

Designing Feedback Algorithms for Controlling the Periodic Motions of Legged Robots

J.P. Ostrowski and J.W. Burdick
Dept. of Mechanical Engineering, Mail Code 104-44, CALTECH Pasadena, CA 91125

Abstract.

This paper develops a nonlinear feedback control algorithm for repetitive legged robotic locomotion whose global dynamic behavior can be encoded in a Poincaré return map. Control synthesis is equivalent to the selection of a nonlinear feedback control law which properly “shapes” the return map. We present a method for return map shaping in a neighborhood of a fixed point. In practice, this neighborhood can be quite large. The theory is illustrated with applications to a simplified model of Raibert’s hopping machines. An added benefit is accurate control of hopping height above undulating terrain.

1. Introduction

Several investigators [KB91, VBC, MB91, MB93, LG90] have recently used dynamical systems concepts to analyze the *global* stability of dynamically stable legged locomotion systems. These recent efforts have considered the question: “given a particular robot mechanism (which may contain nonlinear mechanical elements) and a particular feedback control algorithm, is the closed loop system stable? In particular, is it globally stable?” For example, Koditschek and Bühler [KB91] considered a simple vertical hopping model which is analogous to the physical design and control methodology used by Raibert [Rai]. They showed that in addition to stable uniform vertical hopping motions, an anomalous period-2 hopping behavior could exist. Vakakis and Burdick [VBC] extended the model in [KB91], corroborating the period-2 bifurcation. [VBC] also constructed the global bifurcation diagrams with respect to system and control parameters, showing a period-doubling cascade into chaos. These undesirable higher period motions can arise because Raibert’s algorithm is based largely on kinematic intuition, and may not take into account the possibly nonlinear dynamics of a given mechanism. M’Closkey and Burdick [MB91, MB93] extended this analysis to a model which included forward running dynamics and Raibert’s leg placement algorithm. The anomalous period-doubling behavior exists for this more realistic model as well. This work points to the need for a more probing look into nonlinear legged locomotion control algorithms.

This paper considers a more fundamental and useful question: “for a given legged robot mechanism (which possibly contains nonlinear mechanical elements), how does one design a globally stable and robust feedback control law?” Like [VBC, MB91], our approach is based on the Poincaré return map. This paper suggests ways in which feedback can be used to “shape” the return map in a neighborhood of a fixed point. In

practice, this neighborhood can be quite large. In addition, it is shown that this algorithm can implement stable hopping height control on uneven terrain and is qualitatively robust with respect to some modeling errors. To illustrate the theory, we use the 1-legged model of [VBC]. The methodology can be extended to other legged systems as well.

The Poincaré return map approach contrasts with the literature which employs linearized dynamic models [FM86, FS90, GHM74]. These linearized approaches *can not* ensure global periodic stability properties. Philosophically, linearized approaches are ill-suited to dynamically stable legged locomotion studies.

2. Return Maps for Locomotion Analysis

This section reviews the use of the Poincaré maps by way of an example— the vertical hopping model of [VBC]. The simplified vertical hopping robot model consists of a point mass fixed to a massless leg and actuator combination (Fig. 1). The actuator is a pneumatic cylinder with restoring force [Rai] $F_r = \frac{\eta}{r}$, where r is the height of the point mass and η is a “spring constant” which is a function of the leg cylinder pressure. η may be adjusted by changing the cylinder supply pressure. The piston length does not matter because a “massless” extension may be added and the dimensions rescaled.

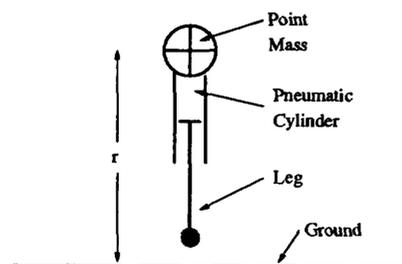


Figure 1: Schematic of Simplified Vertical Hopping Machine

The vertical hopping cycle may be decomposed into four phases, which are the 1-dimensional analogies of Raibert’s higher dimension state machine models for hopping control [VBC].

i. Thrust Phase. The j^{th} cycle begins at time t_j when the leg reaches maximum compression. At t_j , a constant supply pressure is applied to the cylinder for period δ_t , resulting in a constant thrust force, τ (which

is the product of the air pressure and cylinder cross section). The equation of motion during this phase is: $\ddot{r} - \tau + g = 0$ for $t_j \leq t \leq t_j + \delta_t$.

ii. **Decompression Phase.** At $t_j + \delta_t$, the valves are closed, defining a spring constant, τr_{et} , where r_{et} is the body position at the "end of thrust." The equation of motion is:

$$\ddot{r} - \frac{\tau r_{et}}{r} + g = 0 \quad \text{for } r_{et} \leq r \leq r_0 \quad (2.1)$$

where r_0 is the uncompressed pneumatic cylinder reference length. The robot loses ground contact when $r = r_0$.

iii. **Flight Phase.** Since air drag is neglected during flight, the touchdown and lift-off velocities have identical magnitudes. The equation of motion is: $\ddot{r} + g = 0$, for $r \geq r_0$.

iv. **Compression Phase.** At touchdown, the existing pressure in the leg fixes η . The equation of motion is:

$$\ddot{r} - \frac{\eta}{r} + g = 0. \quad (2.2)$$

At the end of this phase, $r = r_{j+1}$ and $t = t_{j+1}$, and the hopping cycle begins anew.

For given initial conditions, the hopping cycle dynamics evolve along a unique trajectory in the 3-dimensional extended phase space: $\mathcal{P} = (r, \dot{r}, t)$. The *Poincaré Section*, Σ , is a hyperplane in \mathcal{P} that transversally intersects this trajectory [MB93] (Fig. 2). Here, as in [VBC], the Poincaré section is situated at $\dot{r} = 0$. The *Poincaré Return Map*, relates successive intersection points with the same orientation. The trajectory will pierce Σ at points A (max. leg compression) and A' (max. hopping height). We sample the map at A .

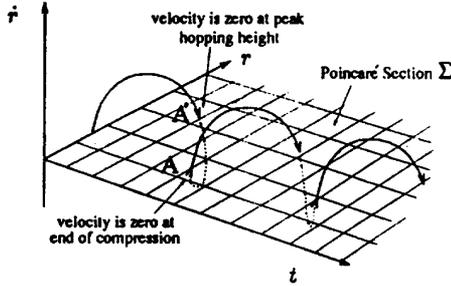


Figure 2: Schematic of Poincaré section in the extended phase space

The return map: (1) reduces the *dimension* of the dynamics by one (e.g., for \mathcal{P} the map has the general form $r_{j+1} = f(r_j, t_j)$, $t_{j+1} = g(r_j, t_j)$); and (2) converts the study of the continuous dynamics to the analysis of a discrete map. For this model, the return map

is fortuitously 1-dimensional. Introducing the non-dimensional leg length, ω_j , and non-dimensional parameters λ and β :

$$\omega_j = \frac{r_j}{r_0}; \quad \lambda = \frac{\tau r_0}{\eta}; \quad \beta = \frac{\tau \delta_t^2}{2r_0},$$

the return map has the form [VBC]:

$$\omega_{j+1} = e^{-\lambda[\beta - (\beta + \omega_j)\eta(\beta + \omega_j)]} \quad (2.3)$$

Equation (2.3) encodes the complete qualitative global dynamics of this system. In deriving the map, gravity forces are considered negligible compared to thrust and stiffness forces during stance [VBC]. This assumption simplifies the equations. The return map including gravity is developed in [MB93], where it is shown that the map without gravity is an excellent qualitative predictor of the global behavior of the system including gravity. For simplicity we initially analyze the map without stance gravity to understand important qualitative system features, but include gravity later in the numerical simulations of hopping over rough terrain in Section 6.

The fixed points of the return map, i.e. $\bar{\omega} = f(\bar{\omega})$, correspond to period-1 oscillations of the dynamical system. Stable fixed points correspond to stable uniform hopping motions. Local stability of $\bar{\omega}$ is determined by the return map slope at $\bar{\omega}$. For 1-dimensional maps: $|f'(\bar{\omega})| < 1$ implies $\bar{\omega}$ is stable, while $|f'(\bar{\omega})| > 1$ implies that $\bar{\omega}$ is unstable.

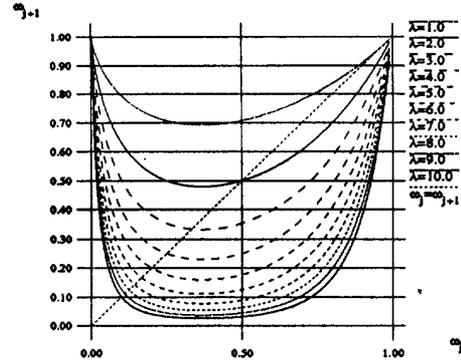


Figure 3: Return Map for $\beta = 0$

Fig. 3 depicts the return map, Equation (2.3), for $\beta = 0$ (instantaneous stance thrust) and for $\lambda = 1, \dots, 10$. $\bar{\omega}$ corresponds to the point where the line of unity slope intersects the graph of the return map. For $\beta = 0$, $\bar{\omega} = 1/\lambda$ (for $\beta \neq 0$, $1/\lambda$ is a very good fixed point approximation). For $\lambda < e^2$, this fixed point is stable [VBC]. As λ increases to e^2 , the fixed point begins to lose stability (increasing λ corresponds to higher commanded hopping height for fixed η). At $\lambda = e^2$, the hopping motion remains bounded, but the period-1

fixed point loses stability by spawning a stable period-2 orbit in a period-doubling bifurcation. The changes in global dynamic behavior with respect to parameter variations can be encapsulated in a "bifurcation diagram." Fig. 4 shows the bifurcation diagram for the case $\beta = 0$, by graphing versus λ^{-1} the "steady state" behavior for repeated iterations of Equation (2.3).

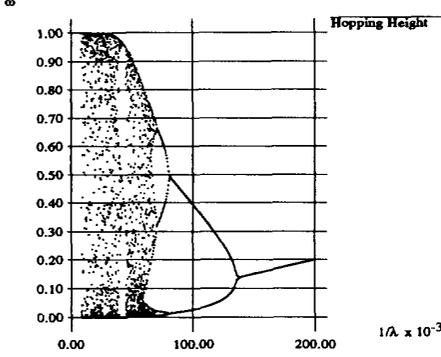


Figure 4: Bifurcation Diagram for $\beta = 0$ (e.g, a plot of $\bar{\omega}$ vs. λ^{-1})

The goal of this work can now be stated formally. Let a legged locomotion system be described by a Poincaré Return map of the form:

$$\bar{x}_{k+1} = f(\bar{x}_k, \bar{p}) \quad (2.4)$$

where \bar{x} is the system state, and \bar{p} is a set of "control" parameters. To improve our ability to control legged locomotion we wish to find a state feedback law, $\bar{p} = \alpha(\bar{x}_k)$ which transforms a given return map (2.4) into

$$\bar{x}_{k+1} = g(\bar{x}_k) \quad (2.5)$$

where $g(\bar{x})$ is a return map that encodes desired global dynamic behavior.

3. An Energetic Viewpoint

In Section 2, the return map was computed by integrating the equations of motion over a complete gait cycle. For this particular model, total system energy is conserved in every phase except for the thrust phase. In such cases, the return map can be more easily computed by examining the net change in system energy over one gait cycle. Let E_1 be the total system energy at the end of the thrust (nonconservative) phase. Let E_2 be the energy at the end of the compression phase (onset of the next nonconservative phase).

Since the motion is conservative between the end of thrust and the end of the following compression phase, the Poincaré Map can be found by equating $E_1(r_j) = E_2(r_{j+1})$. Setting $\tilde{\beta} = \frac{1}{2}(\tau - g)\frac{\delta_t^2}{r_0} = \beta - \frac{1}{2}\frac{g\delta_t^2}{r_0}$, E_1 is:

$$\begin{aligned} E_1 &= T_1 + V_1 = \frac{1}{2}\dot{r}_{et}^2 - \tau r_{et} \ln(\omega_{et}) + g(r_{et} - r_0) \\ &= \tau r_0 [\tilde{\beta} - (\omega_j + \tilde{\beta}) \ln(\omega_j + \tilde{\beta}) + \frac{g}{\tau}(\omega_j + \tilde{\beta} - 1)] \end{aligned} \quad (3.1)$$

where V_1 is the gravitation and leg spring potential energies. Its reference value is chosen so that $V_1(r = r_0) = 0$. Likewise, E_2 , evaluated at $r = r_{j+1}$, is:

$$\begin{aligned} E_2 &= T_2 + V_2 = \frac{1}{2}\dot{r}_{j+1} - \eta \ln(\omega_{j+1}) + g(r_{j+1} - r_0) \\ &= -\eta \ln(\omega_{j+1}) + g r_0 (\omega_{j+1} - 1) \end{aligned} \quad (3.2)$$

with V_2 again evaluated such that $V_2(r = r_0) = 0$. Equating $E_1 = E_2$ gives the transcendental equation which the hopping robot must satisfy on each successive return:

$$\begin{aligned} -\frac{1}{\lambda} \ln \omega_{j+1} + \frac{g}{\tau} \omega_{j+1} &= -(\omega_j + \tilde{\beta}) \ln(\omega_j + \tilde{\beta}) \\ &+ \frac{g}{\tau} \omega_j + \tilde{\beta} \left(1 + \frac{g}{\tau}\right) \end{aligned}$$

Setting $g = 0$ (i.e. neglecting stance gravity) yields the original return map of Equation (2.3).

This viewpoint is valuable because the return map can be derived simply by computing total system energy before and after any non-conservative phase. This obviates the need to integrate the equations of motion during conservative phases. The result is general for running and hopping locomotion systems with a flight phase, which can be assumed to be energetically conservative, followed by a stance phase which is not conservative. This idea also gives us a logical base for attempting to control towards a fixed point, as will be seen in Section 5.

4. Variable Thrust Profile Return Map

In Raibert's vertical hopping height control algorithm, τ is held constant during the thrust period, δ_t . This section derives the return map for a thrust sequence consisting of N sequential thrust pulses. While this result offers the control engineer greater flexibility in the design of a hopping height control algorithm it additionally demonstrates: (1) the utility of the energetic methods in Section 3; and (2) a broader notion of an "accumulated" control parameter, which is a logical basis for the control methodologies of Section 5.

As before, the thrust phase begins when the leg reaches maximum compression. The pulses have equal duration δ_t , but can have variable magnitude, τ_i ($i = 1, \dots, N$). To derive the Poincaré map, we use the approach of Section 3. The thrust phase modification does not affect the energy expression at the end of the conservative phase, Eq. (3.2). E_1 will be slightly modified, as r_{et} and \dot{r}_{et} are now different. It can be shown that the final conditions for this modified thrust phase are:

$$\begin{aligned} r_{et} &= r_{et,N} = r_j + \delta_t^2 \sum_{i=1}^N \frac{1}{2} (2(N-i) + 1) \tau_i; \\ \dot{r}_{et} &= \dot{r}_{et,N} = \delta_t \sum_{i=1}^N \tau_i; \text{ and } \tau = \tau_N \end{aligned}$$

Recalling that $E_1 = T_1 + V_1 = \frac{1}{2}r_{et}^2 - \tau r_{et} \ln(\omega_{et})$ (where stance gravity is neglected), the return map is derived by setting $E_1(r_j) = E_2(r_{j+1})$. As before, the map may be rewritten in nondimensional terms. To retain a relationship to the original map, three parameters are chosen to be the same, except for the fact that τ goes to τ_N : $\omega_j = \frac{r_j}{r_0}$, $\beta = \frac{\tau_N \delta_t^2}{2r_0}$, and $\lambda = \frac{\tau_N r_0}{\eta}$. Then, it can be shown that in the limit as $N \rightarrow \infty$ and $\delta_t \rightarrow 0$ (i.e., as the thrust sequence approaches a continuous curve “profile”, denoted $\tau(t)$), the return map takes the form:

$$\omega_{j+1} = e^{-\lambda(\gamma + (\omega_j + \sigma) \ln(\omega_j + \sigma))} \quad (4.1)$$

where:

$$\begin{aligned} \gamma &= \frac{1}{2\tau_N r_0} \left(\int_{t_j}^{t_{et}} \tau(t) dt \right)^2 \\ \sigma &= \frac{1}{r_0} \int_{t_j}^{t_{et}} (t_{et} - t) \tau(t) dt \end{aligned} \quad (4.2)$$

Remark: This result shows that the effect of *any* thrust sequence on the global qualitative dynamics can be captured by three parameters, λ , γ , and σ . Infinitely many different stance thrust profiles will result in the same values of γ and σ , which respectively encode a total and weighted area under the thrust curve. The different thrust profiles will not change the system’s global asymptotic behavior. However, they can affect local dynamic properties, such as energy use or maximum jerk, during the stance phase. We shall hereafter call γ and σ “accumulated” parameters, as they encode an accumulated effect.

5. Control Towards Fixed Hopping Height

Our goal is to determine useful methods for “shaping” the return map in some global sense. The purpose of this shaping is two-fold. First, we wish to maintain the stability of the fixed point through parameter regions in which a bifurcation to undesirable behavior would have occurred for the unmodified system. Second, by shaping the map we may also improve the transient or robustness properties of the closed loop system.

We consider cyclical dynamical systems which are conservative and continuous in all but a section of their dynamics. This will be true for any legged system which has a flight phase and a stance phase. Let E_1 be the total system energy at the end of the nonconservative phase (beginning of the conservative phase). It is assumed to be a function of r_j and a set of “control” parameters, $p \in \mathbb{R}^p$. These parameters may be “simple” parameters, such as δ_t , τ , and η . They may also be derived parameters, such as β and λ . Or, they could be “accumulated” parameters, such as the parameters associated with a variable thrust profile (Section 4). Let E_2 denote total system energy at some later point in the conservative phase of the cycle, and assume it can be expressed as a function of r_{j+1} and the control parameters, p . Note, for simulations, $p = (\tau, \delta_t, \eta)^T$ is used.

Suppose that a nominal set of constant parameters, p_0 , determines a nominal or desired period-1 fixed point value, $\bar{r}(p_0)$. We assume that either an exact or an approximate expression for \bar{r} can be derived. This also determines an associated value of E_2 . We propose to compute a parameter feedback law so that the energy at the end of thrust, E_1 , equals the energy associated with the peak hopping height for p_0 . In other words, we would like to choose parameters, \hat{p} , so that $E_1(r_{et}, \hat{p}) = E_2(\bar{r}, p_0)$. In this way, the robot will reach its desired hopping height by modifying the control parameters employed during the stance phase.

We can derive a first order approximation for E_1 and E_2 by Taylor expansion about $p = p_0$:

$$E_1 \approx E_1(r_j, p_0) + \frac{\partial E_1}{\partial p} (p - p_0) \quad (5.1)$$

$$E_2 \approx E_2(\bar{r}, p_0) + \frac{\partial E_2}{\partial p} (p - p_0) \quad (5.2)$$

Setting these truncated expansions equal to each other:

$$E_1(r_j, p_0) - E_2(\bar{r}, p_0) = \left(\frac{\partial E_2}{\partial p} - \frac{\partial E_1}{\partial p} \right) (p - p_0)$$

We can then solve for the least norm solution for p by using the pseudo-inverse. Let E^\dagger be the pseudo-inverse of $\left(\frac{\partial E_2}{\partial p} - \frac{\partial E_1}{\partial p} \right)$. This yields an equation for p , the values of the control parameters which should be applied during the stance phase:

$$p = p_0 + E^\dagger (E_1(r_j, p_0) - E_2(\bar{r}, p_0)).$$

If desired, one can introduce a parameter rescaling law to account for the fact that all parameters are not equally effective in control action. Let $\hat{p} = \begin{pmatrix} p_1 \\ p_{10} \end{pmatrix}, \begin{pmatrix} p_2 \\ p_{20} \end{pmatrix}, \dots, \begin{pmatrix} p_p \\ p_{p0} \end{pmatrix}$ be a weighted parameter vector, where p_{i0} is the i^{th} component of p_0 . If $p_{i0} = 0$, we can substitute a nonzero scaling parameter \hat{p}_{i0} to give \hat{p}_i the proper weighting. Taylor expanding in terms of \hat{p} about p_0 gives the following equations to replace (5.1) and (5.2):

$$E_1 \approx E_1(r_j, p_0) + \frac{\partial E_1}{\partial \hat{p}} (\hat{p} - 1) \quad (5.3)$$

$$E_2 \approx E_2(\bar{r}, p_0) + \frac{\partial E_2}{\partial \hat{p}} (\hat{p} - 1) \quad (5.4)$$

and the equation p remains the same, using E^\dagger now defined to be the pseudo-inverse of $\left(\frac{\partial E_2}{\partial \hat{p}} - \frac{\partial E_1}{\partial \hat{p}} \right)$.

There may, however, be bounds on the parameters which arise from physical considerations or limitations of the linearization. For this reason, \hat{p} may be constrained such that $\hat{p}_i \in [1 - \epsilon, 1 + \epsilon]$, where ϵ is “small.” If any parameter value, $\hat{p}_1, \dots, \hat{p}_p$, lies outside that parameter range, it is set to its extremal allowed value.

Fig. 5 shows the return map when parameters τ and δ_t are varied during each stance according to the above

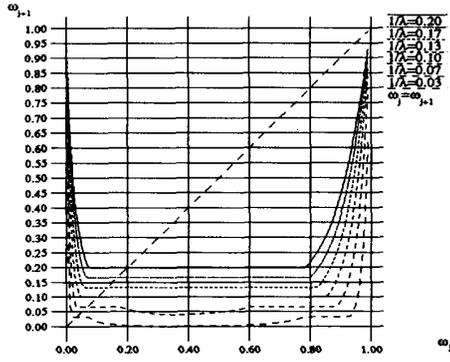


Figure 5: Return map for hopping with feedback. Nominal parameter values are $\tau = 15$, $\delta_t = 0.01$, $r_0 = 1$, and $\epsilon = 0.3$. $\frac{1}{\lambda}$ is varied by changing η .

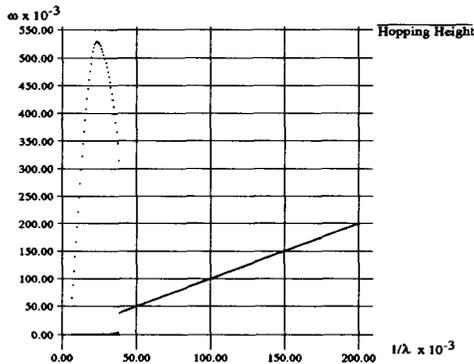


Figure 6: Bifurcation diagram with parameter feedback. Parameters as in Fig. 5

procedure, while η is fixed. Here, $\epsilon = 0.3$. Note that the map is “flattened” in a large neighborhood of the fixed point. As shown below, this flattening causes a wide range of initial conditions to quickly converge to the fixed point. Fig. 6 shows the associated bifurcation diagram for the map of Fig. 5. Note that the region of desirable period-1 behavior is significantly enlarged as compared to the map without feedback, which closely resembles Fig. 4.

In the above example, we assumed $\epsilon = 0.3$, i.e., that the control action is quite limited in range. Allowing a larger range of control action can significantly improve the resulting behavior. Fig. 7 depicts the bifurcation diagram when $\epsilon = \frac{2}{3}$. Note that the hopping height is period-1 for *all* parameter values. In other words, this system is *globally* asymptotically stable for *all* system parameters and commanded hopping height values. We have thus achieved our goal.

Note in Fig. 6 the persistence of a stable period-2 orbit. Because the proposed control is local, it does not preclude the co-existence of period-2 (or higher) orbits

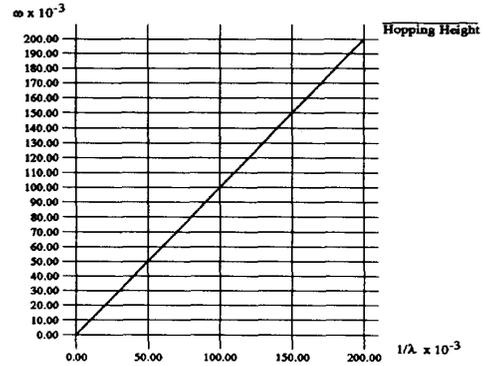


Figure 7: Bifurcation diagram for parameters as in Fig. 5, but with $\epsilon = \frac{2}{3}$.

with a period-1 orbit. That is, initial conditions near the fixed point will converge to the stabilized period-1 orbit, while those far away may converge to a period-2 fixed point (Fig. 8). This behavior is not found in our original return map because it was *unimodal*. For unimodal maps, the local fixed point stability guarantees global fixed point stability. The control law presented here, however, modifies the map so that it is no longer unimodal, and thus allows for the existence of more than one type fixed point at a time. However, this algorithm may indeed have global stability, as seen in Fig. 7.

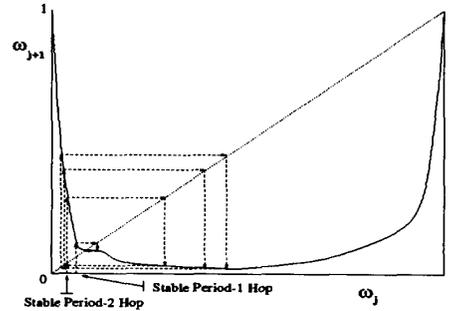


Figure 8: Coexistence of stable period-2 and stabilized period-1 orbits for local feedback

An additional benefit of this control law is that the “flattened” return map converges more quickly towards a stable periodic hopping height. This fact is shown in Fig. 9, which plots the number of iterates required for the Raibert algorithm and our algorithm to bring the hopping height to within 1% of the desired height (from a random initial condition). Note that our control almost always converges to the proper height in one cycle. The number of hops to convergence is saturated at 25 for the Raibert algorithm, since in this region, the robot exhibits higher period motion which can never stabilize to a fixed height.

In summary, the return map can be modified to achieve

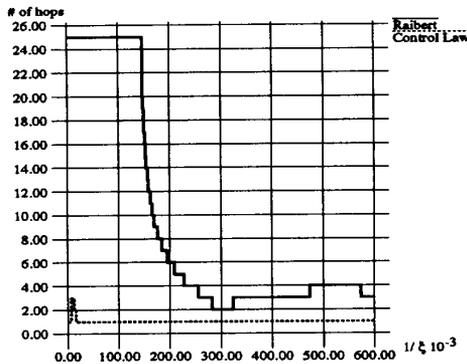


Figure 9: Graph comparing rates of convergence for new control law vs. Raibert's original law. Nominal parameter values same as in Fig. 5, and $\epsilon = \frac{2}{3}$.

desired dynamical properties by making the control parameters a function of the current state. The control law computes "target" values of the control parameters during each gait cycle. The control parameters could be simple ones, such as τ , η , and δ_t . They could also be "accumulated" parameters, such as the area under the stance phase thrust profile (Equation (4.2)). In this case, many different thrust profiles will yield the target value computed by the feedback control law. This freedom in choosing a thrust profile could in turn be used to optimize additional criteria, such as energy use or jerk during stance. We do not address these secondary issues in this paper, but only point out the generality of our approach.

6. The Hopping Robot on Uneven Terrain

Raibert's hopping height algorithm works quite well for flat surfaces. How does it work on uneven terrain? Numerical simulations for the vertical hopper show that while its stability is not greatly affected, the peak hopping height mimics the underlying terrain. That is, if the terrain varies sinusoidally, the peak hopping height will approximate a sinusoid. Also, the robot is still subject to period-doubling bifurcations. This section shows how the feedback algorithm of Section 5 can uniformly control the robot's peak hopping height over uneven terrain. It is impossible to solve analytically for the point at which the robot touches down in this model. Thus, we use numerical simulations to investigate the hopping behavior on rugged terrain. We consider only the simplified vertical hopping system described above, but include gravity during the stance phase. Since there is no forward motion in this simplified model, we assume that the terrain height varies sinusoidally while the hopper is in flight. This variation approximately captures the vertical hopping dynamics of a legged system running over a terrain with a sinusoidal profile.

Shown in Fig. 10 are the peak hopping heights for the vertical hopper controlled initially by the Raibert

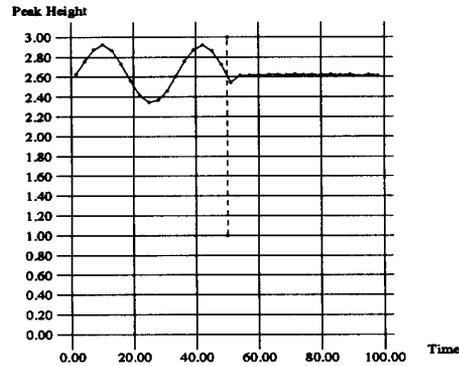


Figure 10: Peak hopping height for $z = 0.3 \sin(\frac{1}{2}t)$. Nominal parameters: $\tau = 60$, $\eta = 30.0$, and $\delta_t = 0.01$. The vertical bar at $t = 50$ denotes activation of the control law.

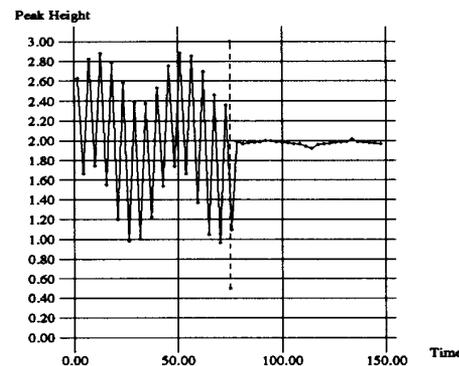


Figure 11: Same values as in Fig. 10, but with $\eta = 10.0$, implying stability of the period-2 hop.

hopping height algorithm. The control algorithm of Section 5 is activated at $t = 50$. Note how the peak hopping height originally mimics the terrain, while our control law stabilizes the system to a constant height. Thus, our method is quite robust to terrain height perturbations. Fig. 11 shows a similar sequence when the Raibert control algorithm exhibits period doubling. The period-2 hop is superimposed on the terrain height variation.

7. Robustness of the Algorithm

Robustness with respect to perturbations is of critical importance to practical applications of this algorithm. Perturbations may arise from measurement noise or errors, modeling errors, or perturbations in state. The last section showed that the algorithm is relatively insensitive to perturbations in ground height.

The method is based on a return map, which encodes

the dynamics of the system. The dynamical models employed in the construction of the return map are idealizations of the real physical system. It is therefore important to determine whether the control, which is formulated using these idealizations, can tolerate small discrepancies between the real system and the idealized system used to construct the return map. Our investigation of these questions is qualitative.

For example, our idealized model assumes no frictional losses during the stance or flight phases. A real system is expected to have some frictional losses during stance. To qualitatively investigate the effect of modeling error on our approach, we added (to the simulated system) a viscous damping term during the compression and decompression phases (the equation for the compression/decompression phases is $\ddot{r} = -g - \dot{r} + \frac{r}{\tau}$). However, damping was not included in the model used to construct the control algorithm.

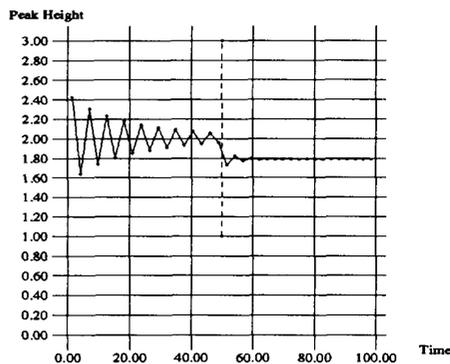


Figure 12: The effect of damping on period-2 hopping, with a demonstration of the control law activated at $t = 50$.

Figure 12 shows the peak hopping height of this system. For $t < 50$, the robot is controlled using the Raibert algorithm and a set of parameters which would normally result in period-doubling. At $t = 50$, the state feedback control algorithm of Section 4 is switched on, and the robot quickly stabilizes to a period-1 hopping behavior. The steady state hopping height is less than desired, however, due to the viscous damping. Since this is an energy based algorithm, modeling errors which add or remove energy during a gait cycle will cause corresponding errors in the desired hopping height. However, the motion appears to be stable and robust.

8. Discussion and Conclusion

This paper presented a feedback control algorithm for improving the control of legged locomotion systems which can be described by a Poincaré Return Map. The method is based on an energy principle which assumes that the gait cycle can be divided into energetically conservative and nonconservative phases. We employ a local parameter state feedback law to “shape” the return map in a neighborhood of a fixed point.

For illustration purposes, we have focused on the Raibert-like vertical hopping model of [VBC], as it contains nonlinear mechanical elements. We chose to “flatten” the return map near the fixed point, so as to improve transient response. We also showed that this control law can uniformly control constant hopping height on rugged terrain, and that our algorithm is robust to some modelling errors and other perturbatory terms.

The dynamics of more complicated systems (such as the more complicated model of [MB91], or multilegged systems) can often be expressed by a Poincaré return map. Our method is in principle generalizable to these systems as well. However, there are many possible ways in which return maps can be shaped in higher dimensions. This is the subject of future work.

9. References

- [Devaney] R.L. Devaney, An Introduction to Chaotic Dynamical Systems, Addison-Wesley Publishing Company, Inc., Redwood City, Calif., 1987.
- [FS90] Furusho, J. and Sano, A. 1990 (Apr). “Sensor-Based Control of a Nine-Link Biped,” *Int. J. Robotics Research*. 9(2):83-98.
- [FM86] Furusho, J. and Masubuchi, M. 1986 (Jun). “Control of a Dynamical Biped Locomotion System for Steady Walking,” *J. Dyn. Sys., Meas., and Control*. 108:111-8.
- [GHM74] F. Gubina, H. Hemami, and R.B. McGhee. 1974 (Mar). “On the Dynamic Stability of Biped Locomotion,” *IEEE Transactions on Biomedical Engineering*. 21(2):102-8.
- [KB91] Koditschek, D. and Bühler, M. 1991. Analysis of a Simplified Hopping Robot. *Int. J. Robotics Research*. 10(6):587-605.
- [LG90] Li, Z. and Ge, J. 1990 (Dec). An Energy Perturbation Approach to Limit Cycle Analysis in Legged Locomotion Systems. *Proc. 29th Conf. on Decision and Control*, Honolulu, Hawaii.
- [MB91] R.T. M'Closkey and J.W. Burdick, “The Periodic Motions of a Hopping Robot with Vertical and Forward Motion,” *Proc. IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, April, 1991.
- [MB93] R.T. M'Closkey and J.W. Burdick, “On the Periodic Motions of a Hopping Robot with Vertical and Forward Motion,” to appear in *Int. J. of Robotics Research*, April, 1993.
- [Rai] M.H. Raibert, Legged Robots that Balance, MIT Press, Cambridge, Mass., 1986.
- [VBC] Vukobratovic, A. F., Burdick, J. W., Caughey, T. K. 1991. An Interesting Strange Attractor in the Dynamics of a Hopping Robot. *Int. J. of Robotics Research*. 10(6):606-618.