Proceedings of the 1997 IEEE
International Conference on Robotics and Automation
Albuquerque, New Mexico - April 1997

Self–Stabilizing Running

Robert Ringrose
MIT Leg Lab
MIT Artificial Intelligence Laboratory
545 Technology Square
Cambridge, MA 02139
ringrose@ai.mit.edu

Abstract

Legged robots can sustain stable dynamic locomotion without sensors or feedback. It is possible to construct a running robot that is inherently stable and needs no sensing to reject minor perturbations; it is therefore self-stabilizing. In contrast, most previous attempts to make robots run have used active, high bandwidth feedback control systems. We use a simplified monopod to give an understandable reason why self-stabilizing running should be possible. Physically realistic simulations running with one, two, or four legs demonstrate self-stabilizing running, as does a physical monopod robot.

Introduction

Running motions can be *self-stabilizing*. That is, with proper design the structure and motion of a robot can automatically cause it to recover from minor disturbances even if it cannot detect them. Many items in common use, such as tables, automobiles, and airplanes are stable about their standard operating conditions. Tables remain standing, automobiles drive forward, and airplanes fly straight and level [5]. This reliable default behavior makes them easier to control. In contrast, most current legged robots rely on high bandwidth sensing and feedback for stability. Self-stabilizing running could eliminate most of the bandwidth normally required to control running. This paper demonstrates some of the possibilities of self-stabilizing running by using it to control simulated one, two, and four legged robots in a variety of gaits, including a gallop. As a test of the simulations' validity, we built a mechanical self-stabilizing monopod.

Controlling most legged robots is like trying to balance a ball on top of a mountain: you must continuously check which way the ball is rolling and push it back to the top. Self-stabilizing running is like trying to balance a ball at the bottom of a valley. Unless the ball is completely out of the valley it will roll back to the bottom, where you want it. There is no need to actively sense which way it is rolling or push it back to the desired position.

In the self-stabilizing running robots we have studied, each joint has a perfect actuator in series with a spring. Over the running cycle, these actuators go through a predetermined pattern. There are no sensors, there is no conditional logic, and the actuator pattern does not change. With properly chosen actuator patterns and physical structure, the robot runs stably as a consequence of its design and the laws of physics. For the self-stabilizing quadruped in this paper, the phases between the legs are arbitrary. Therefore, it is capable of more gaits, including transverse and rotary gallops, than most other controllers such as the virtual leg controller [6]. Previous dynamic legged robots, such as those presented in [7] [8] [4], did not take advantage of self-stabilizing running because the conditions required for stable running were not known.

Other related work includes McGeer's passive dynamic walking and running machines [3] [2]. McGeer's devices locomote downhill by a simple interaction of gravity and inertia, with no actuators, no sensing, and no computers. In some respects, McGeer's work distills walking to its essence: a set of motions generated by a simple mechanism interacting with gravity, although his mechanism has some stability problems.

Juggling is much like walking except that the actuator (paddle) is on the ground, rather than part of the leg. Researchers have accomplished stable juggling with feedback [1] and without feedback [10]. Schaal and Atkeson's work indicates that juggling can also be self-stabilizing.

Simplified Monopod

We have simulations of self-stabilizing quadrupeds, bipeds, and monopods. Numerical methods show that these simulations are stable, but do not give a good intuitive understanding of why the self-stabilizing robots work. Our self-stabilizing quadrupeds are stable because they are formed by pairs of self-stabilizing bipeds, with the back counteracting disturbances and stabilizing the quadruped fore and aft. The bipeds are stable for similar reasons: each biped is two self-stabilizing monopods, with the pelvis and hips counteracting the limited offset between the two monopod's heights. Because the multi-legged robots depends on the monopod's stability, we first present an explaination of why the monopod is self-stabilizing.

The monopod consists of a mass, attached to a foot by a perfect linear actuator in series with a spring (Fig-

0-7803-3612-7-4/97 $5.00 © 1997 IEEE

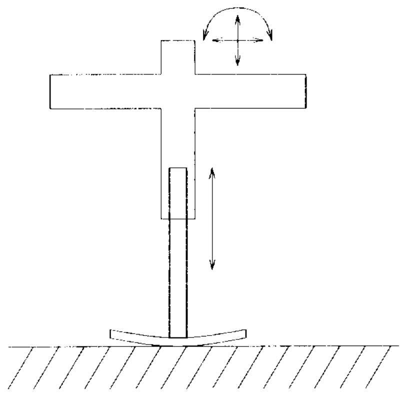

Figure 1: Monopod with curved foot. The model is planar, but can pitch. There is no joint at the hip.

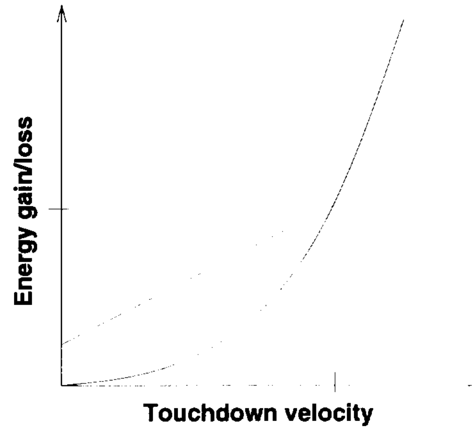

Figure 2: Energy removed by damping (solid) and added by the actuator (dashed) as a function of the touchdown velocity. The point where the two energies cross is the stable touchdown velocity. The total system energy ($E_{tot} = \frac{1}{2}mV_{td}^2$) is related to the touchdown velocity.

ure 1). There is a damper in parallel with the actuator and spring, representing friction in the vertical leg motion. The actuator is driven through a periodic motion with a fixed cycle time λ. The monopod can hop vertically, move horizontally, and pitch side to side. The actual monopod's foot has a mass which is small compared to the body. For this explanation only, we simplify the model by having a massless foot. We also approximate the leg actuator's motions with a single impulsive extension at time t_{extend}, of distance Δx, and a single retraction at some later point when the monopod is in the air. Experimentally, a large number of actuator motions, including sine waves, triangle waves, and square waves, will result in stable hopping motions.

The monopod's motions split into hopping height, touchdown phase, and pitch. First, we show that if the touchdown phase is constant and the monopod can only move vertically there is a stable hopping height. The stable hopping height is the point at which the actuator adds the same amount of energy as the damper removes over the course of a single cycle.

Second, we show that the phase is stable. The stable touchdown phase is such that when the height stabilizes the time on the ground plus the time in the air is one complete actuator cycle. If the monopod touches down late, it will lose a little energy and remain in the air less, bringing the next touchdown closer to the stable touchdown phase (vice versa for an early touchdown time).

Finally, we allow the monopod to pitch side to side and move horizontally, and demonstrate that a properly shaped foot can stabilize the additional degrees of motion. The pitch stabilization relies on an extension of the restoring forces you get from a wheel with an off-center mass.

Height Stabilization

The hopping height reaches equilibrium when the energy added to the hop by the actuator is balanced by the energy removed by the damper, and the total time in the air and on the ground is the same as the actuator's cycle time. We will address the relationship between hopping frequency and actuator frequency later; for now assume that the time of touchdown t_{td} preceeds t_{extend} by a constant amount of time, and that t_{extend} occurs before maximum spring compression.

For the hopping height to be in a *stable* equilibrium, if the total energy is greater than the equilibrium value the energy removed must be greater than the energy added, and vice versa when the total energy is lower. Additionally, the total energy removed or added must be small enough that the hopper will approach the equilibrium value without overshooting too badly. The energy introduced by the actuator motion is linear with respect to the touchdown velocity, while the energy removed from the damping effects is a quadratic function of the touchdown velocity. Where these functions cross, the hopping height is stable (Figure 2).

The remainder of this section is a mathematical justification of the existence of a stable hopping height. We find the approximate energy changes due to damping and leg extension, and find conditions which will allow them to be equal. We then calculate the total energy change due to an error in touchdown velocity, and show that this energy error decreases over successive hops.

$$\Delta E_d = -\int_{t_{td}}^{t_{to}} bV^2(t) \tag{1}$$

$$V(t) \approx \tilde{V}(t)V_{td} \tag{2}$$

$$\Delta E_d = -\mathcal{K}V_{td}^2 \tag{3}$$

Equation 1 is the formula for damping, in terms of velocity and a damping coefficient. For a lightly damped spring, the velocity over time approximately scales linearly with the touchdown velocity as in equation 2. $\mathcal{K}$ is constant with respect to the touchdown velocity ($\mathcal{K} = b\int_{t_{td}}^{t_{to}} \tilde{V}^2(t)$) and makes the formulation for the energy loss in equation 3 have an obvious form.

The energy change due to the leg extension is a bit more complex. Define the height at touchdown as zero, so the total energy upon touchdown E_{td} with velocity V_{td} is completely kinetic. Let $x(t)$ be the spring compression, with x_{max} being the compression at time t_{max}, the time of maximum compression if the actuator were not to extend. At the time the actuator extends, $x_{extend} = fx_{max}$, with $f \in [0..1]$.

$$x_{max} = \frac{1}{k}\left(mg \pm \sqrt{m^2g^2 + kmV_{td}^2}\right) \quad (4)$$

$$x_{max} \approx \frac{mg}{k} + V_{td}\sqrt{\frac{m}{k}} \quad (5)$$

$$\Delta E_a = \frac{k}{2}\left(2\Delta x f x_{max} + \Delta x^2\right) \quad (6)$$

Equation 4 comes from setting the energy at time t_{max} (without actuator extension) equal to the touchdown energy and solving for x_{max}. Under most useful circumstances, the m^2g^2 within the square root in the equation for x_{max} can be approximated away (Equation 5). Subtracting the total energy before and after the leg actuator extends gives the energy change, ΔE_a. The energy ΔE_a added by the actuator's extension is therefore expressable in terms of f and V_{td}.

$$\Delta E_a = \Delta x f\left(mg + V_{td}\sqrt{mk}\right) + \frac{k}{2}\Delta x^2 \quad (7)$$

Setting $\Delta E_a = \Delta E_d$ (equations 3 and 7) and solving for the touchdown velocity V_{td} gives the reentrant touchdown velocity V_0 in terms of f.

$$V_0 = \frac{\sqrt{mk}}{2\mathcal{K}} f\Delta x \pm \frac{1}{2\mathcal{K}}\sqrt{mk\Delta x^2 f^2 + 4\mathcal{K}mgf\Delta x + 2\mathcal{K}k\Delta x^2} \quad (8)$$

Because all the values in the square roots are positive, V_{td} is guaranteed to exist for any given f and Δx. Additionally, to have a positive velocity at touchdown, we must use the positive square root. Therefore, given f there is a touchdown velocity V_0 which corresponds to no energy change over a complete hop.

Showing that this energy level is stable requires two parts. Given a touchdown velocity $V_0 + V_e$, the energy change must have sign opposite the energy error caused by V_e and must result in an energy change of less than twice the energy error caused by V_e. The sign change is necessary to ensure that over time the velocity error disappears, and the maximum velocity change prevents an overall energy error increase from overshooting V_0.

Given a touchdown velocity $V_0 + V_e$, the energy changes due to damping and the actuator can be added to get the energy change over a cycle ΔE.

$$\Delta E = \Delta x f\sqrt{mk}V_e - 2\mathcal{K}V_0V_e - \mathcal{K}V_e^2 \quad (9)$$

For ΔE to have a sign opposite V_e, you need $2\mathcal{K}V_0 + \mathcal{K}V_e > \Delta x f\sqrt{mk}$. Substituting the right side of equation 8 for V_0, a lower bound on V_e is $V_e > -V_0$, and there is no upper bound.

To guarantee that the change in energy is not too great, consider the energy error ΔE_e resulting from a velocity error V_e, set the change in energy over the step to be enough that the energy error magnitude increases, and do some algebra to get a necessary condition for the energy to oscillate unstably.

$$\Delta x f\sqrt{mk} > (\mathcal{K} - m)(2V_0 + V_e) \quad (10)$$

If $\mathcal{K} < m$, the fact that $\Delta x f\sqrt{mk} \geq 0$ results in $0 < 2V_0 + V_e$, an inequality which is true. In the case where $\mathcal{K} > m$, the hopping motion is only stable if $V_e < \frac{\Delta x f\sqrt{mk}}{\mathcal{K} - m} - 2V_0$. As the damping effects increase, the maximum allowable error steadily decreases.

Therefore, as long as the damping is small ($\mathcal{K} < m$) and the touchdown velocity error is less than the touchdown velocity, given f and Δx there exists a touchdown velocity V_0 which is stable.

Phase Stabilization

If the phase of the monopod is stable, it should hop with a touchdown time t_{tds} fixed relative to the actuator motion and a period λ which is the same as the actuator motion's period. Just as with the hopping height, we will show that the phase has a fixed point, and that this fixed point is locally stable.

The existence and stability of the fixed point comes from the relationship between phase and the velocity at takeoff. Earlier touchdowns result in larger vertical velocities, while later touchdowns result in smaller vertical velocities. The time in the air is directly proportional to the vertical velocity, and the time on the ground is a constant determined by the spring and mass (neglecting damping and leg extension). Given an actuator oscillation period λ, there is a touchdown time such that the time in the air plus the time on the ground is λ (there are some limits on λ, which we will discuss later). If the monopod should touch down late, the smaller vertical velocity would decrease flight time, making the next touchdown closer to the fixed point, and vice versa for early touch down times.

To justify this explanation of hopping phase stability, we calculate the amount of time and in the air, as a function of when the monopod touches down, assuming a stable hopping height for that touchdown time. This information gives a function from the touchdown time of one cycle to the touchdown time for the next cycle. The function has a fixed point, and perturbations from the fixed point will be pushed back toward the fixed point.

To show that it will achieve a stable phase, assume that the actuator has a single time t_{extend} at which

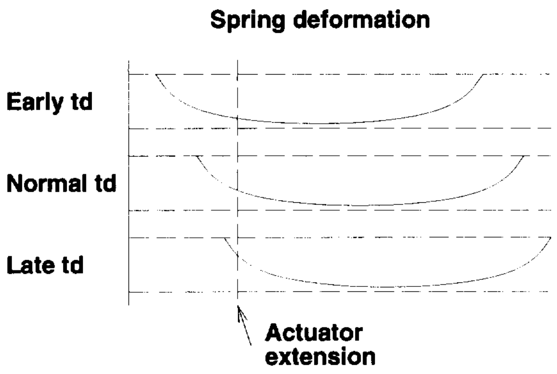

Figure 3: Fraction of maximum compression changes with time of touchdown.

it extends, as before. Equation 8 relates the fraction f of maximum compression at which the actuator extends to a touchdown velocity V_0 which is stable assuming f does not change from hop to hop. While the monopod is on the ground, it follows the sinusoidal path $x(t) = \mathcal{A}\sin\left(\sqrt{k/m}(t - t_{td})\right)$ with $\mathcal{A}$ determined by the touchdown velocity (this equation neglects damping and leg extension). Therefore, as long as the leg extension occurs before the sinusoid reaches maximum compression, the fraction of maximum compression at which the actuator extends (Figure 3) is:

$$f = \cos\left(\sqrt{\frac{k}{m}}\left(t_{td} - t_{extend} + \frac{\pi}{2}\sqrt{\frac{m}{k}}\right)\right) \qquad (11)$$

As a consequence, any t_{td} has a touchdown velocity V_0 which would be stable, assuming that the next touchdown also occurred at t_{td}. If t_{td} increases to $t_{tdmax} = t_{extend}$, where f approaches zero, the appropriate touchdown velocity decreases to V_{tdmax}. At the other end of the spectrum, if t_{td} decreases to $t_{tdmin} = t_{extend} - \frac{\pi}{2}\sqrt{\frac{k}{m}}$, so that f approaches 1, the stable touchdown velocity increases.

We need a relationship between the time of touchdown and the time until the next touchdown. Simplify the problem by assuming that upon touchdown, the velocity immediately changes to the touchdown velocity V_0 which corresponds to the f value associated with t_{td}, with no further energy changes. The monopod bounces on the ground in a sinusoidal manner until the time $t_{td} + \pi\sqrt{\frac{m}{k}}$, when it takes off. By symmetry, at takeoff it has velocity V_0 upwards. It will remain in the air until gravity has completely reversed the velocity. At time $t_{next}(t_{td}, V_0)$, the monopod will touch down again.

$$t_{next}(t_{td}, V_0) = t_{td} + \pi\sqrt{\frac{m}{k}} + \frac{2}{g}V_0 \qquad (12)$$

If t_{td} is as late as possible, the next touchdown will be at $t_{next}(t_{tdmax}, V_{tdmax})$. Similarly, if t_{td} is as early as possible the next touchdown will be at $t_{next}(t_{tdmin}, V_{tdmin})$. These give a bound on λ within which a fixed point t_{tds} exists.

$$\pi\sqrt{\frac{m}{k}} + \frac{2}{g}V_{tdmax} \quad < \quad \lambda < \pi\sqrt{\frac{m}{k}} + \frac{2}{g}V_{tdmin} \qquad (13)$$

$$\pi\sqrt{\frac{m}{k}} + \frac{2}{g}V_{tds} \quad = \quad \lambda \qquad (14)$$

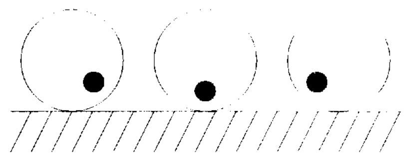

Figure 4: Wheel with an off-center mass. If the wheel rolls to one side, the mass is no longer above the contact point and applies a restoring torque.

If λ is within the given bounds, a fixed point t_{tds} exists. If the monopod touches down at time t_{tds}, the next touchdown will occur at time $t_{tds} + \lambda$. If monopod touches down at time $t_{tds} + t_{error}$, the next touchdown will occur at time $t_{tds} + t_{error} + \pi\sqrt{\frac{m}{k}} + \frac{2}{g}V_{tds+error}$. However, velocity is a decreasing function of of touchdown time. Rewriting in terms of V_{tds} and V_{error}, the velocity at touchdown and the velocity change due to the touchdown time error, the next touchdown time is $t_{tds} + \lambda + t_{error} - \frac{2}{g}V_{error}$. Therefore, if the touchdown time is wrong it will be forced back towards the fixed point. This does not necessarily indicate that the fixed point is stable. A perturbation about t_{tds} could be overcorrected if $\frac{1}{g}V_{error} > t_{error}$. We have not witnessed this behavior in any of our simulations, or in our monopod robot.

Pitch Stabilization

Given a monopod which hops stably vertically, with a constant phase, the remaining unstable degree of freedom is pitch. We base the pitch control on the curvature of the foot. It works much the same way as a wheel with an off-center center of mass: the wheel will roll side to side, until the damping of its rolling motion places the center of mass at its lowest point (Figure 4).

For the monopod, though, the "wheel" is only on the ground part of the time. Moving a wheel's center of mass further off-center makes it more stable, but if the wheel is periodically off the ground moving the center of mass too far could have the opposite effect. If the pitch is strongly corrected while the wheel is on the ground, it may take off with a high angular momentum. While the wheel is in the air, it will continue to rotate, and may land with a much greater pitch offset than when it took off (Figure 5). Therefore, there is a limited set of foot radii which will stabilize hopping motion, but if the parameters of the monopod are chosen properly the

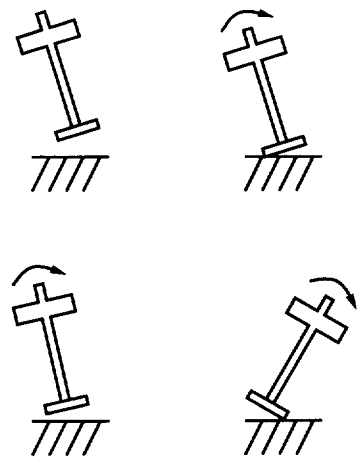

Figure 5: Monopod hopping, overcorrecting. If the foot is too flat, when the monopod comes down with a slight pitch (upper left) it will receive a large torque pitching it towards vertical (upper right). However, if it takes off too soon it may continue rotating in the air (lower left) and land with an even greater pitch error than it had on the previous step (lower right).

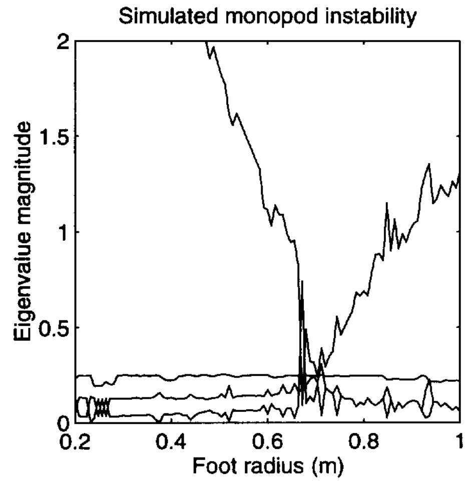

Figure 6: Instability of the monopod simulation as the foot radius changes. This is a plot of the eigenvalue magnitudes of the linearized discrete-time monopod equivalent, as a function of the foot's radius of curvature. If any eigenvalue has magnitude greater than one, the monopod is unstable for that foot radius and will tip over.

set is nonempty. The mathematics are covered in more detail in [9].

Results

For space, this paper contains only results from a hopping monopod, trotting quadruped, and a galloping quadruped. Further examples are available in [9]. All of the simulations are physically realistic and locally stable. Earlier approximations, such as an impulsive actuator and a massless foot, were solely to simplify the explaination of the self-stabilizing properties and are not present in the simulations. We have accoplished self-stabilizing running with a monopod, biped (in phase, out of phase, and various stages in between), and quadruped. The quadruped can trot, pace, bound, rotary gallop, and transverse gallop.

None of these robots and simulations have any sensing or feedback. In accordance with the principles of self-stabilizing running, they have a cycle timer and a fixed translation from timer values to actuator positions.

All of these self-stabilizing robots and simulations are locally stable. Given the entire state of the robot, positions and velocities, one can calculate the state of the robot one cycle later (for complex robots, this simply involves running the simulation for a single cycle). This fact allows one to turn the robot into a discrete-time system, with stable running paths turning into stable fixed points of the discrete-time system. The eigenvalues of a linear approximation about the fixed point of the discrete-time system indicate the local stability. In all cases presented here, the eigenvalue magnitudes were less than one, indicating local stability.

Monopedal Hopping

The monopod simulation consists of a single telescoping leg, with a spring and actuator in series as described earlier (Figure 1). The actuator motion is a pieced-together polynomial chosen for a stable, visually appealing hop. The leg actuator length $l(t)$, given an oscillation period λ and an oscillation magnitude C, is:

$$l(t) = \begin{cases} C((2 * t/\lambda)^2 - 1) & \text{if } t < \lambda/2 \\ -C^{\frac{0.5^2 - t/\lambda + (t/\lambda)^2}{.3^2}} & \text{otherwise} \end{cases} \tag{15}$$

The monopod's foot is a circle with a fixed radius. As the foot radius decreases, the monopod becomes unstable. Similarly, as it increases the monopod begins to overcorrect for pitch errors and, again, becomes unstable. Figure 6 graphs the eigenvalue magnitudes of the discrete-time monopod equivalent and illustrates this behavior. If any of the eigenvalues are greater than one, the monopod is unstable for that foot radius.

For a properly chosen foot radius, the monopod recovers quite well. As a test, we ran a physical monopod (Figure 7) and compared it to the simulation results.[1] Its actuator path is approximately a sinusoid, a path which the simulations predict will be stable. The physical monopod's recovery, using a foot radius predicted by simulation, is consistent with the simulated results.

[1]Thanks to David Robinson, MIT Leg Lab, who built the actual robot.

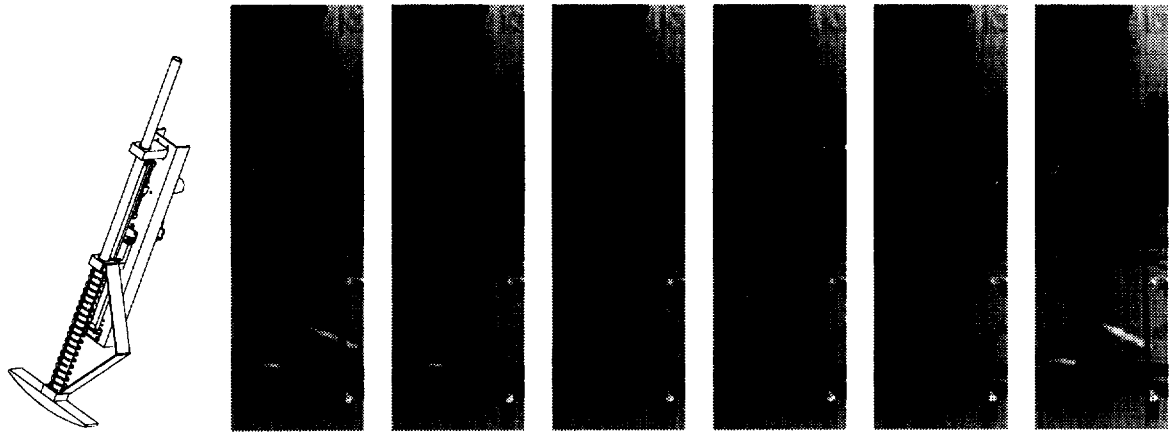

Figure 7: The self-stabilizing monopod hopper. The boom prevents motion towards the camera, but allows motion side-to-side and pitching motion. These frames are taken left to right in succession from videotape, 0.133 seconds apart (four frames).

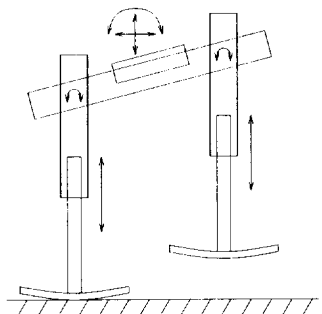

Figure 8: Biped with curved feet. The model is planar. There are pin joints at the hips, and linear joints along the legs. The stability of the monopods helps stabilize the biped locomotion.

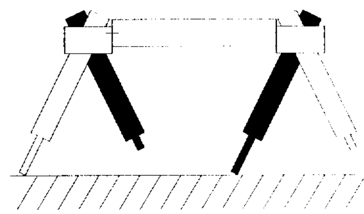

Figure 9: Quadruped with a twisting back. Note the two pins connecting the back to the hips; these pins are rotational joints. The front and back bipeds are stabilized by each other and the spine.

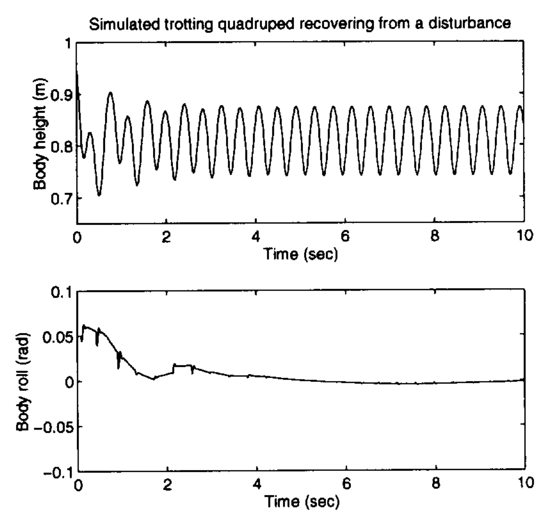

Figure 10: Trotting quadruped recovering from a disturbance in both roll and vertical position. The disturbance occurs at time $t = 0$. The foot radius is 0.75m.

Quadrupedal Trotting

The self-stabilizing quadruped (Figure 9) consists of two self-stabilizing bipeds (Figure 8) connected by a back. The back can rotate along the nose-to-tail axis, allowing the two bipeds to rotate independently. The back is long enough that the two bipeds stabilize each other along the back and do not tip over. Each biped is a pair of monopods connected by rotating hips to the ends of a pelvis, running stably in a plane without feedback. The quadruped itself is fully three dimensional and can move in any direction.

Trotting is a gait which has diagonal pairs of legs on the ground together. Figure 10 has data from a stable trotting quadruped recovering from a disturbance.

Quadrupedal Galloping

A bound has the front and back pairs of legs on the ground alternately. A gallop is similar to a bound, except

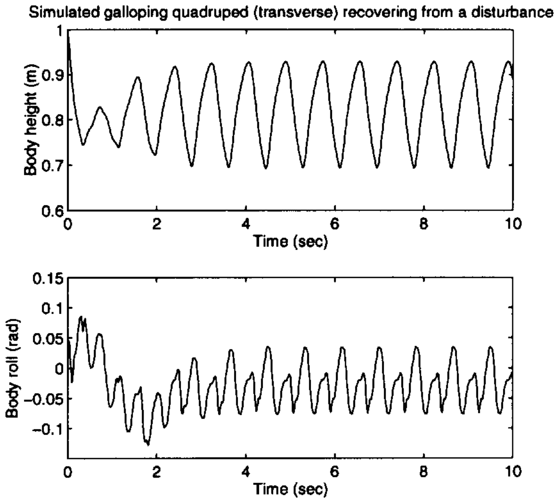

Figure 11: Galloping quadruped (transverse) recovering from a disturbance in both roll and vertical position. The disturbance occurs at time $t = 0$. The foot radius is 0.75m.

that the legs do not touch down at the same time. In a transverse gallop, the legs on one side always come down before the legs on the other side. For example, the foot order could be front left touches down, front right touches down, front left and then right lift into the air, back left touches down, back right touches down. Figure 11 has data from a stable transverse galloping quadruped recovering from a disturbance.

Conclusion

We found that running can be self-stabilizing. It is possible to trot, pace, bound, and gallop without any sensors. We believe that the technology for creating running robots exists, but the control and sensing has a long way to go. A self-stabilizing runner should make a useful base upon which one could layer more complex control systems, expanding the range in which the quadruped runs robustly. Should the more complex control systems fail, the self-stabilizing robot has a reasonably competent default behavior.

Self-stabilizing running, as we have implemented it, is only valid on flat surfaces. Depending on the nature of the terrain, the variances in height could be considered either disturbances to the running motion or problems which must be addressed by a higher level control system. If the ground has little variation, it can be considered a disturbance to the running motion and no modification to the control is necessary. If the ground is irregular enough that further control is necessary, the extra control could be as simple as predicting the touchdown altitude and modifying the leg length and leg angle appropriately.

In the nearer future, we believe that the next generation of active robots will have a self-stabilizing base, with feedback control modifying or overriding the basic stable motion as necessary. This combination could combine the best of feedback control and self-stabilizing running. The self-stability would simplify the feedback control, and the feedback control would give the robot the versitility necessary to make a running robot practical.

References

[1] M. Buhler. *Robotic Tasks with Intermittent Dynamics.* PhD thesis, Yale University, May 1990.

[2] T. McGeer. Passive bipedal running. Technical Report CCS-IS TR 89-02, Simon Fraser University, 1989.

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

[4] S. S. Murthy and M. H. Raibert. 3d balance in legged locomotion: Modelling and simulation for the one-legged case. In *Inter-Disciplinary Workshop on Motion: Representation and Perception.* ACM, 1983.

[5] Pan American Navigation Service, 12021 Ventura Boulevard, North Hollywood, California. *The New Private Pilot*, 9th edition, Jan 1972.

[6] M. H. Raibert, M. Chepponis, and B. Brown, Jr. Running on four legs as though they were one. *IEEE Journal of Robotics and Automation*, RA-2(2), June 1986.

[7] M. H. Raibert, J. K. Hodgins, R. R. Playter, and R. P. Ringrose. Animation of maneuvers: Jumps, somersaults, and gait transitions. In *Imagina*, Jan 1992.

[8] R. Ringrose. Simulated creatures: Adapting control for variations in model or desired behavior. Master's thesis, Massachusetts Institute of Technology, Dec. 1992.

[9] R. Ringrose. *Self-Stabilizing Running.* PhD thesis, Massachusetts Institute of Technology, 1996.

[10] S. Schaal and C. G. Atkeson. Open loop stable control strategies for robot juggling. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 913–918, Atlanta, Georgia, 1993.