Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Speed Control for a Biped Robot

Brhanemedhn Tegegne [*] **Abdollah Homaifar** [**]
Bijan Sayyar-Rodsari [***]

[*] *North Carolina A&T State University, Greensboro, NC (e-mail: bategegn@ ncat.edu).*
[**] *North Carolina A&T State University, Greensboro, NC 27405 USA (e-mail: homaifar@ ncat.edu).*
[***] *Pavilion Technologies, Inc., Austin, TX (e-mail: bijan@ pavtech.com)*

Abstract: A biologically inspired model for a biped robot is developed. Each leg is modeled as a massless spring equipped with one radial and one angular actuator. The two legs in the model are attached to a point mass. The biped locomotion is modeled as a hybrid dynamic system that switches between four operation phases, with distinct dynamic behavior in each phase. A novel, simple yet robust control law that utilizes symmetry is developed to control the robot's speed, hopping height and balance during forward motion. The controller utilizes a feedforward artificial neural network, trained offline, to find approximate symmetric touchdown angles that are used by the control algorithm to determine actuator corrections. Simulation results show that the controller is able to maintain reference forward speed while maintaining balance during forward motion.

1. INTRODUCTION

To date, the most significant research on dynamic legged locomotion was led by Raibert at the Carnegie-Mellon University (CMU) and Massachusetts Institute of Technology (MIT) Leg labs in the 1980s and 1990s Poulakakis and Smith (2005). He decomposed control of running into three parts-hopping control, forward speed control, and balance. Raibert's legged robots generally consisted of a point mass at the end of an actuated, compliant prismatic leg(s). These controllers resulted in a fast and stable dynamic running. Buehler and his students in the Ambulatory Robotics Laboratory (ARL) at McGill University in Montreal, Quebec, Canada designed and constructed ARL monopod I , ARL monopod II, Scout I, and Scout II Buehler (2002); Ahmadi and Buehler (1999). As in Raibert's robots, these robots had compliant prismatic legs except for Scout I which had stiff legs. The robots at ARL were powered by standard electric actuation, instead of tethered hydraulic actuation citeARL. ARL monopod I and II were able to run at up to 1.2m/s Ahmadi and Buehler (1999). Scout II, a four-legged robot, achieved dynamically stable running of up to 1.3m/s on flat ground Poulakakis and Smith (2005). Rhex, a hexapedal robot inspired by research in cockroach locomotion, had six compliant legs and six actuators - one actuator per leg.

This paper presents a new simple and robust control strategy for bipedal running. The robot model used in this work is similar to ARL monopods. It has two compliant legs prismatic legs. Each leg is equipped with two actuators. As in Raibert's controllers symmetry plays a significant role in designing stable running gaits. We have presented the theoretical ground for the empirical observations that the symmetry-based running control is based. Experiments of forward speed profiles tracking show promising results.

The remaining of this paper is organized as follows. In section 2 we describe the mathematical model of the biped robot. Section 3 is devoted to the development of the control laws. It begins with an discussion about the importance of symmetry in legged locomotion. It also presents the formal problem definition for forward speed control. It then discusses the theoretical ground for the running control laws . Finally, it presents the arguments that lead to the development of the form of the forward speed controller. Section 4 summarizes the results of the different experiments performed. Results of forward speed tracking experiments are given in this section. Section 5 presents the conclusion of the paper in which we highlight the contributions of the paper and future research directions.

2. THE MODEL

The SLIP (Spring Loaded Inverted Pendulum) models for running accurately predict the mechanics of running gaits C. T. Farley and McMahon (1993). The model consists of a massless, spring leg attached at a frictionless pin-jointed hip to the center of mass (COM) of a massive rigid body, as shown in Figure **??**n this study, rigid body rotations are ignored; only point-mass dynamics are considered. During locomotion, the SLIP model alternates through two phases: the stance phase, when the foot is on the ground, and the fight phase, when the foot is in the air. Transitions between the two phases occur at discrete touch-down (TD) and lift-off (LO) events, when the governing equations change instantaneously, making the SLIP system a hybrid dynamic system. TD occurs when the leg strikes the ground, which is assumed to be a horizontal plane rough enough to prevent the foot from

[⋆] This work was supported by NSF-RISE program

978-1-1234-7890-2/08/$20.00 © 2008 IFAC **10.3182/20080706-5-KR-1001.3475**

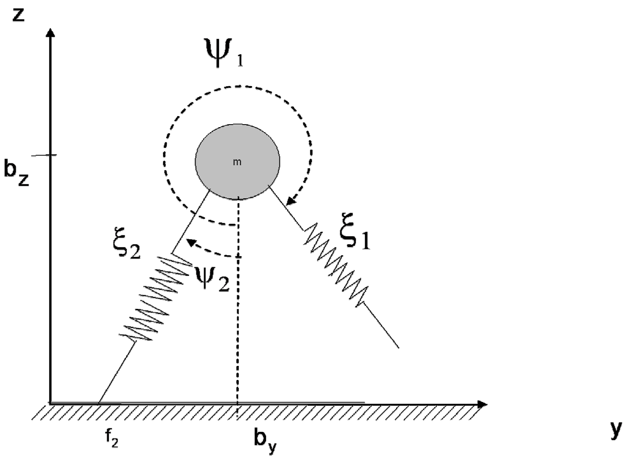

Fig. 1. Planar biped SLIP

sliding during locomotion. After TD, the leg compresses to a point of maximum compression (which is called the bottom) and then unloads until LO, which occurs when the leg has completely unloaded. The mass then travels freely under the influence of gravity until TD occurs again Seipel and Holmes (2005). A single step or stride is defined as the period between two subsequent TD or LO events.

2.1 Bipedal Spring Mass Model

The biped robot model used in this study has two leg springs with equal stiffness constants and rest lengths attached to a point mass representing the body of the robot, as shown in Figure 1. The two legs can freely rotate around the hip joint. The movement of the robot is constrained to the sagittal plane (the longitudinal plane dividing the body into symmetric right and left sections), thus making the robot a planar biped. The two leg springs act independently. During locomotion, each leg passes through two phases – flight and stance. Consequently, the biped robot can be in one of three phases – single-stance, double-stance, or flight. From a systems engineering point of view, the biped robot model can be studied as a switched nonlinear system. A switched system is composed of subsystems and rules that govern the switching among the subsystems. It can be mathematically described as,

$$\delta x\left(t\right)=f_{\sigma}\left(x\left(t\right),u(t),d(t)\right), \qquad x(t_0)=x_0 \tag{1}$$

$$y(t)=g_{\sigma}(x(t),w(t)) \tag{2}$$

where $x(t)$ represents the state, $u(t)$ the control inputs, and $y(t)$ the measured outputs, and $d(t)$ and $w(t)$ stand for external signals such as perturbations. σ, whose value determines which subsystem is active at a given instant, is a piecewise constant signal taking values from an index set $M \triangleq \{1,\ldots,m\}$. $f_k, k \in M$, are vector fields, and $g_k, k \in M$, are vector functions, while the symbol δ denotes the derivative operator in continuous time (i.e., $\delta x(t) = \frac{d}{dt}x(t)$) and the shift forward operator in discrete times (i.e., $\delta x(t) = x(t+1)$).

The switched system representation of the planar biped model comprises four subsystems – the left stance subsystem, the right stance subsystem, the double-stance subsystem, and the flight subsystem. The value of σ is chosen so that $\sigma = 1, 2, 3$, and 4 for the flight, left stance, right stance and double-stance subsystems, respectively. Tables 1 and 2 summarize the states and control inputs of the dynamic model for the planar biped. In these tables and in the rest of this document, variables with subscripts 1 and 2 represent quantities pertaining to the left leg and the right leg, respectively. In general, the problem of locomotion control for the biped robot is equivalent to finding values for the control inputs (which are summarized in Table 2) that, when applied, achieve a desired locomotion behavior. The following sections describe each subsystem in detail.

Table 1. States and their descriptions for the biped robot

State	Description
ζ_i	Length of the left/right ($i = 1/2$) leg
$\dot{\zeta}_i$	Time rate of expansion for the left/right ($i = 1/2$) leg
ψ_i	Angle of the left/right ($i = 1/2$) leg with vertical axis
$\dot{\psi}_i$	Time derivative of ψ_i
b_y	Horizontal displacement of the COM
b_z	Height of the COM
$\dot{b}_y$	Horizontal speed of the COM
$\dot{b}_z$	Vertical speed of the COM

Table 2. Control inputs to the biped robot and their descriptions

Input	Description
υ_i	Radial speed of the left ($i = 1$) or right ($i = 2$) leg
ω_i	Angular speed of the left ($i = 1$) or right ($i = 2$) leg

2.2 Flight Subsystem

The dynamics of the robot in the flight phase is that of a point mass under the action of gravitational force. Mathematically, the dynamics of the flight subsystem can be written as

$$\begin{bmatrix} \ddot{b}_y \\ \ddot{b}_z \\ \dot{\zeta}_1 \\ \dot{\zeta}_2 \\ \dot{\psi}_1 \\ \dot{\psi}_2 \end{bmatrix} = \begin{bmatrix} 0 \\ -g \\ \upsilon_1 \\ \upsilon_2 \\ \omega_1 \\ \omega_2 \end{bmatrix}. \tag{3}$$

2.3 Single-Stance Subsystems

The robot can be in one of two different single-stance phases – the left stance phase and the right stance phase – depending on which leg is on the ground. The dynamic equations governing both phases are identical except for the subscripts distinguishing left from right.

For the left stance phase, If $DU(\zeta_1)$ represents the spring force exerted by the the left leg on the COM when the length of the spring is ζ_1, applying Newton's second law, we have:

$$\begin{bmatrix} \ddot{\zeta}_1 \\ \ddot{\psi}_1 \\ \dot{\zeta}_2 \\ \dot{\psi}_2 \end{bmatrix} = \begin{bmatrix} \zeta_1\dot{\psi}_1^2 - g\cos(\psi_1) - \frac{1}{m}DU(\zeta_1) \\ \dfrac{g\sin(\psi_1) - 2\dot{\zeta}_1\dot{\psi}_1}{\zeta_1} \\ \upsilon_2 \\ \omega_2 \end{bmatrix}. \tag{4}$$

Similarly for a the right stance phase, we have:

$$\begin{bmatrix} \ddot{\zeta}_2 \\ \ddot{\psi}_2 \\ \dot{\zeta}_1 \\ \dot{\psi}_1 \end{bmatrix} = \begin{bmatrix} \zeta_2\dot{\psi}_2^2 - g\cos(\psi_2) - \dfrac{1}{m}DU(\zeta_2) \\ \dfrac{g\sin(\psi_2) - 2\dot{\zeta}_2\dot{\psi}_2}{\zeta_2} \\ v_1 \\ \omega_1 \end{bmatrix}. \tag{5}$$

The challenge in control of the running biped robot comes from the single-stance phase. The time profile of the value of the state vector from TD to LO is called the stance map.

Despite its simple structure, the SLIP model presents difficulties when one wishes to pursue formal analysis and control since it is a hybrid system with nonlinear stance dynamics, which is not closed-form integrable. There has been considerable research aimed at finding an approximate closed-form equivalent of the stance dynamics (see Shih (2000) for a summary). Seipel and Holmes present an approximate stance map by neglecting the force of gravity during the stance phase Seipel and Holmes (2005). The choice to ignore gravity during stance is not uncommon in the robotics literature Shih (2000). This choice is made primarily to simplify the analysis and relies upon the assumption that the spring force dominates the gravitation force and that angular momentum is nearly constant during stance. Due to the high discrepancy between the approximate models and the exact models, the exact model is used in this study.

2.4 Double-Stance Subsystem

The double-stance subsystem is active when both legs are fixed to the ground. Most studies involving biped control concentrate on the single-stance phase because it is the predominant phase of locomotion. In our work, running gaits are generated in such a way that double-stance phases are avoided at all times.

Applying Newton's second law in the vertical and horizontal directions to the free body diagram in Figure 2 gives the following equations of motion for the double-stance phase:

$$\begin{bmatrix} \ddot{b}_y \\ \ddot{b}_z \end{bmatrix} = \begin{bmatrix} 0 \\ -g \end{bmatrix} + \begin{bmatrix} -DU(\zeta_1)\left(\frac{b_y-f_1}{m\zeta_1}\right) - DU(\zeta_2)\left(\frac{f_2-b_y}{m\zeta_2}\right) \\ -DU(\zeta_1)\left(\frac{b_z}{m\zeta_1}\right) - DU(\zeta_2)\left(\frac{b_z}{m\zeta_2}\right) \end{bmatrix}. \tag{6}$$

Since neither leg is in the air, the control inputs do not have any effect. The behavior of the robot in double-stance is determined solely by the initial state handed to the subsystem when it becomes active.

In summary, the dynamics of the biped robot can be compactly written as:

$$\dot{x} = f_\sigma(x, u) \tag{7}$$

where $\sigma = 1, 2, 3,$ or 4. The value of σ changes at TD and LO. The following section discusses switching between subsystems in detail.

2.5 Switching Conditions

The switching signal determines which subsystem is active at any time. Leg i is on the ground if the following condition is satisfied.

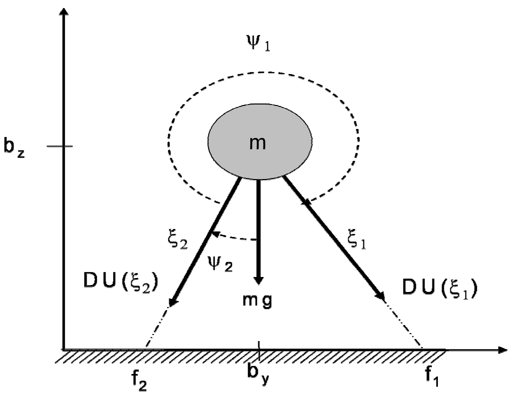

Fig. 2. FBD of the bipedal spring mass system in double-stance

$$|b_z - \zeta_i\cos(\psi_i)| \le \varepsilon \quad AND \quad \left(\zeta_i < \zeta_0 \;\; OR \;\; \dot{\zeta}_i \le 0\right), \tag{8}$$

where ε is a very small positive number that serves as a measure of tolerance, and ζ_0 is the unstretched length of the leg spring. If the function $\sigma_s(b_z, \zeta, \psi, \dot{\zeta})$ is defined as

$$\sigma_s(b_z, \zeta, \psi, \dot{\zeta}) = \begin{cases} 1 & \begin{array}{l} \text{if} \quad |b_z - \zeta\cos(\psi)| \le \varepsilon \\ \text{AND} \;\; \left(\zeta < \zeta_0 \;\; \text{OR} \;\; \dot{\zeta} \le 0\right) \end{array} \\ 0 & \text{Otherwise} \end{cases}$$

then the switching signal for the overall system can be formulated as:

$$\sigma(b_z, \zeta_1, \psi_1, \dot{\zeta}_1, \zeta_2, \psi_2, \dot{\zeta}_2) \tag{9}$$

$$= 2\sigma_s(b_z, \zeta_2, \psi_2, \dot{\zeta}_2) + \sigma_s(b_z, \zeta_1, \psi_1, \dot{\zeta}_1) + 1. \tag{10}$$

TD occurs when one of the two legs goes from a flight to a stance phase. Mathematically, for leg i, TD occurs at time t if

$$\sigma_s(b_z(t-), \zeta_i(t-), \psi_i(t-), \dot{\zeta}_i(t-)) = 0 \tag{11}$$

$$\text{AND} \;\; \sigma_s(b_z(t), \zeta_i(t), \psi_i(t), \dot{\zeta}_i(t)) = 1. \tag{12}$$

where $t-$ denotes a time earlier than t by a vanishingly small amount. Similarly, LO happens when one of the two legs goes from a stance to a flight phase. Therefore, LO occurs at time t if

$$\sigma_s(b_z(t-), \zeta_i(t-), \psi_i(t-), \dot{\zeta}_i(t-)) = 1 \tag{13}$$

$$\text{AND} \;\; \sigma_s(b_z(t), \zeta_i(t), \psi_i(t), \dot{\zeta}_i(t)) = 0. \tag{14}$$

Since the switching signal depends only on the state variables, it is a case of pure-state-feedback switching.

3. LOCOMOTION CONTROL

Control of running can be decomposed into controlling hopping height, forward speed and balance. The controller presented here rely heavily on symmetry. For the legged robot to keep its forward running speed fixed over the duration of a stance phase, the torques and horizontal forces exerted on the body by the legs must integrate to zero. This is true for both running machines and animals

Raibert (1986). Study of animal locomotion shows that, when animals are running at constant speed, the stance leg reaches maximum compression when it is in a vertical posture C. T. Farley and McMahon (1993), which is a requirement for a symmetric stance map.

3.1 Forward Speed Control

Each stride in running is composed of a single-stance phase and a flight phase. Since the horizontal speed of the COM does not change during the flight phase, the horizontal speed can only be controlled by adjusting the TD state of the TD leg so that the horizontal speed of the COM at the next LO is as close as possible to the reference horizontal speed. The straightforward way to achieve this is to find an expression for the horizontal speed at the next LO in terms of the TD parameters. This method is impractical for a number of reasons. The first one is that, since the equations that define the dynamics of the single-stance subsystem (viz., Equations 4 and 5) are not closed form integrable, it is difficult, if not impossible, to find a closed form expression of the TD state in terms of the LO horizontal speed. Furthermore, the mapping from the LO horizontal speed to the TD state is not one-to-one (*i.e.,* there are more than one set of values for the TD state that result in the same horizontal speed at LO).

The problem of forward speed control can be formulated as follows.

Problem 1. Given a reference horizontal speed at LO, $\dot{b}_{yr}(t)$, where $0 \leqslant t \leqslant T$, determine a control input, $u(t) \triangleq [\, v_1(t) \; v_2(t) \; \omega_1(t) \; \omega_2(t) \,]^T$ that, when applied to the switched system, comprised of the four subsystems defined by Equations 3, 4, 5, and 6, with the initial state $x = x_0$ and the switching rule given by Equation 9, minimizes the tracking error p defined by

$$p = \frac{1}{N} \sum_{t_{\rm lo}} \left| \dot{b}_y(t) - \dot{b}_{yr}(t) \right| \qquad \text{for } 0 \leqslant t \leqslant T \tag{15}$$

where $t_{\rm lo}$ is the time when lift-off, defined by Equation 13, occurs, and N is the number of LO events fired during the interval $0 \leqslant t \leqslant T$.

Generally, the horizontal speed of the COM changes only during a stance phase. For certain values of the TD parameters, however, the horizontal speed of the COM at TD equals the horizontal speed of the COM at LO. These TD states are of particular interest for two main reasons. First, they represent steady-state operating points. Under normal operating conditions, the robot is most likely required to travel at a constant horizontal speed. This operation mode requires TD conditions that result in zero net horizontal acceleration. Secondly, operating the robot around those points implies an upright posture for the robot during motion. This is very important in avoiding stumbling and maintaining balance.

The length of the leg spring at TD is assumed to be equal to the rest length of the leg spring. Under this assumption, Theorem 1 below states that checking whether the spring leg angle at TD is the negative of the spring leg angle at LO or not is a sufficient test for symmetry of the stance map. Therefore, as an implication of Theorem 1, solving for TD angles that result in symmetry is reduced to solving for TD angles that result in opposite leg spring angles at TD and LO. Once those TD angles are found, assuming an ideal angular actuator that can rotate the leg at any angular speed at any time during flight, it is always possible to deploy the leg at the symmetric TD angle. It can be shown, however, that it is not always possible to find a TD angle that results in a symmetric stance map. More specifically, for speeds of the COM greater than a certain value, the leg spring cannot generate enough resistance against the inertia and weight of the robot's mass to prevent it from hitting the ground before LO. According to Theorem 2, if the speed of the COM is less than a certain value, there exists a TD angle (called the symmetry angle) that satisfies the condition of Theorem 1.

Theorem 1. Given a SLIP, assuming that the length of the leg spring at TD equals the rest length of the leg spring, if the TD angle and the LO angle are the negatives of each other, then the horizontal speed of the COM at TD equals the horizontal speed of the COM at LO. Also, under the same condition, the vertical speed of the COM at TD and the vertical speed of the COM at LO are equal in magnitude but opposite in sign, i.e.,

$$\dot{b}_{ytd} = \dot{b}_{ylo} \tag{16}$$

$$\dot{b}_{ztd} = -\dot{b}_{zlo}, \tag{17}$$

where $\dot{b}_{ytd}$, and $\dot{b}_{ylo}$ are the horizontal speeds of the COM at TD and LO, respectively, while $\dot{b}_{ztd}$ and $\dot{b}_{zlo}$, respectively, are the vertical speeds of the COM at TD and LO.

Theorem 2. For a given SLIP with a rest spring length ζ_0, body mass m, and leg spring stiffness constant k, for any speed of the COM at TD less than a value $v_{\max}$, there exists a TD angle ψ_{sym} such that the horizontal speed of the COM at TD, $\dot{b}_{ytd}$, is equal to the horizontal speed of the COM at LO, $\dot{b}_{ylo}$.

Theorem 3. Given a SLIP with TD speed less than $v_{\max}$ (as in Theorem 2)*, the partial derivative of the horizontal speed at LO* $\dot{b}_{ylo}$ w.r.t. the TD angle ψ_t evaluated at the symmetric TD angle ψ_{sym} is always positive, i.e.,

$$\left. \frac{\partial}{\partial \psi_t} \dot{b}_{ylo} \right|_{\psi_t = \psi_{sym}} > 0 \tag{18}$$

By Theorem 3, a SLIP with given velocity strikes the ground at an angle slightly greater than the symmetric TD angle corresponding (by Theorem 2) to the TD velocity, then the LO horizontal speed will be slightly greater than the horizontal speed at TD.

Touchdown Control The implementation of the touchdown control algorithm is based on Theorem 3, with the forward speed of the robot controlled using the TD angle. Increasing (decreasing) forward speeds are achieved by touching the ground at angles greater (less) than the symmetry TD angle. Figure 3 shows a horizontal speed stance map of the three qualitatively different scenarios: TD angle less than, equal to and greater than the symmetry TD angle. The vertical coordinate corresponding to the start point in each graph represents the horizontal speed at TD, and that of the end point represents the horizontal speed

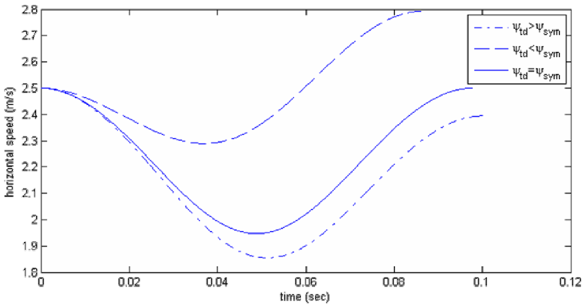

Fig. 3. Effect of asymmetry on horizontal speed

at LO. It can clearly be seen that the middle plot in Figure 3 is symmetric about a vertical axis that passes through the point of lowest horizontal speed. The result presented in Figure 3 confirms Theorem 3.

Let us define f as

$$f : v_{td} \times \phi_{td} \times \psi_{td} \longmapsto \psi_{lo}, \tag{19}$$

where v_{td} is the speed of the COM, ϕ_{td} is the angle the velocity of the COM makes with the vertical axis measured clockwise, ψ_{td} is the TD angle, and ψ_{lo} is the LO angle. Practically, f represents a solution to the set of differential equations defining the single-stance subsystem (viz., Equations 4 or 5). Finding the symmetric TD angle is equivalent to solving the following equation for ψ_{td}.

$$f(v_{td}, \phi_{td}, \psi_{td}) + \psi_{td} = 0 \tag{20}$$

for given values of v_{td} and ϕ_{td}. As mentioned, a closed-form analytic solution of Equation 4 or 5 does not exit, so one must resort to numerical methods. Finding a solution to Equation 20 is computationally expensive, but the controller relies heavily on the solution. In this work, a root-finding tool that uses a combination of bisection, secant, and inverse quadratic interpolation methods is used. The convergence speed is satisfactory for offline calculation but too slow for real-time application. As a work-around to the problem, Equation 20 is numerically solved offline for equally spaced values of v_{td} and ϕ_{td}, and the result is used to train a feedforward neural network. The controller then uses the trained neural network to find approximate values of the symmetric TD angle for different TD velocities. Figure 4 shows a plot of the corresponding symmetric TD angle for different combinations of TD speed and velocity angle. The forward speed controller is a feedback controller that adjusts the speed of the robot by changing the TD angle relative to the symmetric angle. If the horizontal speed at TD is equal to the reference horizontal speed, the TD angle must be equal to the symmetric angle. If it is less (greater) than the reference horizontal speed, then the TD angle must be less (greater) than the symmetric TD angle. At the same time, the deviation from the symmetric angle must be kept small to avoid stumbling.

The control law has the following form.

$$k = 1 + k_1 \frac{\dot{b}_{ytd} - \dot{b}_{yr}}{\dot{b}_{yr}} \tag{21}$$

$$\psi_{td} = k\psi_{sym}, \tag{22}$$

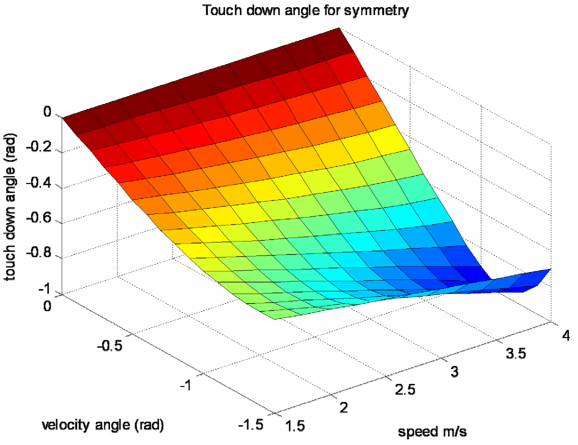

Fig. 4. Symmetric surface for speed range of 1.5 to 4m/s.

where k_1 is the feedback gain and $\dot{b}_{yr}$ is the reference forward speed at LO. In Equation 21, k becomes 1 if $\dot{b}_{ytd} = \dot{b}_{yr}$; substituting 1 for k in Equation 22 gives $\psi_{td} = \psi_{sym}$. And, if $\dot{b}_{ytd} < \dot{b}_{yr}$ in Equation 21, then $k < 1$. Since running with positive forward speed results in negative symmetric TD angles, $k < 1$ implies $\psi_{td} > \psi_{sym}$. A TD angle greater than the symmetric angle results in a net positive horizontal acceleration (as expected), minimizing the difference between $\dot{b}_{ytd}$ and $\dot{b}_{yr}$. Using similar logic, when $\dot{b}_{ytd} > \dot{b}_{yr}$, we have $\psi_{td} < \psi_{sym}$.

Prediction of symmetric TD angles is very sensitive to the values of the SLIP parameters used in training the neural network. In any real-time control system, there is always some amount of external noise. To reduce the effect of noise, the control algorithm needs some feedback correction mechanism. It is assumed that the symmetric TD angle in the presence of noise is some positive constant times the noise-free symmetric TD angle. i.e.,

$$\psi_{sym(noise)} = k_c \psi_{sym(ideal)} \tag{23}$$

where k_c is the correction factor.

Therefore, while controlling the speed of the robot, the control algorithm tries to estimate the value of k_c. The noise rejection algorithm uses error information from a previous stride to estimate the value of k_c. If the symmetric TD angle is predicted correctly, the LO horizontal speed at a current stride must have moved towards the reference LO horizontal speed in reference to a previous stride. A case otherwise shows an error in the prediction of the symmetric TD angle. Based on this information, the correction factor k_c is adjusted in a way that minimizes the effect of the noise. The following iterative algorithm is used to adjust k_c. Initially, $k_c = 1$.

$$if\{\dot{b}_{yr} - \dot{b}_y\}_i > 0 \ AND \ \{\dot{b}_{yr} - \dot{b}_y\}_i > \{\dot{b}_{yr} - \dot{b}_y\}_{i-1}$$

$$k_c = k_c * k_2$$

$$if\{\dot{b}_{yr} - \dot{b}_y\}_i < \ 0 \ AND \ \{\dot{b}_{yr} - \dot{b}_y\}_i < \{\dot{b}_{yr} - \dot{b}_y\}_{i-1}$$

$$k_c = \frac{k_c}{k_2}$$

where i represents the index of the current stride and k_2 is positive constant slightly less than 1. The TD control

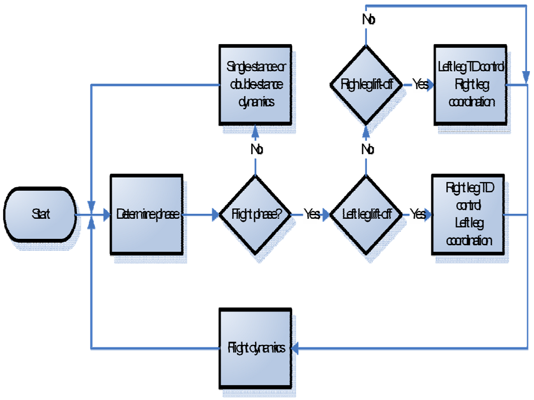
Start
Determine phase
Flight phase?
Yes
No
Single-stance or double-stance dynamics
Left leg lift-off
Yes
No
Righ leg lift-off
Yes
No
Left leg TD control Right leg coordination
Right leg TD control Left leg coordination
Flight dynamics

Fig. 5. Structure of forward speed controller

algorithm given in Equations 21 and 22 is modified to include the disturbance rejection algorithm as follows:

$$k = k_c + k_1 \frac{\dot{b}_{ytd} - \dot{b}_{yr}}{\dot{b}_{yr}} \tag{24}$$

$$\psi_{td} = k\psi_{sym}, \tag{25}$$

The fractional change in horizontal speed behaves similarly for different horizontal speeds around the symmetry point. This observation is used in determining the form of the control law. Instead of the absolute error in horizontal speed, the control law uses the fractional error in speed. With this choice, the same value for the feedback gain constant works for a wider range of values of speed. Simulations confirm that this choice results in a better performance. Comparison of the two methods is presented in a subsequent section.

Off-Turn Leg Controller The main purpose of the off-turn leg controller is to keep the foot of the off-turn leg as far away from the ground as possible to avoid collision. The controller generates input values for the angular actuator of the off-turn leg so that, when TD occurs for the other leg, the position of the off-turn leg is upwards As soon as the flight subsystem is active, the approximate flight time, t_{flight}, before TD for the active leg is calculated using the following formula.

$$t_{flight} = \frac{\dot{b}_z + \sqrt{\dot{b}_z^2 - 2g\left(\zeta_0 \cos\left(\psi_{sym}\right) - b_z\right)}}{g} \tag{26}$$

The control input ω_i for the off-turn leg is then calculated as

$$\omega_i = \frac{(\pi - \psi_i)}{t_{flight}}, \tag{27}$$

where ψ_i is the current leg angle for the off-turn leg.

Figure 5 shows a flow chart for determining the action of the two components of the forward speed controller. The off-turn leg controller is active on a leg when the other leg is on the ground. Alternation between the two legs is achieved using the flow chart illustrated in Figure 5.

4. SIMULATION RESULTS

The results reported here are based on the value of the SLIP parameters shown in Table 3. A two-layer feedforward neural network with 10 neurons in the hidden layer is trained for equally spaced values of the TD speed in the range $[1.5m/s,\ 4m/s]$ and velocity angle in the range $[-\frac{\pi}{2}, 0]$ for 200 epochs. The training set has 176 points. The MSE of the neural network on the training set after training is 5.0×10^{-5}. The performance of the forward speed controller using the trained neural network in tracking different references was tested.

Table 3. SLIP parameter used in experiment

Parameter	Symbol	Value
Spring stiffness constant	k	5000N/m
Mass of robot	m	7Kg
Unstretched leg spring length	ζ_0	0.17m

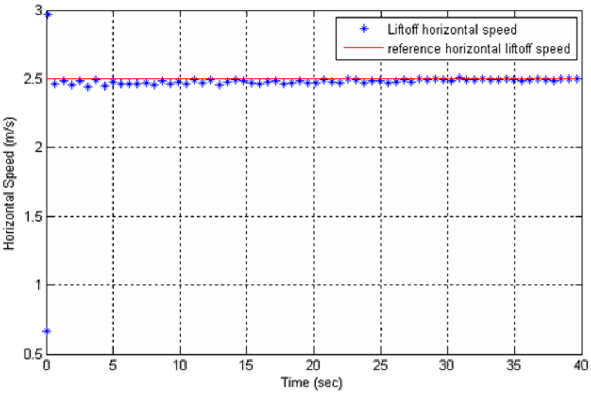

Fig. 6. Speed tracking with a step input reference

The initial value of the state vector for all the experiments is

$$X_0 = [0.1700, 0.1700, 0, 0, -0.8000, 3.1416, 0, 0, 0, 0, 0.2679, 0.6662, -3.0276]. \tag{28}$$

At this initial state, the robot is in the flight phase. Figure 6 shows the result of a speed reference tracking experiment for a step input. As shown in the figure, the horizontal speed came within 2% of the desired value in just two steps. The mean absolute error for the time range $0 < t < 40sec$ is less than $0.05m/s$. Figure 7 shows a result of a speed reference tracking experiment for a time varying reference speed. Comparing the result of this experiment to that of a similar experiment presented in Figure 2.13, page 49 of Raibert (1986) clearly shows that this controller gives a closer performance. We believe that this is mainly due to the crude assumption made by the author that the symmetric TD angle depends only on the horizontal speed at TD. This is stated on page 42 of the same book, which reads, "For each forward speed there is a unique foot position that results in zero net forward acceleration. We call this the neutral point." Figure 8 shows the result of a speed tracking experiment using absolute error instead of fractional error. Comparing Figure 7 with Figure 8 clearly shows that using fractional error results in a better performance than using absolute error. Figure 9 shows the operation region for the biped robot with parameter values shown in Table 3. The vertical axis shows hopping height while the horizontal speed at LO is represented by values on the horizontal axis.

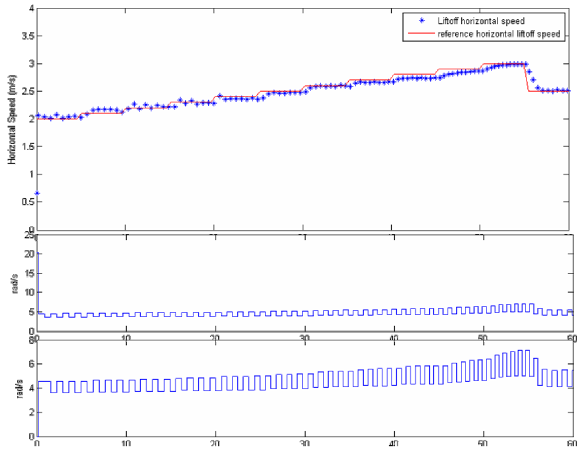

Fig. 7. Forward speed control and inputs w_1 and w_2

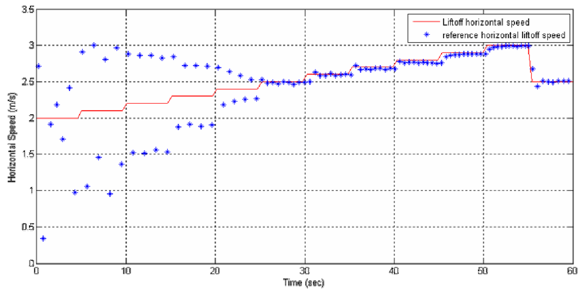

Fig. 8. Controller performance using absolute error

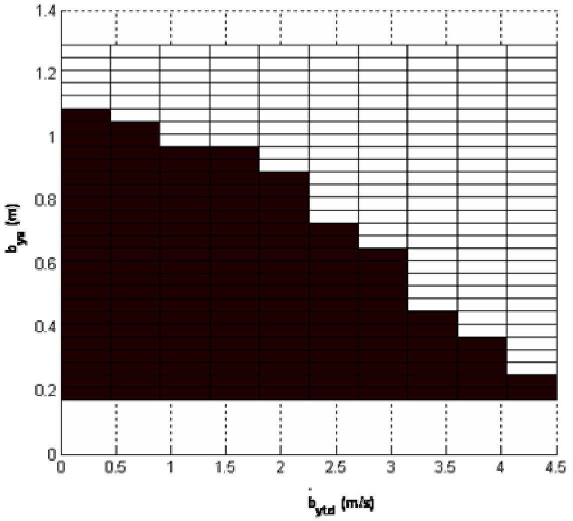

Fig. 9. Operation region for the biped robot

A closer look at the TD control algorithm shows that the control law has both closed-loop and open-loop characteristics. The open-loop behavior is a result of the assumption that the symmetric TD angles calculated using the trained neural network are error-free. The TD control law does not compensate for disturbances in the calculated symmetric TD angles that come from different sources. On the other hand, the control law is a closed-loop controller because it does not assume the magnitude of horizontal acceleration due to deviation from the symmetric TD angle. Due to its open-loop component the controller shows high sensitivity to system disturbances.

5. CONCLUSION

Experimental results of forward speed control have shown improved performances over those reported by Raibert in his book. The major contributions of this work are (a) development of a new control law that uses fractional errors instead of absolute error and, (b) the use of neural networks in approximating values of symmetric TD angles. Recognition that the symmetric TD angle depends not only on the horizontal component of the TD velocity but also its vertical component is also a contribution of this work, (c) laying the theoretical ground for symmetry-based running control.

REFERENCES

M. Ahmadi and M. Buehler. The arl monopod ii running robot: Control and energetics. *IEEE International Conference on Robotics and Automation*, 1999.

M. Buehler. Dynamic locomotion with one, four and six-legged robots. *Journal of Robotics Society of Japan*, 2002.

J. Glasheen C. T. Farley and T. A. McMahon. Running springs: Speed and animal size. *Journal of Experimental Biology"*, 185:71–86, 1993.

Ioannis Poulakakis and James Andrew Smith. Modeling and experiments of untethered quadrapal running with a bounding gait: The scout ii robot. *International Journal of Robotics Research*, 2005.

Marc H. Raibert. *Legged Robots That Balance*. The MIT Press, 1986.

Justin E. Seipel and Philip Holmes. Running in three dimensions: Analysis of a point-mass sprung-leg model. *Int. J. Rob. Res.*, 24(8):657–674, 2005. ISSN 0278-3649. doi: http://dx.doi.org/10.1177/0278364905056194.

W.A. Shih, C.-L. Gruver. Approximating the stance map of a 2-dof monoped runner. *Journal of Nonlinear Science*, 10:533–568, 2000.