Vertical Control for a Mechanical Model of the One-Legged Hopping Machine

Joseph Prosser and Moshe Kam
ECE Department, Drexel University, Philadelphia, PA, 19104

Abstract

A vertical control algorithm is proposed for a one-legged hopping machine. It differs from previous control algorithms in three areas: (i) it uses a near-inverse of the machine's dynamics; (ii) it uses discrete-time output feedback, namely the hopping heights of previous cycles (rather than continuous full state feedback); and (iii) it employs a more realistic model of the hopping machine mechanical components. Numerical simulations are used to assess the ability of the controlled machine's to track a piecewise constant height trajectory. Specifically, we study the sensitivity of the design to deviations in the hopping machine's parameters.

I. Introduction

The one-legged hopping machine [2,5-10] consists of a body and a single springy leg which articulates with respect to the body. The controller dynamically balances the body (i.e. keeps the machine from falling over) and at the same time maintains a particular hopping height and horizontal hopping velocity. Raibert [5] decomposed the control problem into *hopping height control* and *horizontal velocity control*, and demonstrated that separate designs of the two controllers may be robust enough to allow decoupled operation. Raibert has also proposed a tabular control algorithm [9] which uses a large table of pre-computed data and calculates the control signal by interpolation. Raibert's algorithms were applied experimentally to 2-D and 3-D physical prototypes [6,8]. Following Raibert, Sznaier and Damborg [10] modified the vertical actuation scheme and proposed an *adaptive* algorithm for horizontal velocity control. For learning control of hopping height, Helferty and his coworkers [2] have used multiperceptrons which were trained to minimize hopping height errors. Additional approaches to the problem are described by Raibert [7], Todd [11] and Cox and Wilfrong [1].

In this paper, we propose a mechanical design of the hopping machine, and utilize a model which takes into account the actuator dynamics, including physical constraints on the control signal. The hopping machine model is for vertical motion, as it is foreseen that the horizontal control problem can be approached separately. The hopping height control algorithm is based on off-line synthesis of an inverse model for the plant, using numerical simulations of the dynamics. This is an approximate functional relationship between the hopping heights of previous hopping cycles, the height of the next hopping cycleThe 'near-inverse' is used to synthesize a control signal. This scheme exhibits a relatively low sensitivity of the tracking error to parameters of the physical system. Performance of the controlled machine is assessed in numerical simulations, using a piece-wise constant reference signal for the hopping height to track.

The mechanical model of the system is presented in section II. In section III, the control algorithm is described, and in section IV, the system dynamics are simulated numerically. Height tracking performance is assessed under "worst-case" conditions with several mechanical system parameters simultaneously perturbed from their nominal values.

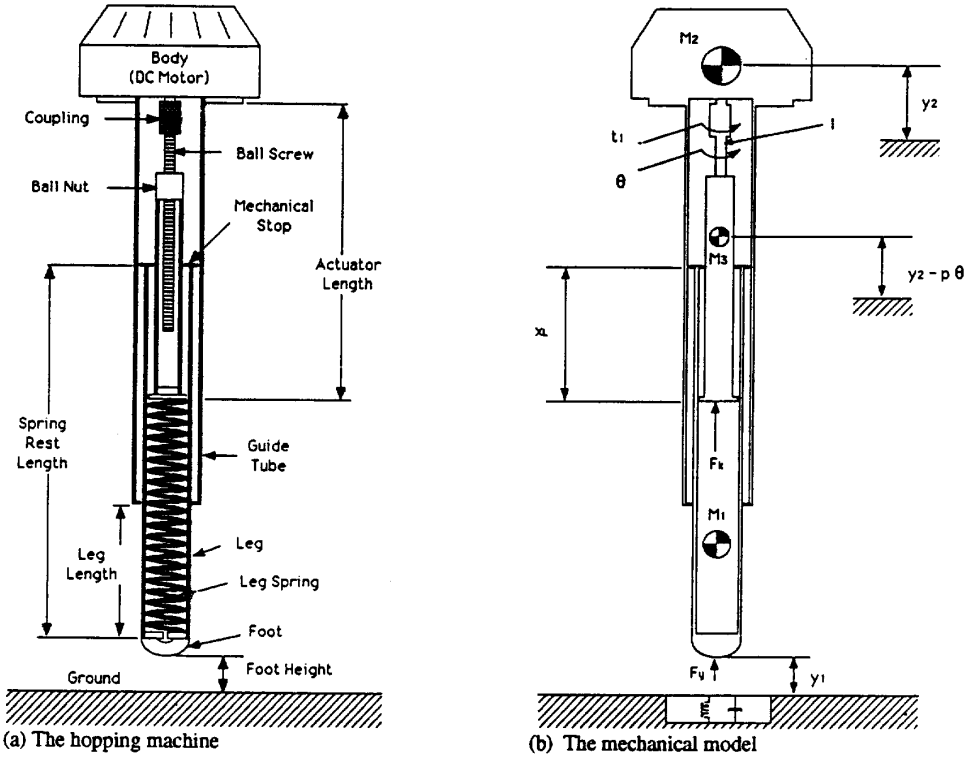
Acknowledgment: This study was supported by the National Science Foundation through grant no. ECS 9057587.

II. The Mechanical System

The one-legged hopping machine consists of a body, a springy leg, and an actuator (figure 1a). The body constitutes most of the mass of the machine, and is dynamically balanced atop the leg. It carries the payload and various sensors, and forms a structure from which forces and torques may be applied to the leg in order to maintain the machine's balance. In general, the leg changes both its length and its orientation with respect to the body during locomotion. In this paper, however, only vertical motion is considered and no reorientation of the leg is allowed.

The body includes an actuator which acts to lengthen and shorten the leg. The actuator consists of a DC motor whose shaft is coupled to a ball screw. As the shaft turns, the actuator changes length. The direction and speed of shaft rotation is controlled with a voltage applied to the motor.

CH3000-7/92/0000-0136 $1.00©1992 IEEE



(a) The hopping machine (b) The mechanical model

Figure 1

The leg consists of a foot, a spring (the leg's energy storage element), and a mechanical stop which limits the leg's length (this enables the leg to be lifted from the ground). The rest length of the leg spring is chosen so that when the spring is relaxed, both the leg spring and the mechanical stop just touch the actuator. With this arrangement, any change in actuator length while the leg spring is compressed will change its stored energy. This ability to manipulate the system's energy enables the control of hopping height; the amount of energy that is added to the leg spring during each hop can be changed.

A resonant interaction between the body mass and the leg spring yield a cyclic hopping motion which alternates between phases of *flight* (when the foot is in the air) and *support* (when the foot is on the ground). The hopping cycle is characterized by the following events:

top: The moment when the center of gravity of the hopping machine is highest in the hopping cycle.

touchdown: The instant the foot strikes the ground.

bottom: The moment when the center of gravity of the hopping machine is lowest in the hopping cycle.

liftoff: The instant the foot leaves the ground.

At *touchdown*, the kinetic energy of the leg is dissipated in the ground, but the body continues to move downward as the leg spring begins to compress. During *support*, the actuator is lengthened, and when the body begins to move upward, the leg spring extends until the actuator strikes the mechanical stop (at *liftoff*). At this moment, a fraction of the body's kinetic energy is lost. The height at *top* (the hopping height) is maintained by adding an amount of energy to the leg spring during *support*, which is equal to the amount of energy that is lost over the entire hopping cycle.

The body, leg, and actuator are represented in the

mechanical model of figure 1b. The body and leg are represented as point masses, with the body mass including that of the motor, guide tube, coupling, and ball screw. The mechanical stop, ground, and leg spring are modeled as springs with damping, with the stiffness of the mechanical stop and ground springs set to a very high value, and the damping of the leg spring including the friction between the leg and actuator. The actuator is modeled as a point mass constrained to move along the shaft of the ball screw.

The following variables describe the state of the hopping machine (all are functions of time):

y_1 foot height (m)
$\dot{y}_1$ foot velocity (m/sec)
y_2 body height (m)
$\dot{y}_2$ body velocity (m/sec)
θ ball screw angle (rad)
$\dot{\theta}$ ball screw angular velocity (rad/sec)
i motor armature current (amps)

The only control input to the system is

v_m voltage applied to the motor (volts).

Hopping height is measured as the height of the center of gravity of the entire machine, at the time when the sign of its derivative changes from positive to negative (*top*) (In practice, the hopping height may be calculated from measurements of the time duration of *flight* and the actuator length at *liftoff*). The foot height is measured from the surface of the ground. The body height and ball screw angle are defined to be zero when (i) the foot is just touching the ground; (ii) the leg spring is at its rest length; and (iii) the actuator is at some "nominal" length. In fig. 1(b), x_L is the leg spring compression, F_k is the force upward on the body due to both the leg spring and mechanical stop, and F_y is the force upward on the leg due to the ground. The motor torque is labeled t_1 and acts to increase the shaft angle. Explicit definitions of these appear in the appendix (eqs. A1-A4)

The following parameters describe the motor and actuator:

K_t motor torque constant (N m/amp)
K_b motor back-emf constant (V / (rad/sec))
L motor armature inductance (H)
R motor terminal resistance (Ω)
i_{max} maximum allowable motor current (amps)
μ_s rotational friction coefficient of ball screw and motor shaft (N m / (rad/sec))
e efficiency of ball screw back-drive
M_3 actuator mass (kg)
I ball screw/motor shaft inertia (kg m^2)
p lead of ball screw (m/rad)

The following parameters are associated with the body, leg, and ground:

M_2 body mass (kg)
μ_f friction coefficient between leg and body (N/(m/sec))
M_1 leg mass (kg)
K_L leg spring force constant (N/m)
B_L leg spring damping coefficient (N/(m/sec))
K_m mechanical stop spring force constant (N/m)
B_m mechanical stop spring damping coefficient (N/(m/sec))
K_g ground spring force constant (N/m)
B_g ground spring damping coefficient (N/(m/sec))
g acceleration due to gravity (m/sec^2)

The differential equations governing the vertical motion of the robot are derived using d'Alembert's principle, and are given in the appendix (eqs. A5-A8).

III. Hopping Height Control

Raibert's approach [5] to controlling hopping height was to use a pseudo-control input which determined the trajectory of the actuator for the entire support phase. This input (which was a change in actuator length) was calculated based on measurements of the machine's state at *bottom*. Using fundamental energy relations, Raibert determined a relationship between the state at *bottom*, the control, and the state at the next *top*. This algorithm was shown in [4] to exhibit relatively high sensitivity to some of the system's physical parameters (namely, the leg and ground spring constants). The sensitivity was mainly due to the fact that the leg spring compression at *bottom* is used to predict the next hop height, and the next hop height is roughly proportional the square of this compression.

In [4] we have proposed to develop a low-order approximation to the relation

$$h_i = f\left(\Delta\chi_i , h_{i-1}\right). \qquad (1)$$

In (1) h_i, the height of the i^{th} hop, is a function of h_{i-1}, the height of the previous hop, and $\Delta\chi_i$, the pseudo-control signal.

To perform control, the approximation to (1) is

used to generate a control signal by solving for $\Delta\chi_i$; h_i is set to the desired hopping height, and h_{i-1} is the measured height of the previous hop. The solution provides a 'near-inverse' relation which exhibits reduced sensitivity to parameters of the leg and the ground.

In [4] and [5], the actuator length was considered to be the control input to the system. In the present study, the mechanical model includes the actuator dynamics, and the voltage applied to the DC motor is the control input. During the support phase of the hopping cycle, a constant voltage is applied to the motor, and during the flight phase, proportional feedback is used to return the actuator to its nominal length:

$$v_m = \begin{cases} v_{gi} & y_1 < 0 \\ -K_\theta\,\theta & y_1 \geq 0 \end{cases} \qquad (2)$$

Here, v_{gi}, the voltage applied during the support phase of the i^{th} hop, is a pseudo-control input, and the relation of interest is

$$h_i = \bar{f}\left(v_{gi}, h_{i-1}\right). \qquad (3)$$

The approximation to (3) takes the form

$$\hat{h}_i = a_0 + a_1 v_{gi} + a_2 h_{i-1} + a_3 v_{gi} h_{i-1} + a_4 v_{gi}^2 + a_5 v_{gi} h_{i-1}^2 \quad (4)$$

and h_i can then be symbolically rewritten as

$$h_i = \hat{h}_i + e\left(v_{gi}, h_{i-1}\right) \qquad (5)$$

with $e(v_{gi}, h_{i-1})$ an additive error term representing the approximation error. The form (4) was found in numerical experimentation to represent a good compromise between the need to obtain low approximation error and manageable computation effort in calculating the coefficients $a_0, a_1, .., a_5$. To obtain these coefficients, the mechanical system was simulated off-line for several hopping cycles with different values used for v_{gi} and initial height. The values used for v_{gi} ranged (in equal increments) from 8 to 50 volts, and the values for the initial height ranged from 0.02 to 0.3 meters.[1] The hopping machine was placed at an initial height, allowed to fall, and equation (2) was used to apply the control. The resulting height of the next hop was recorded, along with the initial height and the control parameter value which produced it. After obtaining 100 points on a 10x10 grid, a least-square error curve fit was used to obtain the coefficients a_0 through a_5 in equation (4) (The coefficient values used in simulations appear in the appendix). The 'near-inverse' is found by substituting a desired height (h_d) for $\hat{h}_i$ in (4) and solving for v_{gi}, i.e.

$$v_{gi} = \frac{-B \pm \sqrt{B^2 - 4\,a_4\left(a_0 + a_2 h_{i-1} - h_d\right)}}{2\,a_4} \qquad (6)$$

where

$B = a_1 + a_3 h_{i-1} + a_5 h_{i-1}^2$

There are two solutions for v_{gi}; either one can (theoretically) be used for control. In practice, we choose the solution which is smaller in magnitude, to minimize the actuation effort.

With (6) used, and with some of the system parameters perturbed, a steady state error in hopping height develops. To reduce it, small corrections to the approximating surface are introduced, in the form

$$a_0^{(i)} = a_0^{(i-1)} + \alpha\left(h_{i-1} - \hat{h}_{i-1}\right) \qquad (7)$$

where, $\alpha < 1$ is a small adaptation parameter. The use of this update requires the storage of the *two* most recent hop heights, due to the dependence of $\hat{h}_{i-1}$ on h_{i-2} (eq. (5)).

IV. Simulation Results

A sample of the hopping machine's simulated state trajectory appears in figure 2. The desired hopping height is 30 cm, and the machine was dropped from an initial height of 2 cm. The solid line indicates the trajectory of the foot, and the dotted line is the trajectory of the body mass. Note that the foot penetrates the ground slightly, and a small oscillation occurs just after *touchdown*. This is due to the ground spring/damper. At this hopping height, the duration of the support phase is about 0.2 seconds and the hopping cycle period is about 0.7 seconds.

In figure 3, successive hop heights were recorded over an 80-second time interval, with Eq. 6 used for control, and the adaptation rule (7) deactivated (α=0). The desired hopping height trajectory (solid lines) took on the values [30, 10, 25, 5, 30, 15, and 20] centimeters at times [0,10, 20, ..., 60] seconds. The upper and lower curves represent the system hopping-height tracking responses under simultaneous parameter perturbations for the two "worst-case"

[1]The DC motor was modeled after PMI model JR12M4CH, and the value used for K_θ was 2.0 volts/rad.

scenarios with a tolerance of 10%:

Parameter	Perturbation from nominal value (upper curve)	(lower curve)
K_t	+ 10%	- 10%
M_2	+ 10%	- 10%
μ_f	- 10%	+10%
M_1	+ 10%	- 10%
K_L	+ 10%	- 10%
B_L	- 10%	+ 10%
K_g	+ 10%	- 10%
B_g	- 10%	+ 10%

In figure 4, the upper and lower response curves were obtained under the same conditions as in figure 3, except that in eq. (7), α=0.3. The use of the two most recent hop heights (as opposed to one) improves the tracking performance. Naturally, there is a tradeoff between noise amplification and speed of convergence in eq. (7). A compromise α value has to be established at each noise level.

Conclusion

A more realistic continuous time model of a hopping machine was used to synthesize a near-inverse of the plant. The model includes a description of the actuator which was obtained directly from a mechanical design. The near-inverse relation allows the use of discrete-time hop-height output feedback. It decreased the sensitivity of the hopping height to the physical parameters of the machine and of the terrain. Small on-line corrections to the near-inverse show improvement in hopping height tracking, though their effectiveness when observations are noisy is still to be investigated. The control is suitable for on-line tuning when parameters change slowly, which should improve operation in realistic environments. In the synthesis of the control signal, past hopping heights were used, and, as expected, the use of more memory (past heights) improved the tracking response of the system. The next topics for study include the performance of this vertical control algorithm in a physical prototype, extension to two dimensions, and the development of a more sophisticated adaptive height-control algorithm.

References

[1] J. J. Cox, G. T. Wilfrong (eds.) , *Autonomous Robot Vehicles*, New York: Springer-Verlag .

[2] J. J. Helferty, J. B. Collins, and M. Kam, "Adaptive Control of a Legged Robot Using an Artificial Neural Network," *Proc. ACC* 1989, pp. 165-168.

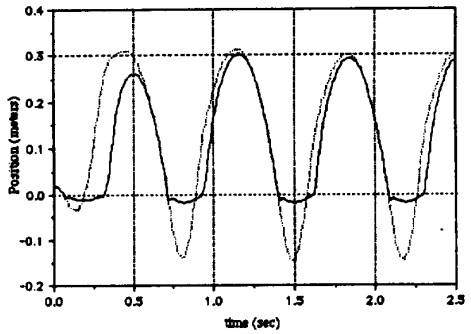


Figure 2. A sample trajectory

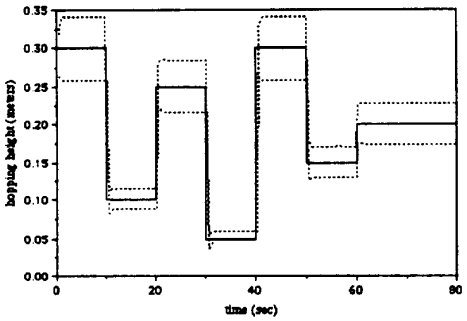


Figure 3. Hopping height tracking response - "worst-case" scenarios.

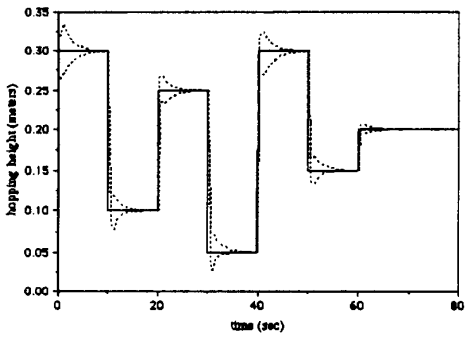


Figure 4. Hopping height tracking under parameter variation with adaptation (α=0.3).

[3] S. H. Lane and R. F. Stengel, "Nonlinear Inverse Dynamics Control Laws - A Sampled Data Approach," *Proc. ACC* 1987, Vol. 2, pp. 1224-1226.

[4] J. Prosser and M. Kam, "Height Control of a One-Legged Hopping Machine Using a Near-Inverse

Model" *Proc. Conference of Information Science and Systems*, Princeton, NJ, March 1992.
[5] M. H. Raibert, "Hopping in Legged Systems - Modeling and Simulation for the Two-Dimensional One-Legged Case," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. SMC-14, No. 3, p.451-463, May / June 1984.
[6] M. H. Raibert, H. B. Brown, and M. Chepponis, "Experiments in Balance with a 3D One-Legged Hopping Machine," *Int'l J. Robotics Research*, Vol. 3, No.2, Summer 1984.
[7] M. H. Raibert, *Legged Robots that Balance* , Cambridge, MA: MIT Press. 1986.
[8] M. H. Raibert, "Experiments in Balance with a 2D One-Legged Hopping Machine," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 106, pp. 75-81, March 1984.
[9] M. H. Raibert and F. C. Wimberly, "Tabular Control of Balance in a Dynamic Legged System," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. SMC-14, No. 2, p.451-463, March / April 1984.
[10] M. Sznaier and M. J. Damborg, "An Adaptive Controller for a One-Legged Mobile Robot," *IEEE Trans. Robotics and Automation*, Vol. 5, No. 2, pp. 253-259, April 1989.
[11] T. J. Todd, *Walking machines*, London: Kogan Page, 1985
[12] B. Widrow, J. M. McCool, and B. P. Medoff, "Adaptive Control by Inverse Modeling," *Proc.. Asilomar Conference on Circuits, Systems, and Computers*, pp. 90-94, 1978.
[13] B. Widrow, D. Shur, and S. Shaffer, "On Adaptive Inverse Control," *Proc. Asilomar Conference on Circuits, Systems, and Computers*, pp.185-189, 1981.
[14] C. Zhang and R. J. Evans, "Offset Elimination in Direct Self-Tuning Control," *IEEE Trans. Autom. Ctrl.*, Vol. 33, No. 6, pp. 603-607, June 1988.

APPENDIX

I. Nominal Parameter Values

K_t	0.17039	M_2	5.45
K_b	0.17	μ_f	7.71
L	0.004	M_1	1.36
R	0.95	K_L	1972.0
i_{max}	20.0	B_L	7.671
μ_s	0.00135	K_m	21917.0
e	0.85	B_m	164.38
M_3	0.91	K_g	21917.0
I	0.0001	B_g	164.38
p	0.00202	g	9.8066

II. Coefficients of eq. (4)

a0	-0.0112072	a3	0.00636197
a1	0.00243704	a4	0.00000942736
a2	0.4.01503	a5	-0.00336151

III. Forces, torque of figure 1(b).

$$x_L = y_1 - y_2 + p\,\theta \tag{A1}$$

$$F_k = \begin{cases} K_m x_L + B_m \dot{x}_L & x_L < 0 \\ K_L x_L + B_L \dot{x}_L & x_L \geq 0 \end{cases} \tag{A2}$$

$$F_y = \begin{cases} -K_g y_1 - B_g \dot{y}_1 & y_1 < 0 \\ 0 & y_1 \geq 0 \end{cases} \tag{A3}$$

$$t_1 = K_t\, i \tag{A4}$$

IV. The differential equations of motion

$$\ddot{y}_1 = -g + \frac{(-F_L + \mu_f(\dot{y}_2 - \dot{y}_1) + F_y)}{M_1} \tag{A5}$$

$$\ddot{y}_2 = \frac{\left(I + e\,p^2 M_3\right)\left(F_L - (M_3 + M_2)\,g + \mu_f(\dot{y}_1 - \dot{y}_2)\right)}{D} + \frac{M_3\,p\left(t_1 - e\,p\,(M_3\,g - F_L) - \mu_s\,\dot{\theta}\right)}{D} \tag{A6}$$

$$\ddot{\theta} = \frac{(M_2 + M_3)\left(t_1 - e\,p\,(M_3\,g - F_L) - \mu_s\,\dot{\theta}\right)}{D} + \frac{e\,p\,M_3\,(F_L - (M_3 + M_2)\,g + \mu_f(\dot{y}_1 - \dot{y}_2))}{D} \tag{A7}$$

where $D = I(M_2 + M_3) + e\,p^2\,M_2\,M_3$ and

$$\dot{i} = \begin{cases} 0 & v_m - K_b\,\dot{\theta} - R\,i < 0 \text{ and } i < -i_{max} \\ 0 & v_m - K_b\,\dot{\theta} - R\,i > 0 \text{ and } i > i_{max} \\ \dfrac{v_m - K_b\,\dot{\theta} - R\,i}{L} & \text{otherwise} \end{cases} \tag{A8}$$

Simulations were performed using a Runge-Kutta fourth order integration scheme with a time step of 0.0001 seconds.