

# Modeling and Control of the Monopedal Robot Thumper

Ioannis Poulakakis and J. W. Grizzle

**Abstract**—A hybrid controller that induces stable running gaits on a monopedal robot is developed. The robot features a rigid leg with a revolute knee and a heavy torso with center of mass located far from the hip. The torso houses a novel powertrain that provides series compliance in the compression direction of the leg. The proposed control law is developed within the hybrid zero dynamics framework and it acts on two levels. On the first level, continuous within-stride control asymptotically imposes (virtual) holonomic constraints reducing the dynamics of the robot to a lower-dimensional hybrid subsystem. On the second level, event-based control stabilizes the resulting hybrid subsystem. The controller achieves the dual objectives of working harmoniously with the system’s natural dynamics and inducing provably exponentially stable running motions, while all relevant physical constraints are respected.

## I. INTRODUCTION

Thumper, Fig. 1(a), is a novel monopedal robot that grew out of a collaborative effort between the University of Michigan and Carnegie Mellon University to study legged locomotion. Two robots were built: a biped called MABEL, [4], and the monopod Thumper. One of the purposes of building these robots is to explore a novel powertrain design that incorporates compliance, with the objective of improving the power efficiency of locomotion, both in steady-state operation and in responding to disturbances. A second purpose is to inspire the development of advanced feedback control algorithms for running on level surfaces and walking on rough terrain.

The salient characteristics of Thumper’s morphology are summarized in Figs. 1 and 2. In particular, the robot features a relatively light leg with a revolute knee and a heavy torso with center of mass (COM) displaced from the hip, and it employs a transmission system that incorporates compliance for shock absorption and power efficiency; a detailed account of the design philosophy can be found in [5].

This paper focuses on control law design. The majority of control laws suitable for monopods with non-trivial torsos are derived under the assumption that the torso COM *coincides* with the hip joint; see [2], [3] and [1] for example. The purpose of this assumption, which is crucial for the success of these controllers, is that it results in trivial coupling between the torso and leg dynamics. However, the COM of Thumper’s torso is located high above the hip, as in a human.

I. Poulakakis and J. W. Grizzle are with the Control Systems Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122, U.S.A. Phone: +1-734-936-3875; fax: +1-734-763-8041; e-mail: {poulakas, grizzle}@eecs.umich.edu.

Additional results and other material supplemental to this work can be found in [4]. A video of the results of Section VII accompanies the paper.

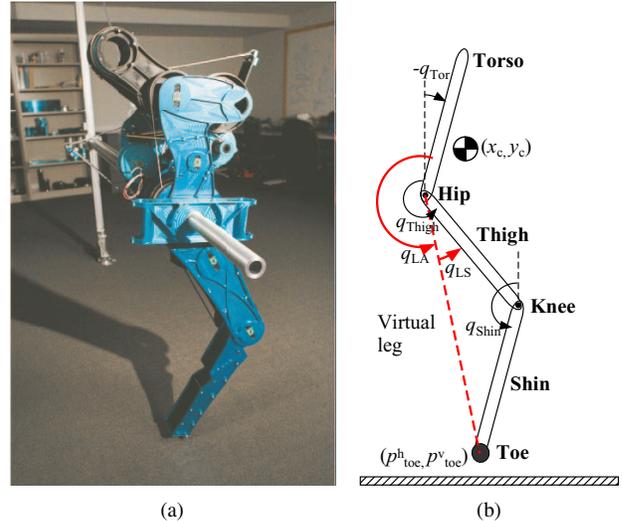


Fig. 1. **Left:** The monopedal robot Thumper (courtesy of Prof. J. Hurst). The leg consists of two links—the thigh and the shin—connected together with a knee joint. The thigh and shin are each 50cm long making the robot one meter tall at the hip. The overall mass is approximately 30kg. The torso contains the transmission mechanism and is attached to the leg via the hip joint, which does *not* coincide with the center of mass of the torso. **Right:** A schematic of Thumper with variables describing linkage configuration, and other important quantities.

To the best of the the authors’ knowledge, the asymmetric case is addressed only in [6].

The feedback law presented here builds upon control ideas developed for a simplified monopedal model, termed the Asymmetric Spring Loaded Inverted Pendulum (ASLIP); see [10] and [11]. The ASLIP is an extension of the Spring Loaded Inverted Pendulum (SLIP) that includes torso pitch dynamics nontrivially coupled to a massless leg, and it differs from Thumper mainly in two ways. First, the presence of leg mass in Thumper leads to energy loss at touchdown and, through conservation of angular momentum during flight, in torso rotation when the leg is positioned to its desired touchdown position. Second, in transmitting the actuator torques to the robot’s joints, Thumper’s powertrain introduces dynamic effects not present in the ASLIP. This paper proposes a controller that confronts these challenges and works in concert with the compliance present in Thumper to produce stable, periodic, natural, running motions.

## II. THE MORPHOLOGY OF THUMPER

Thumper is a planar monopedal robot comprised of three links assembled to form a torso and a kneed leg terminated in a point foot; see Fig. 1(a). It has been designed to incorporate some of the dynamic aspects of a SLIP, [2], while more

faithfully capturing the dynamics of a human in terms of the torso and leg morphology. This requirement has given rise to non-traditional mechanical design solutions, which, in turn, pose new challenges in developing control laws that induce stable hopping motions to Thumper.

Thumper features a novel compliant transmission system in which a collection of differentials is used to connect its two motors to the hip and knee joints so that one motor controls the angle of the *virtual leg* consisting of the line connecting the hip to the toe, and the second motor is connected *in series* with a spring to control the length of the virtual leg. In essence, the purpose of this transmission system is to ensure —mechanically, not by software— that the thigh and shin are coordinated, so that the leg behaves as if it were prismatic, despite the existence of a revolute knee. Another novelty in Thumper’s powertrain is that the spring in series with the leg-shape motor is *unilateral* in the sense that it compresses, but does not extend beyond its nominal rest length; instead, once the spring reaches its rest length, the position of the leg-shape motor,  $q_{mLS}$ , and the leg-shape angle,  $q_{LS}$ , are rigidly connected (i.e. no longer through a compliant element); see Fig. 2 for how this is achieved. This feature provides an advantage in initiating liftoff during hopping, because at liftoff the leg-shape motor does not “fight” the spring that is trying to extend due to the non-zero mass of the shin.

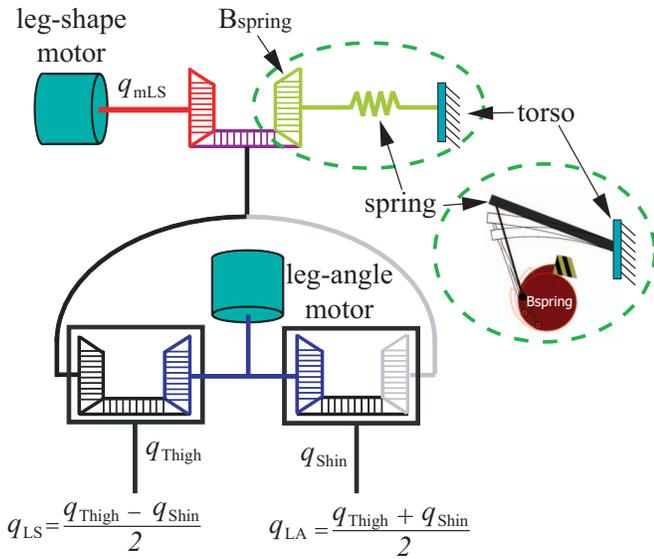


Fig. 2. Thumper’s powertrain system. The motors and spring are connected to the hip and knee joints via three differentials. The differentials are realized via cables and pulleys and they are connected so that the actuated variables are the leg and leg-shape angles,  $q_{LA}$  and  $q_{LS}$ , respectively; see Fig. 1(b). The spring is in series with the leg-shape motor: one end is connected to the torso and the other to the  $B_{spring}$  pulley via a cable, which makes the spring *unilateral*. When the spring reaches its rest position, the  $B_{spring}$  pulley hits a hard stop, as shown in the detail. When this happens, the leg-shape motor is, for all intents and purposes, rigidly connected to leg shape through a gear ratio. Design details of the transmission can be found in [5].

The transmission system introduces dynamic effects that have significant implications on controller design in at least

two ways. First, an immediate consequence of the presence of compliance is the increase of the degree of underactuation of the system. In fact, during the stance phase, Thumper has two degrees of underactuation —one more than the previously studied ASLIP; see [10] and [11]. Second, Thumper exhibits significant torso dynamics that cannot be captured by any point-mass approximation such as the SLIP. Moreover, contrary to most of the existing monopedal robots, Thumper’s leg is attached to the torso so that the hip joint does not coincide with the torso’s COM, resulting in *nontrivial coupling* between the torso and leg dynamics. This particular morphology cannot be addressed by the majority of control laws available for one-legged robot models, which are derived based on the assumption that the torso COM coincides with the hip joint.

### III. MODELING THUMPER

Running in Thumper can be represented by a hybrid system corresponding to the alternation of stance and flight phases. In this section, a mathematical model that captures the dynamics of the system in running is derived. The most general form of the robot model is discussed first. Important special cases, such as those arising when ground contact conditions create holonomic constraints at the leg ends, are presented next, followed by the transition maps relating the stance and flight phases.

#### A. Thumper’s Unconstrained Dynamics

Fig. 1(b) shows a schematic of a planar mechanism composed of three rigid links representing Thumper’s torso, thigh, and shin. A model describing the dynamics of the mechanism of Fig. 1(b) has five degrees of freedom (DOF): one DOF associated with the orientation of each link and two DOF associated with the horizontal and vertical displacements of the torso’s COM. The spring present in the transmission system provides an additional DOF, so that the combined mechanism —transmission plus rigid linkage— has six DOF in total.

Fig. 1(b) includes a convenient choice of generalized coordinates describing the configuration of the mechanism, namely, the leg angle,  $q_{LA}$ , i.e. the relative angle between the torso and the virtual leg connecting the hip with the toe, the leg-shape angle,  $q_{LS}$ , i.e. the relative angle of the virtual leg and the thigh that uniquely determines the length of the virtual leg, the angle of the torso with respect to the vertical,  $q_{Tor}$ , and the cartesian position of the COM,  $(x_c, y_c)$ . Finally, the additional DOF provided by the spring in the transmission will be described by the leg-shape motor position measured relative to the torso,  $q_{mLS}$ .

The corresponding equations of motion are obtained using the method of Lagrange. In computing the Lagrangian, the total kinetic energy is taken to be the sum of the kinetic energies of the transmission and the rigid linkage. Similarly, the total potential energy is computed by adding the corresponding potential energies assuming that the transmission contributes to the potential energy of the system only via

its elastic energy. The resulting model of the unconstrained robot dynamics can be determined as

$$D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e)\dot{q}_e + G_e(q_e) = B_e u, \quad (1)$$

where  $D_e$  is the mass matrix,  $C_e$  contains Coriolis and centrifugal terms, and  $G_e$  is the vector of the configuration dependent forces, gravitational and elastic.

### B. Thumper's Constrained Dynamics

The model (1) can be particularized to describe the flight and stance phase dynamics through the inclusion of proper holonomic constraints, as discussed below. Such constraints can be incorporated in (1) following standard procedures; see [9, pp. 265–275], for example.

1) *Flight Phase Dynamics*: Although the flight phase dynamics can be captured by (1), to simplify the model, it will be assumed that the spring deformation can be neglected during the flight phase. This assumption results in a holonomic constraint between the leg-shape motor position,  $q_{mLS}$ , and the leg-shape angle,  $q_{LS}$ , obtained by setting the spring deflection equal to zero. Including this constraint results in a model similar to (1), in which the configuration variables are selected to be  $q_f = (q_{LS}, q_{LS}, q_{Tor}, x_c, y_c)'$ .

2) *Stance Phase Dynamics*: During the stance phase, the toe is in contact with the ground. The toe-ground interaction is modeled as a frictionless pin-joint rigidly attached to the ground, resulting in a holonomic constraint that reduces the dimension of the unconstrained dynamics by two. This constraint can be incorporated to the unconstrained dynamics, resulting in a form similar to (1), in which the configuration variables are selected to be  $q_s = (q_{LA}; q_{LS}; q_{mLS}; q_{Tor})'$ .

In both the flight and stance phases, the second-order models describing the dynamics can be brought into standard state-space form by introducing the state vector  $x := (q; \dot{q})$ ,

$$\dot{x} = f(x) + g(x)u. \quad (2)$$

### C. Thumper's Transitions

1) *Flight-to-stance Transition*: The flight phase is terminated when the vertical distance of the toe from the ground becomes zero. The threshold function  $H_{f \rightarrow s}(x_f) := p_{toe}^v$ , with  $p_{toe}^v$  denoting the vertical distance between the toe and the ground, see Fig. 1(b), signifies the touchdown event at its zero crossing, and defines a smooth switching surface  $\mathcal{S}_{f \rightarrow s}$  in the flight state space  $\mathcal{X}_f$ , given by

$$\mathcal{S}_{f \rightarrow s} := \{x_f \in \mathcal{X}_f \mid H_{f \rightarrow s}(x_f) = 0\}. \quad (3)$$

The flight-to-stance transition map taking the (final) states  $x_f^-$  of the flight phase to the (initial) states  $x_s^+$  of the ensuing stance phase can be easily derived using the unconstrained dynamics by following the procedure described in [13, pp. 74–75] under the assumption of a rigid impact.

2) *Stance-to-flight Transition*: The stance phase is terminated by imparting a positive (upwards) acceleration to the toe when the vertical component  $F_{toe}^N$  of the ground reaction force becomes zero. This condition can be used to trigger the flight phase. On the other hand, to take full advantage

of the spring, it is desirable that, by the end of stance, the spring has completely returned the energy stored in it through the compression of the leg. Therefore, the nominal motion of Thumper is designed so that when the vertical ground reaction force becomes zero, the spring reaches its natural length.

However, away from the nominal conditions, the two events may not be synchronized. It will be assumed that the flight phase is initiated when either of the events occurs. Hence, if the vertical ground reaction force becomes zero first, then transition to flight occurs and the energy that is stored in the spring is assumed to be dissipated in the form of heat in the leg-shape actuator. On the other hand, if the spring reaches its nominal position first, the flight phase can be initiated by lifting the toe off the ground; this is possible due to the unilateral nature of the spring; see Fig. 2. In both cases, the corresponding events each define a surface  $\mathcal{S}_{s \rightarrow f}$  in the stance state space signifying the end of stance. In computing the transition maps note only that in the later case an “internal” impact occurs due to the pulley “Bspring” hitting a mechanical stop, not allowing the spring to enter its extension phase; see Fig. 2.

## IV. OVERVIEW OF THE CONTROL METHOD

This section outlines the general framework for controller design. In accordance to Raibert's observations for monopodal running, a running gait in Thumper can be divided into three control objectives, namely, regulate torso angle, hopping height, and forward velocity.

The control objectives can be encoded in a set of suitably designed constraints that are imposed on the robot dynamics during the stance and flight phases through its actuators. These constraints are parameterized with respect to monotonic quantities that are functions of the state – not time – and can be interpreted as (*virtual*) *holonomic constraints*, which restrict Thumper's dynamics on lower-dimensional surfaces embedded in the stance and flight state spaces, respectively. Loosely speaking, this *reduction-by-feedback* procedure effectively reduces the feasible motions of the robot by *coordinating* the actuated degrees of freedom of Thumper, so that a lower-dimensional hybrid subsystem “emerges” from the robot's closed-loop dynamics. This lower-dimensional hybrid subsystem governs the existence and stability properties of distinguished periodic orbits that correspond to running motions of interest on Thumper.

More specifically, to achieve the control objectives, the feedback law exploits the hybrid nature of the system by introducing control action on two levels; see Fig. 3. On the first level, continuous-time feedback laws  $\Gamma_\mu^c$  are employed in each of the continuous phases indexed by  $\mu \in \mathcal{M}$ , a finite index set. This stage introduces a set of parameters  $\alpha_\mu$  and  $\beta_\mu$ , and its purpose is to create an invariant and attractive surface  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$  embedded in the state space  $\mathcal{X}_\mu$  of the corresponding continuous phase.

On the second level, event-based updates of the parameters  $\alpha_\mu$  and  $\beta_\mu$  are performed at transitions between continuous phases. The division of the parameters introduced in the

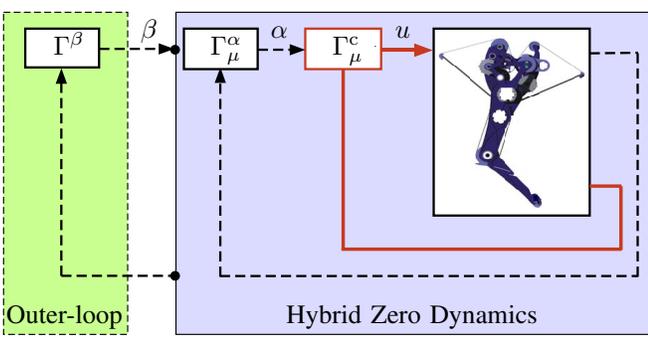


Fig. 3. Feedback diagram presenting the basic structure of the controller. Continuous lines represent signals in continuous time; dashed lines represent signals in discrete time. The control laws  $\Gamma_\mu^c$  and  $\Gamma_\mu^\alpha$  are intended to create a well defined hybrid zero dynamics (HZD), while the controller  $\Gamma^\beta$  ensures that the resulting HZD is exponentially stable.

continuous-time phases in two arrays, namely,  $\alpha_\mu$  and  $\beta_\mu$ , follows the structure of the event-based parameter update law, which is organized in an inner/outer-loop architecture. The inner-loop controller  $\Gamma_\mu^\alpha$  properly updates the parameters  $\alpha_\mu$  to ensure that the initial condition  $x_\mu^+$  of the corresponding continuous phase lies on the surface  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$ , i.e.,  $x_\mu^+ \in \mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$ . This inner-loop controller leads to the creation of a reduced-order hybrid subsystem governing the stability properties of the full-order model of Thumper. This subsystem is called the *Hybrid Zero Dynamics* (HZD). Finally, the outer-loop controller  $\Gamma^\beta$  completes the control design by updating the parameters  $\beta = \{\beta_\mu\}_{\mu \in \mathcal{M}}$  so that the resulting HZD is exponentially stable.

While the procedure described above can be made mathematically precise using the results in [7], [8] and [10], in this paper we will turn our attention only to design and implementation issues. In Sections V and VI we particularize these ideas through explicit constructions of a set of feedback laws  $\Gamma_\mu^c$ ,  $\Gamma_\mu^\alpha$  and  $\Gamma^\beta$  that are organized according to Fig. 3 and achieve the control objectives.

## V. CONTINUOUS-TIME CONTROL OF THUMPER

In this section, the feedback objectives are described in detail. A set of virtual holonomic constraints is devised, which, when enforced on the dynamics of Thumper through zeroing properly selected output functions, guarantees that the objectives are achieved, resulting in periodic running motions on Thumper.

### A. Stance Phase Virtual Holonomic Constraints

The purpose of the continuous-time control action during the stance phase is twofold. First, it ensures that the torso remains upright throughout stance and, in addition, that it enters the flight phase with suitable initial conditions. Second, the controller regulates the energy stored in the leg spring so that a desired hopping height is maintained.

In more detail, to the dynamics

$$\dot{x}_s = f_s(x_s) + g_s(x_s)u \quad (4)$$

of the stance phase associate the output

$$y_s = h_s(q_s, \alpha_s, \beta_s) := q_{c,s} - h_s^d(\theta(q_s), \alpha_s, \beta_s), \quad (5)$$

where  $q_{c,s}$  contains the *controlled* variables, which are selected to be the torso orientation  $q_{\text{Tor}}$  and the motor position  $q_{\text{mLS}}$ , i.e.  $q_{c,s} = (q_{\text{Tor}}, q_{\text{mLS}})'$ . In (5),  $h_s^d$  represents the desired evolution  $q_{\text{Tor}}^d$  and  $q_{\text{mLS}}^d$  of  $q_{\text{Tor}}$  and  $q_{\text{mLS}}$ , respectively. It corresponds to a spline that is parameterized with respect to the strictly monotonic (increasing) quantity  $\theta$  representing the angle formed by the line connecting the toe with the hip relative to the ground, i.e.,

$$\theta(q_s) := -\frac{3\pi}{2} + q_{\text{LA}} + q_{\text{Tor}}. \quad (6)$$

Roughly speaking,  $\theta$  is used to replace time in parameterizing the motion of Thumper in stance. The parameter arrays  $\alpha_s$  and  $\beta_s$  in (5) include polynomial coefficients.

Fig. 4 illustrates the general shape of the commanded constraints on  $q_{\text{Tor}}$  and  $q_{\text{mLS}}$ . The commanded constraint for  $q_{\text{Tor}}$  is composed by a “transient” part, whose purpose is to drive –in a smooth way– the torso angle from its initial value<sup>1</sup>,  $q_{\text{Tor}}^{s+}$ , to a desired final one,  $q_{\text{Tor}}^{s-}$ , in anticipation of liftoff. The transition stage is followed by a part during which  $q_{\text{Tor}}$  is kept constant and equal to  $q_{\text{Tor}}^{s-}$  so that switching to flight occurs with favorable initial conditions. In particular, commanding zero pitch velocity in the late stage of the stance phase ensures that the angular momentum associated with the torso is small when the system switches to flight, so that excessive pitching during flight is eliminated.

The transition part will be parameterized using a sixth order Beziér polynomial spanning a period from the beginning of the stance phase,  $\theta = \theta^{\min} = \theta_{\text{Tor}}^{\min}$ , until the angle  $\theta$  reaches a “settling” value  $\theta = \theta_{\text{Tor}}^{\max}$ . Mathematically,

$$h_{s,1}^d(\theta(q_s)) := \begin{cases} \sum_{k=0}^6 b_k(s_{\text{Tor}})\alpha(k,1), & 0 \leq s_{\text{Tor}} \leq 1 \\ \beta_{s,1}, & s_{\text{Tor}} > 1, \end{cases} \quad (7)$$

where the dependence of  $h_{s,1}^d$  on  $\alpha_s$  and  $\beta_s$  was suppressed. The coefficients  $b_k$  are given by

$$b_k(s) := \frac{M!}{k!(M-k)!} s^k (1-s)^{M-k}, \quad (8)$$

and the normalized independent variable is computed by

$$s_{\text{Tor}} := \frac{\theta - \theta_{\text{Tor}}^{\min}}{\theta_{\text{Tor}}^{\max} - \theta_{\text{Tor}}^{\min}}. \quad (9)$$

On the other hand, as shown in Fig. 4, the desired evolution of the motor position  $q_{\text{mLS}}$  is as follows. First, the motorshaft is kept at a constant angle  $q_{\text{mLS}}^{s+}$  until  $\theta = \theta_{\text{mLS}}^{\min}$ , the point at which the spring is maximally compressed. This ensures that energy is stored in the spring without the actuator performing unnecessary negative work on the leg. When maximum compression is reached, the actuator injects energy through compressing the spring further by rapidly

<sup>1</sup>Notation: “s+” denotes a value at the beginning of stance; “s-” denotes a value at the end of stance. Similar convention is adopted for flight.

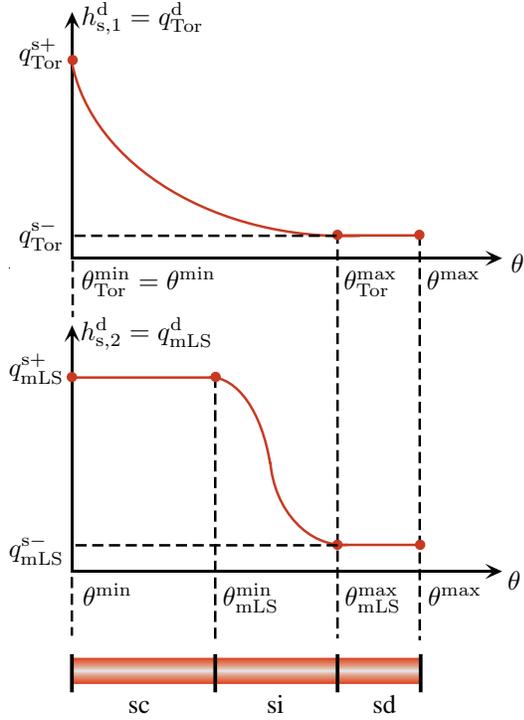


Fig. 4. The general shape of the stance phase commanded constraints. Note that  $\theta_{\text{Tor}}^{\text{max}} = \theta_{\text{mLS}}^{\text{max}}$  to avoid the introduction of an extra stance subphase.

repositioning the motorshaft at a new desired position,  $q_{\text{mLS}}^{s-}$ , which depends on the amount of energy that is required to maintain hopping. Then, the motorshaft is kept at this position until liftoff occurs.

Similarly to the torso angle, the transient period from  $\theta_{\text{mLS}}^{\text{min}}$  to  $\theta_{\text{mLS}}^{\text{max}}$  will be a fixed percentage of the nominal duration of the part of stance between the spring being maximally compressed and liftoff, and will be parameterized using a sixth order Beziér polynomial. The mathematical description of the constraint  $h_{d,2}$  for  $q_{\text{mLS}}$  is

$$h_{s,2}^d(\theta(q_s)) := \begin{cases} \alpha_{(-1,2)}, & s_{\text{mLS}} < 0 \\ \sum_{k=0}^6 b_k(s_{\text{mLS}})\alpha_{(k,2)}, & 0 \leq s_{\text{mLS}} \leq 1 \\ \beta_{s,2}, & s_{\text{mLS}} > 1, \end{cases} \quad (10)$$

where the dependence of  $h_{s,2}^d$  on  $\alpha_s$  and  $\beta_s$  was suppressed,  $b_k$  is given by (8), and

$$s_{\text{mLS}} := \frac{\theta - \theta_{\text{mLS}}^{\text{min}}}{\theta_{\text{mLS}}^{\text{max}} - \theta_{\text{mLS}}^{\text{min}}} \quad (11)$$

is the corresponding normalized independent variable.

This procedure introduces a number of parameters, namely, the coefficients of the polynomials used to describe the various branches of (7) and (10). These coefficients are included in the arrays  $\alpha_s$  and  $\beta_s$  defined as

$$\alpha_s := \{\alpha_{(0,1)}, \alpha_{(1,1)}, \dots, \alpha_{(6,1)}, \theta_{\text{Tor}}^{\text{min}}, \theta_{\text{Tor}}^{\text{max}}, \\ \alpha_{(-1,2)}, \alpha_{(0,2)}, \dots, \alpha_{(6,2)}, \theta_{\text{mLS}}^{\text{min}}, \theta_{\text{mLS}}^{\text{max}}\},$$

and

$$\beta_s := \{\beta_{s,1}, \beta_{s,2}\},$$

respectively. It is noted that, in order to avoid discontinuities in the commanded torques, the polynomial coefficients must satisfy certain requirements, which guarantee that the desired trajectories are  $C^2$  functions of  $s_{\text{Tor}}$  and  $s_{\text{mLS}}$ .

By properties of the Beziér polynomials, [13, p. 139], the coefficients  $\beta_{s,1}$  and  $\beta_{s,2}$  represent the values of the torso pitch angle and the motorshaft position prior to liftoff, i.e.  $q_{\text{Tor}}^{s-}$  and  $q_{\text{mLS}}^{s-}$ , respectively; see Fig. 4. Adjusting  $\beta_{s,1}$  determines the posture of the torso as the robot enters the flight phase, while adjusting  $\beta_{s,2}$  determines how much energy is injected or removed from the leg spring during the stance phase so that the hopping height is regulated at the desired value. Updating these parameters provides a powerful control input for the stabilization of Thumper, as will be detailed in Section VI-B.

Finally, note that the constraints (7) and (10) break the stance phase into three subphases, namely, stance-compression, stance-injection, and stance-decompression, which will be denoted by the indices “sc”, “si” and “sd,” respectively; see Fig. 4. These subphases are described by the same continuous-time dynamics (4) parameterized by the stance phase states,  $x_s$ . Moreover, they contain parameters that are entries of  $\alpha_s$ . The events separating the three stance subphases can be defined as crossings of the surfaces

$$\begin{aligned} \mathcal{S}_{\text{sc}} &:= \{x_s \in \mathcal{X}_s \mid \dot{B}_{\text{spring}}(x_s) = 0\}, \\ \mathcal{S}_{\text{si}} &:= \{x_s \in \mathcal{X}_s \mid \theta(q_s) - \theta_{\text{Tor}}^{\text{max}} = 0\}, \\ \mathcal{S}_{\text{sd}} &:= \mathcal{S}_{s \rightarrow f}, \end{aligned} \quad (12)$$

where  $B_{\text{spring}}$  is the spring deflection and  $\mathcal{S}_{s \rightarrow f}$  is the stance-to-flight switching surface; see Section III-C.2. These events can be utilized to update the stance phase parameters  $\alpha_s$  not only at the beginning of the stance phase, but also at transitions from one subphase to the next. This feature will be used in Section VI-A to update certain parameters when maximum compression is detected.

### B. Flight Phase Virtual Holonomic Constraints

The purpose of the continuous-time controller during the flight phase is to place the leg at a proper configuration in anticipation of touchdown. In more detail, to the continuous dynamics

$$\dot{x}_f = f_f(x_f) + g_f(x_f)u \quad (13)$$

of the flight phase associate the output

$$y_f = h_f(q_f, \alpha_f, \beta_f) := q_{c,f} - h_f^d(\ell(q_f), \alpha_f, \beta_f), \quad (14)$$

where the vector  $q_{c,f}$  contains the *controlled* variables. In selecting  $q_{c,f}$  note that, as was recognized by Raibert in [12] and others, e.g. [2], the position of the foot as it touches the ground at the end of flight has a strong influence on the ensuing stance phase providing an effective means for controlling forward speed. Moreover, when placing the leg, toe stubbing and excessive pitching of the torso should be avoided. These objectives can be achieved through commanding properly designed virtual constraints on the horizontal distance between the toe and the COM and on the leg-shape angle,  $(p_{\text{toe}}^h - x_c)$  and  $q_{\text{LS}}$ , respectively, i.e.,  $q_{c,f} = (p_{\text{toe}}^h - x_c, q_{\text{LS}})'$ .

In (14),  $h_f^d$  corresponds to the desired evolution of  $q_{c,f}$ , which depends on the parameters  $\alpha_f$  and  $\beta_f$ , and is parameterized with respect to the strictly monotonic quantity  $\ell$ . The variable  $\ell$  corresponds to the horizontal distance traveled by the COM during flight, i.e.,

$$\ell(q_f) := x_c - x_c^{f+}, \quad (15)$$

where  $x_c^{f+}$  is the position of the COM at the beginning of the flight phase. The desired evolution of the controlled variables in flight will be parameterized using sixth order Beziér polynomials, i.e.

$$h_f^d(\ell(q_f)) := \begin{bmatrix} \sum_{k=0}^5 b_k(s_f)\alpha_{(k,1)} + b_6(s_f)\beta_f \\ \sum_{k=0}^6 b_k(s_f)\alpha_{(k,2)} \end{bmatrix}, \quad (16)$$

where the dependence of  $h_f^d$  on  $\alpha_f$  and  $\beta_f$  has been suppressed, and

$$s_f := \frac{\ell - \ell^{\min}}{\ell^{\max} - \ell^{\min}} \quad (17)$$

is the normalized independent variable. Note that  $\ell^{\min} = 0$ , since at the beginning of the flight phase  $x_c = x_c^{f+}$ .

As in the stance phase, this procedure introduces a number of parameters corresponding to the coefficients of the polynomials defining  $h_f^d$ , which are grouped in the array

$$\alpha_f := \{\alpha_{(0,1)}, \dots, \alpha_{(5,1)}, \alpha_{(0,2)}, \dots, \alpha_{(6,2)}, \ell^{\min}, \ell^{\max}\}.$$

In order to explain why the coefficient of the highest order monomial of  $h_{f,1}^d$  is denoted by  $\beta_f$  instead of  $\alpha_{(6,1)}$ , note that, by properties of the Beziér polynomials, see [13, p. 139],  $h_{f,1}^d(\ell^{\max}) = \beta_f$ . Hence, the coefficient  $\beta_f$  corresponds to the position of the toe relative to the COM just prior to touchdown, i.e.  $(p_{\text{toe}}^h - x_c)^{f-}$ , and it provides a powerful control input for regulating the forward speed. As a result, following the control diagram of Fig. 3,  $\beta_f$  will be updated in the outer-loop discrete controller to achieve stability of the hybrid zero dynamics, and not in the inner-loop controller like the rest of the coefficients grouped in  $\alpha_f$ .

### C. Continuous-time Controller: Enforcing the Constraints

In what follows, the finite index set  $\mathcal{M} = \{\text{sc}, \text{si}, \text{sd}, \text{f}\}$ , contains indices corresponding to each of the continuous-time phases, namely, stance-compression, stance-injection, stance-decompression, and flight. Hence, for each  $\mu \in \mathcal{M}$  an output function

$$y_\mu := h_\mu(q_\mu, \alpha_\mu, \beta_\mu), \quad (18)$$

has been associated in the previous section with the continuous dynamics

$$\dot{x}_\mu = f_\mu(x_\mu) + g_\mu(x_\mu)u. \quad (19)$$

For given values of  $\alpha_\mu$  and  $\beta_\mu$ , differentiating (18) twice with respect to time results in

$$\frac{d^2 y_\mu}{dt^2} = L_{f_\mu}^2 h_\mu(x_\mu, \alpha_\mu, \beta_\mu) + L_{g_\mu} L_{f_\mu} h_\mu(q_\mu, \alpha_\mu, \beta_\mu)u, \quad (20)$$

where  $L_{g_\mu} L_{f_\mu} h_\mu(q_\mu, \alpha_\mu, \beta_\mu)$  is the decoupling matrix. Under the condition that  $L_{g_\mu} L_{f_\mu} h_\mu(q_\mu, \alpha_\mu, \beta_\mu)$  is invertible,

$$u^*(x_\mu, \alpha_\mu, \beta_\mu) := - \left( L_{g_\mu} L_{f_\mu} h_\mu(q_\mu, \alpha_\mu, \beta_\mu) \right)^{-1} L_{f_\mu}^2 h_\mu(x_\mu, \alpha_\mu, \beta_\mu), \quad (21)$$

is the unique control input that renders the surface

$$\mathcal{Z}_{(\alpha_\mu, \beta_\mu)} = \{x_\mu \in \mathcal{X}_\mu \mid h(q_\mu, \alpha_\mu, \beta_\mu) = 0, L_{f_\mu} h_\mu(x_\mu, \alpha_\mu, \beta_\mu) = 0\} \quad (22)$$

invariant under the flow of the closed-loop dynamics; that is, for every  $z \in \mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$ ,

$$f^*(z, \alpha_\mu, \beta_\mu) := f_\mu(z) + g_\mu(z)u^*(z, \alpha_\mu, \beta_\mu) \in T_z \mathcal{Z}_{(\alpha_\mu, \beta_\mu)}. \quad (23)$$

Essentially, imposing the (virtual) holonomic constraints by zeroing the corresponding outputs reduces the dimension of the system by restricting its dynamics on the surface  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$  embedded in the corresponding continuous-time state space  $\mathcal{X}_\mu$ . Following standard terminology, the surface  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$  is called the *zero dynamics manifold*, and

$$\dot{z} = f^*|_{\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}}(z, \alpha_\mu, \beta_\mu) \quad (24)$$

is the corresponding *zero dynamics*. To establish *attractivity* of  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$ , the input (21) is modified as

$$u = \Gamma_\mu^c(x_\mu, \alpha_\mu, \beta_\mu) := \left( L_{g_\mu} L_{f_\mu} h_\mu(q_\mu, \alpha_\mu, \beta_\mu) \right)^{-1} \left[ v(y_\mu, \dot{y}_\mu, \epsilon) - L_{f_\mu}^2 h_\mu(x_\mu, \alpha_\mu, \beta_\mu) \right], \quad (25)$$

where

$$v(y_\mu, \dot{y}_\mu, \epsilon) := -\frac{1}{\epsilon^2} K_P y_\mu - \frac{1}{\epsilon} K_V \dot{y}_\mu, \quad (26)$$

and  $K_P, K_V$  are appropriately chosen gain matrices, and  $\epsilon > 0$ . Under the continuous-time feedback laws  $\Gamma_\mu^c$ , constructed for each phase  $\mu$ , the solutions of (19) converge to the invariant surface  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$  exponentially fast at a rate that depends on  $\epsilon$ . In other words, the controller ensures that the virtual holonomic constraints, which are defined by (18) in the form of outputs that are zeroed by the control action, are (asymptotically) imposed on Thumper, thus restricting its dynamics in each phase on the corresponding lower-dimensional surface  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$ .

## VI. EVENT-BASED CONTROL OF THUMPER

A key implication of the hybrid nature of Thumper combined with the (trivial) dynamics  $\dot{\alpha}_\mu = 0$  and  $\dot{\beta}_\mu = 0$  governing the parameters in the continuous phases, is the possibility of updating  $\alpha_\mu$  and  $\beta_\mu$  in an event-based manner. In this section, a rule for updating the parameters  $\alpha_\mu$  will be devised, so that at the beginning of each continuous-time phase the initial condition lies on the corresponding surface  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$ . This results in a well-defined hybrid subsystem governing the stability of Thumper, i.e. the hybrid zero dynamics, which can be rendered exponentially stable through updating the parameters  $\beta_\mu$  in an outer-loop, as is schematically shown in Fig. 3.

### A. Inner-loop Event-based Control for Invariance

As shown in Fig. 5, at the beginning of a continuous phase  $\mu$ , the parameters  $\alpha_\mu$  can be updated based on feedback of the corresponding initial state  $x_\mu^+$ , i.e.,

$$\alpha_\mu^+ := \Gamma_\mu^\alpha(x_\mu^+), \quad (27)$$

with the purpose of the feedback law  $\Gamma_\mu^\alpha$  being to achieve invariance in the hybrid setting. This is accomplished by updating  $\alpha_\mu$  to a new value  $\alpha_\mu^+$  so that the surface  $\mathcal{Z}_{(\alpha_\mu^+, \beta_\mu)}$  is locally “deformed” at the beginning of the phase  $\mu$  to include the corresponding initial condition  $x_\mu^+$ , i.e.  $x_\mu^+ \in \mathcal{Z}_{(\alpha_\mu^+, \beta_\mu)}$ . Essentially,  $\alpha_\mu$  is updated so that it affects only the *entry* conditions to the continuous phase  $\mu$ ; the updated values leave the *exit* conditions unaffected. Note here that the exit conditions from phase  $\mu$  depend partly on  $\beta_\mu$ , which will be updated via the outer loop in order to achieve stability, as will be discussed in Section VI-B below.

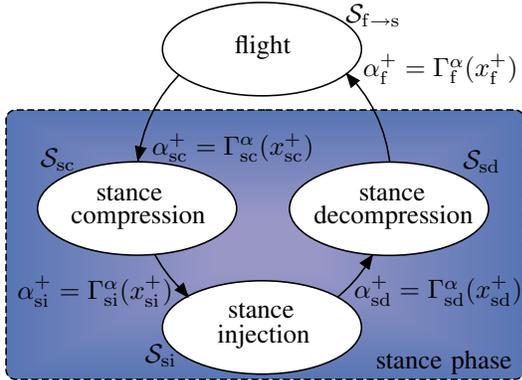


Fig. 5. Inner-loop event-based control for achieving invariance of the surfaces  $\mathcal{Z}_{(\alpha_\mu, \beta_\mu)}$ ,  $\mu \in \mathcal{M}$ . This controller, in combination with the continuous-time control action (25), results in a well defined hybrid zero dynamics, which will be rendered exponentially stable via the outer-loop controller of Section VI-B; see Fig. 6. The symbols  $\alpha_{sc}^+$ ,  $\alpha_{si}^+$ ,  $\alpha_{sd}^+$  represent updated values of  $\alpha_s$  at the beginning of the corresponding stance subphases.

### B. Outer-loop Event-based Control for Stability

Under the effect of the continuous-time controllers  $\Gamma_\mu^c$  and the inner-loop updates of the stance and flight parameters  $\alpha_s$  and  $\alpha_f$  according to Sections V and VI-A, respectively, the resulting closed-loop system takes the form

$$\Sigma : \begin{cases} \begin{bmatrix} \dot{x}_f \\ \dot{\alpha}_f \end{bmatrix} = \begin{bmatrix} f_{f,cl}(x_f, \alpha_f, \beta) \\ 0 \end{bmatrix}, & x_f^- \notin \mathcal{S}_{f \rightarrow s} \\ \begin{bmatrix} x_f^+ \\ \alpha_f^+ \end{bmatrix} = \begin{bmatrix} \Delta(x_f^-, \beta) \\ \Gamma(x_f^-, \beta) \end{bmatrix}, & x_f^- \in \mathcal{S}_{f \rightarrow s}, \end{cases} \quad (28)$$

as shown in Fig. 6. In (28),

$$f_{f,cl}(x_f, \alpha_f, \beta) := f_f(x_f) + g_f(x_f)\Gamma_f^c(x_f, \alpha_f, \beta), \quad (29)$$

$\mathcal{S}_{f \rightarrow s}$  is given by (3), and  $\Delta, \Gamma$  are discrete maps resulting from appropriate compositions of the stance-to-flight and flight-to-stance transition maps, the inner feedback laws and the flow maps associated with the stance phase.

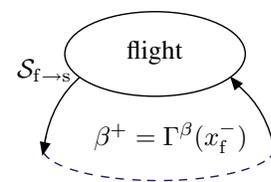


Fig. 6. Outer-loop event-based control for achieving stability of the hybrid zero dynamics.

A critical aspect of (28) is its dependence on the parameter array  $\beta = \{\beta_s, \beta_f\}$ , which can be selected according to an outer-loop feedback law  $\Gamma^\beta$  shown in Fig. 3, whose purpose is to exponentially stabilize (28). One way of designing  $\Gamma^\beta$  is by using discrete LQR techniques. Selecting the Poincaré section to be the surface  $\mathcal{S}_{f \rightarrow s}$ , i.e., the ground surface, the Poincaré map  $\mathcal{P} : \mathcal{S}_{f \rightarrow s} \rightarrow \mathcal{S}_{f \rightarrow s}$  associated with the hybrid system (28) gives rise to the discrete-time control system

$$x_f^-(k+1) = \mathcal{P}(x_f^-(k), \Gamma(x_f^-(k), \beta(k)), \beta(k)), \quad (30)$$

where  $x_f^-(k)$  is the state just prior to  $k$ -th touchdown. Linearizing (30) and implementing a discrete LQR result in the following rule for updating  $\beta$

$$\beta^+ := \beta(k) = \bar{\beta} + K[x_f^-(k) - \bar{x}_f^-] =: \Gamma^\beta(x_f^-), \quad (31)$$

where  $\bar{x}_f^-$  is the nominal value of the state just prior to  $k$ -th touchdown, and  $\bar{\beta}$  the nominal value of the parameters  $\beta$ . The feedback controller (31) guarantees that all the eigenvalues of the linearization of (30) are within the unit circle, and completes the control design.

## VII. SIMULATION RESULTS

Consider the hybrid dynamics of the Thumper in closed-loop with the feedback controllers developed in Sections V and VI. The problem of finding a periodic motion for the closed-loop system can be cast as a constrained minimization problem according to the procedure of [13, Ch. 6]. This procedure results in a fixed point  $\bar{x}_f^-$  of the Poincaré return map (30), and in the nominal values  $\bar{\alpha}$  and  $\bar{\beta}$  of the parameters. Fig. 7 shows the virtual holonomic constraints for two steps corresponding to a periodic running motion of Thumper. The resulting motion respects all the constraints; the input torques are within the capabilities of the actuators, the ground reaction forces respect the friction cone constraints and the unilateral nature of the toe/ground interaction, etc.

To illustrate the orbit’s local stability, the state prior to touchdown is perturbed away from the fixed point  $\bar{x}_f^-$ . An initial error of +2deg is introduced on each angle and an error of +10deg/s and +0.1m/s on each angular velocity and on the velocity of the COM, respectively. The resulting motion is shown in the accompanying video. Fig. 8 presents the constraints imposed on torso angle and leg-shape motorshaft position during stance (top), and the evolution of the errors in torso angle and horizontal velocity in discrete time. It is interesting to note how the event-based controller manipulates the shape of the imposed constraints. For instance, Fig. 8(b) shows how the outer-loop controller

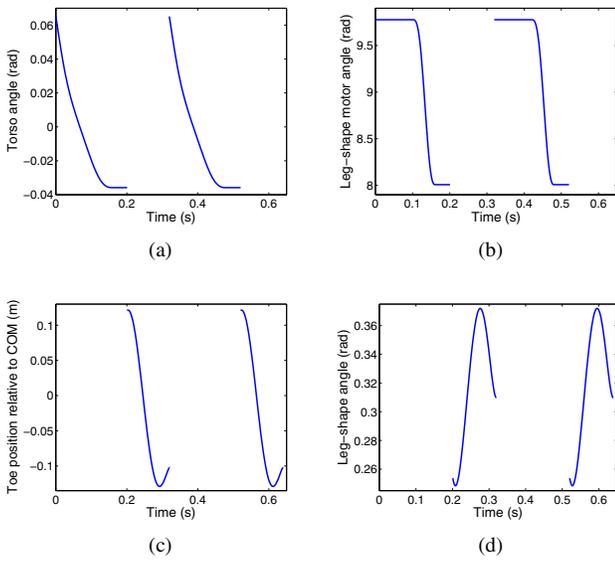


Fig. 7. Virtual constraints during nominal motion. **Top:** Stance phase torso angle, (a), and leg-shape motor position, (b), constraints given by (7) and (10), respectively. Compare with Fig. 4. **Bottom:** Flight phase toe horizontal position relative to the COM, (c), and leg-shape angle uniquely determining virtual leg length, (d).

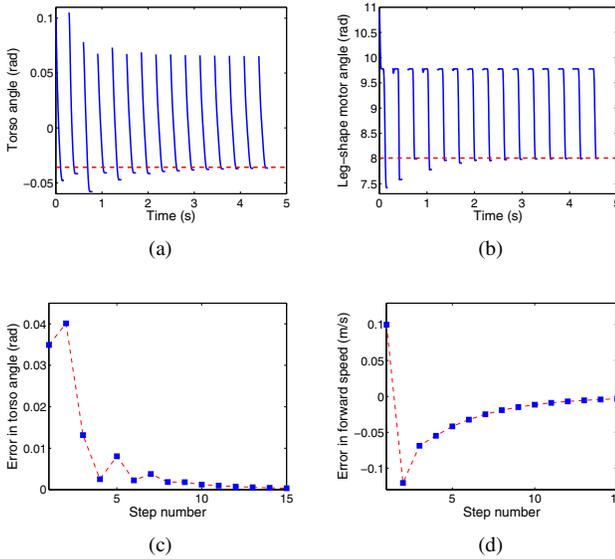


Fig. 8. **Top:** Stance phase virtual constraints during recovery from a perturbation: torso angle, (a), and leg-shape motor position, (b). Dashed line represents nominal liftoff values: The outer-loop controller adjusts the liftoff values for the torso angle and the motor position (corresponding to energy input) to ensure convergence to the nominal orbit. **Bottom:** Discrete errors in pitch angle and forward speed showing convergence to zero. The norm of the maximum eigenvalue is 0.8.

updates the coefficient,  $\beta_{s,2}$ , corresponding to the motorshaft position prior to liftoff,  $q_{mLS}^{s-}$ , in order to manipulate the energy stored at the spring.

## VIII. CONCLUSION

A stabilizing feedback control law for running has been developed for Thumper. The robot is composed of a relatively

light kneed leg mounted on a heavy torso, and the hip joint is displaced from the torso COM. The robot features a novel compliant powertrain, realized through cable differentials that are arranged so that the virtual leg, i.e., the line connecting the hip to the toe, is compliant. The proposed feedback law extends control ideas developed in the context of the ASLIP—a simplified model with a massless leg and an ideal transmission; see [11]—to an elaborate model that constitutes an accurate representation of Thumper’s dynamics. The control law development is systematic and works harmoniously with Thumper’s natural dynamics to induce running motions with provable stability. It is anticipated that these properties will render this controller an attractive alternative to traditional heuristic approaches by avoiding laborious trial-and-error procedures during experimental implementation.

## ACKNOWLEDGMENT

This work was supported by NSF grant ECS 0600869. The general features of the robot (morphology, dimensions, target mass and motor sizing) were set by J.W. Grizzle, J. Hurst, B. Morris, and A. Rizzi, on the basis of preliminary modeling and control law studies. The detailed mechanical design of Thumper was done solely by J. Hurst, who also oversaw the machining of the robot’s parts. Hae-Won Park provided an early version of Fig. 2.

## REFERENCES

- [1] M. Ahmadi and M. Buehler, “Control passive dynamic running experiment with the ARL Monopod II,” *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 974–986, Oct. 2006.
- [2] R. Altendorfer, D. E. Koditschek, and P. Holmes, “Stability analysis of legged locomotion models by symmetry-factored return maps,” *International Journal of Robotics Research*, vol. 23, no. 10-11, pp. 979–999, Oct. 2004.
- [3] N. Chrouvim and E. Papadopoulos, “Single actuator control analysis of a planar 3DOF hopping robot,” in *Robotics: Science and Systems I*, S. Thrun, G. Sukhatme, and S. Schaal, Eds. MIT Press, 2005, pp. 145–152.
- [4] J. W. Grizzle, “Papers on biped robots,” <http://www.eecs.umich.edu/~grizzle/papers/robotics.html>, retrieved on 15 September 2008.
- [5] J. W. Hurst, “The role and implementation of compliance in legged locomotion,” Ph.D. dissertation, The Robotics Institute, Carnegie Mellon University, August 2008.
- [6] S.-H. Hyon and T. Emura, “Symmetric walking control: Invariance and global stability,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 1455–1462.
- [7] B. Morris and J. W. Grizzle, “A restricted Poincaré map for determining exponentially stable periodic orbits in systems with impulse effects: Application to bipedal robots,” in *Proceedings of the IEEE International Conference on Decision and Control*, Seville, Spain, Dec. 2005, pp. 4199–4206.
- [8] —, “Hybrid invariance in bipedal robots with series compliant actuators,” in *Proceedings of the IEEE International Conference on Decision and Control*, San Diego, USA, Dec. 2006, pp. 4793–4800.
- [9] R. M. Murray, Z. Li, and S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.
- [10] I. Poulakakis and J. W. Grizzle, “Formal embedding of the spring loaded inverted pendulum in an asymmetric hopper,” in *Proceedings of the European Control Conference*, Kos, Greece, July 2007.
- [11] —, “Monopodal running control: SLIP embedding and virtual constraint controllers,” in *Proceedings of the IEEE/RSJ International Conference of Intelligent Robots and Systems*, San Diego, Oct. 2007.
- [12] M. H. Raibert, *Legged Robots that Balance*. Cambridge, MA: MIT Press, 1986.
- [13] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*. Taylor & Francis/CRC Press, 2007.