Achieving Bipedal Locomotion on Rough Terrain through Human-Inspired Control

Shishir Kolathaya and Aaron D. Ames
Texas A&M University
College Station, TX, USA
shishirny@tamu.edu, aames@tamu.edu

Abstract — This paper presents a method for achieving robotic walking on rough terrain through Human-Inspired Control. This control methodology uses human data to achieve human like walking in robots by considering outputs that appear to be indicative of walking, and using nonlinear control methods to track a set of functions called Canonical Walking Functions (CWF). While this method has proven successful on a specific well-defined terrain, rough terrain walking is achieved by dynamically changing the CWF that the robot outputs should track at every step. To make the computation more tractable Extended Canonical Walking Functions (ECWF) are used to generate these desired functions instead of CWF. The state of the robot, after every non-stance foot strike, is actively sensed and a new CWF is constructed to ensure Hybrid Zero Dynamics is respected for the next step. Finally, the technique developed is implemented on different terrains in simulation. The same technique is adopted experimentally on the bipedal robot AMBER and tested on sinusoidal terrain. Experimental results show how the walking gait morphs based upon the terrain, thereby justifying the theory applied.

Keywords: *Bipedal robotic walking, rough terrain navigation, human-inspired control*

I. Introduction

Inspired by the capacity to navigate over rough terrain, bipedal walking represents a challenge in itself in disaster scenarios. Bipedal robots are one of the best means of locomotion (if not the best) on any random terrain, since they have a very small footprint and can display a wide variety of motion primitives to address the difficult problems associated with motion planning.

Despite the fact that the problem of walking on a random terrain is seemingly simple to humans, it remains a difficult problem to solve in the context of bipedal robots. Out of the many approaches used, the concept of Zero Moment Point ([11]) is popular and has been used extensively. But, this technique assumes that the robot has feet and they are flat all the time (which is very "un-human-like"). Walking has also been studied by viewing the robot as an inverted pendulum in [6], [10], and [5], [9] uses the idea of passive dynamic walking in robots. [12] uses the concept of Hybrid Zero Dynamics to create a low dimensional representation that yields stable walking with point feet. Motivated by Hybrid Zero Dynamics, Human-Inspired Control was introduced in order to create more human-like walking gaits through the use of human-inspired output functions, and achieve flat-ground walking with AMBER [3], [13]. Human-Inspired Control is heavily influenced by the way humans actuate their joints through Central Pattern Generators (CPG) to realize walking (see [8]). The idea is to track a set of functions (termed *Canonical Walking Functions (CWF)*) which emulate the trajectories of human walking. While these methods have been utilized to achieve rough terrain walking through the inherent robustness of the human-inspired controllers, the current framework lacks a formal way for addressing rough terrain locomotion in the context of human-inspired control.

This paper presents a method to address this gap by using the concept of *Intermediate Motion Transitions* along with Human-Inspired Control. We argue that, depending on the terrain, appropriate patterns are injected during human locomotion to get the "right" kind of walking. Here "right" could mean stability, power consumption, speed of execution, feasibility etc. (flat-ground walking that was achieved with AMBER [13] can be considered to be one such pattern). Since the list of possibilities of a terrain is never ending, it is not possible to store infinitely large number of patterns in the robot. Hence, we claim that for a specific terrain with small variations, we can generate patterns (or CWF) from the reference walking function dynamically, and the extent of the variations are decided by the feasibility of implementation and stability of the robot. In this paper, we choose flat ground, and use the reference CWF for flat ground (obtained from [13]). The walking is achieved in such a way that it respects Hybrid Zero Dynamics; this implies that the actuated outputs of the robot are tracking the desired functions even through impacts. Then, we allow small perturbations in the terrain, i.e., allow small changes in slopes (γ) and generate new CWF in such a way that it brings it back to normal flat ground walking mode.

This paper begins with a description of the bipedal robot AMBER and its model in Section II; we then give a brief overview of the control strategy used to achieve walking in simulation. Section III describes our method for achieving parameters for these human-inspired controllers by reviewing the optimization and related constructions used to find these parameters. In Section IV, our method for computing ECWF that allow for transitions back to nominal walking behaviors is presented. Finally, the method for computing the intermediate surfaces for rough terrain is implemented in AMBER and analyzed in Section V.

II. Robot Modeling and Control

Walking involves alternating phases of continuous and discrete dynamics. Hence it is modeled as a hybrid control system $\mathscr{HC} = (X,U,S,\Delta,f,g)$ (see [3], [4] for a formal definition). For our planar robot, AMBER, which has 5 links (2 calves, 2 thighs and a torso, see Fig. 1), we can define the configuration space Q, defined by the joint angles: $\theta = (\theta_{sf},\theta_{sk},\theta_{sh},\theta_{nsh},\theta_{nsk})^T$, where θ represents the joint angles of stance and non-stance feet, stance and non-stance knees, stance and non-stance hips. Note that the walking will be achieved in a 2D robot, and therefore it is supported in the lateral plane via a boom; this boom does *not* provide support to the robot in the sagittal plane. This means that the torso, through which the boom supports the robot, can freely rotate around the boom. The boom is fixed rigidly to a sliding mechanism (Fig. 2). In addition, AMBER's feet are not actuated and is thus an underactuated robot.

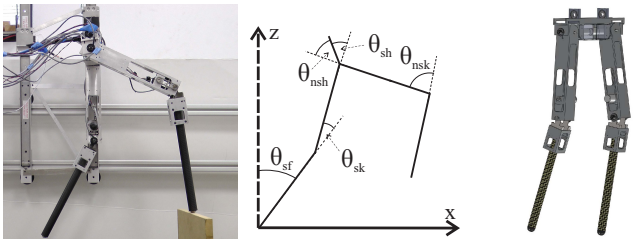

Fig. 1: The biped AMBER (left), the angle conventions (center), and the SolidWorks model of AMBER (right).

For the hybrid control system considered, $\mathscr{HC}$, $X \subset TQ$ is the domain given by the constraint $h_R(\theta) \geq 0$, where h_R is the height of the non-stance foot, $U \subset \mathbb{R}^4$ is the set of admissible controls, $S \subset X$ is the guard given by $h_R = 0$, Δ is the reset map which provides an instantaneous change in velocity at foot strike, and $\dot{x} = f(x) + g(x)u$, with $x = (\theta^T, \dot{\theta}^T)^T \in \mathbb{R}^{10}$

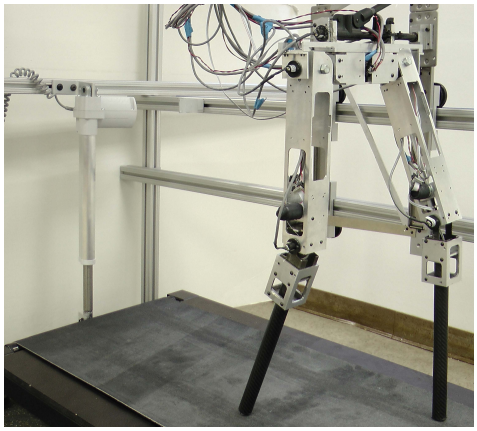

Fig. 2: AMBER Experimental Setup showing the boom and the treadmill. The treadmill is raised up and down to emulate the rough terrain through a linear actuator shown in the back.

and u the torque input, is a control system obtained from the Lagrangian of the robot:

$$L(\theta,\dot{\theta}) = \frac{1}{2}\dot{\theta}^T D(\theta)\dot{\theta} - V(\theta). \tag{1}$$

Including the DC motor model results in a new control system: $\dot{x} = f_v(x) + g_v(x)V_{in}$, with voltage, V_{in}, being the control input. See [14] for more details.

Human-Inspired Control. This section reviews *human-inspired control* which is the approach we use to achieve walking both in simulation and experiment on a given known terrain. For simplicity, we will describe the control law for flat ground walking which are described in detail in [4] (also see [3] for related results in the case of full actuation and [14] for results on other kinds of terrains). The idea will be to extend flat ground walking to rough terrain with minimal variations.

Human Outputs and Walking Functions. As suggested by the name, Human-Inspired Control involves using human data to inspire the development of controllers that provably result in robotic walking. While there are several ways to use the human data, we pick simple linear combinations of joint angles and specifically term them "human outputs" (see [3] for details). The idea is to produce a low-dimensional representation for bipedal walking by using these outputs. Since AMBER has 5 links, it is necessary to chose to pick 5 human outputs which are completely going to represent AMBER walking. The five outputs are:

1) The linearization of the x-position of the hip, p_{hip}:
$$\delta p_{\text{hip}}(\theta) = L_c(-\theta_{sf}) + L_t(-\theta_{sf} - \theta_{sk}), \tag{2}$$
2) The linearization of the slope of the non-stance leg m_{nsl}, (the tangent of the angle between the z-axis and the line on the non-stance leg connecting the ankle and hip):
$$\delta m_{nsl}(\theta) = -\theta_{sf} - \theta_{sk} - \theta_{sh} + \theta_{nsh} + \frac{L_c}{L_c + L_t}\theta_{nsk}. \tag{3}$$
3) The angle of the stance knee, θ_{sk},
4) The angle of the non-stance knee, θ_{nsk},
5) The angle of the torso from vertical,
$$\theta_{tor}(\theta) = \theta_{sf} + \theta_{sk} + \theta_{sh}. \tag{4}$$

A similar idea was also used in [7] where the authors use the terminology "constraints" as opposed to "outputs", and which, evidently, are constrained to realize bipedal walking.

Having defined the outputs, we make the following observations from the human data: the linearized position of the hip is a linear function of time:

$$\delta p^d_{\text{hip}}(t,v) = v_{hip}t, \tag{5}$$

Inspection of the other outputs indicates that they appear to act like the time solution to linear mass-spring-damper systems (see [3]), which motivates the introduction of the *Canonical Walking Function (CWF)*:

$$y_H(t,\alpha) = e^{-\alpha_4 t}(\alpha_1\cos(\alpha_2 t) + \alpha_3\sin(\alpha_2 t)) + \alpha_5. \tag{6}$$

Human-Inspired Outputs. Having defined the outputs, we can construct a controller that drives four of the outputs of the robot (hip position is omitted since AMBER is an underactuated robot and it is not possible to drive five outputs with four actuators) to the outputs of the human, as represented by the *CWF*: $y_a(\theta(t)) \to y_d(t,\alpha)$, with:

$$\begin{aligned} y_d(t,\alpha) = &\quad [y_H(t,\alpha_{nsl}), y_H(t,\alpha_{sk}), y_H(t,\alpha_{nsk}), y_H(t,\alpha_{tor})]^T, \\ y_a(\theta) = &\quad [\delta m_{nsl}(\theta), \theta_{sk}, \theta_{nsk}, \theta_{tor}(\theta)]^T, \end{aligned} \tag{7}$$

where $y_H(t,\alpha_i)$, $i \in \{nsl, sk, nsk, tor\}$ is the *CWF* (6) but with parameters α_i specific to the output being considered. Grouping these parameters with the velocity of the hip, v_{hip}, that appears in (5), results in the vector of parameters $\alpha = (v_{hip}, \alpha_{nsl}, \alpha_{sk}, \alpha_{nsk}, \alpha_{tor}) \in \mathbb{R}^{21}$.

We can remove the dependence of time in $y_d(t,\alpha)$ based upon the fact that the (linearized) position of the hip is accurately described by a linear function of time:

$$\tau(\theta) = (\delta p^R_{\text{hip}}(\theta) - \delta p^R_{\text{hip}}(\theta^+))/v_{\text{hip}}, \tag{8}$$

where $\delta p^R_{\text{hip}}(\theta^+)$ is the linearized position of the hip at the beginning of a step. θ^+ is the configuration where the height of the non-stance foot is zero, i.e., $h_\gamma(\theta^+) = 0$. Using (8), we define the following *human-inspired* output:

$$y_\alpha(\theta) = y_a(\theta) - y_d(\tau(\theta), \alpha). \tag{9}$$

Control Law Construction. The outputs were chosen so that the decoupling matrix, $A(\theta,\dot\theta) = L_g L_f y_\alpha(\theta,\dot\theta)$ with L the Lie derivative, is nonsingular. Therefore, we can define the following torque controller:

$$\begin{aligned} &u_{(\alpha,\varepsilon)}(\theta,\dot\theta) = \\ &-A^{-1}(\theta,\dot\theta)\left(L_f^2 y_\alpha(\theta,\dot\theta) + 2\varepsilon L_f y_\alpha(\theta,\dot\theta) + \varepsilon^2 y_\alpha(\theta)\right). \end{aligned} \tag{10}$$

In other words, we can apply feedback linearization to obtain the linear system on the human-inspired output: $\ddot y_\alpha = -2\varepsilon\dot y_\alpha - \varepsilon^2 y_\alpha$. More details can be found in [13]

III. Human-Inspired Hybrid Zero Dynamics (HZD)

The goal of the human-inspired controller (10) was to drive the outputs of the robot to the outputs of the human: $y_a \to y_d$ as $t \to \infty$. Yet due to the occurrence of the impact with every step, this may not be true all the time. Therefore, the goal of this section is to find the CWF for which this is satisfied; the end result is that these outputs will define the hybrid zero dynamics surface.

Problem Statement. Since AMBER is underactuated, θ_{sf} and $\dot\theta_{sf}$ cannot be controlled. This leads to the notion of using the zero dynamics surface which has a dimension of 2. The control law is applied in such a way that the controllable outputs are forced to 0 as $t \to \infty$. As long as these outputs are exponentially stable, we can realize a surface which represents bipedal robotic walking. Therefore, with the human-inspired controller applied, we say that the controller renders the *zero dynamics surface*:

$$\mathbf{Z}_\alpha = \{(\theta,\dot\theta) \in TQ : y_\alpha(\theta) = \mathbf{0},\ L_f y_\alpha(\theta,\dot\theta) = \mathbf{0}\} \tag{11}$$

exponentially stable; moreover, this surface is invariant for the continuous dynamics. Note that here $\mathbf{0} \in \mathbb{R}^4$ is a vector of zeros and we make the dependence of $\mathbf{Z}_\alpha$ on the set of parameters explicit. It is at this point that continuous systems and hybrid systems diverge: while this surface is invariant for the continuous dynamics, it is not necessarily invariant for the hybrid dynamics. In particular, the discrete impacts in the system cause the state to be "thrown" off of the zero dynamics surface. Therefore, a hybrid system has *Hybrid Zero Dynamics* (HZD) if the zero dynamics are invariant through impact: $\Delta(S \cap \mathbf{Z}_\alpha) \subset \mathbf{Z}_\alpha$.

The goal of *human-inspired HZD* is to find parameters α^* that solve the following constrained optimization problem:

$$\begin{aligned} \alpha^* = \underset{\alpha\in\mathbb{R}^{21}}{\text{argmin}}\ &\text{Cost}_{\text{HD}}(\alpha) \\ \text{s.t}\quad &\Delta(S\cap\mathbf{Z}_\alpha) \subset \mathbf{Z}_\alpha \end{aligned} \tag{12}$$

(HZD)

with Cost_{HD} the least squares fit of the CWF with the human data. This determines the parameters of the CWF that gave the best fit of the human walking functions to the human output data, but subject to constraints that ensure HZD. To get provable and physically realizable walking, other constraints are imposed like non-stance foot height clearance, torque and velocity constraints (see [13]). The optimization also produces a fixed point $(\theta(\vartheta(\alpha)), \dot\theta(\vartheta(\alpha)))$ of the periodic gait which can be used to compute the transitions on rough terrain. Space constraints limit the explanation of the guard configuration of the robot, $\vartheta(\alpha)$, but it can be found in [4].

Zero Dynamics. With the control law ensuring HZD, we can explicitly construct the zero dynamics surface. In particular, we utilize the constructions in [12], reframed in the context of canonical human walking functions. In particular, we define the following coordinates for the zero dynamics:

$$\begin{aligned} \xi_1 &= \delta p^R_{\text{hip}}(\theta) =: c\theta \\ \xi_2 &= D(\theta)_{1,1}\dot\theta =: \gamma_0(\theta)\dot\theta \end{aligned} \tag{13}$$

where $c \in \mathbb{R}^{5\times 1}$ is obtained from (2), and $D(\theta)_{1,1}$ is the first row of the inertia matrix in (1). Moreover, since ξ_1 is just the linearized position of the hip, which was used to parameterize time (8), we can write $y_d(\tau(\theta)) = y_d(\xi_1)$.

Due to the fact that we considered linear output functions, from (2)-(4) we can write $y_a(\theta) = H\theta$ for $H \in \mathbb{R}^{4\times 5}$ with full row rank. Therefore, defining

$$\begin{aligned} \Phi(\xi_1) &= \begin{bmatrix} c \\ H \end{bmatrix}^{-1} \begin{pmatrix} \xi_1 \\ y_d(\xi_1) \end{pmatrix} \\ \Psi(\xi_1) &= \begin{bmatrix} \gamma_0(\Phi(\xi_1)) \\ H - \frac{\partial y_d(\xi_1)}{\partial \xi_1} c \end{bmatrix}^{-1} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{aligned} \tag{14}$$

it follows that for $\theta = \Phi(\xi_1)$ and $\dot\theta = \Psi(\xi_1)\xi_2$ that $(\theta,\dot\theta) \in \mathbf{Z}_\alpha$. Finally, the zero dynamics evolve according to the ODE:

$$\begin{aligned} \dot\xi_1 &= \kappa_1(\xi_1)\xi_2 & \kappa_1(\xi_1) &:= c\Psi(\xi_1) \\ \dot\xi_2 &= \kappa_2(\xi_1) & \kappa_2(\xi_1) &:= \left.\frac{\partial V(\theta)}{\partial \theta_{sf}}\right|_{\theta=\Phi(\xi_1)} \end{aligned} \tag{15}$$

with V the potential energy of the robot obtained from (1).

IV. Walking on Rough Terrain

This section will introduce the methods adopted to achieve Hybrid Zero Dynamics on rough terrain. On flat ground, the zero dynamics surface ($\mathbf{Z}_\alpha$) derived from the CWF (α) is respected i.e., the robot will exhibit Hybrid Zero Dynamics. When the non-stance foot hits the ground, we say that the robot strikes the guard, S. But, on an uneven terrain, the robot could strike a different guard, S^{int}, altogether, i.e., when the non-stance foot hits the ground at a height $h_R \neq 0$, then the resulting post-impact state, $x^+ = (\theta^+, \dot{\theta}^+)$, may not be on the same surface. This calls for defining a new set of CWF in such a way that the post-impact state of the robot resides on the resulting new zero dynamics surface (call it the intermediate zero dynamics surface, $\mathbf{Z}_\alpha{}^{int}$). This effectively reduces the problem to finding the new CWF, the resulting intermediate zero dynamics surface for which contains the post-impact state. Given the post-impact state obtained due to the non-zero height (h_R) of non-stance foot, x^+, we find the new CWF by computing the new set of parameters for the next step, α^{int}.

Extended Canonical Walking Function (ECWF). Given the canonical walking function in (6), we have the robot outputs and their derivatives:

$$y_a(\theta^+) = H\theta^+,\ dy_a(\dot{q}^+) = \frac{H\dot{\theta}^+}{\dot{\xi}_1^+}, \tag{16}$$

where $\dot{\xi}_1^+ = c\dot{\theta}^+$ (from (13)) is the post-impact hip velocity and dy_a is effectively the derivative of the outputs divided by the post-impact hip velocity. The reason for dividing by the hip velocity is to make the outputs independent of the hip velocity and based on the fact that if HZD is respected for one hip velocity on the guard ($S \cap \mathbf{Z}_\alpha$), then it must be respected for all hip velocities on the same guard ($S \cap \mathbf{Z}_\alpha$). This is proved in [12] for the case of underactuated walking.

We can design α^{int} in such a way that it brings the state of the robot back to the zero dynamics surface for flat ground walking. To be more precise, if the next step were to occur on flat ground, then the intermediate surface, $\mathbf{Z}_\alpha{}^{int}$, will lead the state of the robot back to the original surface $\mathbf{Z}_\alpha$. In other words, the construction of α^{int} will allow the guard of $\mathbf{Z}_\alpha{}^{int}$ to intersect with the guard of $\mathbf{Z}_\alpha$.

The end of the step for flat ground walking, $x^- = (\theta^-, \dot{\theta}^-)$, can be computed from, $x^- = (\theta(\vartheta(\alpha)), \dot{\theta}(\vartheta(\alpha)))$. The corresponding outputs and their derivatives are:

$$y_a(\theta^-) = H\theta^-,\ dy_a(\dot{\theta}^-) = \frac{H\dot{\theta}^-}{\dot{\xi}_1^-}, \tag{17}$$

with $\dot{\xi}_1^- = c\dot{\theta}^-$ being the hip velocity at the end of the step (again, the end of this intermediate step is assumed to be on flat ground).

Since the desired outputs should equal the outputs of the robot to ensure hybrid invariance, the goal will be to find α^{int} for which the following is satisfied:

$$\begin{aligned} y_a(\theta^+) &= y_d(\tau(\theta^+), \alpha^{int}),\ dy_a(\dot{\theta}^+) = \frac{\partial y_d(\tau(\theta^+), \alpha^{int})}{\partial \xi_1^+}, \\ y_a(\theta^-) &= y_d(\tau(\theta^-), \alpha^{int}),\ dy_a(\dot{\theta}^+) = \frac{\partial y_d(\tau(\theta^-), \alpha^{int})}{\partial \xi_1^-}, \end{aligned} \tag{18}$$

where τ is the time parameterized by hip position given in (8).

Since there are four outputs, and $y_a(\theta) \in \mathbb{R}^4$, (18) has 16 equations with the unknown being $\alpha^{int} \in \mathbb{R}^{21}$. Due the way the CWF was constructed, finding a solution (or one of the solutions) to the 16 nonlinear complex equations will be time consuming and may not be guaranteed. This motivates introducing a new function called the *Extended Canonical Walking Function* (ECWF):

$$\begin{aligned} y_{He}(t,\alpha) = e^{-\alpha_4 t}(\alpha_1 \cos(\alpha_2 t) + \alpha_3 \sin(\alpha_2 t)) + \alpha_5 \cos(\alpha_6 t) + \\ \frac{2\alpha_5\alpha_4\alpha_2}{\alpha_2^2 + \alpha_4^2 - \alpha_6^2} \sin(\alpha_6 t) + \alpha_7, \end{aligned} \tag{19}$$

which has seven parameters and four of the parameters can be written in vector form, $\alpha_v = [\alpha_1, \alpha_3, \alpha_5, \alpha_7]^T$ which are linear in the expression and can be separated out in the following manner:

$$\begin{aligned} LTV(t,\alpha) &= \begin{bmatrix} e^{-\alpha_4 t}\cos(\alpha_2 t), \\ e^{-\alpha_4 t}\sin(\alpha_2 t), \\ \cos(\alpha_6 t) + \frac{2\alpha_4\alpha_6}{\alpha_2^2+\alpha_4^2-\alpha_6^2}\sin(\alpha_6 t), \\ 1 \end{bmatrix}, \\ y_{He}(t,\alpha) &= LTV^T(t,\alpha)\alpha_v, \end{aligned} \tag{20}$$

where LTV is the linear transformation vector containing the remaining expression which becomes a constant vector by keeping $\alpha_2, \alpha_4, \alpha_6$ constant. Similarly, we can define the partial derivative of the desired extended outputs:

$$\begin{aligned} LTV_d(t,\alpha) &= \begin{bmatrix} -\alpha_4 e^{-\alpha_4 t}\cos(\alpha_2 t) - \alpha_2 e^{-\alpha_4 t}\sin(\alpha_2 t), \\ -\alpha_4 e^{-\alpha_4 t}\sin(\alpha_2 t) + \alpha_2 e^{-\alpha_4 t}\cos(\alpha_2 t), \\ -\alpha_6 \sin(\alpha_6 t) + \frac{2\alpha_4\alpha_6^2}{\alpha_2^2+\alpha_4^2-\alpha_6^2}\cos(\alpha_6 t), \\ 0 \end{bmatrix}, \\ dy_{He}(t,\alpha) &= \frac{\partial y_{He}(t,\alpha)}{\partial \xi_1} = LTV_d^T(t,\alpha)\frac{\partial t}{\partial \xi_1}\alpha_v. \end{aligned} \tag{21}$$

LTV_d is the time derivative of LTV; and from (8) and (13), it follows that $\frac{\partial t}{\partial \xi_1} = v_{hip}^{-1}$.

We can now use the ECWF, to replace (7) with:

$$\begin{aligned} y_{de}(\tau,\alpha) &= \begin{bmatrix} y_{He}(\tau,\alpha_{nsl}) \\ y_{He}(\tau,\alpha_{sk}) \\ y_{He}(\tau,\alpha_{nsk}) \\ y_{He}(\tau,\alpha_{tor}) \end{bmatrix}, \\ \frac{\partial y_{de}(\tau,\alpha)}{\partial \xi_1} &= \begin{bmatrix} dy_{He}(\tau,\alpha_{nsl}) \\ dy_{He}(\tau,\alpha_{sk}) \\ dy_{He}(\tau,\alpha_{nsk}) \\ dy_{He}(\tau,\alpha_{tor}) \end{bmatrix}, \end{aligned} \tag{22}$$

which is obtained from the new set of parameters, $\alpha = (v_{hip}, \alpha_{nsl}, \alpha_{sk}, \alpha_{nsk}, \alpha_{tor}) \in \mathbb{R}^{29}$. Accordingly, the new desired outputs (22) will replace the desired outputs in (18) resulting in 16 linear equations with 29 unknowns. Note that the control law in (10) will bear no change with y_d replaced with y_{de} and the hybrid invariance will still be valid. By keeping v_{hip} and $\alpha_2, \alpha_4, \alpha_6$ of all outputs constant, (18) will contain 16 unknowns, and can be solved for in a straightforward manner. In other words, α^{int} will contain the new set of parameters where $\alpha_2, \alpha_4, \alpha_6$'s of outputs are same as in α, and others are obtained by solving for (18) through (22). To illustrate, we will consider one of the outputs, the stance knee angle, θ_{sk}, resulting in:

$$y_{sk}(\theta^+) = \theta_{sk}^+,\ dy_{sk}(\dot{\theta}^+) = \frac{\dot{\theta}_{sk}^+}{\dot{\xi}_1^+},$$
$$y_{sk}(\theta^-) = \theta_{sk}^-,\ dy_{sk}(\dot{\theta}^-) = \frac{\dot{\theta}_{sk}^-}{\dot{\xi}_1^-}, \tag{23}$$

and (18) yields:

$$y_{sk}(\theta^+) = y_{He}(\tau(\theta^+), \alpha_{sk}^{int}),\ dy_{sk}(\dot{\theta}^+) = dy_{He}(\tau(\theta^+), \alpha_{sk}^{int}),$$
$$y_{sk}(\theta^-) = y_{He}(\tau(\theta^-), \alpha_{sk}^{int}),\ dy_{sk}(\dot{\theta}^-) = dy_{He}(\tau(\theta^-), \alpha_{sk}^{int}). \tag{24}$$

Applying (20) and (21) to (24) yields:

$$\begin{bmatrix} y_{sk}(\theta^+) \\ dy_{sk}(\dot{\theta}^+) \\ y_{sk}(\theta^-) \\ dy_{sk}(\dot{\theta}^-) \end{bmatrix} = \begin{bmatrix} LTV^T(\tau(\theta^+), \alpha_{sk}) \\ LTV_d^T(\tau(\theta^+), \alpha_{sk}) v_{hip}^{-1} \\ LTV^T(\tau(\theta^-), \alpha_{sk}) \\ LTV_d^T(\tau(\theta^-), \alpha_{sk}) v_{hip}^{-1} \end{bmatrix} \alpha_v, \tag{25}$$

which yields a unique solution for α_v, if the matrix on the R.H.S. of (25) is invertible. The invertibility is maintained by the choice of $\alpha_2, \alpha_4, \alpha_6$. This procedure remains the same for the remaining output parameters.

It is important to note that we can apply this transformation even when $h_R^+ = 0$, resulting in the set of parameters α^{int} matching with the original parameters α. This means that we can repeatedly apply this transformation irrespective of the height after every step resulting in HZD on flat ground too.

The suggested method was implemented on two kinds of terrain. The first kind was generated from a pseudo-random number generator varying between -2^0 and 2^0. The phase portrait and the outputs of the resulting walking are shown in Fig. 3. The second kind was a sinusoidal terrain with an amplitude of 2^0 and a frequency of 0.1Hz. The phase portrait and the outputs are shown in Fig. 3. Simulated videos for both random and sinusoidal terrains can be found at [2].

V. Experimental Implementation and Results

AMBER is powered by DC motors, and therefore the type of control law suggested in (10) requires knowing the model parameters of the robot. Instead, by looking at the joint angles obtained from the absolute encoders, we adopt a simple linear control law: $V_{in} = -K_p y_\alpha(\theta)$, where V_{in} is the vector of voltage inputs to the motors and K_p is the proportional constant. More details can be found in [13]. Since we adopt a different control

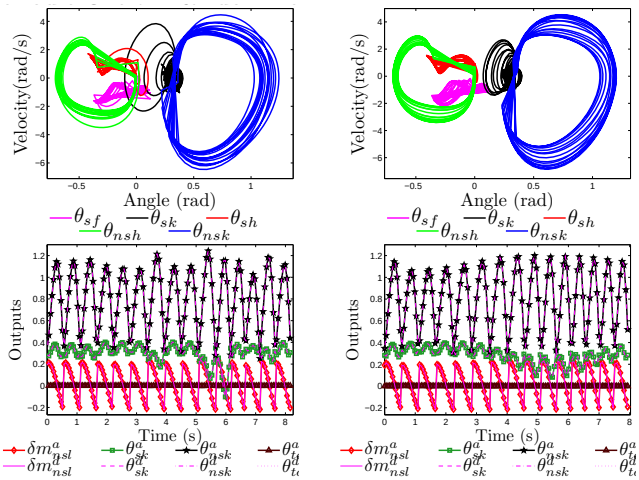

Fig. 3: Phase portraits and outputs of AMBER walking over a random terrain (left) and a sinusoidal terrain (right).

law, the walking in AMBER will not exhibit hybrid zero dynamics. Yet, we will pick the hip position of the robot, ξ_1 and find corresponding state of the robot on the surface and get the post-impact state. The reasoning we apply is that if the canonical walking functions picked give walking with AMBER, then its variants around the region specified by the hip position and the resulting intermediate ECWF should also yield walking with AMBER. Therefore, if $\xi_1^* = c\theta^*$ is the hip position when the non-stance foot strikes the uneven terrain, we scale the hip position to match with the guard on the zero dynamics surface, $k\xi_1^*$. With this hip position, the state, x^+, is reconstructed from the zero dynamics surface by using (14):

$$x^+ = \Delta\left(\begin{bmatrix} \Phi(k\xi_1^*) \\ \frac{\partial \Phi(k\xi_1^*)}{\partial \xi_1^*} v_{hip} \end{bmatrix}\right), \tag{26}$$

where Δ gives the post-impact map from the reconstructed state. Accordingly, the intermediate transition matrix is obtained as described in Section IV. It is important to note that this transition matrix is dynamically updated in AMBER after every non-stance foot strike. Tiles of AMBER walking over a rough terrain for one step are shown in Fig. 4, and Fig. 6 shows the variation of outputs of the robot on the sinusoidal terrain (see video [1]). Fig. 5 shows all the ECWF computed from the updated α^{int} after every step, and they are compared with the human data.

VI. Conclusions

This paper presented a method for dynamically updating the parameters of the canonical walking function utilized in human-inspired control in order to realize robotic walking on rough terrain. In particular, the extended canonical walking function was introduced and a novel method for determining the parameters of this function was given. Then end result was the ability to dynamically create a hybrid zero dynamics surface that returns the robot to its nominal walking gait. Applying this method to the bipedal robot AMBER experimentally resulted in walking on rough terrain.

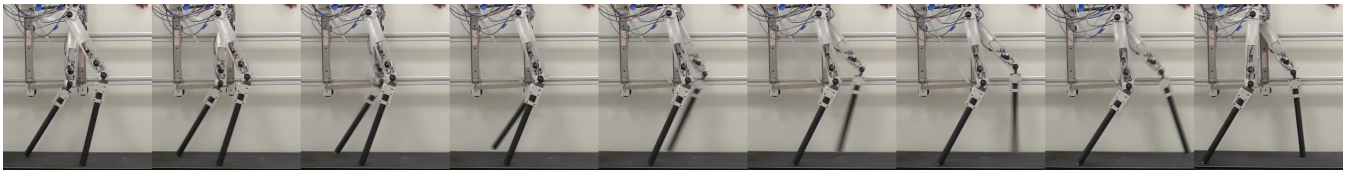

Fig. 4: Tiles showing AMBER taking one step on the rough terrain. It can be observed that the configuration of the robot at the end of the step is different from the beginning of the step. In other words, AMBER is doing the transition from its intermediate pose to its normal flat ground walking pose.

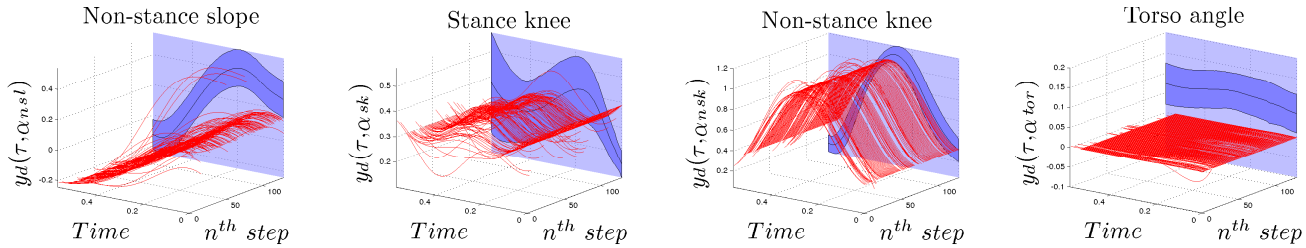

Fig. 5: Desired output functions from the intermediate transition matrix α^{int} computed at all the steps (red waveforms) and compared with the human data (shown in blue).

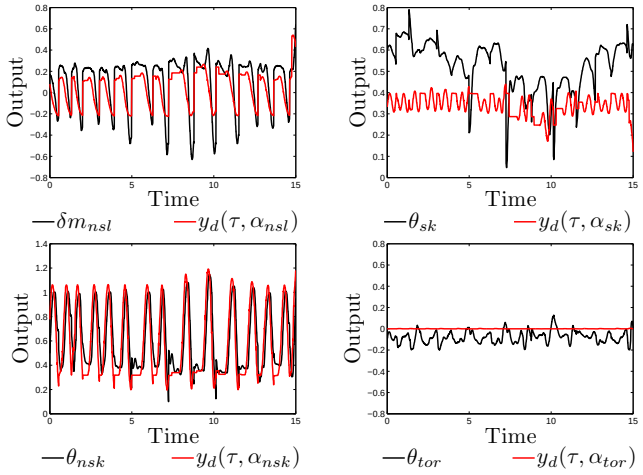

Fig. 6: Comparison of outputs of the robot with the desired outputs. Since the stance knee takes the weight of the robot, it does not match with the desired output well.

It is important to note that the walking results presented represented significant improvements in the ability of the robot to walk on rough terrain. Since the intermediate walking functions yield different steady state walking speeds with the robot, it will move backward and forward w.r.t. the treadmill which is set at constant speed. In addition, the improvement in walking is observed thereby justifying the advantage of applying intermediate motion transitions. In fact, applying the transitions makes the walking on rough terrain not just smooth, but also more human-like.

Acknowledgment

The work is supported by NSF grants CNS-0953823 and CNS-1136104, NHARP project 000512-0184-2009 and NASA contract NNX12AB58G. The authors would like to thank National Instruments and Andy Chang for being instrumental in supporting us with all the necessarily hardware and software.

References

[1] Amber walking on sinusoidal terrain. `http://youtu.be/aUiEXt8otrY`.

[2] Simulated walking on random and sinusoidal terrains. `http://youtu.be/OtW1I4kRWj8` `http://youtu.be/p7bdqmYhzcc`.

[3] A. D. Ames. First steps toward automatically generating bipedal robotic walking from human data. In *Robotic Motion and Control 2011*, volume 422 of *LNICS*, pages 89–116. Springer, 2012.

[4] A. D. Ames. First steps toward underactuated human-inspired bipedal robotic walking. In *2012 IEEE Conference on Robotics and Automation*, St. Paul, Minnesota, 2012.

[5] S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307:1082–1085, 2005.

[6] P. Holmes, R. J. Full, D. Koditschek, and J. Guckenheimer. The dynamics of legged locomotion: Models, analyses, and challenges. *SIAM Rev.*, 48:207–304, 2006.

[7] Y. Hurmuzlu. Dynamics of bipedal gait; part i: Objective functions and the contact event of a planar five-link biped. *ASME Journal of Applied Mechanics*, 60(2):331–336, 1993.

[8] A. J. Ijspeert. Central pattern generators for locomotion control in animals and robots: a review. *Neural Networks*, 21(4):642–653, 2008.

[9] T. McGeer. Passive dynamic walking. *Intl. J. of Robotics Research*, 9(2):62–82, April 1990.

[10] M. H. Raibert. Legged robots. *Communications of the ACM*, 29(6):499–514, 1986.

[11] M. Vukobratović, B. Borovac, D. Surla, and D. Stokic. *Biped Locomotion*. Springer-Verlag, Berlin, March 1990.

[12] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris. *Feedback Control of Dynamic Bipedal Robot Locomotion*. CRC Press, Boca Raton, 2007.

[13] S. Nadubettu Yadukumar, M. Pasupuleti, and A. D. Ames. From formal methods to algorithmic implementation of human inspired control on bipedal robots. In *Tenth International Workshop on the Algorithmic Foundations of Robotics (WAFR)*, Cambridge, MA, 2012.

[14] S. Nadubettu Yadukumar, M. Pasupuleti, and A. D. Ames. Human-inspired underactuated bipedal robotic walking with amber on flat-ground, up-slope and uneven terrain. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Portugal, 2012.