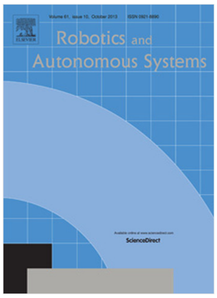


Adaptive height controller for an agile hopping robot

Jawaad Bhatti, Matthew Hale, Pejman Iravani, Andrew Plummer,
Necip Sahinkaya

PII: S0921-8890(16)30770-9
DOI: http://dx.doi.org/10.1016/j.robot.2017.07.004
Reference: ROBOT 2874

To appear in: *Robotics and Autonomous Systems*

Received date : 2 December 2016
Revised date : 4 June 2017
Accepted date : 7 July 2017

Please cite this article as: J. Bhatti, M. Hale, P. Iravani, A. Plummer, N. Sahinkaya, Adaptive height controller for an agile hopping robot, *Robotics and Autonomous Systems* (2017), http://dx.doi.org/10.1016/j.robot.2017.07.004

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Adaptive height controller for an agile hopping robot

Jawaad Bhatti[a], Matthew Hale[b], Pejman Iravani[b], Andrew Plummer[b], Necip Sahinkaya[c]

[a] *Blatchford Group, Basingstoke, UK*
[b] *Department of Mechanical Engineering, University of Bath, Bath, UK*
[c] *Mechanical and Automotive Engineering, Kingston University, London, UK*

Abstract

This paper addresses the problem of controlling the hopping height and stride length of a monoped hydraulic robot. Hopping over discontinuous, rough terrain with limited surfaces suitable for foot placement requires a controller capable of adjusting the hop height and landing foot position of the robot on each step. This motivates the need for an agile controller that uses the short window of time while the foot is on the ground (the stance phase) to exert the required action to reach the next landing position.

This paper contributes a simple yet effective adaptive controller capable of changing the flight time within a single hop. The controller does not require force feedback and is capable of self-tuning its feedback gain parameters in response to changing ground parameters using the results of previous hops. The main contribution of the paper is the development of an analytical understanding of why the controller is capable of adjusting the height in a single step and how the errors in the achieved height can be used to tune automatically the controller gains. This allows the controller to be successfully implemented even if the conditions or parameters are initially unknown, automatically correcting for errors.

The controller is first derived for height control of hopping vertically, with no horizontal motion, from an analytical approximation. This is tested in simulation, using a spring-damper model and a more detailed model with a foot mass and compliant ground. The controller is then applied to a hydraulic spring-loaded hopper monoped. An extension allows the control of running on a treadmill, with constant horizontal speed.

Keywords: Hopping controller, Legged locomotion, Running, Hydraulically actuated robot, Adaptive control

---

1. Introduction

Dynamically stable legged locomotion aims to build systems and controllers that exploit the passive dynamics of the mechanism while ensuring they remain stable and achieve their desired gait. This approach contrasts with statically stable gaits where the centre of gravity of the robot remains within the contact points, such as in many multi-legged robots [1]. The main advantages of the dynamically stable approach are their energy efficiency [2, 3], agility and in some cases their self-stabilisation properties.

Legs have a significant advantage over wheels or tracks when tackling rough terrain because a continuous support surface is not required. This means that terrain with isolated footholds can be traversed if foot placement can be precisely controlled. For statically stable robots with large bases of support or multiple legs, foot placement is purely a kinematic problem *i.e.* positioning legs to achieve the desired gait. If a robot is required to cross rough terrain quickly or jump large gaps or heights then the problem involves the dynamic control of foot placement *i.e.* taking the right control action during ground contact in order to launch into the flight phase with a ballistic trajectory that will lead to the desired next foot placement spot.

One of the main obstacles to the development of control strategies for dynamic legged locomotion has been the difficulty in solving the dynamics of the stance phase. This makes it difficult to find control strategies to select inputs which will result in the desired next foot landing position. Section 2 discusses the main approaches taken in related works, which broadly either use an analytical approximation to the solution or use numerical integration.

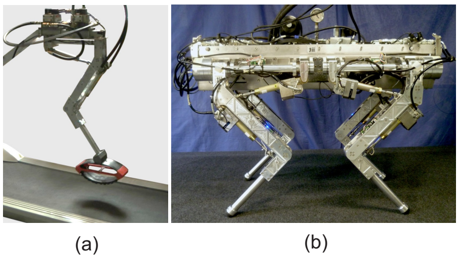

(a) (b)

Figure 1: (a) The hydraulic hopping monoped robot used in this research and (b) HyQ quadruped robot from which it is derived [27]

This paper presents a novel approach for controlling the hopping height of a monoped instantaneously, *i.e.* during the flight phase the controller computes the required action to achieve the demand height of the next step. The controller is experimentally validated with the robot shown in Fig. 1(a). The monoped is developed from a HyQ quadruped leg (Fig. 1(b)).

Email addresses: `Jawaad.Bhatti@blatchford.co.uk` (Jawaad Bhatti), `m.hale2@bath.ac.uk` (Matthew Hale), `p.iravani@bath.ac.uk` (Pejman Iravani), `a.r.plummer@bath.ac.uk` (Andrew Plummer), `m.sahinkaya@kingston.ac.uk` (Necip Sahinkaya)

The controller uses the height of the previous hop, together with the height demand for the next hop, to compute the actuator's demand during stance. It does not rely on any explicit model of the dynamics, thus the approach is able to adjust to changing conditions; for example in this paper the compliance of the ground is varied.

After the discussion of previous work in the area (Section 2), a novel control strategy is derived and tested in simulation in section 3. The experimental setup is introduced and the control strategy adapted for implementation in section 4. Results are presented for stationary hopping on ground of varying compliance and hopping at a constant speed on a treadmill in section 5.

2. Previous work

With some exceptions discussed here, research on legged hoppers has been mainly focused on achieving stable, steady state running [4]. The goal has been to approach a desired hopping height and running speed over a number of hops in a way that is robust to disturbances, for instance, unforeseen changes in the ground height. Agile hopping, meaning the ability to perform rapid changes in speed and direction has not been studied in detail.

Classic work on monoped hopping from Raibert et. al. at MIT's leg lab [5] included several impressive dynamic robots able to hop up and down steps and perform acrobatics. These robots operated on a very simple strategy, where the control problem is broken down into height, horizontal velocity and body orientation, termed the "three part controller". So long as these three problems can be considered decoupled, computationally simple controllers can successfully control dynamic gaits in real time. An extension of this concept showed that step length could be controlled by varying any one of the forward speed, flight duration or stance duration [6].

Recently, there have been several impressive humanoid biped robots, such as the work at Honda (which became the famous ASIMO robot) [7] and Boston Dynamics' ATLAS [8]. Much of this work has been built around maintaining the 'Centre of Pressure' within the base of support for most or all of the gait.

By contrast, monoped robots, by their nature, must use a flight phase to reposition their foot in order to make progress. Most designs also assume a point contact with the ground, and so must rely on dynamic stability even during stance phase. The Spring Loaded Inverted Pendulum (SLIP) [9] has become the standard model for investigating monoped locomotion in nature and robotics. It is important to note that gaits and strategies for this model are equally applicable to a biped running [10], which also alternates between single foot stance and flight phases. Much research has been carried out investigating and stabilising gaits for the SLIP model where each step is the same length (e.g. [11, 12, 13, 14, 15]). These gaits have the potential to greatly improve efficiency and speed, and allow traversal of terrain with large discontinuities.

In recent years, there has been an increased interest in developing controllers to allow a SLIP based hopping model to traverse more difficult terrain, such as limited, unevenly spaced footholds, by varying the step length. The primary obstacle to doing so has been that, despite the apparent simplicity of the SLIP model, there is no closed form solution for the stance phase dynamics. This makes it difficult to predict what input values (usually touchdown angle and/or force input during stance) will produce the lift-off conditions required to achieve a desired trajectory.

Motivated by the potential for a control strategy computationally simple enough to be easily operated in real time, several researchers have attempted to find analytical approximations for the stance phase dynamics which will provide closed form solutions. For example, Geyer et. al. derive an approximation assuming small angular sweep and small spring compression [16]. Yu et. al. present a more accurate approximation which is formulated using a truncated Taylor series expansion [17]. This again makes the assumption that the spring compression is a relatively small fraction of the leg length (equivalent to a stiff spring), but performs much better for large angles and in particular asymmetric cases where the magnitude of touchdown and lift-off angles are different. This approximation was used to create a controller and tested in simulation, and able to successfully vary the forward velocity of a SLIP model on level ground. Another analytical approximation [18] has been demonstrated for generating an apex return map which can then be used to control a hopper with the full simulated SLIP dynamics to traverse height varying terrain with gaps [19].

Degani et. al. applied a simple analytical model based on the assumption of an instantaneous stance phase [13]. This simplified model was used to find open-loop stable periodic trajectories which were then applied to a physical robot in reduced gravity. Another simple approximation, commonly used, is to ignore gravity during the stance phase, which is the source of the non-integrable term (e.g. [20]).

An early example of a controller able to traverse terrain with limited footholds is presented by Zeglin and Brown [21, 22]. Here an approximated analytical model based on the assumption of instantaneous impact, "with ad hoc but physically motivated corrections" [21] is used to approximate future hops. This model is used for a graph search to find suitable trajectories, creating a feedforward controller, and is combined with a feedback controller to keep the robot close to the computed trajectory. The graph search, being computationally expensive, is not performed on every hop, but only if the actual position becomes too far from the desired trajectory, in which case a new trajectory must be computed starting at the current position.

An alternative approach is, instead of trying to find an analytical solution, to use numerical integration to solve the stance dynamics. An example of such an approach, using Model Predictive Control (MPC), is presented by Rutschmann et. al. [23]. The critical choice of the number of future steps to simulate becomes a compromise between accuracy of the controller and time to execute because of the computationally intensive numerical simulation. Using a desktop computer, they find the MPC optimisation requires in the order of a few hundred milliseconds (carried out during the flight phase) to achieve foot placement within a centimetre of target.

On-line numerical simulations are used by Piovan and Byl [24] to choose actuator displacement during stance phase for a simulated SLIP hopper, in order to reach a desired lift-off state. Since the controller is to be operated in real time during the stance phase, a low order integration method is used which is fast to compute.

These previous works show a general trend towards greater complexity and computational cost to reduce approximation errors; the aim of this work is to demonstrate that a controller with a very simple form can achieve good performance, through tuning of the gains to match the conditions. As such, this paper contributes, based on a simplified analytical approach, an original controller capable of *(i)* performing agile hopping height control and *(ii)* automatically tuning controller gains to adapt to environmental changes. The analytical controller is demonstrated in simulation and forms the inspiration for an experimental implementation tested on a hydraulic monoped.

3. Adaptive Hopping Height controller

This section of the paper develops a method to control the period or hopping height of the monoped robot, which are directly related for parabolic flight. Firstly, an analytical derivation of the controller gains for a mass-spring-damper hopper with a linear actuator is presented. The proposed controller sets the actuator velocity as a function of ground contact and take-off velocities.

The analytic controller and its gains are then built within a feed-forward and self-adjusting (or adaptive) control architecture which ensures that perturbations, such as different ground properties, can be seamlessly taken into account. The adaptation of the controller gains is based on the error between consecutive hops, so it normally requires two hops to adjust to a sudden change.

3.1. Controller and gains derivation

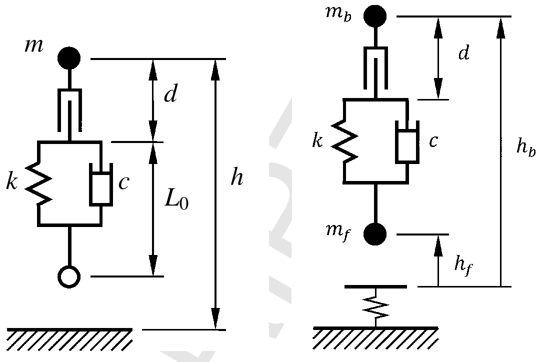
The hopping height controller developed here is based on the principle of adjusting the required energy to reach the next demand height. The kinetic energy at take-off is a square function of the take-off velocity. The simplified hopper shown in Fig. 2(a) will be analysed.

In this 1-dimensional model, the free length of the spring does not affect dynamics so can be set to zero to simplify the equations of motion (free length, $L_0 = 0$) giving stance and flight dynamics of:

$$\begin{aligned} &\text{for } h - d < 0\text{: } \ddot{h} = -g - \left(2\zeta\omega_n(\dot{h} - \dot{d}) + \omega_n^2(h - d)\right) \\ &\quad \text{otherwise: } \ddot{h} = -g \end{aligned} \tag{1}$$

where $d = d(t)$ is the actuator extension, $\omega_n = \sqrt{k/m}$ is the system's natural frequency, and $\zeta = c/2\sqrt{km}$ is the damping ratio.

Energy can be added to the system by moving the actuator during stance; in our case by extending it at a constant velocity q. The actuator is then retracted back to its starting position during the flight phase ready for the next hop. It is important to note that there is no requirement for constant speed actuation, only the quantification of the energy added into the system during the stance period, this could be done in various other ways, including the average actuation speed.



(a) (b)

Figure 2: Hopping monoped robots, (a) shows a 1D hopper with a mass-less foot and stiff ground, (b) a hopper with a mass on the foot and elastic ground.

In order to make the stance phase more amenable to mathematical analysis, gravity will be neglected, assuming $g \ll \ddot{h}$. The inaccuracy introduced by this assumption will later be mitigated through on-line modification of controller gains. With this assumption, Eq. 1 simplifies to:

$$\ddot{h} + 2\zeta\omega_n(\dot{h} - \dot{d}) + \omega_n^2(h - d) = 0 \tag{2}$$

The actuator will be extended at a constant velocity q throughout stance so:

- $d = qt$

Letting the impact with the ground occur at time $t = 0$ with a speed v_1 the initial conditions are:

- $h(0) = 0$
- $\dot{h}(0) = -v_1$

The above can be solved giving the motion during stance, $0 \le t < t_{lo}$:

$$h(t) = -\left(\frac{v_1 + q}{\omega_d}\right) e^{-\zeta\omega_n t} \sin\omega_d t + qt \tag{3}$$

$$\dot{h}(t) = -\left(\frac{v_1 + q}{\omega_d}\right) e^{-\zeta\omega_n t} \left(\omega_d \cos\omega_d t - \zeta\omega_n \sin\omega_d t\right) + q \tag{4}$$

Lift-off will occur at $t = t_{lo}$. While there is no closed form solution for t_{lo} in the real system, neglecting gravity allows a solution to be found as half the period of oscillation, $t_{lo} = \frac{\pi}{\omega_d}$. The lift-off speed, which will be equivalent to the touch-down speed of the next hop v_2, can be found from Eq. 4:

$$v_2 = \dot{h}(t_{lo}) = \dot{h}\left(\frac{\pi}{\omega_d}\right) \tag{5}$$

$$\Rightarrow\ v_2 = C_R(v_1 + q_1) + q_1 \tag{6}$$

where C_R is defined as:

$$C_R = \exp\left(\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}\right) \tag{7}$$

and q_1 is the input for this stance phase.

The simplifying assumptions (namely $g \ll \ddot{h}$ and $t_{lo} \approx \frac{\pi}{w_d}$) will create some error in the approximation given by equation 6. The extent of this error can be assessed by comparison to simulated results, which are not constrained by these assumptions, as shown in Figure 3. The error is reasonably small, and suggests this approximation will be useful for developing a control strategy.

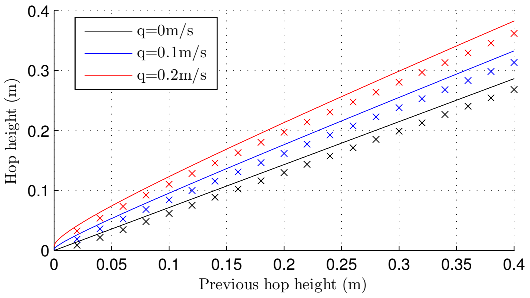


Figure 3: Comparison between the analytical approximation in equation 6 (line) and a numerical simulation with massless foot. This is the hop height as a function of the previous hop height for an example inputs of $q = 0, 0.1, 0.2$m/s.

The change in speed from touch-down to lift-off is $\Delta v_1 = v_2 - v_1$. By substituting $v_2 = v_1 + \Delta v_1$ into Eq. 6, rearranging for q_1 it can be seen that:

$$q_1 = \underbrace{K_L v_1}_{(i)} + \underbrace{K_\Delta \Delta v_1}_{(ii)} \tag{8}$$

$$K_L = \frac{1 - C_R}{1 + C_R} \tag{9}$$

$$K_\Delta = \frac{1}{1 + C_R} \tag{10}$$

A controller for the stance phase is provided by Eq. 8 relating the actuator velocity during stance, q_1, to impact velocity, v_1, and the desired change in velocity, Δv_1. As previously mentioned, take-off velocity and hopping height are related by $h = v_2^2/2g$. The control signal, q_1, is the sum of: (*i*) a term proportional to the impact velocity and thus accounting for linear losses and, (*ii*) a term proportional to the change in height required to reach the next hopping height. The term (*i*) could be seen as a steady-state height controller, whereas the second term (*ii*) deals with the dynamic changes.

Generalising, the actuator extension velocity needed during the stance phase of the n^{th} hop is q_n. The desired change in speed between touch-down and lift-off is labelled Δv_n. This can also be written in terms of the desired touch-down velocity v_{n+1} of the next, $(n+1)^{\text{th}}$, hop:

$$q_n = K_1 v_n + K_2(v_{n+1} - v_n) \tag{11}$$

The gains in Eq. 11 can be initially set from Eq. 9 and Eq. 10 so $K_1 = K_L$ and $K_2 = K_\Delta$. However, the error caused by the analytical approximation will cause K_L and K_Δ to differ from the optimum values. To mitigate this problem, the gain values will be tuned using the results of previous hops. In this way the controller becomes more robust to errors in the modelling, including the modelled effects of ground compliance.

Motion during the flight phase is parabolic which means there are simple relationships between the touch-down speed v_n, flight time T_{fn} and hopping height h_n:

$$v_n = \frac{1}{2} g T_{fn} \tag{12}$$

$$v_n = \sqrt{2 g h_n} \tag{13}$$

This means that the control logic of Eq. 11 can similarly be written in terms of T_{fn} and $\sqrt{h_n}$:

$$q_n = K_a \sqrt{h_n} + K_b \left(\sqrt{h_{n+1}} - \sqrt{h_n}\right) \tag{14}$$

$$q_n = K_\alpha T_{fn} + K_\beta (T_{f(n+1)} - T_{fn}) \tag{15}$$

The gains K_a and K_b or K_α and K_β take analagous roles to K_L and K_Δ above.

It should be noted that q_n can take a negative value. This results in the leg retracting to remove energy from the system to reduce lift-off speed more than would be otherwise possible with damping alone.

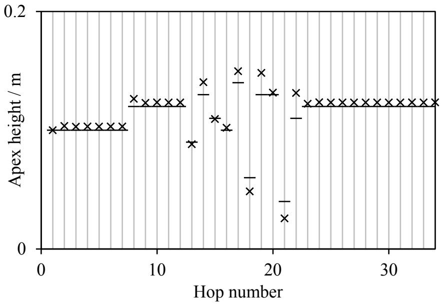


Figure 4: Height tracking performance for dynamic controller $q = K_1\sqrt{h_0} + K_2(\sqrt{h_d} - \sqrt{h_0})$. Dashes show demand height and crosses actual height. Gains are fixed at $K_1 = 0.70$, $K_2 = 2.56$.

Simulation results for hopping height control using Eq. 14 are plotted in Fig. 4. The gains have been set analytically such that: $K_a = \sqrt{2g}K_L$ and $K_b = \sqrt{2g}K_\Delta$.

This feedback controller offers good tracking capability and errors can be further reduced by tuning the control gains, as shown in the adaptive self-tuning controller, which uses touch-down speeds and actuator actions from previous hops to improve controller gains as explained in the next section.

3.2. Adaptive gain hopping height controller

Writing Eq. 11 for the previous two hops in matrix, it is expected that:

$$\underbrace{\begin{bmatrix} v_{n-1} & \Delta v_{n-1} \\ v_{n-2} & \Delta v_{n-2} \end{bmatrix}}_{\mathbf{V}} \underbrace{\begin{pmatrix} K_1 \\ K_2 \end{pmatrix}}_{\mathbf{k}} = \underbrace{\begin{pmatrix} q_n \\ q_{n-1} \end{pmatrix}}_{\mathbf{q}} \tag{16}$$

where $\Delta v_n = v_{n+1} - v_n$. In (16), the required inputs ($\mathbf{q}$) are found as a function of the desired liftoff/touchdown speeds ($\mathbf{v}$), assuming the correct gains. However, if the measured speeds are instead used (denoted by $\bar{\mathbf{V}}$), it is possible to calculate the gains which would have been required to successfully predict these measured outcomes:

$$\mathbf{k} = \bar{\mathbf{V}}^{-1}\mathbf{q} \tag{17}$$

This allows a new set of gains to be found, used in (11) for the next hop.

Solving for the gains $\mathbf{k}$ requires $|\mathbf{V}| \neq 0$. This can be computed:

$$|\bar{\mathbf{V}}| = \bar{v}_{n-1}\bar{v}_{n-2}\underbrace{\left(\frac{\Delta\bar{v}_{n-2}}{\bar{v}_{n-2}} - \frac{\Delta\bar{v}_{n-1}}{\bar{v}_{n-1}}\right)}_{\rho} \tag{18}$$

In certain circumstances, the previous two hops may not contain enough information to infer the values of K_1 and K_2, and the solution will be ill-conditioned ($|\mathbf{V}| \approx 0$). This might occur, for instance, if $v_{n-3} = v_{n-2} = v_{n-1}$, in which case there would be no information to find K_2. In order to avoid this a threshold condition, $|\rho| > 0.01$, is checked. If below the threshold then the gains are left unchanged, with the consequence of removing the adaptive nature of the controller. In simulation it was also found that the controller can fail if $K_2 \approx 0$. A simple solution is not to update the value of K_2 where this would be the case.

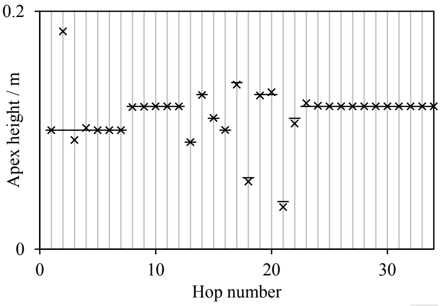


Figure 5: Simulated height tracking performance for adaptive dynamic controller $q = K_1\sqrt{h_0} + K_2(\sqrt{h_d} - \sqrt{h_0})$. Dashes show demand height and crosses actual height. Initially gain values are chosen to be $K_1 = 1.5$, $K_2 = 3.0$ (Note that these are very different from those analytically determined to demonstrate the adaptability).

The results of a simulation where the gains were self-tuned in this way are plotted in Fig. 5. Initial values of controller gains are selected to be poor. This results in hops 2 and 3 with large errors. Thereafter, the controller has enough information from previous hops to keep the gains correctly tuned. The self-tuning results in better performance than the analytically derived gains in Fig. 4 as it corrects for the simplifying assumptions.

3.3. *Adapting to changing ground compliance*

Self-tuning control gains are useful, for example, when running over ground with changing properties. This can be demonstrated by simulations using a compliant ground model, as shown in Fig. 2(b). This is similar to the simple model used previously, Fig. 2(a), but includes a foot mass m_f in addition to the body mass m_b and a non-rigid ground; all parameters are defined in Table 1.

Table 1: Fig. 2(b) model simulation parameters.

	Parameter	Value	
m_b	Body mass	10	kg
m_f	Foot mass	1	kg
k	Spring stiffness	8000	N m^{-1}
c	Damping coefficient	30	N s m^{-1}
$h_b(t = 0)$	Initial body height	0.15	m
$h_f(t = 0)$	Initial foot height	0.15	m
	High stiffness model:		
F_0	Reference spring force	10000	N
δ_0	Reference spring displacement	0.01	m
c_{gr}	Damping coefficient	10	N s m^{-1}
	Low stiffness model:		
F_0	Reference spring force	100	N
δ_0	Reference spring displacement	0.01	m
c_{gr}	Damping coefficient	10	N s m^{-1}

The ground is a modelled as a non-linear spring-damper, giving a ground reaction force, F_{gr}, of:

$$F_{gr} = F_0\left(\frac{-h_f}{\delta_0}\right)^{\frac{3}{2}} + c_{\mathrm{gr}}(-\dot{h}_f) \tag{19}$$

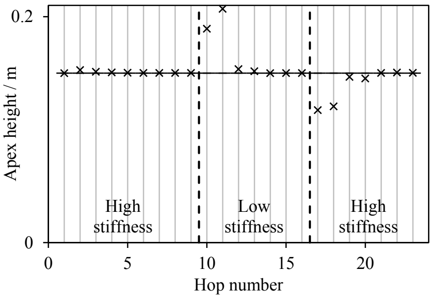


Figure 6: Results for model with compliant ground (Fig. 2(b)) with changing ground properties. Ground is soft after hop 9 and returns to hard after hop 16. Height demand is kept constant and adaptive controller is used.

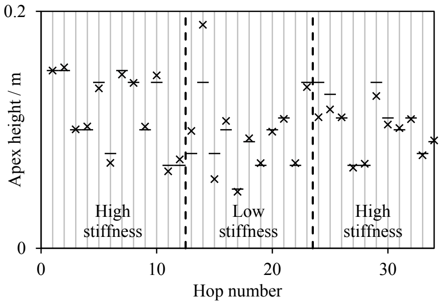


Figure 7: Results for model with compliant ground (Fig. 2(b)), changing to soft ground after hop 12 and returning to hard ground after hop 23. Randomly varying height demand with adaptive controller.

Results for two simulations are plotted in Fig. 6 and Fig. 7. In the first simulation, the demand hopping height is kept constant. The ground properties are changed after hop 9 and 16. Within a couple of hops, the gains are tuned to the new ground.

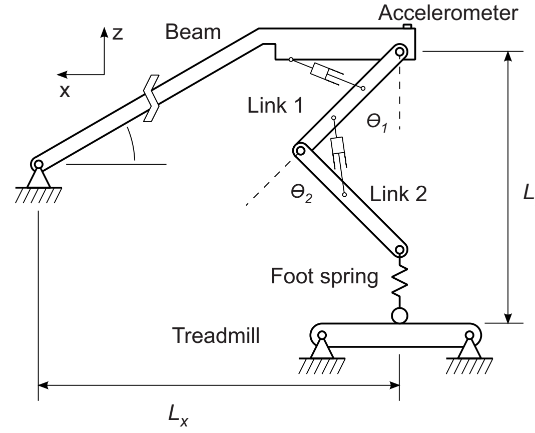


Figure 8: Schematic of experimental rig: emulates a two-link hydraulically actuated springy leg constrained to hop vertically. Degrees of freedom include the beam angle, hip angle, knee angle, spring displacement and treadmill motion. The angles θ_1 and θ_2 are relative angles. The beam angle is absolute with respect to the ground which can be used to get the full state for the robot.

The second simulation is similar but presents a more challenging height demand. It is randomly varied between 0.05 m and 0.15 m. With a variable demand, the controller can still adapt to changing ground properties within a few hops illustrating the capabilities of the adaptive controller.

4. Experimental test-bed and hopping controller

This section describes the robotic apparatus used to validate the controllers proposed and illustrates how position (during flight phase) and velocity (during stance phase) control are achieved. A schematic drawing of the experimental hopping leg (Fig. 1(a)) is shown in Fig. 8. This is a leg from the HyQ robot [27] that has been constrained to hop approximately vertically on a treadmill using a pivoting beam. The leg consists of two rigid links and a compliant foot. The leg is actuated by two hydraulic actuators as shown. Encoders are used to measure relative joint angles θ_1, θ_2 and a string potentiometer to measure the beam angle with respect to ground. This information is sufficient to calculate the state required for the controller, *i.e.* body and leg position and velocity. Additionally an accelerometer is positioned above the hip. Key parameters for this experimental setup have been listed in Table 2.

Table 2: Experimental rig parameters

Parameter	Value	
Link 1 length, Hip-knee	0.35	m
Link 1 mass	1.772	kg
Link 2 length, Knee-foot	0.33	m
Link 2 mass	0.808	kg
Aluminium box beam width	38.1	mm
Aluminium box beam thickness	3.2	mm
Total mass	18	kg
Approximate foot stiffness	10000	$\mathrm{N\,m^{-1}}$
Hip-beam pivot distance	2	m
Hydraulic supply pressure	160	bar
Actuator stroke	80	mm
Actuator bore	16	mm
Servovalve rated flow	2.5	L/min

4.1. Hopping controller

This section presents the controller used for precise and agile height control of the hydraulic robotic leg. The controller is based on the analytical developments presented in Section 3 although with some implementation differences to account for system non-linearity and to ensure the stability of the adaptive controller gains.

The system is programmed with a two-state controller:

- During the flight phase, position control is used to return the foot to a predefined home position.
- During the stance phase, the actuator pushes downwards with a demand velocity calculated based on the height demand for the next hop and the impact velocity.

Switching between states is triggered as follows:

- From flight to stance a threshold crossing on the accelerometer.
- From stance to flight automatically when the stance phase duration has exceeded the estimated stance period, (approximately 0.14 s in our system).

4.2. Flight phase controller

During the flight phase, the leg is essentially off the ground which means that the foot moves freely in the air. The lack of any external force, the speed of the motion and the compliance of hydraulic systems means that during the point to point motion the leg will oscillate.

In order to remove these unwanted oscillations external damping will be imposed using the concept of Closed Loop Signal Shaping, CLSS [29]. In short, CLSS uses a signal shaper, such as Zero Vibration and Derivative (ZVD) [30], in the forward path of conventional feedback controller. This architecture is shown in Fig. 9.

For this experiments, the articulated leg was lifted off the ground. The initial displacements of the upper actuator cylinders were selected such that the foot was vertically below the hip joint and the leg was not close to any kinematic limit, the lower actuator was fixed at a stretched out position to maximise the effect of the leg inertia.

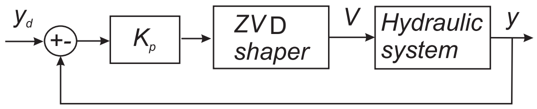


Figure 9: The diagram illustrates the ZVD input shaper block in the forward controller path of a position controller with proportional gain K_p. The demand signal is y_d, V stands for the control voltage and y the measured response.

Closed-loop proportional controllers and CLSS were implemented on the upper hip actuator. The position response of the actuator to step changes in demand position of 6.25% of stroke is shown in Fig. 10, which is representative of a leg repositioning action.

A step position input was given to the upper actuator demand position while keeping the demand to the lower one steady.

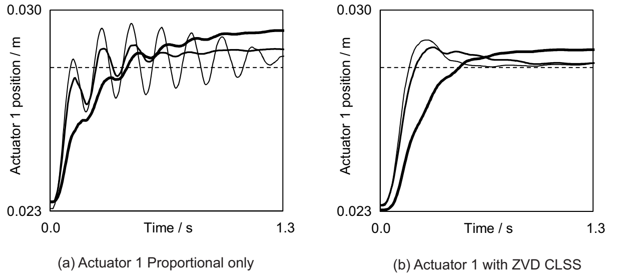


Figure 10: Experimental results: Response of actuators to step change in demand with and without ZVD in closed loop.

This is done for three values for the proportional gain K_P, resulting in the 12 time series results plotted in Fig. 10. As it can be seen in Fig. 10(b) the CLSS delivers a much improved response eliminating the majority of the oscillations. The benefit of this method was not observed on the lower-leg actuator due to the smaller inertia, thus a proportional controller is sufficient for this application.

4.3. Stance phase controller

The stance controller proposed in Section 3 computes the demand actuator speed as calculated equivalently by Equations 8, 11, 14 or 15. As previously described, the controller has two terms. The first one accounts for steady-state hopping *i.e.* sufficient energy is added to maintain a constant hopping height. The second term, accounts for the differential energy required to make a sudden change in the demanded hopping height or hop period.

The block diagram for the experimentally implemented stance controller is shown in Fig. 11. Each of the parts for this controller are described in the following sections.

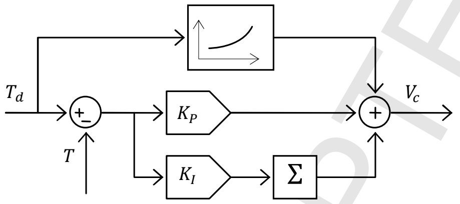


Figure 11: Block diagram of the proposed hopping height controller. Where K_P and K_I are controller gains, V_c is the valve control signal, T_d and T the demand and measured hopping period respectively. The block in the top loop represents the steady state look up function. Immediately following the K_I gain is a summation block.

4.3.1. Steady state hopping height controller

In the simulations presented in section 3, the actuator velocity was used as a control input. On the rig used here the actuator velocity is controlled through a hydraulic servo valve which introduces non-linear dynamics meaning the control input does not directly correspond to the linear extension velocity of the foot. Additional significant non-linear dynamics are introduced due to different kinematics (two links), the distribution of mass, gravity, fluid dynamics, contact dynamics, friction, etc. The assumption made by term (*i*) in Equation 8 of a linear relationship between the control input and the steady-state hopping lift-off velocity or period therefore somewhat breaks down. A simple solution is to substitute for term (*i*) an experimentally derived look-up function. In the block diagram in Fig. 11 this is the feed-forward block at the top.

A set of experiments were carried out to plot the relationship between the control input V_c, which maps to valve control voltages extending the foot during stance, and steady state hop period. Results are shown in Figure 12 showing the non-linear relationship. It can be seen that ground properties affect the steady-state hopping period (and height).

A cubic equation was fitted to the hard ground data (crosses) giving a look-up function giving the control action V_c as a function of a desired steady state hopping period T_{ss}:

$$f_{ss}(T) = 437.12T^3 - 517.4T^2 + 210.08T - 26.36 \qquad (20)$$

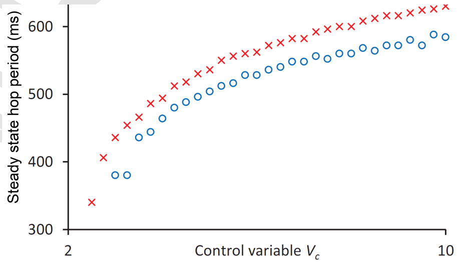


Figure 12: Relationship between control variable V_c and steady-state hopping period time on hard (red crosses) and soft (blue circles) ground.

4.3.2. Varying hopping height controller

The controller in Eq. 8 deals with sudden changes in hop period demand using the second term (*ii*). The gain for term (*ii*) in Eq. 8 or K_β in Eq. 15 is equivalent to the gain K_P in the block diagram (Fig. 11). This is a proportional gain acting on the difference between the current hop period and the new desired value for the next hop.

4.3.3. Adapting to varying ground properties

The look-up function used in our height controller was fitted to experimental data for hopping on hard ground. It can be seen in Fig. 12 that changes in ground properties affect the steady state hopping height for a given control input. A steady state error results when hopping on ground properties other than that from which the look-up function is derived. We found that the addition of a small integral gain K_I could deal with this steady state error allowing the hopping machine to adapt to different ground properties over a number of hops.

4.3.4. Adapting controller gains

The analytical controller described in Section 3 can adapt the values of its gain given two consecutive steps by measuring the errors. It was found in experiment that adapting the gains over two hops was too unstable so a more cautious approach was taken to adapt the controller gains. On each hop where the demand had changed the gain K_P was incrementally updated using the error in the previous hop by applying the following formula:

$$K_{P(n+1)} = K_{P(n)} + \delta(T_{d(n-1)} - T_{(n-1)})\text{sign}(T_{d(n-1)} - T_{(n-1)}) \quad (21)$$

Where the value of δ is small enough that it makes an incremental correction to K_P.

5. Results on hopping height control

The Section presents the experimental results achieved with the control method described in Section IV. Initially the experiments demonstrate hopping over stationary ground and then the effect of hopping on a ground with horizontal velocity (running treadmill).

5.1. Hopping on stationary ground

Figure 13 shows the results of the monoped robot (Fig. 1) hopping over random step changes in height demand. The results show that the controller is capable of reaching a step-change in height demand within a single hop. Moreover, the different ground properties hard (concrete) and soft (padded) grounds do not show any significant effect.

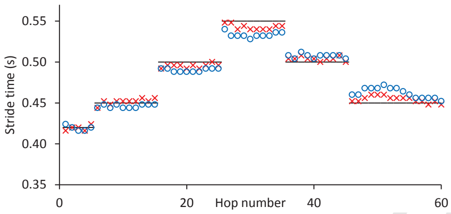


Figure 13: Agile monoped hopping control on different grounds. Hard ground (crosses); soft ground (circles).

A more challenging demand is shown in Fig. 14. Here random hopping periods are demanded in the range 0.38 s to 0.57 s. This corresponds to hopping heights from 0.077 m to 0.237 m. It can be seen that the large shortfalls on hops 15, 18 and 31 occur because the control signal had reached saturation. This may be avoided by limiting demanded hopping periods to within the performance envelope of the robot. Additionally, it should be noted that some hops require a negative value for the control variable V_c. This means that the leg has to actively flex to absorb more energy than damping alone would accomplish.

5.2. Hopping with horizontal ground speed

To test the effects of running on hopping control while neglecting considerations of balance, experiments were carried out with the treadmill (Fig. 8) in motion.

When running, it is desirable to begin sweeping the foot backwards before touch-down with the ground. This reduces the severity of the impact with the ground because the foot's relative horizontal motion to the ground is removed. In order to begin sweeping the foot before touch-down the next touch-down time $t_{\text{td}(n+1)}$ has to be anticipated. Additionally, the foot needs to be positioned slightly ahead of the desired foot position on touch-down so that as it sweeps backwards in the air it reaches the desired foot position upon impact.

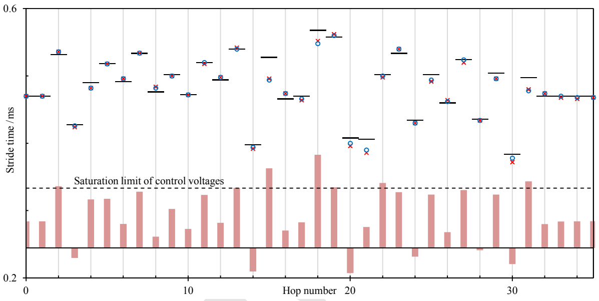


Figure 14: Results for the proposed controller with random hopping demands. Stride times range from 0.38 s to 0.58 s which corresponds to hopping heights of 0.08 m and 0.24 m respectively. Controller was auto tuned before beginning random demand input (horizontal lines). The same experiment was performed first on hard (crosses) ground then on soft ground (circles). The control variable V_c has also been plotted for the case of hard ground.

Figure 15 shows the results of the controller when applied to a hopping while a running speed of 0.37 m s^{-1}. As it can be observed the treadmill velocity influences the accuracy of the hoping height controller. This is expected as the controller is designed assuming decoupling between height and forward velocity, which is not true when running at relatively high speeds, mainly due to the fact the touch down and lift-off leg angles are considerably different than when hopping on the spot.

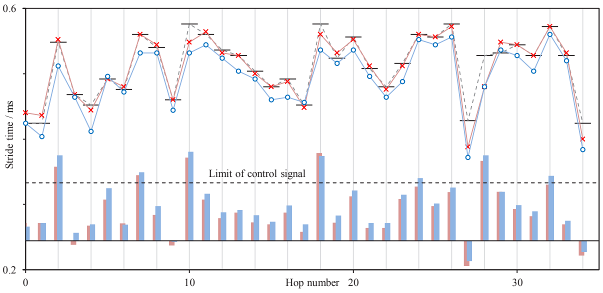


Figure 15: Experimental results using the FF+PI control on treadmill at speeds 0 (red crosses) and 0.37 m s^{-1} (blue circles).

5.3. Hopping at different ground speeds

As shown in Fig. 15 the treadmill motion results in hop periods consistently lower than desired. A possible explanation for this is that while stationary, leg extension forces are directed vertically whereas running requires energy to be expended in swinging the leg, accelerating and decelerating the foot horizontally on each step. To compensate for this extra energy input required, a simple linear factor is added to the control signal. Figure 16 illustrates the linear relation required.

Experimental results of adding this speed based compensation are plotted in Fig. 17 show a reduced error when compared to errors assuming no running speed.

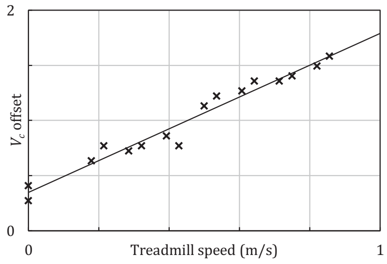


Figure 16: Corrective offset required to V_c at different running speeds to achieve 0.472 s steady state hopping period.

Larger changes in the demand result in greater error due to actuator saturation. Logically, saturation occurs more frequently at high speeds making the errors on extreme hops larger. It should be noted that for all of these results, the same feed-forward function and PI gains were used.

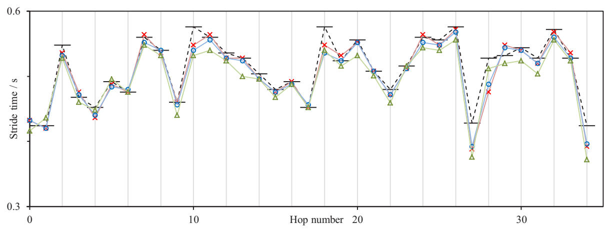


Figure 17: Experimental results using the extended FF+PI control at different running speeds: red crosses, blue circles, green triangles are 0, 0.37, 0.71 ms^{-1} respectively.

6. Conclusion

The paper has described an effective and computationally lightweight controller capable of adjusting the hopping height of a hydraulic monoped robot in a single hop. The method does not require accurate models of the robot or the environment, but relies on energy relationships that remain constant but with changing parameters to account for variations in the environment or the robot. The simulated and experimental results show a good performance of the controller, the main source of error seems to be the saturation of the actuators which is a physical and unavoidable limitation.

The main contribution of the paper is the development of an analytical understanding of why the controller is capable of adjusting the height of a hopping robot in a single step and how the errors in the achieved height can be used to tune automatically the controller gains. Due to experimental phenomena the fast gain adaptability could not be shown to work, but instead a slower integral loop had to be used. Further work on this important area is required.

The control laws developed here are very different to alternative approaches which employ actuators and sensors allowing for high speed force loops for model-based controllers. The work here provides an example of how, with limited sensing and computation, it is still possible to achieve agile performance over different terrains. This can be done, given favourable passive dynamics, by stacking laws to excite, maintain and perturb those dynamics.

The overall approach to control hopping taken here has been:

1. Use a machine with a passive hopping motion. Here this is due to a springy foot.
2. Formulate a variable to impart a vertical impulse, V_c, to be controlled discretely once per hop. This can then be used to form a discrete hop control loop executed once per hop.
3. Generate a look-up table/function for open loop, steady state control of hop periods.
4. Improve steady state and dynamic performance by closing the loop with a simple proportional and integral action.

This approach could be applied to machines with different mechanical designs. For example:

- Pneumatic or electrical actuation might be used instead of hydraulics.
- It is not necessary that the leg is articulated. It could equally well be telescopic or some other design.
- Impact with the ground was detected as a spike in the force sensor at the knee but different sensors placed elsewhere would serve equally well.
- A passive hopping motion is required but this does not have to be provided by a springy foot. Indeed, elasticity might be emulated by the actuators. With real elasticity however, energy is stored and released from one hop to the next. This means that for steady state locomotion, actuators only need to make up energy losses between hops. And to change hop size, actuators need to make up (or dissipate) the energy difference. Actuators typically will not store energy so emulating elasticity would be inefficient. It would also require much more powerful actuators capable of responding to impact forces.

Balance was not a consideration in this paper because the machine's body orientation was constrained but, as shown by Raibert et al [5], height control can be considered decoupled from body orientation and horizontal velocity. Also, state estimation of a hopping robot is very challenging and not discussed in this paper, but this is an area of critical importance for further developing this technolgy.

References

[1] T. Bretl and S. Lall, "Testing static equilibrium for legged robots," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 794–807, 2008.
[2] Y. Yesilevskiy, W. Xi, and C. D. Remy, "A comparison of series and parallel elasticity in a monoped hopper," pp. 1036–1041, 2015.
[3] M. Reis and F. Iida, "An energy-efficient hopping robot based on free vibration of a curved beam," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 1, pp. 300–311, 2014.
[4] A. Sayyad, B. Seth, and P. Seshu, "Single-legged hopping robotics researchA review," *Robotica*, vol. 25, no. 05, pp. 587–613, 2007. [Online]. Available: http://www.journals.cambridge.org/abstract_S0263574707003487

[5] M. H. Raibert, *Legged Robots That Balance*. MIT Press, 1986.

[6] J. K. Hodgins and M. H. Raibert, "Adjusting Step Length for Rough Terrain Locomotion," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 289–298, 1991.

[7] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "Honda humanoid robots development." in *IEEE Conference on Robotics and Automation*, 1998, pp. 1321–1326.

[8] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, "Optimization based full body control for the atlas robot," *IEEE-RAS International Conference on Humanoid Robots*, pp. 120–127, 2014.

[9] R. Blickhan, "The spring-mass model for running and hopping," pp. 1217–1227, 1989.

[10] Rummel, Juergen, and Andre Seyfarth. "Stable running with segmented legs" The International Journal of Robotics Research 27.8 (2008): 919-934.

[11] H. Geyer, R. Blickhan, and A. Seyfarth, "Natural dynamics of spring-like running: Emergence of selfstability," *International Conference on Climbing and Walking Robots (CLAWAR)*, pp. 1–6, 2002.

[12] S. Hyon and T. Emura, "Quasi-periodic gaits of passive one-legged hopper," *Intelligent Robots and Systems, 2002. IEEE . . .*, no. October, pp. 2625–2630, 2002. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1041666

[13] A. Degani, A. W. Long, S. Feng, H. Benjamin Brown, R. D. Gregg, H. Choset, M. T. Mason, and K. M. Lynch, "Design and open-loop control of the parkourbot, a dynamic climbing robot," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 705–718, 2014.

[14] N. Shemer and A. Degani, "Analytical control parameters of the swing leg retraction method using an instantaneous SLIP model," in *IEEE International Conference on Intelligent Robots and Systems*, no. Iros, 2014, pp. 4065–4070.

[15] C. François and C. Samson, "A New Approach to the Control of the Planar One-Legged Hopper," *The International Journal of Robotics Research*, vol. 17, no. 11, pp. 1150–1166, 1998.

[16] 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.

[17] H. Yu, M. Li, P. Wang, and H. Cai, "Approximate Perturbation Stance Map of the SLIP Runner and Application to Locomotion Control," *Journal of Bionic Engineering*, vol. 9, no. 4, pp. 411–422, 2012.

[18] U. Saranli, O. . Arslan, M. M. Ankarali, and O. . Morgül, "Approximate analytic solutions to non-symmetric stance trajectories of the passive Spring-Loaded Inverted Pendulum with damping," *Nonlinear Dynamics*, vol. 62, no. 4, pp. 729–742, 2010.

[19] O. Arslan and U. Saranli, "Reactive Planning and Control of Planar Spring-Mass Running on Rough Terrain," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 567–579, 2012. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6112244

[20] U. Saranli, W. J. Schwind, and D. E. Koditschek, "Toward the control of a multi-jointed, monoped runner," *Proceedings of the 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146*, no. May, pp. 2676–2682, 1998. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=680750

[21] G. Zeglin and B. Brown, "Control of a bow leg hopping robot," *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 1, no. May, pp. 793–798, 1998.

[22] G. Brown, Ben Zeglin, "The Bow Leg Hopping Robot," *Proceedings - IEEE International Conference on Robotics and Automation*, no. May, 1998.

[23] M. Rutschmann, B. Satzinger, M. Byl, and K. Byl, "Nonlinear model predictive control for rough-terrain robot hopping," in *IEEE International Conference on Intelligent Robots and Systems*, 2012, pp. 1859–1864.

[24] G. Piovan and K. Byl, "Reachability-based Control for the Active SLIP Model," *International Journal of Robotics Research*, vol. 34 (3), pp. 270–287, 2015.

[25] ——, "Enforced symmetry of the stance phase for the Spring-Loaded Inverted Pendulum," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2012, pp. 1908–1914.

[26] ——, "Two-element control for the active SLIP model," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2013, pp. 5656–5662.

[27] C. Semini, N. G. Tsagarakis, E. Guglielmino, M. Focchi, F. Cannella, and D. G. Caldwell, "Design of HyQ - a hydraulically and electrically actuated quadruped robot," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 225, no. 6, pp. 831–849, 2011.

[28] K. W. Wait and M. Goldfarb, "A Pneumatically Actuated Quadrupedal Walking Robot," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 1, pp. 339–347, 2014.

[29] J. Bhatti, A. Plummer, P. Iravani, and M.N Sahinkaya, "Implementation of closed loop signal shaping in a hydraulic system", Mechatronics 2012: The 13th Mechatronics Forum International Conference, 2012

[30] J. Vaughan, Y. Aika, and W. Singhose. "Comparison of robust input shapers." *Journal of Sound and Vibration* 315, no. 4, pp.797-815, 2008.

Pejman Iravani: Is a robotics lecturer and heads the robotics lab at the University of Bath. His research interests are in computer vision, machine learning, and legged locomotion. He received his PhD in 2005 on a machine learning architecture for collaborative robots.

Andrew Plummer: Professor Plummer is Director of the Centre for Power Transmission and Motion Control. He has a variety of research interests in the field of motion and force control, including inverse-model based control of electrohydraulic servo systems, control of parallel kinematic mechanisms, hybrid hydraulic/piezoelectric actuation, and active vehicle control.

Jawaad Bhatti: Is a research engineer with the Blacthford Group. His PhD thesis was on the control of hopping height of a monoped hydraulic robot.

Matthew Hale: Is a PhD student at the department of mechanical Engineering, University of Bath. His research interests are on dynamic hopping of small monoped robots.

Necip Sahinkaya: is a Professor of Mechanical Engineering. He received his doctorate in 1979 (University of Sussex). Following periods at the Universities of Sussex, Strathclyde and Bath, he moved to industry in 1988. He re-joined the University of Bath in 2000 as a Reader in Mechanical Engineering, and then moved to Kingston University London in 2013 as a Professor of Mechanical Engineering. He is Fellow of IMechE and Editor-in-Chief for the IMechE Proceedings Part I.