Proceedings of the 2014 International Conference on
Advanced Mechatronic Systems, Kumamoto, Japan, August 10-12, 2014

Deadbeat control with (almost) no sensing in a hybrid model of legged locomotion

George Council
Dept. Electrical Engineering and Computer Science,
University of Michigan,
Ann Arbor, United States
Email : gcouncil@umich.edu

Shiyi Yang
Dept. Electrical Engineering and Computer Science,
University of Michigan,
Ann Arbor, United States
Email : yshiyi@umich.edu

Shai Revzen
Dept. Electrical Engineering and Computer Science, Dept. Ecology and Evolutionary Biology,
University of Michigan,
Ann Arbor, United States
Email : shrevzen@umich.edu

***Abstract*—Hybrid systems often appear as models of mechatronic devices, and are used to express the discontinuous transition in dynamics that occurs when mechanical contacts are made or broken. Contact state changes can coincide with dramatic shifts in control authority, and become sources of unwanted disturbance when they are exogenously driven. In the natural world, such systems appear in rapid legged locomotion. We present an insight derived from the analysis of running animals – namely that dynamics can be partially or fully controlled through geometric encoding of the desired control law in the feed-forward trajectories of the appendages. The resulting systems use no sensing, or nearly so, yet can exhibit strong (deadbeat) stability. We provide a theoretical proof of a general result, then a simulation study of a simplified model of vertical hopping which rejects ground-height changes without sensing the ground. We thereby show that mechatronic controllers for non-trivial tasks such as manipulation and legged locomotion could be implemented mechanically, with little or no sensing, by encoding the control laws in the open-loop motions chosen. Our results highlight that identification of a control mechanism in an existing animal or machine must take into account that such geometric stabilization may exist without neural or computational feedback.**

***Keywords*—legged locomotion; deadbeat; SLIP; geometric encoding**

I. Introduction

Many mechatronic systems perform tasks where mechanical contacts are intermittent, central among these are robotic manipulation tasks [1]. Hybrid dynamical systems – ordinary differential equations with vector fields that can change discontinuously – are a common approach to modeling the change in the equations of motion as mechanical contacts are established and broken. Similar hybrid systems are used to model legged locomotion [2] of animals and machines.

In both manipulation and locomotion, a perfect model of both self and environment is usually absent. As a consequence, contact state changes can occur at unexpected times – sooner or later than intended, or at different angles than planned. From the perspective of the actor (robot or animal), these are exogenously generated transitions from one continuous dynamic regime to another.

Typically, limbs and manipulators have a payload capacity. As a consequence they are often easy to control before contact, but become load-limited when in contact. This change expresses itself in a dramatic reduction in control authority. It is only natural to conjecture that a substantial amount of sensing and control is needed if disturbances from exogenous contact changes are to be rejected.

The essence of our contribution is showing this conjecture to be false both in an example, and almost always under mild assumptions. We demonstrate a means whereby a desired control law for the outcome in the poorly controllable and load limited region can be encoded in a manifold constraining motions in the region where the uncertain transitions may occur. If the high pre-transition control authority can be used to meet these constraints, the actor may remove most or all the uncertainty generated by an exogenously driven hybrid transition.

Our results imply that mechatronic systems can often reject disturbances through judicious choice of feed-forward design. Biologists and biomechanists who seek to identify the mechanisms of control in animals should be wary of the possibility that critically important control mechanisms may be encoded in the geometry of limbs and the choice of feed-forward limb trajectory, partially or completely bypassing the nervous system.

We motivate our key observation in §II, then formalize and prove it in §III-B. In §IV we apply our result to a simplified model of human (or animal) vertical hopping; similar to Blickhan's classical hopper model [3], this model may be of independent interest to the biomechanics community.

II. Biological inspiration

The main challenge of legged locomotion comes from uncertain footing – how can we ensure the outcome of a step given uncertain terrain and traction? In dynamically moving animals [4, 5] leg muscles must be able to produce sufficient forces to support the body. Since in most species (humans being a conspicuous exception) leg mass is only a small fraction of overall body mass, the muscles in each leg can typically move that leg very rapidly and precisely relative to the body whenever that leg is not supporting weight. Once the leg is loaded, and in particular when only one leg is loaded in a multilegged animal, the leg muscles have little to no control authority – since they are now producing the forces between the entire body and the ground.

We model such extreme changes in control authority as the problem of controlling the final output state of a dynamical system exhibiting a single hybrid transition – "touchdown" – guaranteed to occur in every execution (i.e. for every initial condition). The touchdown is associated with both a discontinuous change in the vector field and a complete loss of control authority. Such properties are typical of legged locomotion models [6, 3, 7, 8, 9]: gravity ensures that all bodies eventually hit the ground, at which time their equations of motion change from those of ballistic flight to those of ground contact interaction.

We wished to examine whether we could control the outcomes of such a model with little to no sensing, by choosing appropriate feed-forward actions while a leg is still in the air. Representing these motions in a body-centric frame of reference, foot touchdown can typically occur anywhere within the legs' workspace. We therefor consider hybrid systems with an open "guard" – a region in which an exogenous transition may occur – rather than the more commonly used formulation of guards as submanifolds with boundary in the boundary of a domain [10].

Choosing, as many legged locomotion papers do [11, 12, 6, 13, 14], to analyze the periodic motion of our hopping model using Poincaré return maps to the apex of the hop, we will show a means to select foot motion in the neighborhood of an expected touchdown location so that the model performs deadbeat perturbation rejection for ground height changes. This means that our model will perfectly track changes in ground height by adjusting the subsequent apex height by the same amount – but it will do so *without sensing the ground* – relying entirely on the interaction of the touchdown event with our choice of foot trajectory to produce the "control". The only sensing our model uses is knowledge of how far it has fallen since apex. We will show this result as special case of the more general result described in the next section.

III. Encoding control as a pre-transition constraint manifold

Numerous definitions of hybrid dynamical systems appear in the literature[15, 16, 10]. Since we seek to show a local result, we will focus on the flow across a single "hybrid transition". By stating our result with respect to a single possibly discontinuous change in the equations of motion governing the dynamics we avoid the need for mathematical machinery defining a specific, possibly overly restrictive, class of hybrid systems.

A. Hybrid control system definition

Consider a system of the form:

$$\dot{x} = \begin{cases} f_{\mathcal{A}}(x,u), & x \in \mathcal{A},\ f_{\mathcal{A}} : \mathcal{A} \times \mathbb{R}^k \to \mathsf{T}\mathcal{A} \\ f_{\mathcal{B}}(x), & x \in \mathcal{B},\ f_{\mathcal{B}} : \mathcal{B} \to \mathsf{T}\mathcal{B} \end{cases} \tag{1}$$

with vector fields $f_{\mathcal{A}}$, $f_{\mathcal{B}}$ Lipschitz in x on the closures of their respective precompact domains $\mathcal{A}$, $\mathcal{B}$, which map into respective tangent bundle, and a control input u entering into $f_{\mathcal{A}}$ from $\mathbb{R}^k$. Assume it is furthermore known that all executions[1] starting in an open set $U_0 \subseteq \mathcal{A}$ will flow into an open set $O \subseteq \mathcal{A} \cap \mathcal{B}$ and somewhere therein switch to the B domain, to continue with the B dynamics into an open set $U_1 \subseteq \mathcal{B}$. We will presume that the switching between $\mathcal{A}$ and $\mathcal{B}$ dynamics may be exogenous: it might not be a function of state, nor of time, nor be in any other way amenable to prediction; we are only given that exactly one such switching event will occur for every execution starting at U_0, and that this transition will occur while the state is in O.

B. Controller design

Assume that for the system of III-A we are furnished with Poincaré section $S \subseteq U_1$ given as the 0 level set of a smooth function $\sigma : U_1 \to \mathbb{R}$, such that $S := \sigma^{-1}(0)$, and $\mathsf{D}_x\sigma \cdot f_{\mathcal{B}} > 0$ in the entire domain. The section S will be used to define the output we wish to control, via a smooth function $g : O \times S \to \mathbb{R}^d$ whose Jacobians $\mathsf{D}_S g$ and $\mathsf{D}_O g$ with respect to both S and O are maximal rank everywhere on S and O respectively. We will use g to define an implicit relationship between transition states O and the desired eventual states in S, such that for each execution y with transition at y_τ and arrival at $y_1 \in S$, we drive the value of $g(y_\tau, y_1)$ to zero.

As one final assumption, we must develop our local control scheme in the neighborhood of a known "desirable" execution of the system $x : [0,1] \to (\mathcal{A} \cup \mathcal{B})$ for which the control u is identically 0. x starts at $x(0) = x_0 \in \mathcal{A}$, moves into the open set $O \subseteq \mathcal{A} \cap \mathcal{B}$ in which it transitions over at time τ_* and state $x_* := x(\tau_*)$ to $\mathcal{B}$, then continues in $\mathcal{B}$, finishing at $x(1) = x_1 \in S$. It is "desirable" in the sense that $g(x_*, x_1) = 0$.

Although we have no control over the dynamics in B, nor over the instant at which f_B takes over the dynamics, we will show that we can *locally deadbeat stabilize* the output $g = 0$ by restricting the trajectories in O to a smooth submanifold passing through x_*, using the control u. This follows from the observation that the flow $\Phi_{\mathcal{B}} : \mathbb{R}^+ \times \mathcal{B} \to \mathcal{B}$ in domain $\mathcal{B}$ is smooth in $\mathcal{B}$, as is the "impact time map" $T_{\mathcal{B}} : O \to \mathbb{R}^+$ [17, 18] which maps initial conditions to their arrival times on S, i.e. $\forall o \in O : \Phi_{\mathcal{B}}(T_{\mathcal{B}}(o), o) \in S$. We pull S and thus also g back to O by defining:

$$g_*(x) := g(x, \Phi_{\mathcal{B}}(T_{\mathcal{B}}(x), x)) \quad g_* : O \to \mathbb{R}^d. \tag{2}$$

Observe that g_* maps every state in O to its "eventual" output value, presuming that it will be carried by $\Phi_{\mathcal{B}}$ from here on.

Consider a path $z(t)$ taking values in O, and which is at least $\mathcal{C}^1$. The derivative of g_* along the path z is:

$$\mathsf{D}_{g_*}\dot{z} = [\mathsf{D}_O g + \mathsf{D}_S g \cdot (\mathsf{D}_x \Phi_{\mathcal{B}} + f_{\mathcal{B}} \cdot \mathsf{D}_x T_{\mathcal{B}})] \cdot \dot{z}. \tag{3}$$

If $\mathsf{D}_{g_*}\dot{z} = 0$ then the path $z(t)$ is g_*-invariant with respect to time. Under generic circumstances (with respect to the standard $\mathcal{C}^1$ topology), g_* will be of maximal rank, which

[1] a "solution" of a hybrid system is termed an "execution"

we label d. We will assume g_* is of rank d in a neighborhood X_* of x_* and conclude that the set

$$\mathcal{G} := \{x \in X_* | g_*(x) = 0\} \quad (4)$$

is an immersed submanifold of co-dimension d in O, which passes through x_*, since $g_*(x_*) = 0$.

If a controller can bring trajectories z in $\mathcal{A}$ close to $\mathcal{G}$, i.e. to $\|g_*(z)\| \leq \varepsilon$, and thereafter sustain the $\mathsf{D}_{g_*}\dot{z} = 0$, then any transition that occurs in X_* will lead to a state with $\|g\| \leq \varepsilon$. In particular, this is true even for $\varepsilon = 0$ – finite time convergence to $\mathcal{G}$ will ensure that $g = 0$ after transition.

A variety of methods of nonlinear control [19, 15] may be employed to achieve this goal in $\mathcal{A}$. The choice of controller is not germane to the core observation we present – namely that $\mathcal{G}$ is known in advance, independently of any need for sensing in real time, and that in its geometry it encodes a relationship between the transition state in O and the outcome on the next crossing of S in a feed-forward fashion.

As a corollary, note that if g is full rank, i.e. if $d = \dim \mathcal{B} - 1 = \dim \mathcal{S}$ and so is g_* then $\mathcal{G}$ is a single smooth trajectory passing through x_*. When moving along this trajectory, the system is guaranteed to reach x_1 in S regardless of where in O the transition occurs. If, furthermore, the flow carries $\mathcal{B}$ back into $\mathcal{A}$ such that the example execution is periodic (stable or not), this process renders the controlled execution z stable with respect to the Poincaré section defined by S – any deviations from the nominal z will be removed by the controller acting in $\mathcal{A} \setminus O$. We now proceed to construct such a deadbeat control example in simulation.

IV. Vertical hopping can be Deadbeat controlled without sensing

As a motivating example, we present a model of a vertically hopping organism. In his classical paper [3] Blickhan showed a vertical spring-mass hopper and proceeded to derive relationships for the performance envelopes of human hoppers and runners. In line with Blickhan, Farley [20] has argued that the vertical hopping limit of the Spring Loaded Inverted Pendulum (SLIP) model of running [2, 3, 7] – running with zero forward speed – is informative in understanding bouncing gaits in animals. We suggest some simple elaborations of this model can provide a platform for demonstrating our key result, while at the same time providing a more general model capturing some of the essential nature of control of legged systems with muscles.

A fundamental problem of both SLIP and Blickhan's hopper is that they are energy conserving systems. As such, even if a periodic execution exists, it cannot be asymptotically stable – any perturbation changing total energy must perforce shift the system to a new level-set of the total energy function, never to return. Changes in average ground height, which are of paramount relevance to any legged locomotion in the real world, will require changes in average potential energy. We are thus lead to require a dynamical model which allows for energy to be both added and removed from the system.

Since our goal is to provide an example with no sensing of the ground, the approach of energy injection by triggering some actuation whilst on the ground cannot serve us. Instead, we hark back to biology to note that biological actuation uses muscles, whose intrinsic dynamics already allow for them to add energy and not only remove it. Numerical experimentation with our model has shown this to require some "toe-off" – the liftoff height in hopping must be greater than the landing height. This result of our model may deserve further biomechanical investigation; it seems plausible for human hopping that ground reaction forces only become substantial at heel strike, but persist to toe-off.

The overall scheme of our model is illustrated in fig. 1, showing a hopper hopping over changing ground. We envision the hopper to have an actuator allowing the length of the leg to be freely controlled in flight, but presume this leg length to "lock in" at touchdown, and have dynamics of stance governed entirely by a muscle model, until the length of the leg exceeds the toe-off length, or until the relative height is 0 (i.e : the leg ”crashed”).

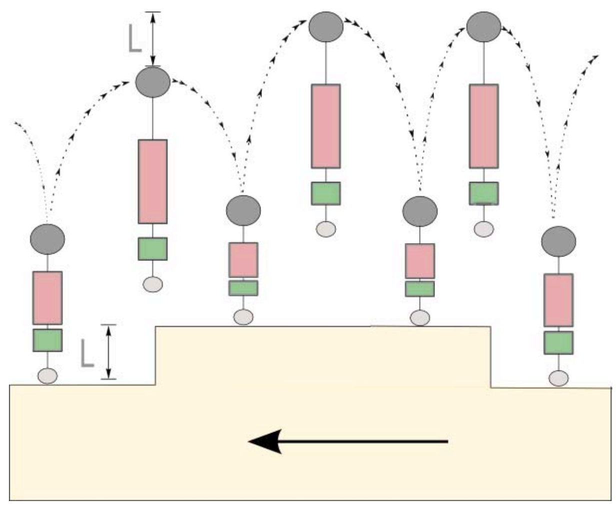

Fig. 1. Vertical Hopper bouncing over a moving plate (height plotted against time), with an actuator (active only in flight) changing the rest length of the leg (pale red) and a muscle-like element (green) supporting the payload mass (dark grey). While the foot (light grey) is in contact with the ground, ground height remains constant; it changes instantaneously when the hopper is at its apex height.

A. Equations of motion

The state of our hopper is defined by the (absolute) vertical position x and velocity $\dot{x}$. Its leg length, in flight, is governed directly by the control input u. Ground height h is piecewise constant, changing only when the hopper is at "apex" – in the flight domain, at the instant when $\dot{x} = 0$. We use the apex as our section S, with $-\dot{x}$ playing the role of σ from the definition of S.

In flight, dynamics are ballistic:

$$\ddot{x} = -g \qquad \text{flight} \quad (5)$$

In stance, dynamics are governed by a combination of gravity and "muscle dynamics".

$$\ddot{x} = -g + K(L - x)(1 + \eta\dot{x}) - \mu\dot{x} \qquad \text{stance} \qquad (6)$$

The length L represents the length of the leg at touchdown. The parameters K and η provide averaged approximations to the length dependent and velocity dependent terms (respectively) of the Hill muscle model; μ adds some dissipation, capturing the overall energy consuming nature of the task.

The Hill muscle model [21] has been postulated to be of sufficient accuracy to be useful for simulating human musculoskeletal behavior [22, 23]. We base our treatment of this model on the presentation in [24] §2. With respect to their notation, K is an average slope for F_L near the operating point of the hopper, and η is an average slope for F_V around 0, and within the typical range of velocities.

Touchdown transitions triggering a change from flight to stance are induced when:

$$x_{TD} - u = h \qquad \text{(touchdown)} \qquad (7)$$

At that moment, the value of L is set to equal $x_T D - h$ for the following stance.

Liftoff transitions are triggered when toeoff occurs:

$$x_{LO} = L + l_{to} \qquad \text{(toeoff)} \qquad (8)$$

A "toeoff" length $l_{to} := 0.1$ was used for all our simulations.

B. Simulation environment

The system was simulated in Python 2.7.5 using the NumPy and SciPy open-source numerical libraries. ODE-s were integrated using a pure-python port of the `dopri5` integrator from [25]. A custom integrator was used because this integrator provides "dense output" – i.e. polynomial patches between time-points. These were used to implement a bisection based event detector, allowing for accurate and speedy simulation of hybrid systems.

C. Constructing the goal function g_*

As a first step towards constructing and example of our control scheme, we produced a periodic solution of the hopping simulation by numerically solving for a fixed point of the apex return map.

Parameters of this solution are found in Table I.

TABLE I
PARAMETERS

Parameter	Definition	Value
η	F_V average slope	-0.03
μ	dissipative loss	0.3
K	F_L average slope	80
y_1	desired relative apex	2
$y_*(y, \dot{y})$	known touchdown	(1.57, -2.87)
l_{to}	toe-off height	0.1

It is convenient to visualize the hopper in terms of its motions relative to the ground, rather than its dynamics with respect to absolute ground height. For this we define auxiliary coordinates $y := x - h$, whose dynamics are identical to those of x except for being discontinuously remapped by the change in ground height at apex.

Our goal for control is to ensure that hopping dynamics remain the same with respect to ground height, which for this very simple system implies that we require y at the following apex to be unchanged even if the landing height L was unexpected. This is expressed by the goal function:

$$g(y_{TD}, \dot{y}_{TD}, y_{next}) = (y_{TD} - y_*) - (y_{next} - y_1) \qquad (9)$$

where y_{TD}, $\dot{y}_{TD}$, y_{next} are the touchdown state and the next apex height, and y_* and y_1 are the touchdown and apex heights on the desired periodic execution. This simple $g = 0$ implies that touchdown height changes exactly with the subsequent apex height.

Using g and the flow map, we sampled g_*, to obtain its values in a neighborhood of the expected touchdown state $(\dot{y}_*, y_*)$ (see fig. 2).

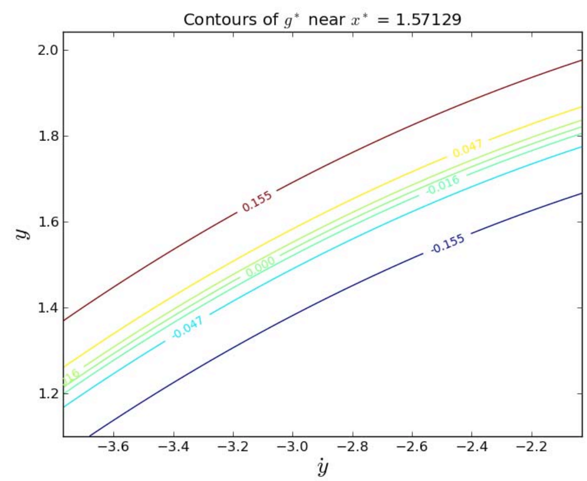

Fig. 2. The pullback of g into g_* into the region X^* around $y_* = 1.571$. The zero contour is the control objective in this region. Values were sampled on a grid with step size 0.012

D. Constructing the feed-forward control law

Given values of g_*, we selected only points whose g_* value was less than 0.01 in magnitude, and least-squares fit a quadratic to obtain y as a function of $\dot{y}$ and thereby indirectly as a function of time since last apex, or distance below last apex. To extend this function outside the sampled region, we held the extremal values using an analytical approximation to a step function. The final control law, written in terms of time since last apex t, the polynomial model `uModel`, the minimal and maximal sampled y values `y_min` and `y_max` and the minimal and maximal $\dot{y}$ as `v_min`, `v_max` was:

```
def aStep( x ):
  return (tanh(x)+1)/2
```

```
def u(t):
  v = -g*t
  x0 = y1 - g*t*t/2
  w0 = aStep((v_min-v)/0.01)
  w1 = aStep((v-v_max)/0.01)
  w = (1-w0-w1)
  return (polyval(uModel,v)*w
      +y_min*w0+y_max*w1)+x0
```

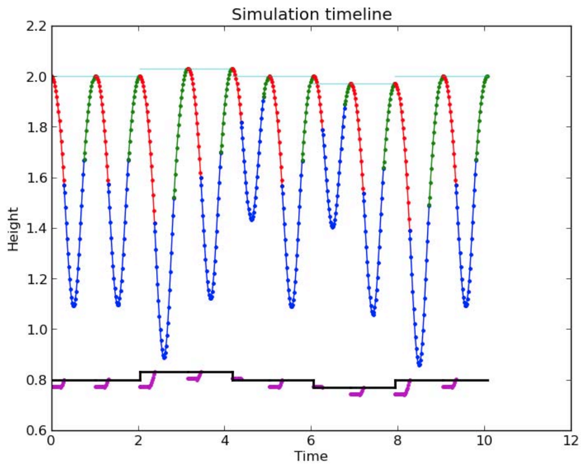

Fig. 3. Hopper simulation with our controller and randomly changing ground height. Each hop begins with a descent (red) associated with a changing "virtual" foot position (purple), ending when the foot intercepts the ground, triggering a transition into stance (blue), which persists until toe-off. After toe-off the hopper ascends (green) to a new apex; upon reaching it the ground height (black) may change. To demonstrate the deadbeat nature of this controller, the desired apex height for the next hop is indicated (light blue), and each ground height persists for two hops to show that hopping height reaches a new steady state after a single hop.

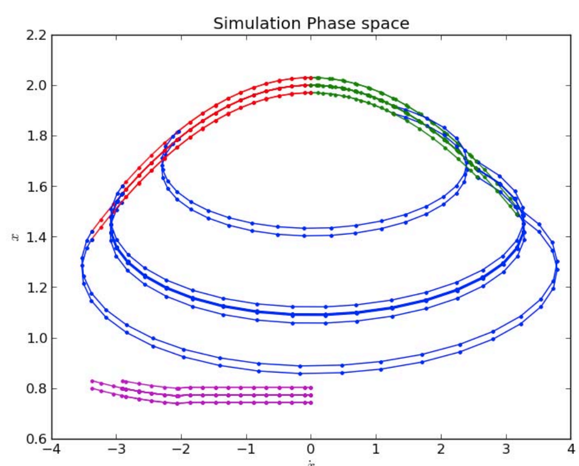

Fig. 4. Phase-space plot of the same simulation as fig. 3

Figs. 3 and 4 show typical simulation results.

The non-conservative nature of the model is evident in the simulation results depicted in figs. 3 & 4. The hopper is gaining and losing energy from hop to hop at the precise amount needed to maintain a fixed *relative* displacement with respect to the ground at apex.

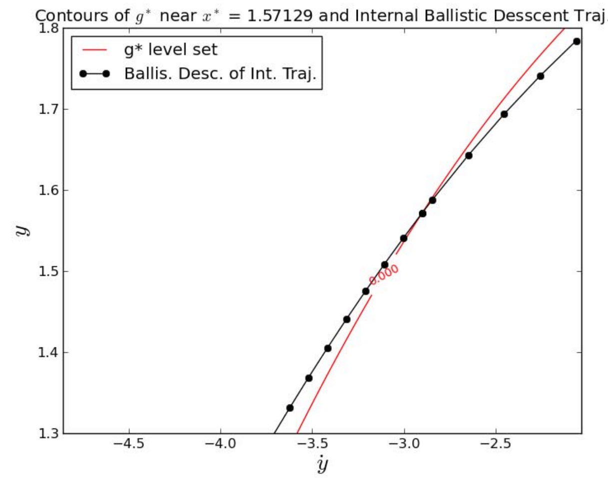

Fig. 5. The relative position of the internal state and the pre-image of 0 under g_*. The difference between the two illustrates the necessary leg length to shift the descent trajectory pointwise to the contour.

E. Physical realization of the model

Fig. 5 illustrates both the control objective graphically, and a potential difficulty with physical realization. As the internal state is *below* the g_* contour, which therefore requires the leg to be shortened. Viewed naively, such a controller requires the toe to start "inside" the ground, and have the leg continually shortening until touchdown. Since the leg length we use is the length of the leg at the onset of generation of ground reaction forces, x_*, one may equally consider this shortening leg as a change of internal configuration in a mechanism longer than L, which causes it to start producing force at the desired location.

V. Conclusion

The above demonstration of the local deadbeat stabilization of a hopping model illustrates that accurate sensing of the ground is not a prerequisite for producing strong stability guarantees with respect to ground height changes. Given the existence of a periodic solution, a controller that only has limited sensing information – resets a time-course at apex events – is able to provide a deadbeat stabilizing region of attraction and reject variations in ground height.

Importantly, given the "open-loop" nature of our control scheme one might consider that mechanical implementation of trajectories tangent to $\mathcal{G}$ would provide strong self-stabilization without need for any overt sensing. Because such stabilization is encoded indirectly in the dynamics of the system, it would not be obvious even to a relatively sophisticated observer of the animal or robot system. While in our example the control objective was achieved entirely via trajectory design, a low value of d would merely act to produce some strongly stable (e.g. deadbeat) directions – allowing partial encoding of the control objective.

Maintaining $\mathsf{D}_{g_*}\dot{z} = 0$ in a neighborhood of x_* renders g_* level sets invariant, but does not make $\mathcal{G}$ attractive. Other control mechanisms that need not be feedforward might be used to move trajectories toward the manifold $\mathcal{G}$. To an investigator hoping to identify the control mechanisms of such a system, only the control rendering $\mathcal{G}$ attractive would be apparent. The mechanisms generating the fast rejection of some perturbations encoded into the shape of $\mathcal{G}$ itself might remain unexplained at best, or be mistakenly attributed to other mechanisms at worst.

The essence of our observation is not a surprising mathematical insight. Rather it is that elementary mathematical tools offer the possibility that in the geometry of approach nearing a hybrid transition one may encode a control law which exploits the transition itself to enact a feedback policy, bypassing the need for conventional sensing and feedback calculations. This key observation we believe to be novel, and to be of potential use in the design of mechatronic systems and in the analysis of animal locomotion.

We have provided a proof that the necessary conditions hold in the generic case (i.e. "almost always"), and shown results for a specific simulated example: the deadbeat stabilization of a vertical, non-conservative, spring-mass hopper which can track exogenous disturbances in ground height without sensing the ground.

References

[1] R. Murray, S. Sastry, and Z. Li, *A mathematical introduction to robotic manipulation*, 1st ed. CRC Press, 1994.

[2] P. Holmes, R. Full, D. Koditschek, and J. Gukenheimer, "The dynamics of legged locomotion : models, analyses, and challenges," *SIAM Review*, vol. 48, no. 2, pp. 206–304, 2006.

[3] R. Blickhan, "The spring-mass model for running and hopping," *J. Biomechanics*, vol. 22, no. 11-12, pp. 1217–1227, 1989.

[4] D. Dudek and R. Full, "Passive mechanical properties of legs from running insects," *J. of Exp. Bio.*, vol. 209, pp. 1502–1515, 2006.

[5] K. Sasaki and R. Neptune, "Differences in muscle function during walking and running at the same speed," *J. Biomechanics*, vol. 39, pp. 2005–2013, 2005.

[6] O. Arslan and U. Saranli, "Reactive planning and control of planar spring-mass running on rough terrain," *IEEE Trans. Roboitics*, vol. 28, pp. 567–579, 2012.

[7] R. Ghigliazza, R. Altendorfer, P. Holmes, and D. Koditscheck, "A simply stabilized running model," *SIAM J. App. Dyn. Systems*, vol. 2, no. 2, pp. 187–218, 2004.

[8] R. Altendorfer, D. Koditschek, and P. Holmes, "Stability analysis of legged locomotion models by symmetry-factored return maps," *Int. J. Robot. Res.*, vol. 32, no. 10-11, pp. 979–999, 2004.

[9] I. Poulakakis and J. Grizzle, "Formal embedding of the spring load inverted pendulum in an asymmetric hopper," in *European Control Conference*, July 2007.

[10] R. Goebel, R. Sanfelice, and A. Teel, "Hybrid dynamical systems," *Control Systems, IEEE*, vol. 29, no. 2, pp. 28–93, 2009.

[11] A. Seyfarth, H. Geyer, M. Gunther, and R. Blickhan, "A movement criterion for running," *J. Biomechanics*, vol. 35, pp. 649–655, 2002.

[12] A. Seyfarth, H. Geyer, and H. Herr, "Swing-leg retraction : a simple control model for stable running," *J. of Exp. Bio.*, vol. 206, pp. 2547–2555, 2006.

[13] S. Carver, N. Cowan, and J. Guckenheimer, "Laterial stability of the spring-mass model suggests a two-step contrl strategy for running," *CHAOS*, vol. 19, 2009.

[14] M. Ankarali and U. Saranli, "Stride-tostride energy regulation for robust self-stabilty of torque-actuated dissipative spring-mass hopper," *Chaos*, vol. 20, no. 033121, 2010.

[15] S. Sastry, *Nonlinear systems : analysis, stability, and control*. Springer, 1999.

[16] J. Guckenheimer and S. Johnson, *Hybrid systems II*. London, UK: Springer-Verglag.

[17] V. Arnold, *Ordinary differential equations*. Cambridge, Massachusetts: The MIT PRess, 1973.

[18] J. Goncalves, A. Megrestski, and A. Dahleh, "Global analysis of piecewise linear systems using impact maps and surface lyapunov functions," *IEEE. Tras. Automat. Contr.*, vol. 48, no. 12, pp. 2089–2106, 2003.

[19] H. Khalil, *Nonlinear systems*, 3rd ed. Upper Saddle River, New Jersey: Prentice Hall, 2002.

[20] C. Farley, R. Blickhan, and C. Taylor, "Mechanics of human hopping : model and experiments," *Am. Zool.*, vol. 25, p. 54A, 1985.

[21] A. Hill, "The heat of shortening and the dynamic constants of muscle," *Proc. R. Soc. Lond. B*, vol. 126, no. 843, pp. 136–195, 1938.

[22] J. Winters, "Hill-based muscle models: a systems engineering perspective," in *Multiple Muscle Systems*, J. Winters and S.-Y. Woo, Eds. Springer New York, 1990, pp. 69–93.

[23] A. Bogert, K. Gerritsen, and G. Cole, "Human muscle-modelling from a user's perspective," *J. Elecromyogr. Kines.*, vol. 8, no. 2, pp. 119–124, 1998.

[24] R. M. Ghigliazza and P. Holmes, "Towards a neuromechanical model for insect locomotion: Hybrid dynamical systems," *Regular & Chaotic Dynamics*, vol. 10, no. 2, pp. 193–225, 2005.

[25] E. Hairer, G. Wanner, and P. Norsett, "Solving ordinary differential equations i – nonstiff problems," ser. Springer Series in Computational Mathematics. Springer-Verlag, 1993.