

# Series-Elastic Actuation Prototype for Rough Terrain Hopping

Katie Byl, Marten Byl, Martin Rutschmann, Brian Satzinger, Louis van Blarigan, Giulia Piovan, Jason Cortell  
Dept. of Electrical and Computer Engineering  
Dept. of Mechanical Engineering  
University of California at Santa Barbara  
Santa Barbara, CA, 93106, USA  
Email: katiebyl@ece.ucsb.edu

**Abstract**—In this paper, we describe development and modeling of a prototype hopping robot. The objective of our work is to create a test platform to verify control theory for fast, legged locomotion with limited sensor knowledge of upcoming, rough-terrain characteristics. Resulting running gaits aim to maximize the magnitude of unanticipated height variation,  $\Delta h$ , for which the hopper prototype can maintain reasonable control authority of the next apex state (and subsequent foothold). Toward quantifying controllability, we identify and quantify the range of states that our actuation strategy allows us to reach at the next apex height. We present the simplified model we use to calculate these reachable regions and provide corresponding system identification results, to justify the modeling assumptions used. This paper provides the following contributions to the field of practical robotics. First, we quantify the effectiveness of a real-world series-elastic actuation (SEA) strategy in allowing for simultaneous foothold planning and mitigation of perturbations. Second, we report on the deployment of a proof-of-concept, legged robot prototype being developed to test theoretical stance-phase control strategies.

## I. INTRODUCTION

Legged modalities should intuitively provide an advantage in achieving fast, agile robot locomotion across rough terrain, where discrete obstacles such as ditches or mole hills must be effectively negotiated. Recent development of BigDog [1], [2] shows that legged robots can respond to extreme perturbations by employing practical principles for maintaining balance. However, the trotting gaits employed by BigDog essentially model the surrounding terrain as locally flat and continuous, with a constant slope. Foothold selection is used to control balance but not explicitly employed as way to avoid “bad” footholds on terrain.

The spring-loaded inverted pendulum (SLIP) has been widely used to model the dynamics of such running and trotting gaits [3], from limit cycle analyses [4], to understanding human running characteristics [5], to an extensive body of work on the control of running gaits [3], [6], [7], [8], [9]. Recovery from perturbations due to sensor noise [10] or variability in terrain height or stiffness [11] has been investigated, as has step length planning [12]. In general, state-of-the-art planners can achieve either careful foot placement

or dynamic gaits for a particular legged robot [13]. Achieving both robustness to perturbations and reasonable foothold planning simultaneously is a significant on-going challenge for legged locomotion which motivates the present work.

In our research, we aim to decouple the problem of foothold planning from that of responding to environmental perturbations to the greatest extent practical. In running, the center of mass (COM) trajectory is not controllable during airborne, ballistic flight. Additionally, since perturbations are often due to unsensed terrain variability, a successful locomotion strategy must sometimes respond after a commitment has been made to command a particular touchdown angle and parameters such as spring pre-compression. Therefore, we focus in particular on control approaches during the *stance* phase. Our goal is to develop strategies to respond during the stance phase to correct for perturbations due to unsensed terrain variability or other external, impulsive events, while still allowing for a significant maneuverability in foothold placement, to exploit any information we do have about upcoming terrain.

The rest of the paper is organized as follows. In Section II, we introduce a simplified dynamic model that includes losses, due to an unsprung mass at the foot, and a series actuator to adjust the force across a passive spring during the stance phase. We present two different theoretical control strategies and quantify the practicality of each, based on real-world limitations we identify and anticipate for our prototype hopper. Then, in Section III, we introduce some overall design trade-offs and discuss two prototypes, a ballscrew driven leg (B) and cable driven leg (C), currently under development in our lab. Section IV describes our testing and control implementation. We then focus on a vertically-constrained prototype of Hopper B and present results from various time and frequency response tests performed in order to identify and verify our system dynamics model in Section V. Although for practical reasons, the presentation is given in an artificially sequential manner, keep in mind that Sections II, III, and IV are all integrally related and have been developed in conjunction with one another. Finally in Section VI, we summarize the design requirements and limitations of a practical hopper design in decoupling the problems of perturbation recovery and foothold selection.

This work is supported in part by DARPA (Contract W911NF-11-1-0077).

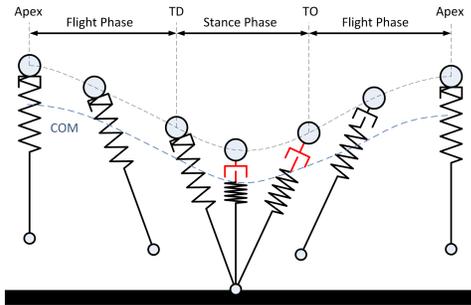


Fig. 1. Apex to apex transition with illustration of phases and events.

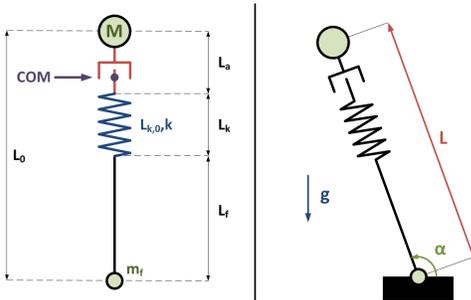


Fig. 2. Left side: Illustration of model parameters.  $L_0$ : leg length during flight [m],  $L_f$ : length of the fixed part of the leg [m],  $L_{k,0}$ : rest length of the spring [m],  $k$ : spring stiffness [N/m],  $M$ : point mass representing the body [kg],  $m_f$ : mass of the foot [kg]. The length  $L_a$  is controlled directly while  $L_k$  is a function of the commanded actuator motion and system dynamics. Right side: Illustration of the coordinate system during stance phase.

## II. SIMPLIFIED DYNAMICS AND CONTROL MODELING

### A. Approximate Dynamic Model

For our control and trajectory simulations, we use a slightly modified version of the widely used SLIP model. First, we include an unsprung mass located at the foot to account for energy losses. Second, we include an actuator in series with the spring which allows us both to replace lost energy and to trade off between forward velocity and apex height in adding this energy. Lastly, we assume that the leg angle may be set arbitrarily in flight by rotating the leg around its COM.

The trajectory of the model is separated into two different phases (Fig. 1): the stance phase when the foot is touching the ground and the ballistic flight phase without ground contact. For this paper we will refer to the changeover from the flight to the stance phase as touchdown (TD) and as takeoff (TO) for the transfer from stance phase to flight.

During flight, the COM follows a ballistic trajectory which can be described in a Cartesian coordinate system with the following equation of motions:

$$\ddot{x} = 0 \quad , \quad \ddot{y} = -g \quad (1)$$

The state where the maximum height  $y$  is reached is referred to as the apex state. Since the velocity in  $y$  direction is always zero at apex, the apex can be described using just three states  $[x, \dot{x}, y]$ . During the stance phase, we use a polar coordinate system (Fig. 2) to describe the position of the body. The state

of the hopper consist of the length of the leg  $L$ , the rate of change of the leg  $\dot{L}$ , the angle  $\alpha$  (measured CCW from the ground), and the angular velocity  $\dot{\alpha}$ . By using the Lagrangian,  $\mathcal{L}$  (2), one can derive the equation of motions for the stance phase (3), (4).

$$\mathcal{L} = \frac{1}{2}M\dot{L}^2 + \frac{1}{2}M(L\dot{\alpha})^2 - LMg \sin(\alpha) - \frac{1}{2}k(L_{k,0} + L_f + L_a - L)^2 \quad (2)$$

$$\ddot{\alpha} = \frac{-g \cos(\alpha) - 2\dot{L}\dot{\alpha}}{L} \quad (3)$$

$$\ddot{L} = -g \sin \alpha + L\dot{\alpha}^2 - \frac{k}{M}(L - L_f - L_a - L_{k,0}) \quad (4)$$

We assume that we can change the length of actuator  $L_a$  immediately. To take the limits of the actuator into account, we later will introduce limits on the acceleration and velocity of the commanded actuator motion. The length  $L$  will be  $L_0$  both at the beginning of the stance phase and at its end, when the hopper takes off again. The touchdown of the foot is modelled as an inelastic collision which means the kinetic energy of the foot is lost.

$$\vec{v}_{COM}^+ = \frac{M}{M + m_f} \vec{v}_{COM}^- \quad (5)$$

where  $\vec{v}_{COM}^-$  is the velocity of the COM just before TD, and  $\vec{v}_{COM}^+$  is the averaged velocity of the body mass and the now-stationary foot mass. The velocity of the body point mass itself does not change due to the impact, i.e.,  $\vec{v}_M^+ = \vec{v}_M^-$ . After TO, the hopper system will have both linear and angular velocity. The rotational part of the kinetic energy is considered to be lost since the leg angle will be actively driven to its new, desired touchdown angle.

### B. Stance-Phase Control of Next Apex State

Control of both foothold placement and forward dynamics are achieved through a combination of the selection of upcoming touchdown angle and control of the series elastic actuator during stance. In this work, we focus on the problem of control during stance. Our goals are (1) to allow for variable step size during fast, running gaits and (2) to remain robust to variability (both anticipated and unsensed) of the ground height at impact. The actuated SLIP model, described in the preceding subsection, is a hybrid dynamic system, with alternating phases in stance and in a ballistic trajectory. To quantify the step-to-step controllability, we look at the range of apex states that can be achieved.

To determine this range, we simulate the hopper using the same, representative initial conditions and the same touchdown angle but enforce different actuator motions during the stance phase. Regardless of the complexity of the chosen actuator motion,  $L_a(t)$ , that we attempt to employ (e.g., sinusoidal, etc.), the range of achievable apex states for the model appears to lie on nearly the same 2D manifold within the 3D space of

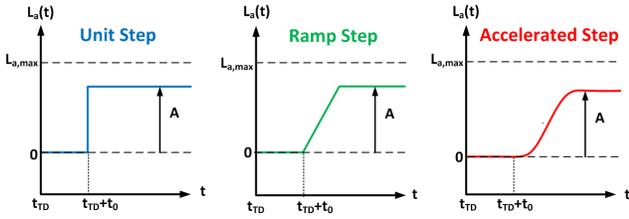


Fig. 3. Actuator movement definitions.

TABLE I  
COMMON SIMULATION PARAMETERS

Parameter	$L_0$ [m]	$g$ [ $m/s^2$ ]	$M$ [kg]	$m_f$ [kg]	$k$ [N/m]
Value	0.5	9.81	7.5	1	2000

possible apex,  $[x, \dot{x}, y]$ . For this reason we discuss only two output states, here forward speed and height  $[\dot{x}, y]$  are chosen, to illustrate the reachable space for each class of motion, and we test only a minimal class of motion types,  $L_a(t)$ .

Specifically, three different kinds of motions are considered. The first one is a unit step, illustrating the boundary introduced by the limiting the actuator range of travel. Taking a ramp step will identify the bound introduced by limits in actuator velocity. Ultimately, an acceleration-limited version of the ramp step (“accelerated step”) will consider the limited acceleration of the actuator. Since we can add the most energy if we have fast actuator movements, we will always apply the fastest possible acceleration to change  $L_a$  as rapidly as possible. All three step types are parametrized by a start time  $t_0$  and an amplitude  $A$ , as shown in Fig. 3.

We use two different simulation parameter sets: an initial parameter set, corresponding to lower-power operating conditions currently used (for safety reasons) during initial testing and control development, and a new parameter set, used to predict the performance of our final hopper design. Table I contains parameters which are fixed and do not change across all simulations, and Table II contains values unique to each parameter set. The higher accelerations and speeds of the new parameter sets are the result of higher motor voltage, larger ballscrew pitch, and increased motor current limits.  $a_{max}$  and  $v_{max}$  are estimated values the actuator can achieve regardless of the spring compression (i.e., in a “worst-case” condition). The new parameter set also includes a pre-compression of the spring at TD, to compensate for the mass of the body. This has the advantage of reducing the spring compression without reducing the total stance time significantly.

Using the initial parameter set, we can reach the apex states shown in Figure 4. For Fig. 4 and 5, the y-axis shows the apex height ( $y$ ) of the COM and the x-axis shows the apex

TABLE II  
DIFFERENT PARAMETER SETS

	$L_f$ [m]	$v_{max}$ [m/s]	$a_{max}$ [ $m/s^2$ ]	$L_{k,0}$ [m]
Initial set	0.37	0.5	10	0.13
New set	0.3	1.5	60	0.237

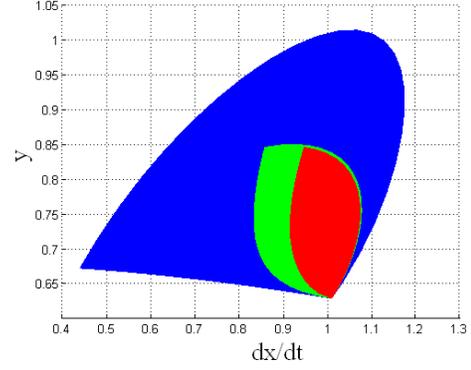


Fig. 4. Reachable apex states with different actuator motions for initial parameter sets. Blue corresponds to the Unit Step, Green to the Ramp Step and Red to the Accelerated Step.

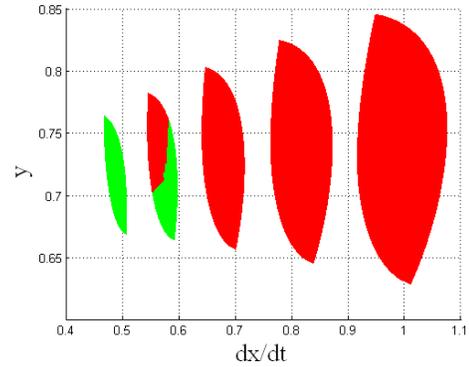


Fig. 5. Different precompressions for initial parameter set. Red and green: reachable areas without considering the limit of spring compression. Green: reachable area with limit. Precompression values for shapes from left to right: 8cm, 6cm, 4cm, 2cm, 0cm.

velocity ( $\dot{x}$ ). For these plots, the initial apex height is 0.7 [m] and the initial apex speed is 1 [m/s]. The touch down angle is selected such that with no actuation ( $L_a(t) = 0$ ), the apex velocity would be constant (i.e., the potential energy at apex is reduced while the kinetic energy at apex is constant).

The simulation used to produce Fig. 4 does not limit the spring travel, and many of the states are achievable only with a negative spring length  $L_k$  during stance. Since negative lengths are not physically achievable in our designs, one possible solution is to use a pre-loaded spring, with larger  $L_{k,0}$ , thus increasing the total compression (and storable potential energy) possible before bottoming out the spring travel. As can be seen in Fig. 5 with the spring constant  $k$  from Table I, the spring needs to be preloaded a minimum of 6 [cm] before we are able to perform a successful jump without bottoming out the spring. Pre-loading the spring has the negative effect of reducing the stance time, which reduces the energy that can be added into the system. This is observed in the shrinkage in the area of the achievable apex states. The same effect can also be observed by increasing the spring stiffness.

Roughly speaking, the broadness of the 2D region of possible next apex states can be mapped to the ability to control

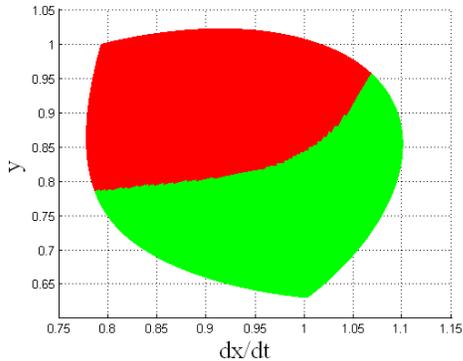


Fig. 6. Simulation with new parameter set. Red and green: reachable areas without considering the limit of spring compression. Green: reachable area with limit.

two factors, step length and total energy, of the corresponding touchdown state. For a narrow area, these two capabilities are too strongly coupled to effectively adjust step length and to negotiate terrain of variable height independently. Updating the actuator performance parameters to higher voltages and alternate components expands the range of reachable states (Figure 6). This expansion is primarily a result of the faster actuator velocity increasing the amount of energy added during the stance phase. The achievable states are still limited by the allowable spring travel. To achieve higher apex heights, either the spring needs to be stiffer or the allowable spring compression length needs to be increased.

Current applications of the stance phase control are noise rejection for our step size controller and a dead beat controller for constant speed running. The dead beat controller always tries to reach the same apex speed and height with respect to the last foothold. For a desired apex speed of 1 [m/s] and a desired apex height of 0.7 [m] disturbances in ground height of approximately 0.08 [m] in both direction can be handled by the controller.

### C. Enforced Symmetry of the Stance Phase

In the previous section, we evaluated the achievable apex states by defining a specific actuator input,  $L_a(t)$ . An alternate approach is to develop a control law that forces the COM to follow a desired trajectory during stance. Specifically, we can enforce a symmetric sinusoidal trajectory, as in [14], where the angular velocity of the COM is defined as:

$$\dot{\alpha}(t) = a \sin \alpha(t) + b. \quad (6)$$

This trajectory allows us to express the stance phase hopper dynamics in closed form. The parameters  $a$  and  $b$  are uniquely determined by the COM position and velocity at touchdown. The resulting leg-length dynamics follow from the equations of motion of the stance-phase (3),(4), and the required actuator displacement is given by:

$$L_a = \frac{M}{k} (\ddot{L} + g \sin \alpha - L \dot{\alpha}^2) + L - L_{k,0} - L_f. \quad (7)$$

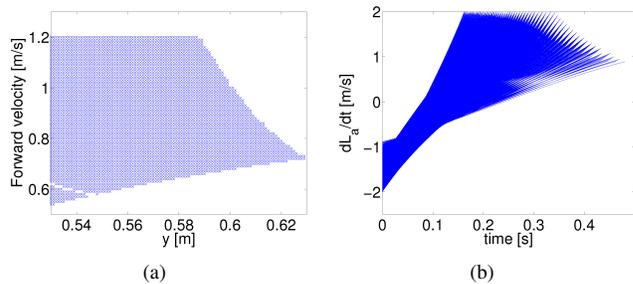


Fig. 7. The blue area in (a) refers to the set of initial conditions considered. Plot (b) shows the resulting actuator velocity during stance.

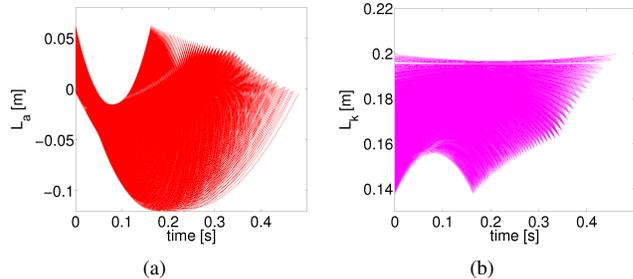


Fig. 8. This figure shows (a) the actuator displacement and (b) the spring length during stance

The advantages of computing a closed-form solution are manifold. For example, motion planning given a set of feasible footholds can be performed with a significantly reduced computation time compared to the classic SLIP model case. Although there have been several attempts in recent years to approximate the SLIP dynamics [15], [16], the resulting estimates are not accurate enough for achieving careful foothold planning [14]. Also, by having access to the analytical solution, several dead-beat controller strategies for hopping successfully in the presence of unknown or faulty measurements on rough terrain can be developed: e.g., to keep the forward velocity and the apex height constant at each step. In this subsection, we investigate operating conditions for which the actuation and spring travel requirements of this method are practical for implementation on our real-world hopper.

TABLE III  
SLIP PARAMETERS

$m$ [kg]	$k$ [N/m]	$L_0$ [m]	$L_{a,0}$ [m]	$L_{k,0}$ [m]	$L_f$ [m]
7.5	2000	0.5	0	0.25	0.3

Using the system parameters as defined in Table III, and starting from initial apex height in the range  $y \in [0.53, 0.7]$  [m], forward velocity in  $\dot{x} \in [0.5, 1.2]$  [m/s] and touch-down angle  $\alpha_{TD} = 100$  [deg], we compute the actuator displacement (and respective spring length) necessary to enforce the desired symmetric gait, according to (7). To consider the limits imposed by the actuator, only the subset of initial apex states that required an actuator velocity not exceeding 2 [m/s] were considered, as shown in Fig. 7(a). Fig. 7(b) shows the resulting velocity of the actuator. For completeness, we want to point

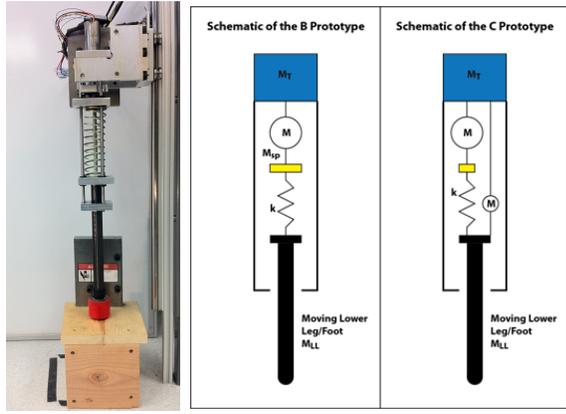


Fig. 9. Hopper B hardware prototype, with motion constrained to vertical hopping (left). One dimensional schematics of Hopper B (middle) and C (right).

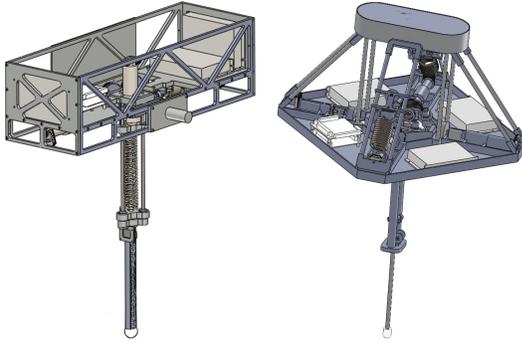


Fig. 10. Hopper B (left) and C (right) CAD designs.

out that, for initial conditions outside of this range, a faster actuator is required. Fig. 8(a) shows the actuator displacement for the set of initial conditions used, while Fig. 8(b) shows the respective spring length during the stance-phase. The actuator displacement required is between  $0.049 [m]$  (spring compression) and  $0.12 [m]$  (spring decompression).

As we can see from Fig 8(a) and 7(b), an instantaneous actuator displacement is required at the beginning of the stance phase. On a real system, however, we can expect a delay in the actuator response, and its consequences will have to be taken into account.

### III. HOPPER DESIGN AND STANCE DYNAMICS

For our research, we are considering two possible hopper designs which we have named Hopper B and Hopper C. A 1D schematic of the two designs is shown in Figure 9 while 3D CAD models are shown in Figure 10. The B-Design is similar to that described in [17] and [18], where an actuator is placed in series with a spring between the moving leg/foot and the robot body. Hopper C, somewhat similar to that in [9], uses two actuators: the primary actuator is placed in series with a spring and principally supports the mass of the torso with a second smaller actuator directly connected to the leg/foot. Design C offers the advantage of enhanced

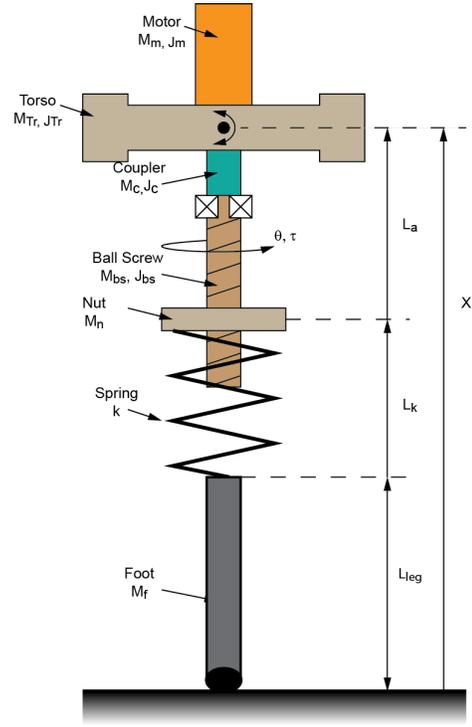


Fig. 11. B-Hopper Model Schematic

dynamic performance and agility at the expense of added design complexity.

To bound our design, we use some assumptions for the desired robot performance and size. Based upon the available space in our laboratory, we have sized our robot to nominally have a  $0.5 \text{ m}$  leg length, be capable of a vertical hop of  $\Delta X_{hop} = 0.25 \text{ m}$ , and have a maximum leg deflection equal to half the leg length ( $\Delta L_{max} = 0.25 \text{ m}$ ). We would like to have the spring be capable of storing all of the potential energy of the hop which leads to the following relationship for the nominal spring stiffness:

$$M_s g (\Delta X_{hop} + \Delta L_{max}) = \frac{1}{2} k \Delta L_{max}^2 \rightarrow \quad (8)$$

$$k = \frac{2 M_s g (\Delta X_{hop} + \Delta L_{max})}{\Delta L_{max}^2}$$

where  $M_s$  is the sprung mass. Preliminary analysis for both designs shows  $M_s \approx 6 \text{ kg}$  which leads to a desired spring rate of  $1600 \text{ N/m}$ .

We have built a one dimensional alpha prototype of the B hopper leg for design evaluation (photo in Figure 9). The hopper is  $60 \text{ cm}$  tall with a leg length of  $46 \text{ cm}$  and has a maximum permissible leg displacement of  $16 \text{ cm}$ . It is attached to a vertical linear guide to limit the motion to one dimension. The leg is actuated with a  $200\text{W}$  DC motor through a ball screw with a lead of either  $2 [\text{mm/rev}]$  or  $5 [\text{mm/rev}]$ . The displacement between the upper and lower legs, and the vertical displacement of the torso are measured using magnetic linear encoders, while the angular displacement of the motor is measured using a capacitance based rotary encoder. The output

$L_a$		Actuator Displacement
$L_k$		Spring Displacement
$\theta$		Motor Angle
$\tau$		Motor Torque
$M_m$	0.30 kg	Motor Mass
$M_c$	0.022 kg	Coupler Mass
$M_{Fr}$	7.310 kg	Frame Mass
$M_{bs}$	0.084 kg	Ball Screw Mass
$M_T$	7.716 kg	$M_T = M_m + M_c + M_{Fr} + M_{bs}$
$M_n$	0.23 kg	Nut Mass
$M_f$	TBD	Foot Mass
$J_m$	$3.33e-6 \text{ kg}\cdot\text{m}^2$	Motor Inertia
$J_c$	$1.18e-6 \text{ kg}\cdot\text{m}^2$	Coupler Inertia
$J_{bs}$	$0.65e-6 \text{ kg}\cdot\text{m}^2$	Ball Screw Inertia
$k$	1000–2000 N/m	Spring Stiffness
$L_{k0}$	$20 \leq L_{k0} \leq 25 \text{ cm}$	Free length of the spring
$P$	2 or 5 mm/rev	Ball Screw Pitch
$N_{bs}$	$2\pi/P$	
$K_t$	0.0276 Nm/A	Motor Torque Constant
$R_m$	0.386 $\Omega$	Motor Resistance
$\eta$	$\approx 0.9$	Ball Screw Efficiency
$V_s$	37-70 V	Maximum Supply Voltage
$\eta_{mc}$	$\approx 0.9$	Motor Voltage Efficiency

TABLE IV  
TABLE OF EXPECTED PROPERTIES

of the motor power electronics is intentionally limited to 60% of the nominal output to limit the likelihood of damaging the leg during testing. Since one of the design variables we wished to evaluate as part our research is the effect of varying the spring pre-load on the range of achievable foot placements, we procured an array springs with nominal spring constants between 800 and 1600 N/m, and lengths between 20 and 25 cm. The spring used during most of the initial testing was the most compliant with a length of 20 cm.

Figure 11 shows a schematic drawing of the B hopper design, illustrating the key design components as well as the variables used to derive the hopper dynamics. Table III lists the design variables and their expected values.

#### A. Stance Dynamics

We have derived the stance dynamics using a Lagrangian approach, where the ball screw displacement ( $L_a$ ) and spring displacement ( $L_k$ ) are the state variables. The stance phase dynamics can be expressed as

$$\ddot{L}_a = \left( \frac{1}{M_1 M_2 + M_n^2} \right) \begin{pmatrix} M_T k (L_k - L_{k0}) \\ +(M_T + M_n)(F_A - B_A) \end{pmatrix} \quad (9)$$

$$\ddot{L}_k = - \left( \frac{M_T}{M_T + M_n} \right) \ddot{L}_a - g - \frac{k}{M_T + M_n} (L_k - L_{k0}) - \frac{B_k}{(M_T + M_n)} \quad (10)$$

where

$$M_T = M_{Tr} + M_m + M_c + M_{bs} \quad (11)$$

$$M_1 = M_T + M_n \quad (12)$$

$$M_2 = M_T + N_{bs}^2 (J_m + J_c + J_{bs}) \quad (13)$$

$$F_A = N_{bs} \eta \tau = N_{bs} \eta K_t i \quad (14)$$

$$B_A = \text{Actuator friction losses} \quad (15)$$

$$B_k = \text{Spring friction losses.} \quad (16)$$

Some straight forward algebra is required to express these in state space format but that is left to the reader.

#### IV. CONTROL DESIGN AND IMPLEMENTATION

The existing hopper design is run using a PC-104 industrial computer. The motherboard is a VersaLogic Tiger Board, with a 1.6 GHz dual core Intel Atom processor. Digital and analog I/O is achieved using a Sensoray 526 PC-104 add-on board. The driving motor on the hopper is a Maxon 4-pole EC motor driven by a sinusoidal commutation device that accepts an analog current reference signal generated by the Sensoray board. Motor position is obtained from a rotational encoder that provides 512 cycles per revolution, corresponding to a linear resolution of 0.98 microns for the 2mm pitch ball screw design. Body height and leg compression are measured with 15 micron resolution.

Real-time control is implemented using xPC target software running a proprietary real time kernel from The Mathworks. The sample rate is 1ms. The xPC target computer and a host running Simulink communicate over an Ethernet connection to facilitate transferring programs, data, and parameter values. An independent hardware watchdog timer, implemented using an Arduino, monitors a square wave signal generated by the xPC target to ensure timing constraints are obeyed. If a sample update is late by more than 100 microseconds, the motor is disabled until the watchdog timer is manually reset. The watchdog timer allowed us to discover timing glitches which were eventually attributed to a BIOS feature called Firmbase, which preempts the CPU. Disabling Firmbase removed the problem.

Power is provided to the system by a TDK-Lambda DC power supply capable of delivering 16A at 60 V. The motor controller is run directly from this power supply, while all logic circuits are run from a switching mode DC-DC converter that converts the high voltage supply down to 24 V. Initial tests indicated that we were able to drive the power supply into an overvoltage shutdown condition by applying braking when the motor was moving quickly. To prevent this overvoltage condition from damaging equipment, the power supply voltage was reduced to 37.6 V for testing. A shunt regulator has since been installed to dissipate excess braking energy, but all tests were conducted with the supply set to 37.6 V.

#### V. SYSTEM IDENTIFICATION

To identify the parameters in our equations of motion, several tests were performed. To identify the parameters of the actuation degree of freedom, current input step tests were performed to establish time constants, natural frequency, damping ratio, ball screw efficiency, and demonstrate linearity of the system around small perturbations. To extract the parameters associated with the free motion of the hopper, tests were performed where the hopper was released from the equilibrium leg length and allowed to settle to its resting position. These tests were performed more than 25 times with two different masses, and an average of all drops was used as demonstrative of the system. This average is plotted in Fig. 12. These tests

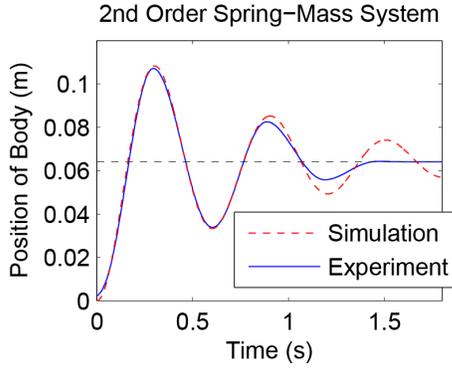


Fig. 12. Hopper B passive spring-mass time response.

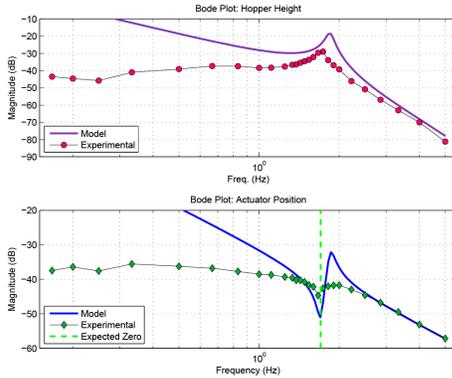


Fig. 13. Hopper B frequency response magnitude data.

allowed us to extract the spring constant, pre-compression length, natural frequency, the sprung mass of the system, and the damping ratio. These parameters are identified in Table V.

TABLE V  
SYS ID PARAMETERS ESTIMATES

$k$ (N/m)	$L_{precomp}$ (m)	$M$ (kg)	$M_{nut}$ (kg)	$\omega_n$ (rad/s)	$\zeta$
$985 \pm 30$	$0.025 \pm 0.003$	$8.6 \pm 0.1$	$0.3 \pm 0.05$	10.7	0.13

Additionally, a frequency response was gathered for both acutation/input and body position/input, shown in Fig. 13. Poles and zeroes (for the actuator position plot) are close to predicted values. Overall, the system is more damped that predicted, however, and the cables used are likely responsible for the flat experimental magnitude data at lower frequencies.

#### A. Experimental Verification of Stance-Phase Control

Fig. 14, we verify our identified parameters by performing current control step response tests for two different values of body mass. In each test, the system began at rest, with the mass pre-compressing the spring statically, due to gravity. A small, 5 Amp current input was given to the actuator for 0.2 seconds. This was long enough to reach the steady-state velocity of the system, and to get a small airborne period in the response. This steady state value depends on the voltage supply and is very repeatable. The initial slope of the velocity (i.e., the

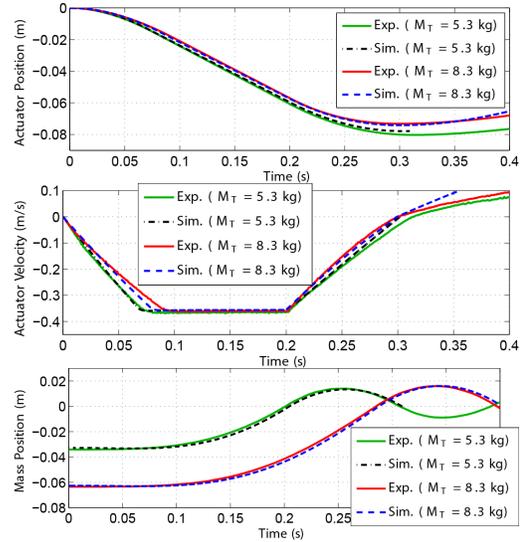


Fig. 14. Verification of system identification via current control step responses. With Hopper B constrained to vertical motion and initially standing at rest, a 5 Amp current step was applied for 0.2 sec for each of two load masses.

acceleration) is nearly linearly and, as previously mentioned in Section II-B, we use a worst-case acceleration to estimate overall performance. Note that  $y_M = 0$  in the mass position plot corresponds to the point at which the leg is fully extended and the foot is no longer touching the ground (ballistic motion) for  $y_M > 0$ . All responses match very well with our modeled dynamics, which include both dry friction and viscous damping.

## VI. CONCLUSIONS AND FUTURE WORK

In this paper, we examine the theoretical consequences of real-world design limitations on the ability of a hopping robot to plan variable-length steps and to recover from perturbations in terrain height at each step. Our goals are to decoupled the ability to adjust step length from the ability to add energy to the system, thus allowing one simultaneously to select isolated footholds and to cope with changes in terrain height, which require changes in potential energy of the hopper. Achieving this flexibility requires the ability to obtains large actuator travel,  $A$ , in Figure 3, and to time the start of this motion,  $t_{TD} + t_0$  with respect to the time of touchdown. Faster and larger actuator motions in turn require long spring travel and high acceleration and peak velocity of the actuator. Given our system identification, modeling, and simulation, we anticipate that Hopper B will be able to survive up to 16% of the leg length (8 cm) in unexpect terrain height change, and that it will allow for fast transitions in speed and step length, based on the region covered in Fig. 6. Performance for Hopper C should be even better, although there are also more risks involved in obtaining good performance from this more unique hopper design. We will be testing both Hopper B and Hopper C in the coming months, to determine whether the degree of flexibility in maneuverability is as great as we predict.

## REFERENCES

- [1] R. Playter, M. Buehler, and M. Raibert, "Bigdog," in *Proc. SPIE*, vol. 6230, 2006.
- [2] M. Raibert, K. Blankespoor, G. Nelson, R. Playter, and the Big-Dog Team, "Bigdog, the rough-terrain quadruped robot," *Proceedings of the 17th World Congress, The International Federation of Automatic Control*, 2008.
- [3] M.H.Raibert, M. Chepponis, and H. Brown, "Running on four legs as though they were one," *IEEE Journal of Robotics and Automation*, vol. 2, no. 2, pp. 70–82, 1986.
- [4] R. M. Ghigliazza, R. Altendorfer, P. Holmes, and D. Koditschek, "A simply stabilized running model," *SIAM Review*, vol. 47, no. 3, pp. 519–549, 2005.
- [5] T. A. McMahon and G. C. Cheng, "The mechanics of running: How does stiffness couple with speed?," *Journal of Biomechanics*, vol. 23, no. Supplement 1, pp. 65–78, 1990.
- [6] I. Poulakakis and J. Grizzle, "Monopedal running control: Slip embedding and virtual constraint controllers," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2007.
- [7] U. Saranli, W. Schwind, and D. Koditschek, "Toward the control of a multi-jointed, monoped runner," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2676 – 2682, 1998.
- [8] A. Sato and M. Buehler, "A planar hopping robot with one actuator: design, simulation, and experimental results," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 4, pp. 3540–3545, 2004.
- [9] G. Zeglin and B. Brown, "Control of a bow leg hopping robot," in *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, vol. 1, pp. 793 – 798, IEEE, May 1998.
- [10] K. Harbick and G. S. Sukhatme, "Robustness experiments for a planar hopping control system," *International Conference on Climbing and Walking Robots*, Sep 2002.
- [11] D. Koepf, K. Kemper, and J. Hurst, "Force control for spring-mass running and walking," in *Proc. IEEE Conference on Advanced Intelligent Mechatronics*, July 2010.
- [12] J. Hodgins and M. Raibert, "Adjusting step length for rough terrain locomotion," *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 289–298, June 1991.
- [13] K. Byl, A. Shkolnik, S. Prentice, N. Roy, and R. Tedrake, "Reliable dynamic motions for a stiff quadruped," in *Proc. of the 11th Int. Symposium on Experimental Robotics (ISER)*, 2008.
- [14] G. Piovani and K. Byl, "Enforced symmetry of the stance phase for the spring-loaded inverted pendulum." unpublished, Sept. 2011.
- [15] H. Geyer, A. Seyfarth, and R. Blickhan, "Spring-mass running: simple approximate solution and application to gait stability," *Journal of Theoretical Biology*, vol. 232, no. 3, pp. 315–328, 2005.
- [16] U. Saranli, O. Arslan, M. Ankarali, and O. Morgul, "Approximate analytic solutions to non-symmetric stance trajectories of the passive spring-loaded inverted pendulum with damping," *Nonlinear Dynamics*, vol. 64, no. 4, pp. 729–742, 2010.
- [17] H. Rad, P. Gregorio, and M. Buehler, "Design, modeling and control of a hopping robot," in *In Proc. IEEE/RSJ Conf. Intelligent Systems and Robots (IROS)*, pp. 1778–1785, 1993.
- [18] M. Ahmadi and M. Buehler, "Controlled passive dynamic running experiments with the arl-monopod ii," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 974–986, 2006.