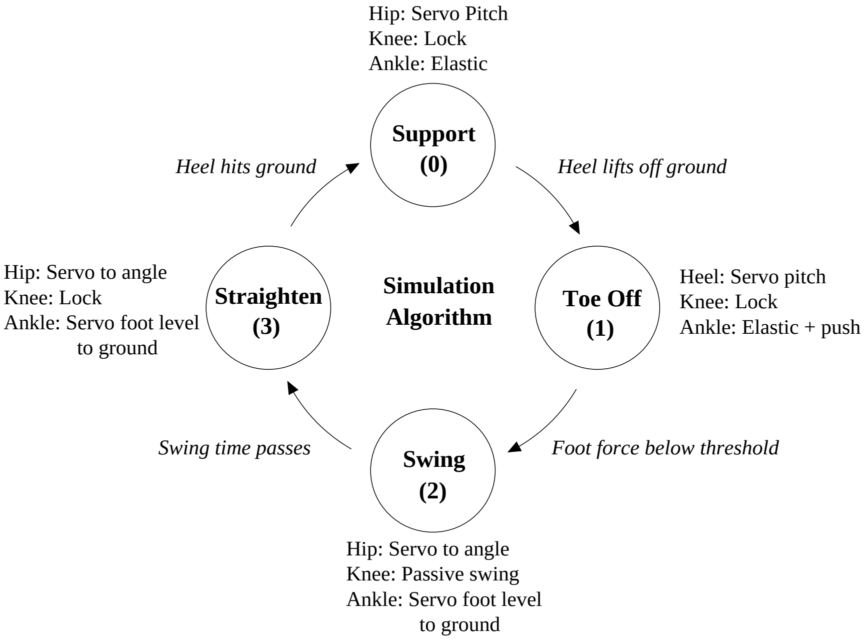

Figure 4: Simulation Algorithm. Each leg has a state machine which is in one of four states. State transition conditions and actions in each state are shown.

to prevent it from banging into the knee cap and finally it is locked once it hits the knee cap.

In order for the natural swing property to exist on Spring Flamingo, we had to invert the direction of knee bend. Previously, the robot had a backward bent joint, meant to model the ankle of a bird. However, the robot did not have the joint corresponding to a birds knee. By giving the robot a human style leg configuration, it obtains passive swing characteristics.

In the next sections, we describe algorithms which exploit the three natural mechanisms, described above, in the control of a simulated and a physical bipedal robot.

3 Simulation Algorithm

We use the natural dynamic mechanisms described above in the control of a simulated seven link bipedal robot. The simulation has an actuated hip, knee, and ankle on each

Table 1: Physical parameters and controller parameters of the simulated planar bipedal walker. The controller parameters were tuned both manually and with a genetic algorithm.

Physical Parameter	Value
Total mass	14.2 kg
Body mass	12.0 kg
Hip to body center of mass	0.20 m
Body moment of inertia	0.10 kg m^2
Upper leg mass	0.46 kg
Upper leg moment of inertia	0.13 kg m^2
Upper leg length	0.42 m
Lower leg mass	0.31 kg
Lower leg moment of inertia	0.0095 kg m^2
Lower leg length	0.42 m
Foot mass	0.35 kg
Foot moment of inertia	0.0014 kg m^2
Foot height	0.04 m
Foot length forward	0.17
Foot length back	0.06

Controller Parameter	Manual	GA
hip swing angle	0.5 rad	0.56
hip gain	15.0	12.9
hip damp	1.0	1.76
hip hold angle	0.35	0.40
knee lock gain	1.0	0.68
knee damping	0.5	0.68
foot force threshold	20.0	22.5
swing time	0.4	0.45
ankle push gain	4.0	5.07
ankle compliance	1000	384
pitch gain	100	(same)
pitch damp	20	(same)
knee stance gain	30	(same)
knee stance damp	10	(same)

leg. It is confined to walk in the sagittal plane. The simulation parameters are listed in Table 1. Moments of inertia are measured about the center of mass. The center of mass of each leg link is approximately in the center of the link.

The simulation algorithm is summarized in Figure 4. Each leg acts separately and has a simple state machine. The leg can be in either Support, Toe Off, Swing, or Straighten states. In Support and Toe Off states, the hip is used to servo body pitch to maintain balance and the knee is locked to maintain height. In Support state, the ankle is unactuated - only the passive ankle compliance is present. During Toe Off state, the ankle is servoed to an angle using a Proportional-Derivative (PD) controller in addition to its passive compliance. The transition from Support to Toe Off occurs when the heel lifts off the ground due to the passive compliance of the ankle.

The robot transitions from Toe Off to Swing when the force on the foot falls below a certain threshold. In both Swing and Straighten states the hip is servoed to an angle using a PD controller and the foot is servoed to be level with the ground so that the robot does not stub its toe. In Swing state, the knee is damped while in Straighten state the knee is locked straight using a PD controller.

The robot transitions from Swing to Straighten state after a constant amount of time passes. Finally, the robot transitions from Straighten to Support state when the heel of the swing leg hits the ground.

The simulation parameters were first manually tuned, and then fine tuned using a genetic algorithm with efficiency as its cost function. Efficiency was computed as distance travelled divided by total joint energy after ten seconds of walking. Total joint energy was computed by integrating the total joint power which is the sum of the absolute values of the mechanical power at each joint:

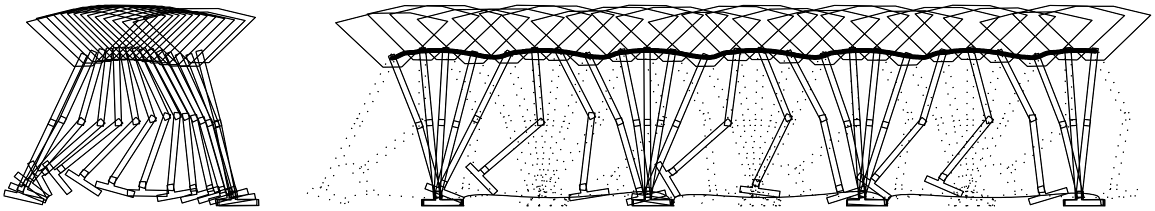

Figure 5: Elapsed time snapshot of the simulated robot walking data. The drawings on the left are spaced approximately 0.08 seconds apart and show the swing phase of one leg. The drawings on the right are spaced approximately 0.25 seconds apart. The right leg is dotted while the left leg is solid. Lines show the path of the tips of the feet and the hip trajectory. The robot walks from left to right.

$$E_{tot} = \int P_{tot}\, dt, \quad P_{tot} = \sum_{joints} |P_{joint}|, \quad P_{joint} = \tau_{joint}\, \dot{\theta}_{joint} \tag{1}$$

After a couple generations, naturally looking simulations resulted. Some of them achieved a smooth transfer of support by coordinating toe off and heel strike. One of those solutions was chosen to be presented here as it appeared quite natural. A time elapsed animation is shown in Figure 5. The drawings on the left show the swing phase of one leg. The drawings on the right show several steps. The right leg is dotted while the left leg is solid. Lines show the path of the tips of the feet and the hip trajectory. Several variables are plotted graphically in Figure 6. In the first row, left to right, are the state of the left leg (solid) and right leg (dotted), forward distance travelled (x), and velocity. In the second row are total power, body height, and pitch. In the last row are hip, knee, and ankle power of the left leg.

We see that the simulated robot walked at a moderate speed (approximately 0.8 m/s) and had a natural looking gait. The simulation requires less than 20 watts of joint power on average. Most of the energy consumption comes from the hips at the beginning of swing phase. It is interesting that the algorithm does not contain any explicit speed control mechanism, yet speed is stabilized. We speculate that this is due to the natural system dynamics, in the same way that speed is naturally stabilized in the passive dynamic walkers.

4 Robot Algorithm

Figure 7 contains photographs of Spring Flamingo, a seven link planar bipedal walking robot. The photo on the left is a close up of the knee joint and kneecap. The photo on the right is a close up of the compliant ankle stop. We exploit these two natural mechanisms, and the passive swing in the control of the robot.

The robot algorithm is summarized in Figure 8. As in the simulation algorithm, each leg acts separately and has a simple state machine. The robot algorithm differs from the simulation algorithm in two ways. First, because the robot does not have ground force sensors yet, we use geometric state transition conditions for initiation of swing. Second, instead of controlling each joint separately, in a decoupled manner, we use Virtual Model Control [14, 15] in the control of pitch and height.

The control parameters for the physical robot were manually tuned over the course of two days. Stable solutions were found that were fairly robust such that the robot

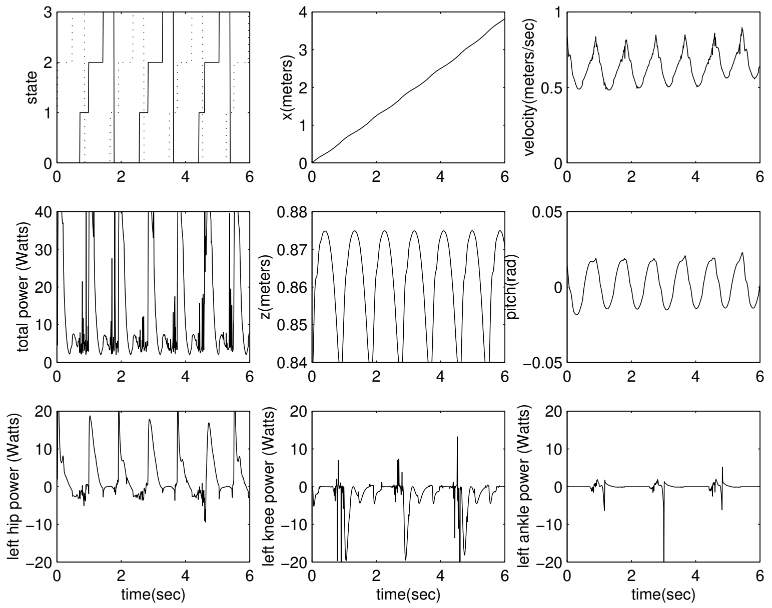

Figure 6: Simulation data. The first row contains, left to right, state of the legs, forward distance travelled, and forward velocity. The second row contains total power, body height, and body pitch. The last row contains joint power at the hip, knee, and ankle of the left leg.

walked continuously on level ground. A time elapsed animation of the robot is shown in Figure 9. The drawings on the left show the swing phase of one leg. The drawings on the right show several steps. Several variables are plotted graphically in Figure 10. In the first row, left to right, are the state of the left leg (solid) and right leg (dotted), forward distance travelled (x), and velocity. In the second row are total power, body height, and pitch. In the last row are hip, knee, and ankle power of the left leg.

We see that the physical robot walked at a casual speed (approximately 0.5 m/s) and had a natural looking gait. The robot requires less than 10 watts of joint power on average. As in the simulation, most of the energy consumption comes from the hips at the beginning of swing phase. Again, as in the simulation, it is interesting that the algorithm does not contain any explicit speed control mechanism, yet speed is stabilized.

5 Conclusions and Future Work

Spring Flamingo presently walks continuously by exploiting the natural dynamics of a kneecap, compliant ankle, and passive swing leg. Two very simple algorithms are used in simulation and on the physical robot. The resultant motion is fairly smooth and efficient. This work may help bridge the gap between passive dynamic walkers and powered bipedal robots.

Both the simulation and the robot settle on a stable speed of walking. The simulation walks approximately 0.8 m/s while the robot walks approximately 0.5 m/s. However, nowhere in either controller is speed explicitly controlled. We believe that the speed is stabilized in a similar way to passive dynamic walking machines. That is, if the robot goes too fast, it naturally takes a longer step due to the natural swing leg dynamics and hence slows down on the next step. Similarly, if the robot moves too slowly, it naturally takes a shorter step and hence speeds up on the next step. Speed is dependent

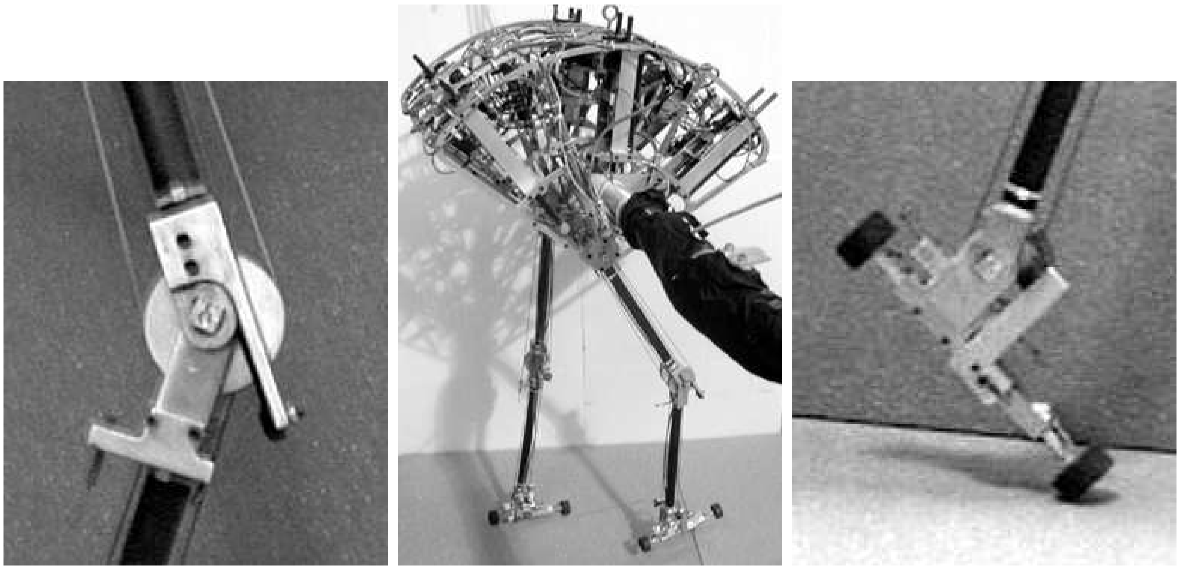

Figure 7: Spring Flamingo, a planar bipedal walking robot. There are six force controlled actuators attached to the body. Power is transmitted to the hips, knees, and ankles via cables. A boom prevents motion in the lateral, roll, and yaw directions.

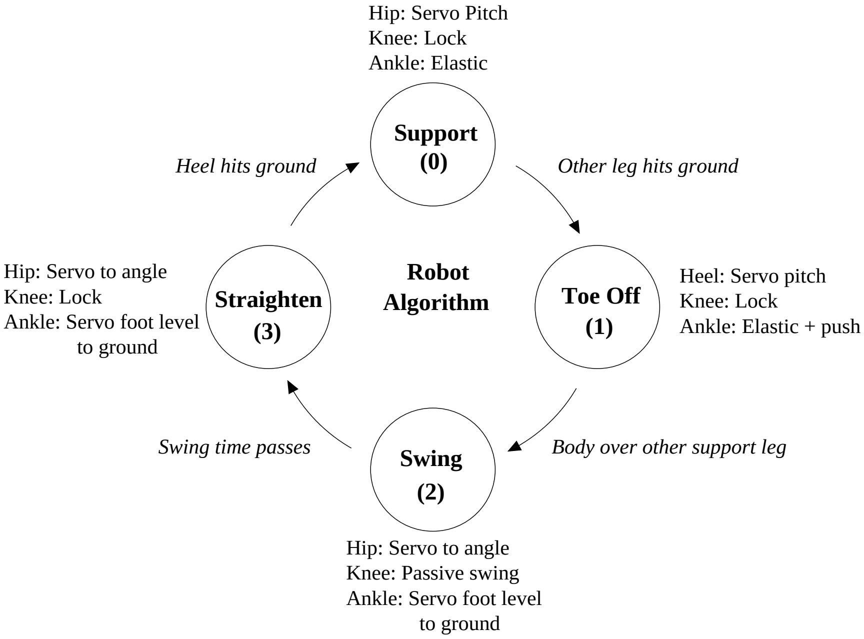

Figure 8: Physical Robot Algorithm. Each leg has a state machine which is in one of four states. State transition conditions and actions in each state are shown.

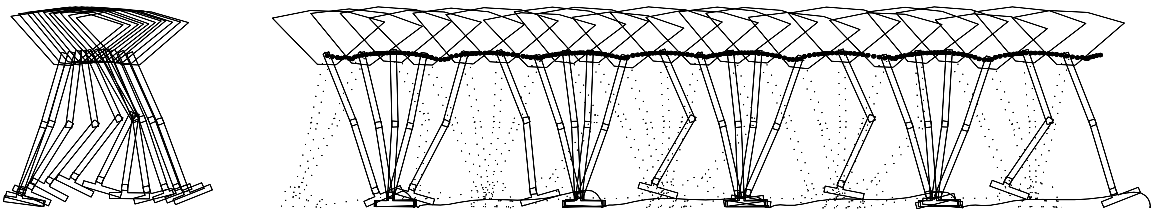

Figure 9: Elapsed time snapshot of the physical robot walking data. The drawings on the left are spaced approximately 0.07 seconds apart and show the swing phase of one leg. The drawings on the right are spaced approximately 0.4 seconds apart.

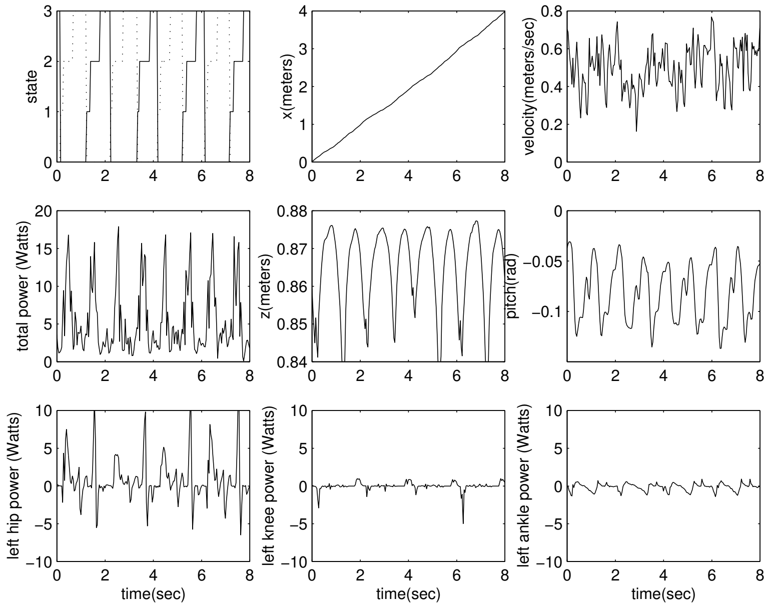

Figure 10: Physical robot walking data.

on parameter values. We plan on controlling speed and achieving faster walking through the modulation of these parameter values.

In order to exploit the natural dynamics of a walking robot, it is important that the inertia and friction of the actuators does not dominate the dynamics of the legs. On Spring Flamingo we use Series Elastic Actuators which result in very little unwanted dynamics. These actuators have good force dynamic range and low force offset which are important in natural and efficient walking. We are continuing to use these actuators in the design of several new walking robots.

Both the simulation and the physical robot required tuning of parameters in order to walk. The simulation was tuned first manually and then fine tuned using a genetic algorithm with efficiency as the cost function. The genetic algorithm found a set of parameters which nicely synchronized toe off of the support leg with heel strike of the swing leg such that a smooth transfer of support occurred and minimal energy was lost. The robot was only tuned manually but still walked fairly well. Its joint power averaged less than 10 watts. We are also considering tuning the robot using a genetic algorithm or other automatic techniques.

Spring Flamingo demonstrates that natural mechanisms can be exploited to make control of bipedal robots simple and the resultant motion natural looking and efficient. With minor changes to the algorithm, the robot should have the capability to start and stop and traverse slopes and go over obstacles. These results give us confidence that we can develop similar algorithms for three dimensional bipedal walking.

Acknowledgements

Dan Paluska designed and built the compliant ankle for Spring Flamingo. Dave Robinson designed and built the feet and force sensors. Spring Flamingo's actuators are based on a design by Mike Wittig. Dan, Dave, and Mike helped in many other aspects of the robot's design and construction.

References

[1] J. Adolfsson, H. Dankowicz, and A. Nordmark. 3-d stable gait in passive bipedal mechanisms. *Proceedings of 357 Euromech*, 1998.

[2] E. Dunn and R. Howe. Foot placement and velocity control in smooth bipedal walking. *IEEE Conference on Robotics and Automation*, pages 578–583, 1996.

[3] J. Fowble and A. Kuo. Stability and control of passive locomotion in 3d. *Proceedings of the Conference on Biomechanics and Neural Control of Movement*, pages 28–29, 1996.

[4] M. Garcia, A. Chatterjee, and A. Ruina. Speed, efficiency, and stability of small-slope 2d passive dynamic bipedal walking. *IEEE International Conference on Robotics and Automation*, pages 2351–2356, 1998.

[5] A. Goswami, B. Espiau, and A. Keramane. Limit cycles ini a passive compass gait biped and passivity-mimicking control laws. *Journal of Autonomous Robots*, 1997.

[6] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of honda humanoid robot. *IEEE Conference on Robotics and Automation*, 1998.

[7] L. Jalics, H. Hemami, and B. Clymer. A control strategy for adaptive bipedal locomotion. *IEEE Conference on Robotics and Automation*, pages 563–569, 1996.

[8] S. Kajita and K.Tani. Adaptive gait control of a biped robot based on realtime sensing of the ground profile. *IEEE Conference on Robotics and Automation*, pages 570–577, 1996.

[9] A. Kun and W. T. Miller. Adaptive dynamic balance of a biped robot using neural networks. *IEEE Conference on Robotics and Automation*, pages 240–245, 1996.

[10] Tad McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 9(2):62–82, 1990.

[11] H. Miura and I. Shimoyama. Dynamic walk of a biped. *International Journal of Robotics Research*, 3(2):60–74, 1984.

[12] Simon Mochon and Thomas A. McMahon. Ballistic walking: An improved model. *Mathematical Biosciences*, 52:241–260, 1979.

[13] Gill A. Pratt and Matthew M. Williamson. Series elastic actuators. *IEEE International Conference on Intelligent Robots and Systems*, 1:399–406, 1995.

[14] J. Pratt, P. Dilworth, and G. Pratt. Virtual model control of a bipedal walking robot. *IEEE Conference on Robotics and Automation*, pages 193–198, 1997.

[15] J. Pratt and G. Pratt. Intuitive control of a planar bipedal walking robot. *IEEE Conference on Robotics and Automation*, 1998.

[16] Marc H. Raibert. *Legged Robots That Balance.* MIT Press, Cambridge, MA, 1986.

[17] M. Vukobratovic, B. Borovac, D. Surla, and D. Stokic. *Biped Locomotion: Dynamics, Stability, Control, and Applications.* Springer-Verlag, Berlin, 1990.

[18] J. Yamaguchi, A. Takanishi, and I. Kato. Development of a biped walking robot adapting to a horizontally uneven surface. *IEEE International Conference on Intelligent Robots and Systems*, pages 1156–1163, 1994.

[19] K. Yi and Y. Zheng. Biped locomotion by reduced ankle power. *IEEE Conference on Robotics and Automation*, pages 584–589, 1996.