BIPEDAL ROBOTIC WALKING ON FLAT-GROUND, UP-SLOPE AND ROUGH

TERRAIN WITH HUMAN-INSPIRED HYBRID ZERO DYNAMICS

A Thesis

by

SHISHIR NADUBETTU YADUKUMAR

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Aaron D. Ames
Committee Members,	Shankar P. Bhattacharyya
	Mehrdad Ehsani
	Igor Zelenko
Head of Department,	Ohannes Eknoyan

December 2012

Major Subject: Electrical and Computer Engineering

ABSTRACT

The thesis shows how to achieve bipedal robotic walking on flat-ground, up-slope and rough terrain by using Human-Inspired control. We begin by considering human walking data and find outputs (or virtual constraints) that, when calculated from the human data, are described by simple functions of time (termed canonical walking functions). Formally, we construct a torque controller, through model inversion, that drives the outputs of the robot to the outputs of the human as represented by the canonical walking function; while these functions fit the human data well, they do not apriori guarantee robotic walking (due to do the physical differences between humans and robots). An optimization problem is presented that determines the best fit of the canonical walking function to the human data, while guaranteeing walking for a specific bipedal robot; in addition, constraints can be added that guarantee physically realizable walking. We consider a physical bipedal robot, AMBER, and considering the special property of the motors used in the robot, i.e., low leakage inductance, we approximate the motor model and use the formal controllers that satisfy the constraints and translate into an efficient voltage-based controller that can be directly implemented on AMBER. The end result is walking on flat-ground and up-slope which is not just human-like, but also amazingly robust. Having obtained walking on specific well defined terrains separately, rough terrain walking is achieved by dynamically changing the extended canonical walking functions (ECWF) that the robot outputs should track at every step. The state of the robot, after every non-stance foot strike, is actively sensed and the new CWF is constructed to ensure Hybrid Zero Dynamics is respected in the next step. Finally, the technique developed is tried on different terrains in simulation and in AMBER showing how the walking gait morphs depending on the terrain.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER I

INTRODUCTION

One of the key characteristics that distincts humans from other animals is the stance posture. Human evolution has made the two legs as the only tools for locomotion in humans. In other words, human evolution has made and continuously improved bipedal (two legged) walking to an extent that it can exhibit amazingly robust behaviors over a wide variety of terrains in the environment. This is one of the factors why emulating human walking has been a continued objective for a majority of bipedal walking researchers. Humans are believed to use their spinal cords to generate simple patterns, which excite the muscles in a periodic manner producing a rhythm, resulting in walking (see [16, 21]). This allows people to walk without needing to think about it. This was one of the motivating factors behind conceiving the idea of human-inspired walking in [6], and this thesis essentially shows how to achieve human-like walking with AMBER on three kinds of terrains–flat-ground, up-slope and rough terrain.

Some of the first fundamental work in bipedal robotic walking was by Marc Raibert, with the idea of achieving locomotion through the use of inverted pendulum models to create single-legged hoppers [24], and Tad Mcgeer who introduced the concept of passive walking [19] (which has also been realized in robots with efficient actuation [12]). Passive walking lead to the notion of controlled symmetries [27] which allows for low energy walking, and the Spring Loaded Inverted Pendulum (SLIP) models [15, 23] for running robots. Walking has also been looked at as a metastable process and extended to stochastic rough terrain walking [11]. In addition to these "minimalist" approaches, several methods have been proposed to directly bridge the gap between biomechanics and control theory by looking at human walking data to build models for bipedal robotic walking (see [13,

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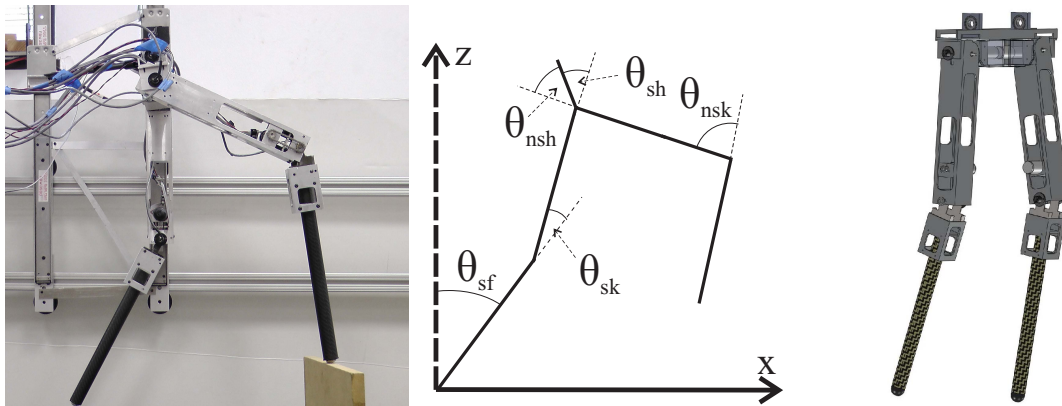
Fig. i.: The biped AMBER (left), the angle conventions (center), and the SolidWorks model of AMBER (right).

28] to name a few). Finally, combining many of the above approaches, significant strides have been made in underactuated bipedal walking (no feet) by using the idea of virtual constraints and Hybrid Zero Dynamics (HZD) [29, 18], which resulted in amazingly robust walking even on rough terrain. HZD has indeed represented bipedal walking in a very elegant fashion, but implementing a HZD controller on a biped involves the determination of the parameters of the robot through identification experiments [22] which are not only very exhaustive and time consuming but also not scalable to changes in hardware or robot structure.

Similar to [29], the technique that we adopt also considers trajectory tracking of a special set of functions called canonical walking functions (CWF) (see [6, 9, 26, 8, 7]. The CWF are obtained from running an optimization problem such that the resulting walking obtained is as close to human data as possible. One of the essential components of this thesis that stands out is that there are no low level control loops running in the robot. The walking algorithm is solely based on the control voltage that is provided by the controller purely as a function of the CWF. This is as close as we get toward emulating the central

pattern generators in humans and claim that the simple excitation of the motor actuators by these CWF based on the configuration of the robot results in walking.

The main contribution of this thesis is to design experimentally realizable control laws to achieve walking with AMBER (Fig. i) for three different kinds of motion primitives: flat-ground, up-slope and rough terrain. We begin by introducing a formal model of AMBER, including both its mechanical and electrical components in Chapter II; the fidelity of this model is essential for predicting experimental behavior through simulation. Using the human-inspired control suggested in Chapter III, steady state walking is achieved on a constant slope, γ, in simulation. This is tried in AMBER by using proportional voltage control, which justifies the control methodology suggested. Chapter IV explains this experimental implementation. Specifically, we do it by using the parameters of outputs obtained from the formal optimization problem that provably results in stable robotic walking, and define a simple voltage-based proportional (P) feedback control law on the human-inspired outputs (similar ideas have been explored for robotic manipulators [10, 17]). Since the actuators of AMBER are powered by DC motors, this naturally lends itself to simple implementation on the physical robot. The end result is that the voltage applied to the motors is directly proportional to the error between the desired and actual outputs of the robot, as represented by the canonical walking functions. This stable walking on a plane of slope γ is then extended to a terrain of varying slopes in Chapter V. In other words, formal controllers are developed for achieving provably stable walking on a rough terrain. We start with flat ground, and use the reference CWF for flat ground (it is obtained from [30]). The walking is achieved in such a way that it respects Hybrid Zero Dynamics. Hybrid Zero Dynamics (HZD) is the equivalent of zero dynamics in hybrid systems. In other words, this means 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.

CHAPTER II

BIPEDAL ROBOTIC MODEL

This chapter explains the mechanical and electrical model of AMBER. The importance of making an accurate model lies in the application of the linearizing controller on the robot to get provable walking results.

II.1.   Biped Description

AMBER is a 2D bipedal robot with five links (two calves, two thighs and a torso, see Fig. ii). AMBER is 61 cm tall with a total mass of 3.3 kg (see Fig. ii). It is made from aluminum with carbon fiber calves, powered by 4 DC motors and controlled through Lab-VIEW software by National Instruments. The robot has point feet, and is thus underactuated at the ankle. In addition, since this robot is built for only 2D walking, 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 (see Fig. ii), which allows the boom and consequently the biped, to move its hip front, back, up and down with minimum friction. The sliding mechanism is rested on a pair of parallel rails. In addition, the whole robot ambulates on a treadmill. In this manner, we can also achieve slope walking by just changing the slope (γ) of the treadmill.

Define the configuration space Q with $\theta = (\theta_{sf}, \theta_{sk}, \theta_{sh}, \theta_{nsh}, \theta_{nsk})^T \in Q$ containing the relative angles between links as shown in Fig. i. When the foot hits the ground, the stance and non-stance legs are swapped. Formally, we represent the robot as a hybrid

system (see [6, 7] for details) indexed by ground slope γ:

$$\mathcal{HC}_\gamma = (X_\gamma, U, S_\gamma, \Delta, f, g), \tag{2.1}$$

where $X_\gamma \subset TQ$ is the domain, $U \subset \mathbb{R}^4$ is the set of admissible controls, $S_\gamma \subset X_\gamma$ is the guard, and Δ is the reset map. $(f(x), g(x))$ forms a control system, i.e., $\dot{x} = f(x) + g(x)u$.
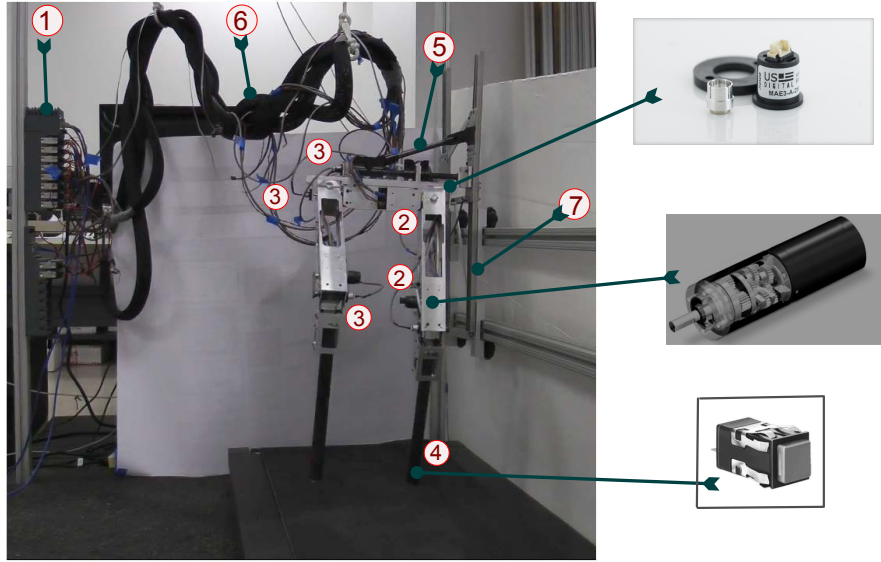
Model Parameters				
Parameter	Mass	Length	Inertia x-axis	Inertia z-axis
	g	mm	$\times 10^3\ g\,mm^2$	$\times 10^3\ g\,mm^2$
Stance calf	213.79	312.27	1967.37	119.69
Stance knee	606.15	282.37	6494.94	418.37
Torso	804.83	9.97	3730.23	3577.19
Non-stance knee	606.15	282.37	6494.94	418.37
Non-stance calf	213.79	312.37	1967.37	119.69

Fig. ii.: AMBER experimental Setup. Parts marked are (**1**): NI cRIO, (**2**): Maxon DC Motors located in the calf and the torso, (**3**): Encoders on boom and the joints, (**4**): Contact switch at the end of the foot, (**5**): Boom, (**6**): Wiring with sheath protection, (**7**): Slider for restricting the motion to the sagittal plane. The table contains the properties of each link.

Continuous Dynamics. Calculating the inertial properties of each link of the robot (Fig. i) yields the Lagrangian,

$$L(\theta, \dot{\theta}) = \frac{1}{2}\dot{\theta}^T D(\theta)\dot{\theta} - V(\theta). \tag{2.2}$$

Explicitly, this is done symbolically through the method of exponential twists (see [20]). The inertias of the motors and boom are also included. Let I_r, I_g, I_m be the rotational inertias of the rotor, gearbox, and motor, respectively. Then, $I_m = I_r r^2 + I_g$, where $r = 157$ is the gear ratio. Because r is large, I_g can be ignored. Each joint is connected to a motor through a metal chain. Therefore, the axis of rotation of the rotor has an offset w.r.t. that of the link. Using the parallel axis theorem: $I_p = I_m + m_m d_m^2$, where I_p is the motor inertia shifted to the joint axis, m_m is the mass of the rotating motor parts, and d_m is the distance between axes. Again, since $m_m = 0.011$ kg, $I_p \approx I_m$.

The boom is rigidly bolted to the sliders and thus does not rotate w.r.t. the world frame. Therefore, only the translational inertia of the boom is considered. Let m_x (resp. m_z) denote the mass of the parts in the boom-slider moving forward and backward (resp. upward and downward). Then, the resulting mass matrix is $M_{boom} \in \mathbb{R}^{6 \times 6}$ where $(M_{boom})_{1,1} = m_x$ and $(M_{boom})_{3,3} = m_z$ with all other entries equal to zero. The combined inertia matrix, D_{com}, used in the Lagrangian is

$$D_{com}(\theta) = D(\theta) + \text{diag}(0, I_{m,sk}, I_{m,sh}, I_{m,nsh}, I_{m,nsk})$$
$$+ J(\theta)^T M_{boom} J(\theta), \tag{2.3}$$

where $I_{m,sk}$, $I_{m,sh}$, $I_{m,nsh}$, $I_{m,nsk}$ represent the motor inertias of the links and $J(\theta)$ is the body Jacobian of the torso center of mass. The Euler-Lagrange equation yields a dynamic model:

$$D_{com}(\theta)\ddot{\theta} + H(\theta, \dot{\theta}) = B(\theta)u, \tag{2.4}$$

6

where the control, u, is a vector of torque inputs. Since AMBER has DC motors, we need to derive equations with voltage inputs. Since the motor inductances are small, we can realize the electromechanical system:

$$V_{in} = R_a i_a + K_\omega \omega, \tag{2.5}$$

where V_{in} is the vector of voltage inputs to the motors, i_a is the vector of currents through the motors, and R_a is the resistance matrix. Since the motors are individually controlled, R_a is a diagonal matrix. K_ω is the motor constant matrix and ω is a vector of angular velocities of the motors. Representing (2.5) in terms of currents, the applied torque is $u = K_\varphi R_a^{-1}(V_{in} - K_\omega \omega)$, with K_φ the torque constant matrix. Thus, the Euler-Lagrange equation takes the form $D_{com}(\theta)\ddot{\theta} + H(\theta, \dot{\theta}) = B(\theta)K_\varphi R_a^{-1}(V_{in} - K_\omega \omega)$.

Converting this model to first order ODEs yields the affine control system (f_v, g_v) with inputs V_{in}:

$$
\begin{aligned}
f_v(\theta, \dot{\theta}) &= \begin{bmatrix} \dot{\theta} \\ -D_{com}^{-1}(\theta)(H(\theta, \dot{\theta}) + B(\theta)K_\varphi R_a^{-1} K_\omega \omega) \end{bmatrix}, \\
g_v(\theta) &= \begin{bmatrix} \mathbf{0} \\ D_{com}^{-1}(\theta)B(\theta)K_\varphi R_a^{-1} \end{bmatrix}.
\end{aligned} \tag{2.6}
$$

Discrete Dynamics. The domain, X_γ, describes the allowable state space restricted by the guard, h_γ. For AMBER, the non-stance foot must be above a slope of γ, i.e., $h_\gamma \geq 0$:

$$X_\gamma = \left\{ (\theta, \dot{\theta}) \in TQ : h_\gamma(\theta) \geq 0 \right\},$$

The guard is just the boundary of the domain with the additional assumption that h_γ is decreasing:

$$S_\gamma = \left\{ (\theta, \dot{\theta}) \in TQ : h_\gamma(\theta) = 0 \text{ and } dh_\gamma(\theta)\dot{\theta} < 0 \right\},$$

7

with $dh_\gamma(\theta)$ the Jacobian of h_γ at θ. When the non-stance foot impacts the ground, the angular velocities change. Hence we define a reset map, $\Delta : S_\gamma \to X_\gamma$ which maps the pre-impact state to the post-impact state; see [6] for details.

CHAPTER III

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 [7] (also see [6, 8] for related results in the case of full actuation and [31] for results on other kinds of terrains). The idea will be to extended from flat-ground walking to up-slope walking and then finally to walking on rough terrain.

III.1.   Human-Inspired Functions

By considering human walking data (as described in [6]), we discover that certain outputs (or virtual constraints), computed from the human joint data, display simple behavior; this core observation will be central to the design of human-inspired controllers. With the goal of picking outputs that elucidate the underlying structure of walking through a low-dimensional representation, or "virtual model," we pick outputs that represent the human (and bipedal robot) as a compass-gait biped [12, 19] and the SLIP model [15]. In particular, the following collection of outputs yields such a representation (as illustrated in Fig. iii): the linearization of the *x*-position of the hip, p_{hip}, given by:

$$\delta p_{\text{hip}}(\theta) = L_c(-\theta_{sf}) + L_t(-\theta_{sf} - \theta_{sk}), \tag{3.1}$$

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), given by:

$$\delta m_{nsl}(\theta) = -\theta_{sf} - \theta_{sk} - \theta_{sh} + \theta_{nsh} + \frac{L_c}{L_c + L_t}\theta_{nsk}, \tag{3.2}$$

9

$$L_{nsl} = \sqrt{L_t^2 + L_c^2 - 2L_t L_c \cos(\pi - \theta_{nsk})}$$

$$\theta_{sl} = \arcsin\left(\frac{p_{hip}}{L_{sl}}\right)$$

$$\theta_{nsl} = \arctan(m_{nsl})$$

$$L_{sl} = \sqrt{L_t^2 + L_c^2 - 2L_t L_c \cos(\pi - \theta_{sk})}$$
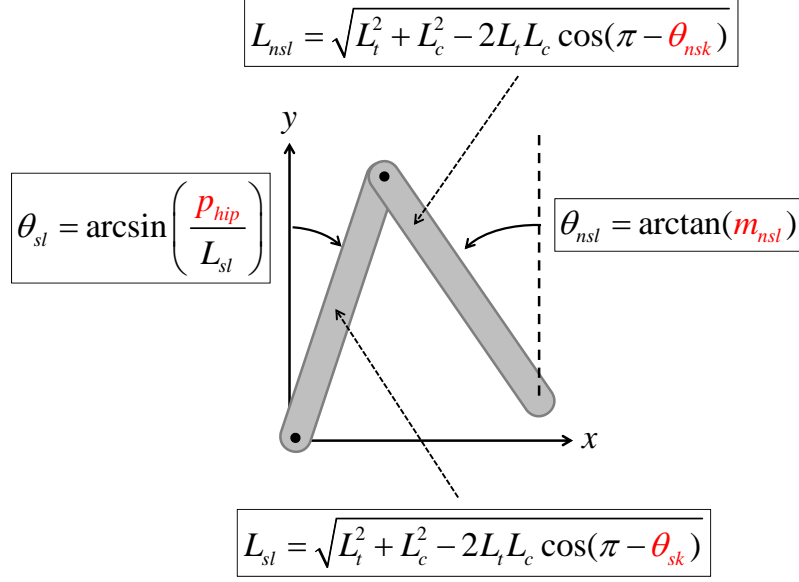
Fig. iii.: Compass gait showing the origin of the idea behind choosing the outputs

the angle of the stance knee, θ_{sk}, the angle of the non-stance knee, θ_{nsk}, the angle of the torso from vertical, $\theta_{tor}(\theta) = \theta_{sf} + \theta_{sk} + \theta_{sh}$. It is important to note that the linearized form of these outputs, rather than their original nonlinear form [6], is considered to allow for simpler implementation. Inspection of these outputs, as computed from the human data and shown in Fig. iv, reveals that they appear to display very simple behavior. In the case of the (linearized) position of the hip, it appears to essentially be a linear function of time:

$$\delta p_{\text{hip}}^d(t, v) = v_{hip} t, \tag{3.3}$$

The remaining outputs, (the non-stance slope δm_{nsl}, the stance knee θ_{sk}, the non-stance knee θ_{nsk} and the torso angle θ_{tor}) appear to act like a second order linear system. This motivated the introduction of the *canonical walking function* [6, 7]:

$$y_H(t, \alpha) = e^{-\alpha_4 t}(\alpha_1 \cos(\alpha_2 t) + \alpha_3 \sin(\alpha_2 t)) + \alpha_5. \tag{3.4}$$

which is simply the time solution to a linear mass-spring damper system, with $\alpha_4 = \zeta \omega_n$, where ζ is the damping ratio and ω_n is the natural frequency, $\alpha_2 = \omega_d$, where $\omega_d =$

(a) Linear hip position

(b) Non-stance slope

(c) Stance Knee

(d) Non-stance Knee
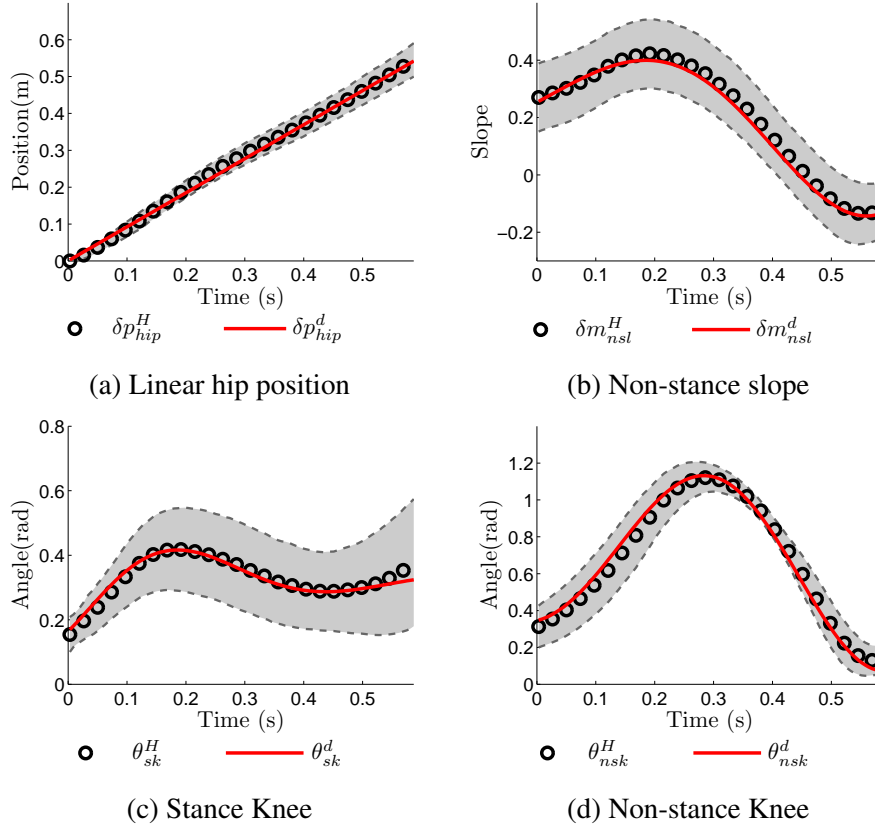
Fig. iv.: The black circles indicate the mean of the human output data (see [8]). The grey shaded area indicates the standard deviation from the mean trajectory. The red solid lines are the fits of the canonical functions to the mean human data.

$\omega_n \sqrt{1 - \zeta^2}$ is the damped natural frequency, $\alpha_1 = c_0$ and $\alpha_3 = c_1$, where c_0, c_1 are determined by the initial conditions of the system and $\alpha_5 = g$, where g is the gravity related constant. Performing a least squares fit of the human output data with these functions results in near unity correlations (see [8]); implying that for the specific outputs chosen humans appear to act like linear mass-spring-damper systems. This is an important conclusion because it illustrates the simplicity in behavior that humans display when walking.

III.2. Human-Inspired Controller

Having defined the outputs, we can construct a controller that drives four of the outputs of the robot (one output, which is the 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*: $y_a(\theta(t)) \rightarrow y_d(t, \alpha)$, with:

$$
\begin{aligned}
y_d(t, \alpha) &= [y_H(t, \alpha_{nsl}), y_H(t, \alpha_{sk}), y_H(t, \alpha_{nsk}), y_H(t, \alpha_{tor})]^T, \\
y_a(\theta) &= [\delta m_{nsl}(\theta), \theta_{sk}, \theta_{nsk}, \theta_{tor}(\theta)]^T,
\end{aligned}
\tag{3.5}
$$

where $y_H(t, \alpha_i), i \in \{nsl, sk, nsk, tor\}$ is the *CWF* (5.4) 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 (3.3), 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:

$$
\tau(\theta) = (\delta p_{hip}^R(\theta) - \delta p_{hip}^R(\theta^+))/v_{hip},
\tag{3.6}
$$

where $\delta p_{hip}^R(\theta^+)$ 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 (3.6), we define the following *human-inspired* output:

$$
y_\alpha(\theta) = y_a(\theta) - y_d(\tau(\theta), \alpha).
\tag{3.7}
$$

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, the outputs

have (vector) relative degree 2 and we can define the following torque controller:

$$u_{(\alpha,\varepsilon)}(\theta,\dot{\theta}) = \tag{3.8}$$

$$-A^{-1}(\theta,\dot{\theta})\left(L_f^2 y_\alpha(\theta,\dot{\theta}) + 2\varepsilon L_f y_\alpha(\theta,\dot{\theta}) + \varepsilon^2 y_\alpha(\theta)\right).$$

In other words, we can apply feedback linearization to obtain the linear system on the human-inspired output: $\ddot{y}_\alpha = -2\varepsilon\dot{y}_\alpha - \varepsilon^2 y_\alpha$. This system is exponentially stable, implying that for $\varepsilon > 0$ the control law $u_{(\alpha,\varepsilon)}$ drives $y_\alpha \to 0$ as $t \to \infty$ (see [25]).

III.3.   Human-Inspired Hybrid Zero Dynamics (HZD)

The goal of the human-inspired controller (3.8) was to drive the outputs of the robot to the outputs of the human: $y_a \to y_d$. And as $t \to \infty$, $y_a \to y_d$. But, 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 and represent the zero dynamics surface.

Problem Statement. Since the robot is underactuated, some of the variables in the state will not be controllable; and since AMBER is a mechanical system, the number of variables is 2. 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*:

$$\mathbf{Z}_\alpha = \{(\theta,\dot{\theta}) \in TQ : y_\alpha(\theta) = \mathbf{0}, \ L_f y_\alpha(\theta,\dot{\theta}) = \mathbf{0}\} \tag{3.9}$$

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* 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:

$$\alpha^* = \operatorname*{argmin}_{\alpha \in \mathbb{R}^{21}} \mathrm{Cost}_{\mathrm{HD}}(\alpha) \tag{3.10}$$

$$\text{s.t} \quad \Delta(S \cap \mathbf{Z}_\alpha) \subset \mathbf{Z}_\alpha \tag{HZD}$$

with $\mathrm{Cost}_{\mathrm{HD}}$ 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 [30]). 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, $\vartheta(\alpha)$, and it can be found in [7].

Zero Dynamics. With the control law ensuring HZD, we can explicitly construct the zero dynamics surface. In particular, we utilize the constructions in [29], reframed in the context of canonical human walking functions.

$$\begin{aligned} \xi_1 &= \delta p_{\mathrm{hip}}^R(\theta) =: c\theta \\ \xi_2 &= D(\theta)_{1,1}\dot{\theta} =: \gamma_0(\theta)\dot{\theta} \end{aligned} \tag{3.11}$$

where $c \in \mathbb{R}^{5 \times 1}$ is obtained from (3.1), and $D(\theta)_{1,1}$ is the first entry of the inertia matrix in (2.2). Moreover, since ξ_1 is just the linearized position of the hip, which was used to

parameterize time (3.6), we can write $y_d(\tau(\theta)) = y_d(\xi_1)$.

Due to the fact that we considered linear output functions, from (3.1)-(3.2) we can write $y_a(\theta) = H\theta$ for $H \in \mathbb{R}^{4 \times 5}$ with full row rank. Therefore, picking the coordinates

$$\eta_1 = y_a(\theta) = H\theta, \quad \eta_2 = L_f y_a(\theta, \dot{\theta}) = H\dot{\theta} \tag{3.12}$$

and defining

$$
\begin{aligned}
\Phi(\xi_1) &= \begin{bmatrix} c \\ H \end{bmatrix}^{-1} \begin{pmatrix} \xi_1 \\ y_d(\xi_1) \end{pmatrix} \\
\Psi(\xi_1) &= \begin{bmatrix} \gamma_0(\Phi(\xi_1)) \\ H - \frac{\partial y_d(\xi_1)}{\partial \xi_1} c \end{bmatrix}^{-1} \begin{pmatrix} 1 \\ 0 \end{pmatrix}
\end{aligned} \tag{3.13}
$$

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:

$$
\begin{aligned}
\dot{\xi}_1 &= \kappa_1(\xi_1)\xi_2 & \kappa_1(\xi_1) &:= c\Psi(\xi_1) \\
\dot{\xi}_2 &= \kappa_2(\xi_1) & \kappa_2(\xi_1) &:= \left. \frac{\partial V(\theta)}{\partial \theta_{sf}} \right|_{\theta = \Phi(\xi_1)}
\end{aligned} \tag{3.14}
$$

with V the potential energy of the robot obtained from (2.2).

III.4.   Optimization Theorem

We now present the main theorem (originally introduced in [6, 7, 8]) that will be used to generate the control parameters and experimentally implemented on AMBER to obtain robotic walking. From the mean human walking data, we obtain discrete times, $t^H[k]$, and discrete values for the human output data, $y_i^H[k]$ and the canonical walking functions, $y_i^d(t, \alpha_i)$ for $i \in \text{Output} = \{hip, nsl, sk, nsk, tor\}$; for example, $y_{nsl}^H[k] = y_H(kT, \alpha_{nsl})$, where T is the discrete time interval and $k \in \mathbb{Z}$. We can now define the following human-data cost

function:

$$\text{Cost}_{\text{HD}}(\alpha) = \sum_{k=1}^{K} \sum_{i \in \text{Output}} \left(y_i^H[k] - y_i^d(t^H[k], \alpha_i) \right)^2 \qquad (3.15)$$

which is simply the sum of squared residuals. To determine the parameters for the human walking functions, we need only solve the optimization problem:

$$\alpha^* = \underset{\alpha \in \mathbb{R}^{21}}{\text{argmin}} \ \text{Cost}_{\text{HD}}(\alpha) \qquad (3.16)$$

which yields the least squares fit of the mean human output data with the canonical walking functions. While this provides an α^* that yields a good fit of the human data (see Fig. iv), these parameters will not result in robotic walking due to the differences between the robot and a human. Therefore, the goal is to determine these parameters which provide the best fit of the human data while simultaneously guaranteeing stable robotic walking for AMBER [1] This motivates the following theorem:

---

[1]It is important to note that [28] also presents an optimization problem that results in the least squares fit of human walking data subject to constraints that ensure HZD. Yet the theorem presented here is a substantial departure from the results in [28] in several important ways: [28] considers human joint angles, while we consider output functions, [28] fits high degree (9th order) polynomials to this data to create virtual constraints while we utilize the canonical walking function (which is nonlinear, and has far fewer parameters), [28] defines a configuration at the end of the step a priori and uses these to constrain the parameters of the outputs to ensure HZD while we define the point in terms of the parameters and allow it to change with the parameters as a result. All of these considerations require the use of different methods and theory and, fundamentally, changes the walking achieved.

Theorem 1 *The parameters α^* solving the constrained optimization problem:*

$$\alpha^* = \underset{\alpha \in \mathbb{R}^{21}}{\text{argmin}} \ \text{Cost}_{\text{HD}}(\alpha) \qquad (3.17)$$

$$\text{s.t} \quad y(\vartheta(\alpha)) = \mathbf{0} \qquad (C1)$$

$$dy_\alpha(\Delta_\theta \vartheta(\alpha))\Delta_{\dot\theta}(\vartheta(\alpha))\dot\vartheta(\alpha) = \mathbf{0} \qquad (C2)$$

$$dh(\vartheta(\alpha))\dot\vartheta(\alpha) < 0 \qquad (C3)$$

$$\mathscr{D}_{\mathbf{Z}}(\vartheta(\alpha)) < 0 \qquad (C4)$$

$$0 < \Delta_{\mathbf{Z}}(\vartheta(\alpha)) < 1 \qquad (C5)$$

yield Hybrid Zero Dynamics: $\Delta(S \cap \mathbf{Z}_{\alpha^*}) \subset \mathbf{Z}_{\alpha^*}$. Moreover, there exists an $\hat\varepsilon > 0$ such that for all $\varepsilon > \hat\varepsilon$ the hybrid system $\mathscr{H}^{(\alpha^*, \varepsilon)}$, obtained by applying the control law (3.8) to the hybrid control system (2.1), has a stable periodic orbit with fixed point $(\theta^*, \dot\theta^*) \in S \cap \mathbf{Z}_{\alpha^*}$ that can be explicitly computed.

It is not possible to introduce all of the elements utilized in Theorem 1 since they are not a part of the thesis, but a detailed explanation can be found in [7]. Of particular importance is the point $(\vartheta(\alpha), \dot\vartheta(\alpha)) \in S \cap \mathbf{Z}_\alpha$ on the intersection of the zero dynamics surface and the guard that can be explicitly computed in terms of the parameters α (this point will later be used in additional constraints that will yield physically realizable walking). In other words, the configuration and velocities at the beginning and end of a step can change with the parameters allowing for a better translation of the outputs to robots which have different mass and length parameters from humans. In addition, $(\vartheta(\alpha), \dot\vartheta(\alpha))$ allows for the constraints in the optimization to be framed only in terms of the parameters, α. For these constraints, (C1) and (C2) (when coupled with the way $(\vartheta(\alpha), \dot\vartheta(\alpha))$ is computed from the outputs) ensure that the state of the robot is restricted to the zero dynamics surface even through the impacts. (C3) ensures that the the non-stance foot intersects the guard

transversally. (C4) and (C5) imply the existence and stability of a periodic orbit in the Hybrid Zero Dynamics. In particular, $\mathscr{D}_{\mathbf{Z}}(\vartheta(\alpha))$, which is a function of the energy contained in the zero dynamics, determines the existence of a step (which in turn determines the existence of the periodic orbit). $\Delta_{\mathbf{Z}}(\vartheta(\alpha))$, gives the post impact velocity in the zero dynamics from pre-impact velocity, and therefore (C5) indicates the stability of the resulting periodic orbit. Finally, following from the results in [29], the existence and stability of a periodic orbit in the Hybrid Zero Dynamics implies the stability of a periodic orbit in the full-order dynamics for sufficiently large ε, i.e., the end result is a stable walking gait.

Additional Constraints. The walking that we achieve using Theorem 1 should be physically realizable, which necessitates the additional constraints that ensure that the resulting control parameters will experimentally result in walking with AMBER:

(C6) *Foot scuff prevention*: The height of the swing foot at any point of time, must be such that it is greater than a quadratic polynomial, $h_R(\theta) > P(\theta)$, where $P(\theta) = ax_f(\theta)^2 + bx_f(\theta) + c$ with $x_f(\theta)$ being the horizontal position of the swing foot w.r.t. the stance foot and

$$a = -\frac{4h_{max}}{\text{SL}(\alpha)^2}, \quad b = \frac{4h_{max}\text{SL}(\alpha)}{\text{SL}(\alpha)^2}, \quad c = -\frac{4h_{max}x_f(\vartheta(\alpha))x_f(\Delta(\vartheta(\alpha)))}{\text{SL}(\alpha)^2},$$

where $\text{SL}(\alpha) = x_f(\vartheta(\alpha)) - x_f(\Delta(\vartheta(\alpha)))$ is the step length of the robot, computed from α through $\vartheta(\alpha)$. These constants, therefore, can be adjusted based on the required maximum stance foot height, h_{max}, and step length, $\text{SL}(\alpha)$.

(C7) *Peak torque*: The maximum torque delivered by the motors is limited. Therefore, the peak torque during a walking gait must be: $max(u_{(\alpha,\varepsilon)}(\theta,\dot{\theta})) < u_{max}$. Here $u_{(\alpha,\varepsilon)}$ is dependent on the parameters α and ε, given in (3.8) and u_{max} is the maximum torque of the motors (for AMBER, $u_{max} = 2$Nm).

(C8) *Hip-Velocity*: The desired hip velocity of the biped must be within reasonable

limits. Therefore, we introduce the constraint: $v_{min} < v_{hip} < v_{max}$. For AMBER, $v_{min} = 0.1\mathrm{m/s}$, $v_{max} = 0.6\mathrm{m/s}$.

(C9) *Angular velocities of joints*: The maximum angular velocities with which the joints can turn are limited by the maximum angular velocities of the motors. The motors used in AMBER have a maximum angular velocity of $6.5rad/s$.

CHAPTER IV

FLAT-GROUND AND UP-SLOPE WALKING IN AMBER

The control law proposed in the previous section requires us to linearize the dynamics of AMBER through model inversion, which requires exact values of masses, inertias and dimensions of the robot. This is not only complex to implement but realizing the control law (3.8) on AMBER could potentially consume both time and resources, and achieving walking may still not be guaranteed due to a potentially inexact model. We, therefore, take the different approach by arguing that due to the "correct" choice of output functions—and specifically the human-inspired outputs—it is possible to obtain walking through simple controllers that are easy to implement and inherently more robust. Specifically, we present a proportional voltage controller on the human-inspired outputs, and demonstrate through simulation that robotic walking is obtained on AMBER. The simplicity of this controller implies that it can be efficiently implemented in software, and the details of this implementation are given. Finally, experimental results are presented showing that bipedal robotic walking is obtained with AMBER that is both efficient and robust.

IV.1.   Human-Inspired Voltage Control

Even if walking is obtained formally through input/output linearization, i.e., model inversion, the controllers are often implemented through PD control on the torque (see, for example, [22]). Since AMBER is not equipped with torque sensors, we sought an alternative method for feedback control implementation. Because AMBER is powered by DC motors, the natural input to consider is voltage, V_{in}, which indirectly affects the torques acting on the joints.

    Let V_{sk} be the voltage input to the stance knee motor, V_{nsk} be the voltage input to the

non-stance knee motor, V_{nsl} be the voltage input to the non-stance hip motor and finally V_{tor} be the voltage input to the stance hip of the motor. Then the following proportional control law is defined:

$$
\begin{aligned}
V_{nsl}(t,\theta) &= -K_{p,nsl}(\theta_{sk} - \theta_{sk}^d(t,\alpha_{sk})), & (4.1) \\
V_{sk}(t,\theta) &= -K_{p,sk}(\theta_{nsk} - \theta_{nsk}^d(t,\alpha_{nsk})), \\
V_{nsk}(t,\theta) &= -K_{p,nsk}(\delta m_{nsl}^R(\theta) - m_{nsl}^d(t,\alpha_{nsl})), \\
V_{tor}(t,\theta) &= -K_{p,tor}(\theta_{tor}^R(\theta) - \theta_{tor}^d(t,\alpha_{tor})),
\end{aligned}
$$

where $K_{p,nsl}, K_{p,sk}, K_{p,nsk}, K_{p,tor}$ are the diagonal entries of the matrix K_p. The non-diagonal entries are 0. If time is parameterised w.r.t hip position, then the variable t in (4.1) will be replaced with $\tau(\theta)$ from (3.6). Therefore, (4.1) can be written in this form:

$$
V_{in} = \begin{bmatrix}
V_{nsl}(\tau(\theta),\theta) \\
V_{sk}(\tau(\theta),\theta) \\
V_{nsk}(\tau(\theta),\theta) \\
V_{tor}(\tau(\theta),\theta)
\end{bmatrix} = -K_p y(\theta), \tag{4.2}
$$

It can be seen that the control law (proportional control) solely depends on the generalized coordinates of robot (angles), θ, and not on the angular velocities. This marks a drastic change from the traditional ways of computing control. Evidently, and importantly, this avoids computation of angular velocities of the joints, which would have been computationally expensive and inaccurate.

It is important to note that the P voltage control law (4.2) is equivalent to a PD torque controller, where the derivative (D) constant is specified by the properties of the motor. Given the voltage input, the torque vector acting on the links is obtained by:

$$
u(\theta) = -K_\varphi R_a^{-1} K_p y(\theta) - K_\varphi R_a^{-1} K_\omega \omega. \tag{4.3}
$$

We can see that P-control with voltage as the input actually leads to P-control with torque as an input with an added velocity term. Hence, the control being applied is not markedly different from the conventional torque control methods adopted in literature (see [14]).

IV.2. Simulation and Experimental Results

By applying the new voltage control law on each of the motors we get the following results.

IV.2.1. Flat-ground walking

For implementing flat-ground walking, we numerically solve the optimization algorithm in Theorem 1 subject to the additional constraints **(C6)**-**(C9)** to obtain the optimized parameter α^* corresponding to zero slope $\gamma = 0$. By applying the control law given in (4.2), parameterized by α^*, to the control system $(f_v(x), g_v(x))$ yields walking on flat ground. Periodic orbit of the walking gait obtained is shown in Fig. v(a), tiles of the walking obtained in simulation can be seen in Fig. v(e). In Fig. vi(a)-(d) the outputs of the robot are compared with the mean human data. Again, we can deduce that the output functions are amazingly close to that of the humans. Figs. v(c),v(d) show the variation of torques and voltages respectively with time resulting from the voltage being applied to the motors. Being within the limits of 2 N-m further confirms our choice of canonical walking functions and the torques do not exceed the limits even after changing the control law to the voltage method.

It was mentioned in Section III that if the human outputs of the robot match with the human output data, then we can say that the walking obtained is human-like. Figs. vi(a)-vi(d) show the comparison between the outputs and the human mean data. Video link to the simulation can be found at [2].

IV.2.2. Walking on a slope

Walking on a slope is obtained by solving the optimization problem for a non-zero value of the slope angle, γ. We considered a particular value of $\gamma = 2^0$, which is a reasonable slope for a robot of the size and power capabilities of AMBER. Again, we get α_γ^*, the solution to the optimization problem and apply the voltage control law resulting in stable up-slope walking. The periodic orbit for the walking obtained is shown in Fig. vii. A comparison between the human output data and simulated robot output data is shown in Fig. viii. Considering the fact that the slope 2^0 is small, it is fair to compare the simulated outputs with flat ground walking data from humans. Fig. ix shows the comparison between the simulated and experimental tiles for both flat-ground and up-slope walking. Fig. x and Fig. xi show how well the outputs of the robot track the actual outputs obtained from the CWF for flat-ground and up-slope walking respectively.
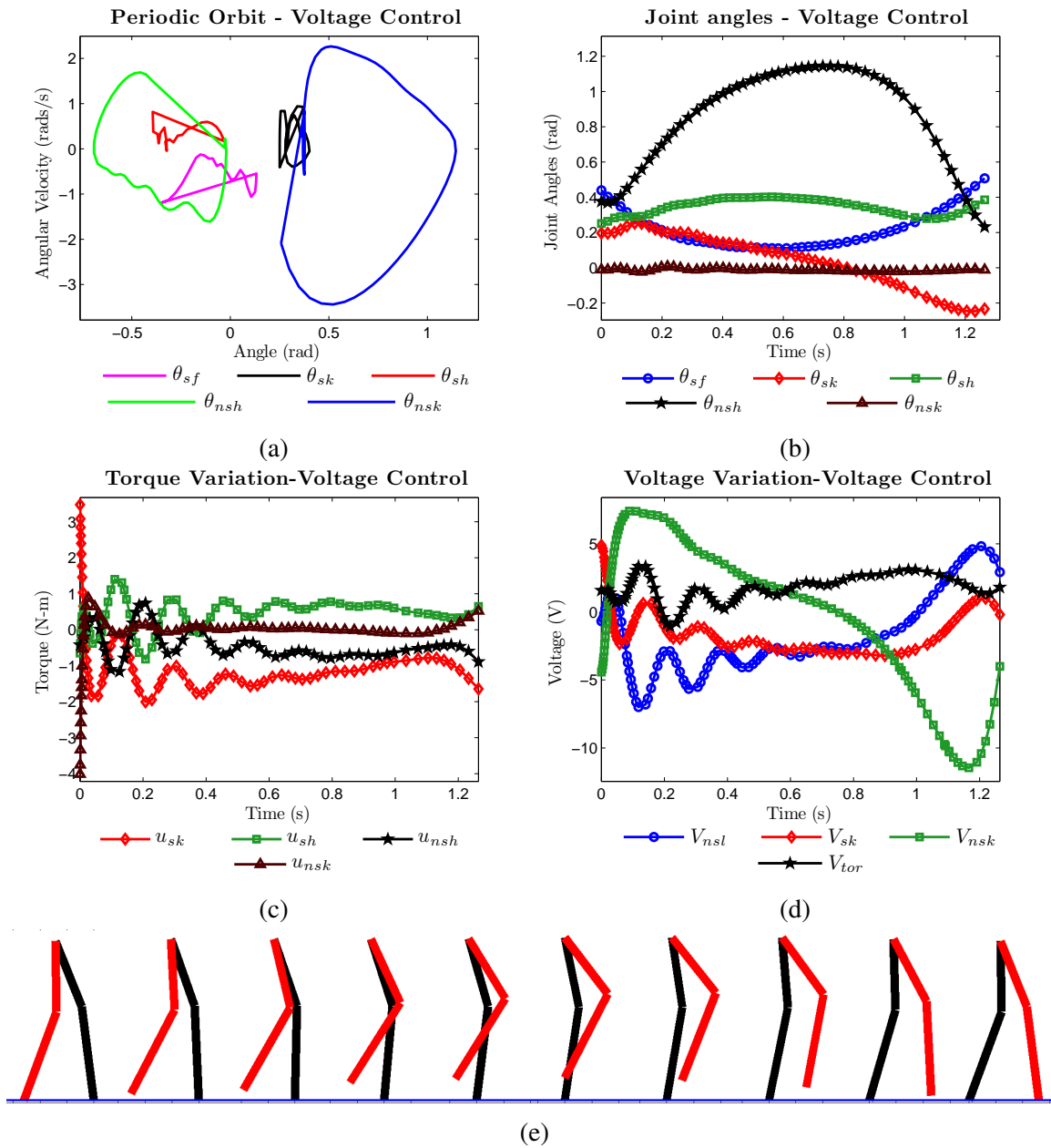
Fig. v.: Phase portrait of the walking gait is shown in (a), joint angles are shown in (b), variation of torques acting with time are shown in (c), variation of voltages with time are shown in (d) and the walking tile is shown in (e).
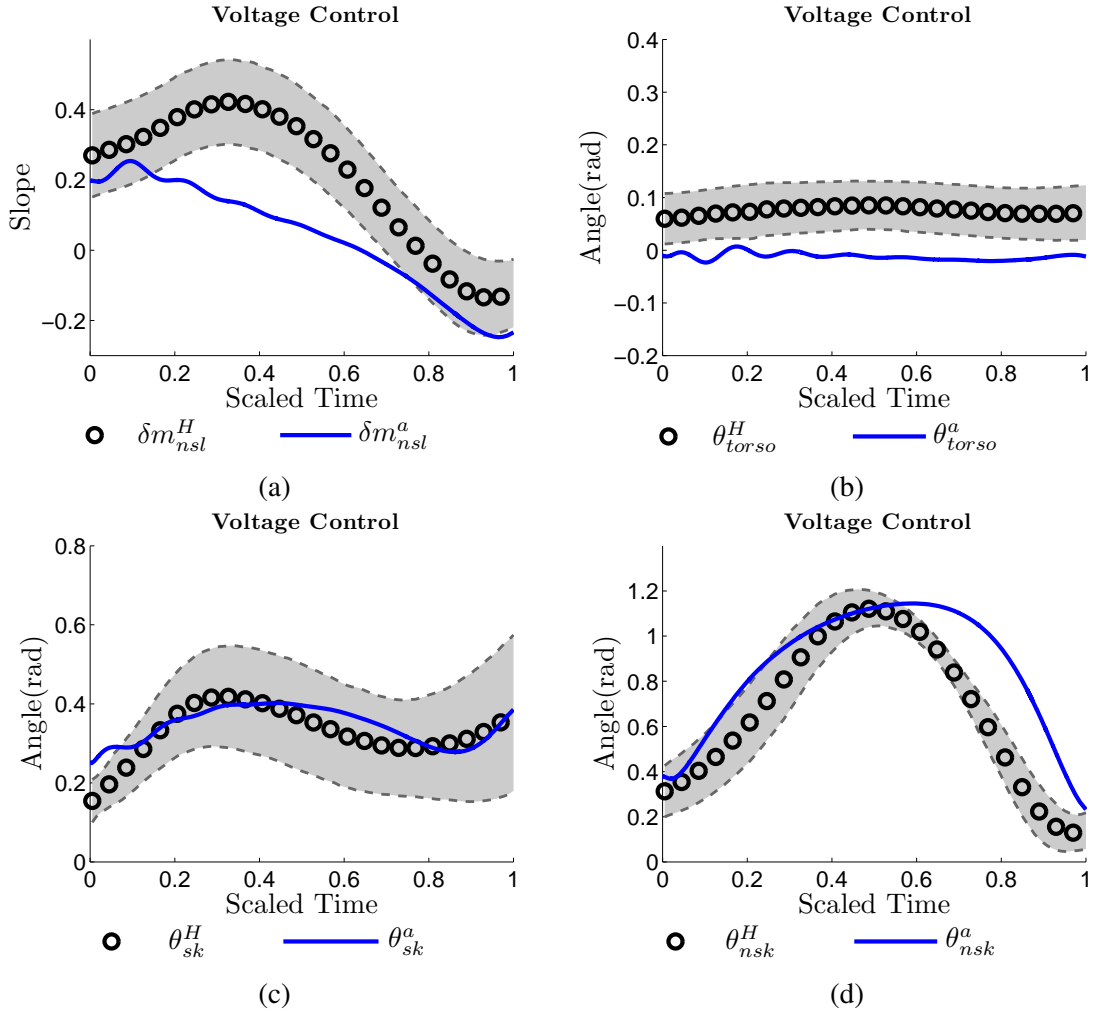
Fig. vi.: Comparisons between human data and the human outputs of the robot for flat-ground walking are shown here.
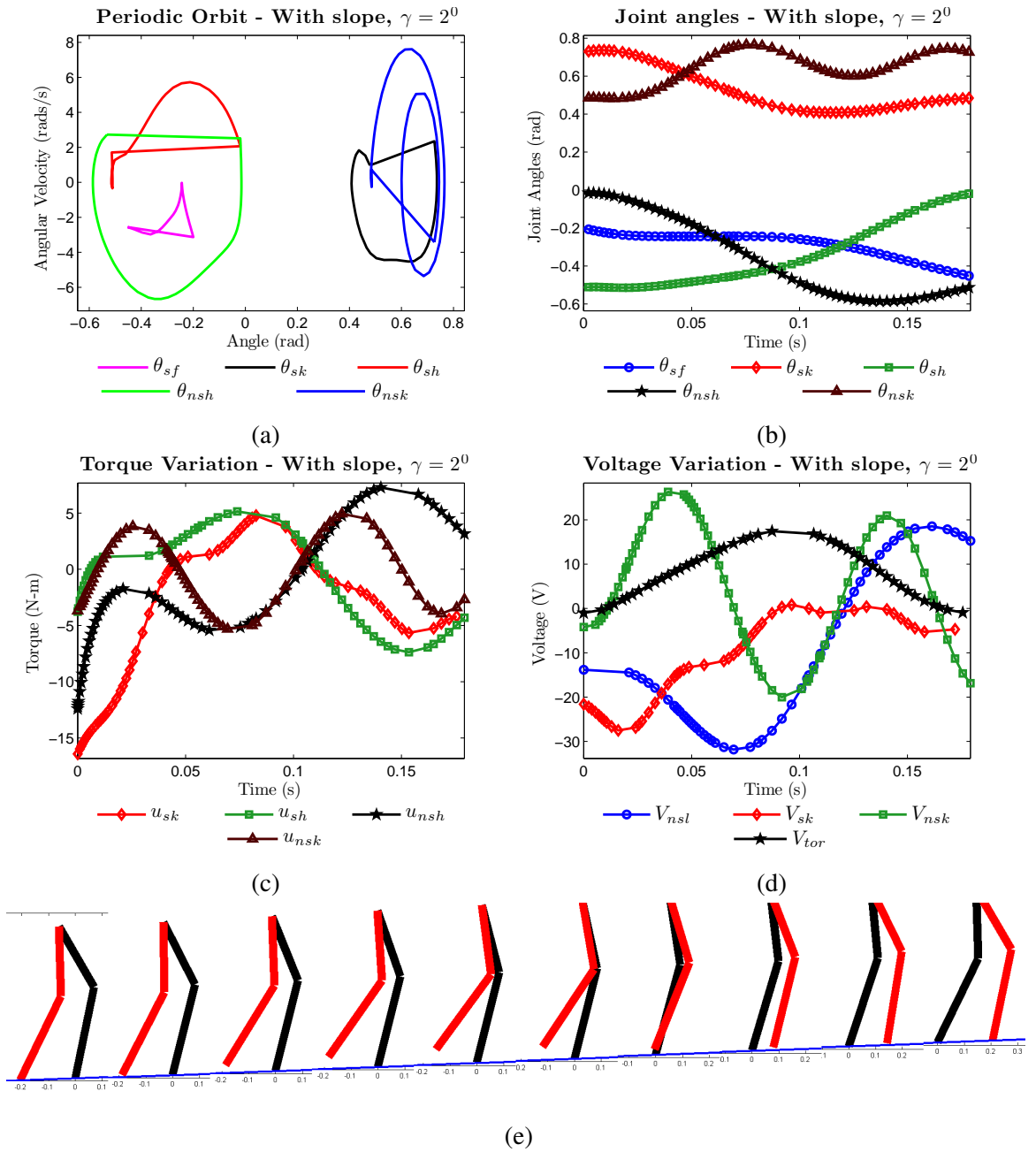
Fig. vii.: Phase portrait of the walking gait is shown in (a), joint angles are shown in (b), variation of torques acting with time are shown in (c), variation of voltages with time are shown in (d) and the walking tile is shown in (e).
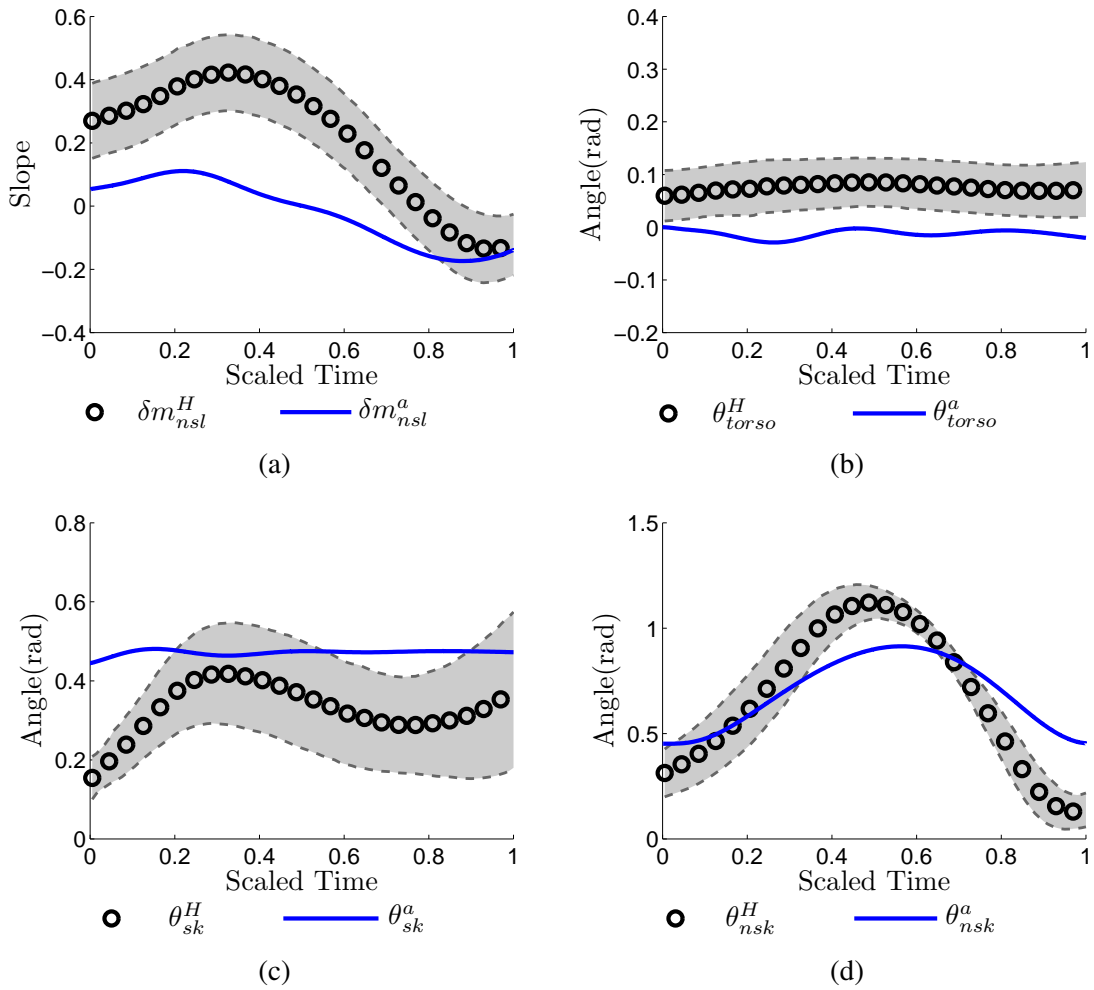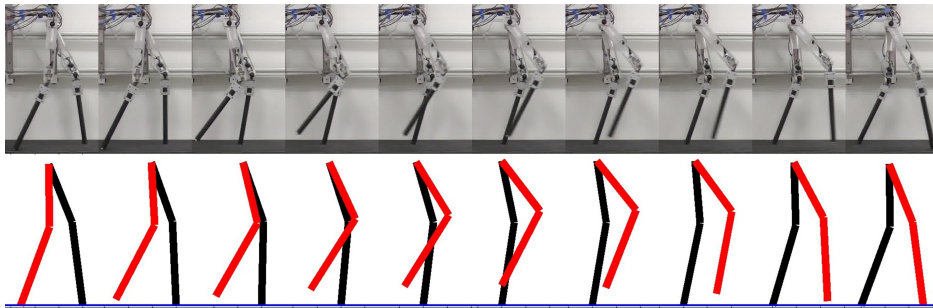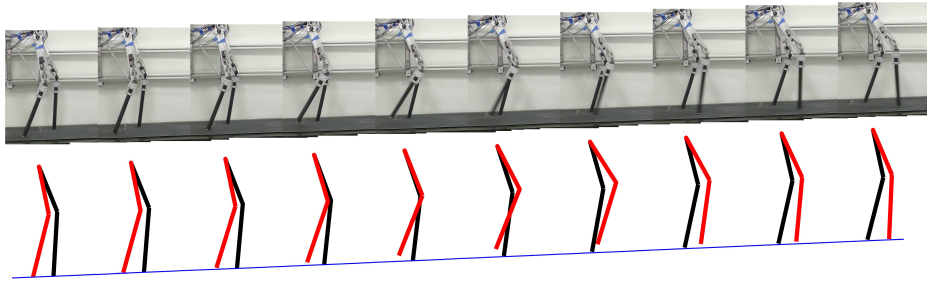
Fig. viii.: Comparisons between human data and the human outputs of the robot for slope walking are shown here.

(a) Walking on flat ground, $\gamma = 0^0$

(b) Walking up a slope, $\gamma = 2^0$

Fig. ix.: Simulation (bottom) vs Experiment (top) for: (a) flat-ground walking, (b) up-slope walking. Video of the experiment can be found in [4]
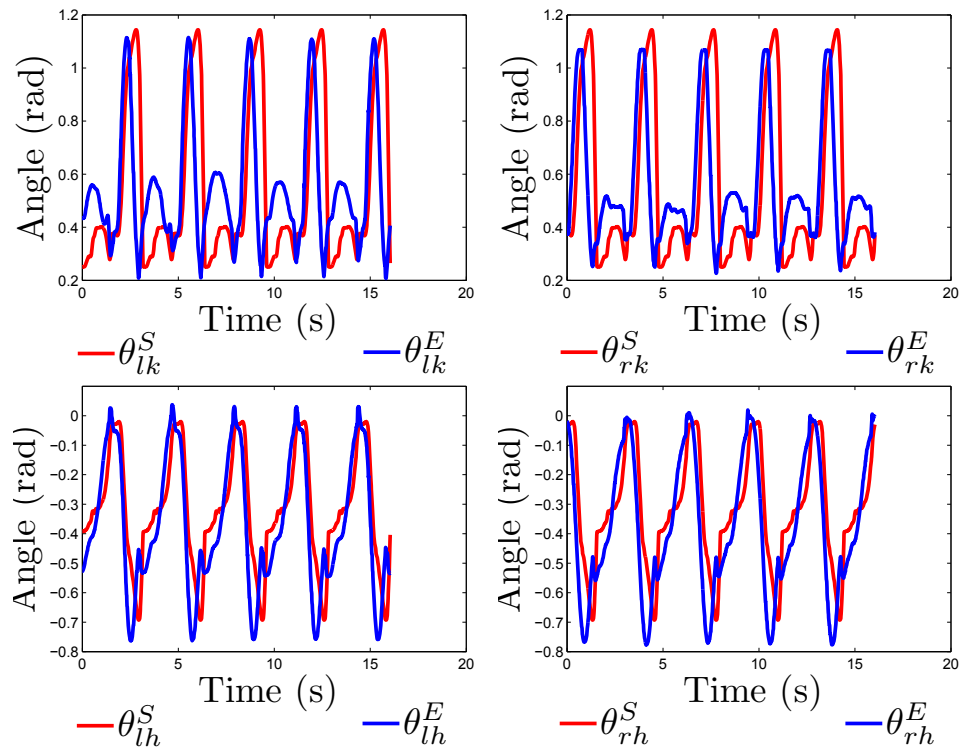
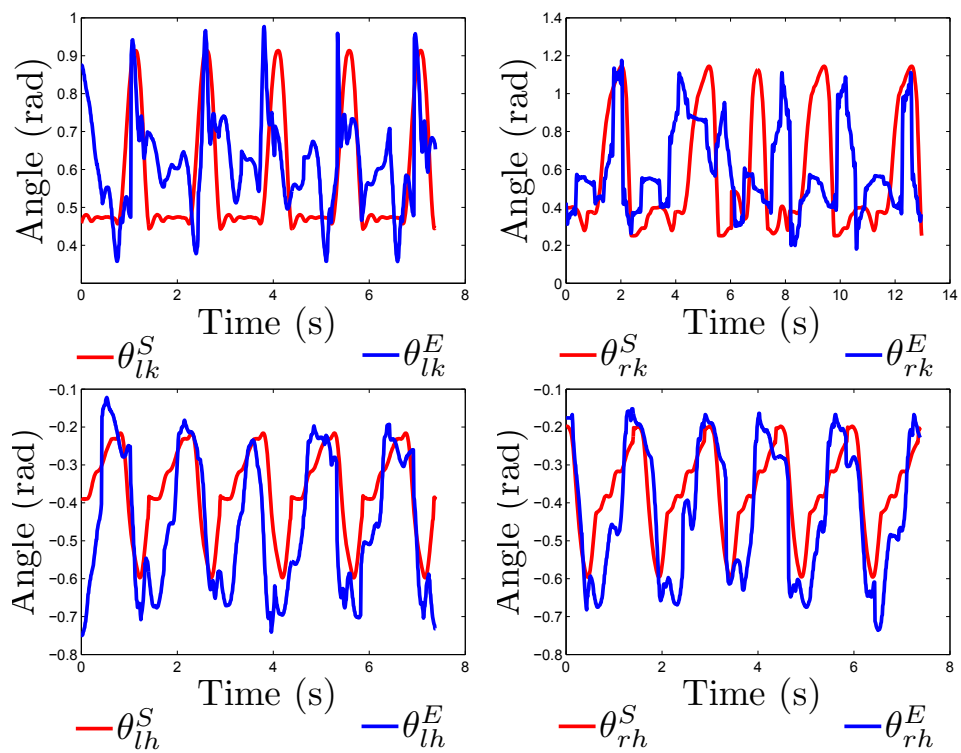Fig. x.: Experimental (blue) vs. simulated (red) angles over 10 steps for flat-ground walking.

Fig. xi.: Experimental (blue) vs. simulated (red) angles over 10 steps for up-slope walking.

30

CHAPTER V

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. But, on an uneven terrain, 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 of which contains the post-impact state. Given the post-impact state obtained due the non-zero height(h_R) of non-stance foot, x^+, we find the new CWF by computing the new set of parameters of walking, α^{int}.

V.1. Extended Canonical Walking Function (ECWF) and Intermediate Motion Transitions (IMT)

Given the canonical walking function in (5.4), we have the robot outputs and their derivatives:

$$y_a(\theta^+) = H\theta^+, \; dy_a(\dot{\theta}^+) = \frac{H\dot{\theta}^+}{\dot{\xi}_1^+}, \tag{5.1}$$

where $\dot{\xi}_1^+ = c\dot{\theta}^+$ (from (3.11)) 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 joint angle velocities and

based on the fact that if HZD is respected for one point on the guard, then it must be respected for all the points on the zero dynamics surface. This is proved in [29] 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 be flat ($h_R = 0$), 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:

$$y_a(\theta^-) = H\theta^-, \ dy_a(\dot\theta^-) = \frac{H\dot\theta^-}{\dot\xi_1^-}, \tag{5.2}$$

with $\xi_1^- = c\dot\theta^-$ 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:

$$y_a(\theta^+) = y_d(\tau(\theta^+), \alpha^{int}), \ dy_a(\dot\theta^+) = \frac{\partial y_d(\tau(\theta^+), \alpha^{int})}{\partial \xi_1^+},$$
$$y_a(\theta^-) = y_d(\tau(\theta^-), \alpha^{int}), \ dy_a(\dot\theta^+) = \frac{\partial y_d(\tau(\theta^-), \alpha^{int})}{\partial \xi_1^-}, \tag{5.3}$$

where τ is the time parameterized by hip position given in (3.6).

Since there are four outputs, and $y_a(\theta) \in \mathbb{R}^4$, (5.3) 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):

$$y_{He}(t, \alpha) = e^{-\alpha_4 t}(\alpha_1 \cos(\alpha_2 t) + \alpha_3 \sin(\alpha_2 t)) + \alpha_5 \cos(\alpha_6 t) +$$

$$\frac{2\alpha_5 \alpha_4 \alpha_2}{\alpha_2^2 + \alpha_4^2 - \alpha_6^2} \sin(\alpha_6 t) + \alpha_7, \tag{5.4}$$

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:

$$LTV(t, \alpha) = \begin{bmatrix} e^{-\alpha_4 t} \cos(\alpha_2 t), \\ e^{-\alpha_4 t} \sin(\alpha_2 t), \\ \cos(\alpha_6 t) + \frac{2\alpha_4 \alpha_6}{\alpha_2^2 + \alpha_4^2 - \alpha_6^2} \sin(\alpha_6 t), \\ 1 \end{bmatrix},$$

$$y_{He}(t, \alpha) = LTV^T(t, \alpha)\alpha_v, \tag{5.5}$$

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:

$$LTV_d(t, \alpha) = \begin{bmatrix} -\alpha_4 e^{-\alpha_4 t} \cos(\alpha_2 t) - \alpha_2 e^{-\alpha_4 t} \sin(\alpha_2 t), \\ -\alpha_4 e^{-\alpha_4 t} \sin(\alpha_2 t) + \alpha_2 e^{-\alpha_4 t} \cos(\alpha_2 t), \\ -\alpha_6 \sin(\alpha_6 t) + \frac{2\alpha_4 \alpha_6^2}{\alpha_2^2 + \alpha_4^2 - \alpha_6^2} \cos(\alpha_6 t), \\ 0 \end{bmatrix},$$

$$dy_{He}(t, \alpha) = \frac{\partial y_{He}(t, \alpha)}{\partial \xi_1} = LTV_d^T(t, \alpha) \frac{\partial t}{\partial \xi_1} \alpha_v. \tag{5.6}$$

LTV_d is the time derivative of *LTV*; and from (3.6) and (3.11), it follows that $\frac{\partial t}{\partial \xi_1} = v_{hip}^{-1}$.

We can now use the ECWF, to replace (3.5) with:

$$y_{de}(\tau, \alpha) = \begin{bmatrix} y_{He}(\tau, \alpha_{nsl}) \\ y_{He}(\tau, \alpha_{sk}) \\ y_{He}(\tau, \alpha_{nsk}) \\ y_{He}(\tau, \alpha_{tor}) \end{bmatrix},$$

$$\frac{\partial y_{de}(\tau, \alpha)}{\partial \xi_1} = \begin{bmatrix} dy_{He}(\tau, \alpha_{nsl}) \\ dy_{He}(\tau, \alpha_{sk}) \\ dy_{He}(\tau, \alpha_{nsk}) \\ dy_{He}(\tau, \alpha_{tor}) \end{bmatrix}, \tag{5.7}$$

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 (5.7) will replace the desired outputs in (5.3) resulting in 16 linear equations with 29 unknowns. Note that control law in (3.8) 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, (5.3) will contain 16 unknowns, and can be solved for in a straightforward manner. To illustrate, we will consider one of the outputs, the stance knee angle, θ_{sk}, resulting in:

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

$$y_{sk}(\theta^-) = \theta_{sk}^-, \ dy_{sk}(\dot{\theta}^-) = \frac{\dot{\theta}_{sk}^-}{\dot{\xi}_1^-}, \tag{5.8}$$

and (5.3) yields:

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

$$y_{sk}(\theta^-) = y_{He}(\tau(\theta^-), \alpha_{sk}^{int}), \ dy_{sk}(\dot{\theta}^-) = dy_{He}(\tau(\theta^-), \alpha_{sk}^{int}). \tag{5.9}$$

Applying (5.5) and (5.6) to (5.9) yields:

$$\begin{bmatrix} y_{sk}(\theta^+) \\ dy_{sk}(\dot{\theta}^+) \\ y_{sk}(\theta^-) \\ dy_{sk}(\dot{\theta}^-) \end{bmatrix} = \begin{bmatrix} LTV^T(\tau(\theta^+), \alpha_{sk}) \\ LTV_d^T(\tau(\theta^+), \alpha_{sk})v_{hip}^{-1} \\ LTV^T(\tau(\theta^-), \alpha_{sk}) \\ LTV_d^T(\tau(\theta^-), \alpha_{sk})v_{hip}^{-1} \end{bmatrix} \alpha_v, \tag{5.10}$$

which yields a unique solution for α_v, if the matrix on the R.H.S. of (5.10) 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.

A graphical illustration of the concept of intermediate motion transitions is given in Fig. xii. 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 rough terrain.

The suggested method was implemented in 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. xiii. The second kind was a sinusoidal terrain with an amplitude of 2^0 and a frequency of 0.1Hz. Phase portrait and the outputs are shown in Fig. xiii. Simulated videos for both random and sinusoidal terrains can be obtained in [5].
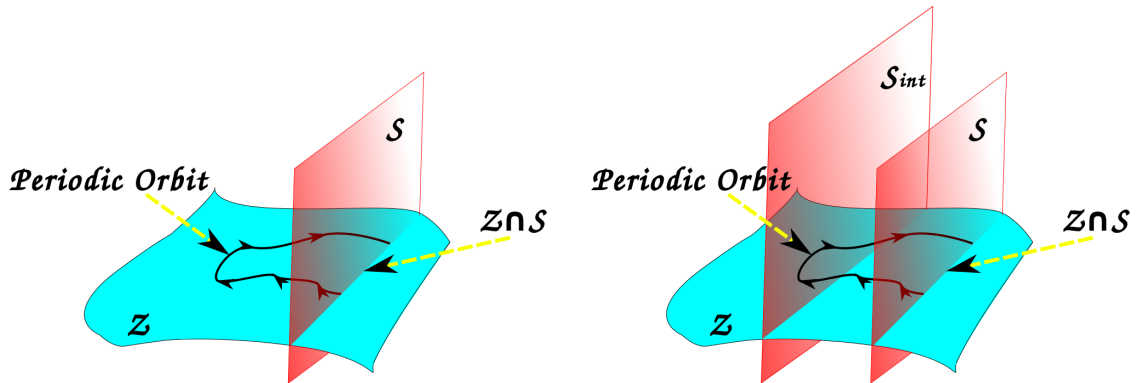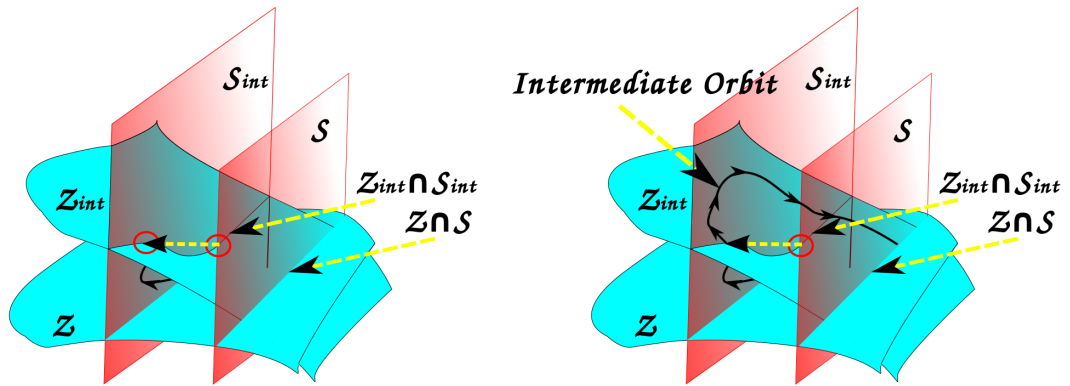
(a) AMBER is walking on flat ground.

(b) Suddenly, it encounters an uneven surface.
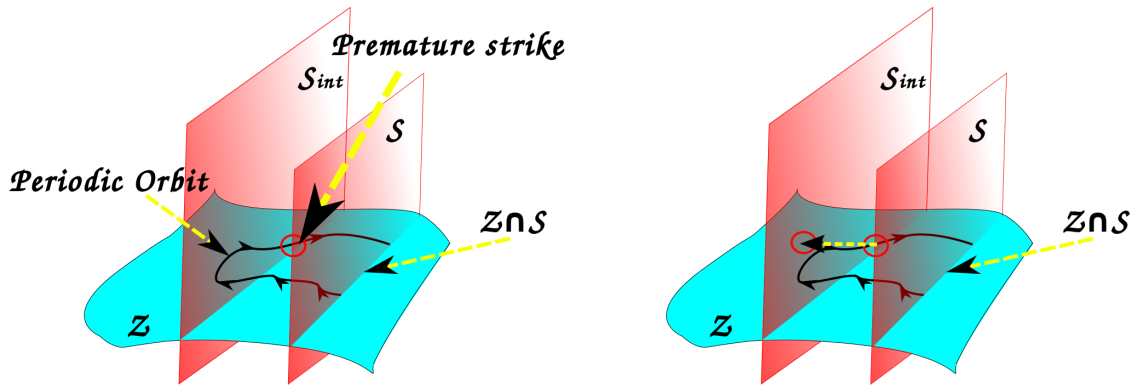
(c) In other words, it strikes a different guard.

(d) There is a jump in state due to the impact.

(e) An intermediate surface is created which constrains the new state and the pre-impact state of flat ground walking orbit.

(f) The intermediate transition allows for the biped to get back to flat ground walking mode, even through disturbances.

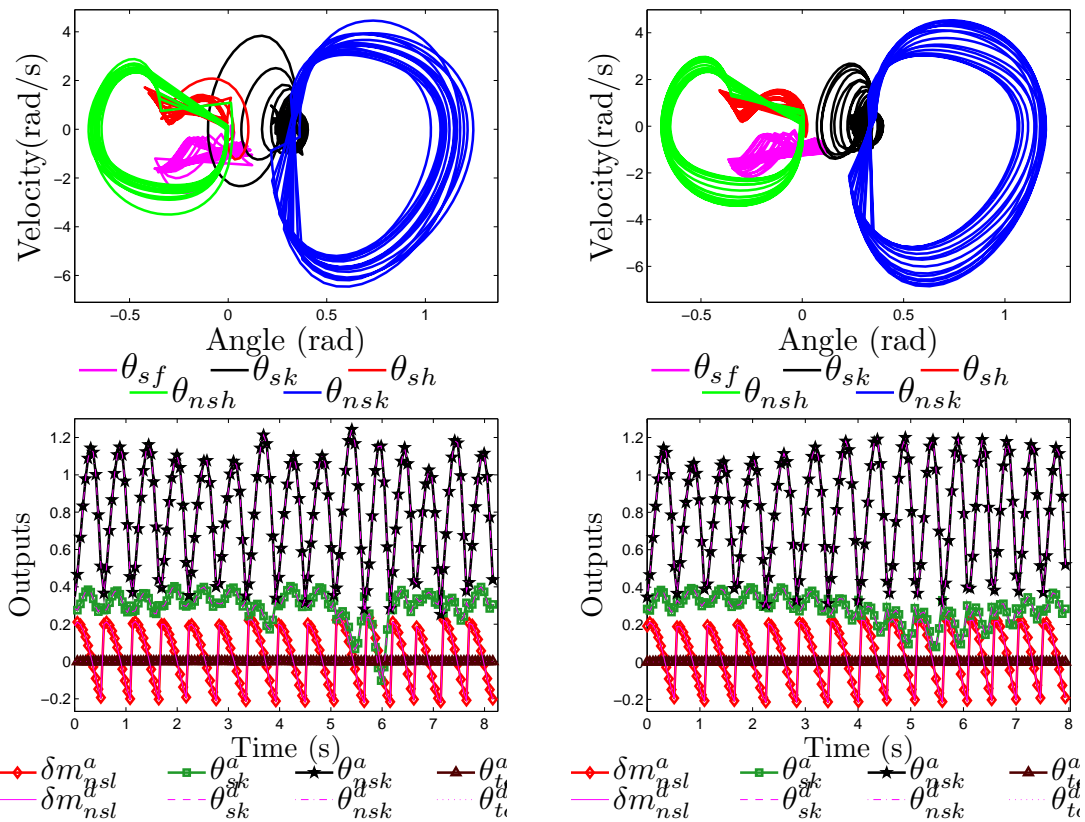Fig. xii.: Graphical representation of IMT is shown here.

Fig. xiii.: Phase portraits and outputs of AMBER walking over a randomly generated terrain (left) and a sinusoidal terrain (right).

V.2. Experimental Implementation of IMT

AMBER is powered by DC motors, and therefore the type of control law suggested in (3.8) requires knowing the model parameters of the robot. Instead, by looking at the joint angles obtained from the absolute encoders, we adopted a simple linear control law: $V_{in} = -K_p y_\alpha(\theta)$, V_{in} is the vector of voltage inputs to the motors and K_p is the proportional constant. More details can be found in [30].
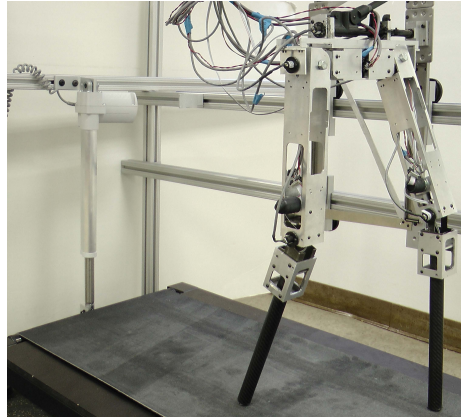

Fig. xiv.: AMBER with the treadmill, and the linear actuator (back) used to vary the slope of the terrain.

Since we adopt a different control 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 CWF 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 (3.13):

$$x^+ = \Delta \left(\begin{bmatrix} \Phi(k\xi_1^*) \\ \frac{\partial \Phi(k\xi_1^*)}{\partial \xi_1} v_{hip} \end{bmatrix} \right),$$ (5.11)

where Δ gives the post-impact map from the reconstructed state. Accordingly, the inter-mediate transition matrix is obtained as described in Section V. It is important to note that this transition matrix is dynamically updated in AMBER after every non-stance foot strike. The sinusoidal terrain is created by sinusoidally exciting the linear actuator connecting the treadmill (see Fig. xiv). Tiles of AMBER walking over a rough terrain for one step are shown in Fig. xv and Fig. xvi shows the variation of outputs of the robot on the sinusoidal terrain (see video [3]). Fig. xvii shows all the ECWF computed from the updated α^{int} after every step, and they are compared with the human data.
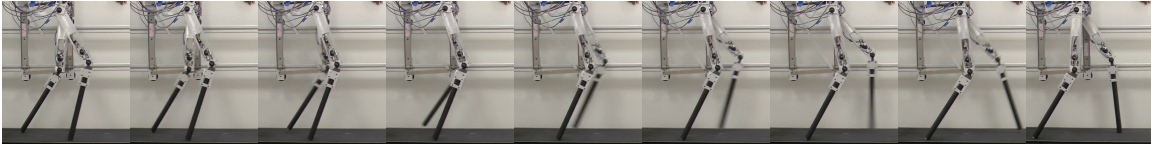
Fig. xv.: Tiles showing AMBER taking one step on the rough terrain. It can be observed that the configuration of the robot at the end of the step is different from the beginning of the step.
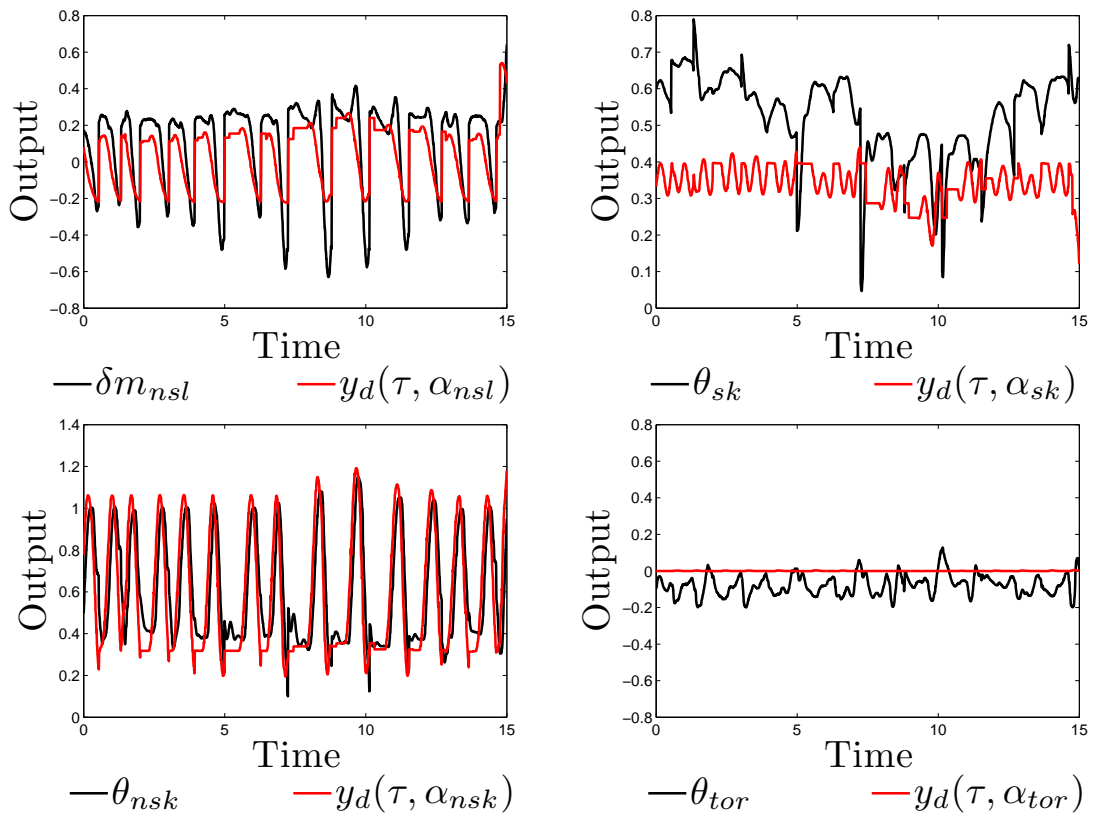

Fig. xvi.: Comparison of outputs of the robot with the desired outputs. Since the stance knee takes the weight of the robot, it does not match with the desired output well.
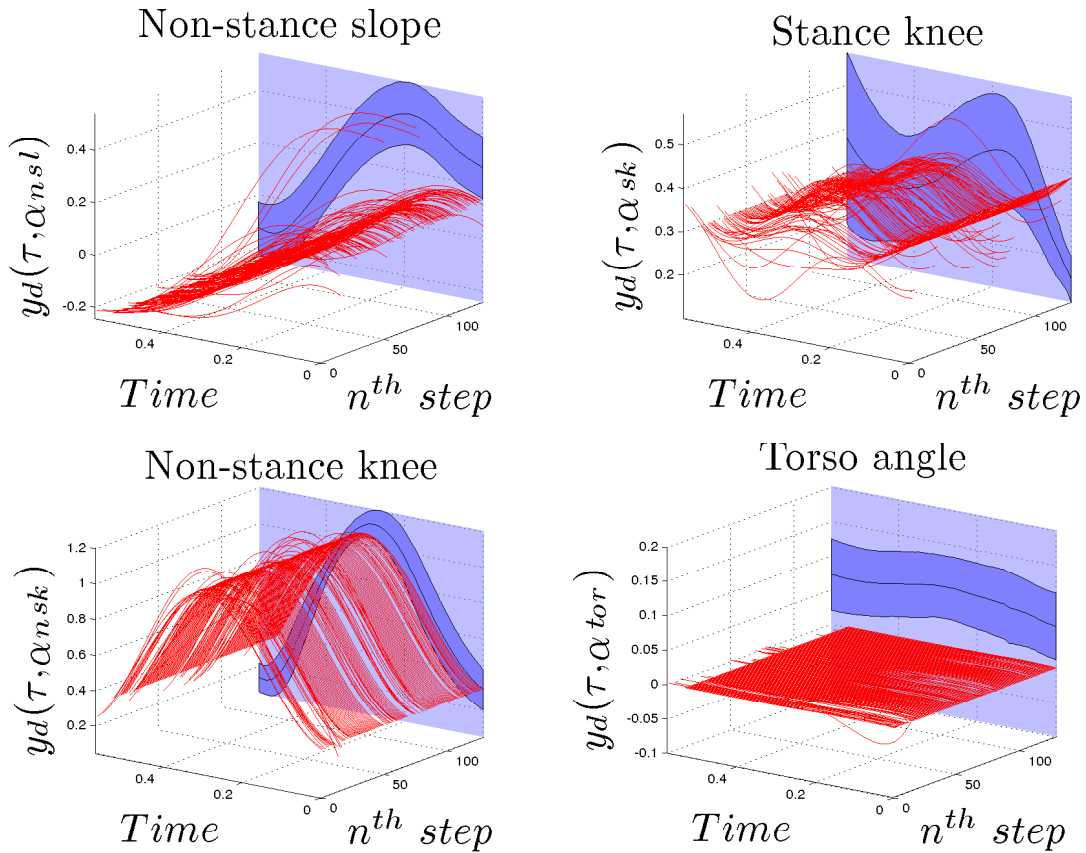
Fig. xvii.: Desired output functions from the intermediate transition matrix α^{int} computed at all the steps (red waveforms) and compared with the human data (shown in blue).

41

CHAPTER VI

CONCLUSIONS

This thesis successfully achieved walking on three motion primitives by using human-inspired control which is not only stable, but also remarkably robust (see video [1]). From walking tiles and joint angle tracking behavior considered, it can be stated that the physically realizable walking obtained from the simulation is mirrored in the walking obtained in physical world. This speaks a lot about the strength of the formal theory on human-inspired control. In fact, the walking was achieved with minimal computation overhead requiring less than 100 lines of code in LabVIEW. One of the reasons that can be attributed to the success is the choice of the CWF chosen. The fact that human walking is an end product of thousands of years of evolution, and that the robustness is inherently built into human walking, shows the extent to which AMBER can be pushed and still walk stably on the platform. It must also be stressed that the feet are not actuated, which makes the problem to realize natural human-like motion primitives even harder. The concept of using motion transitions made the walking even more robust, resulting in stable walking even on an unknown random terrain. The application of the computation for rough terrain transitions is limited by the maximum height at which the robot hits the ground. In other words, since we are using one reference CWF, the time parameter τ^+ must be within the limits to realize the transformation. The limits are decided by the restriction on hyper extension of the knee and also the maximum achievable speed on the robot. For the canonical walking function that was picked for AMBER, $0.532 < \tau^+ < 0.574$. With the transformation suggested and with the limits specified, preliminary experiments showed significant improvement in the walking obtained. 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. Inspite of that, the improvement in walking is observed, thereby justifying the application of intermediate motion transitions. The ability to obtain provably stable walking gaits in the simulation and then translate it in the form of proportional voltage controller in AMBER establishes a delicate and yet reliable nexus between theory and experiment, which inscribes the formula for success.

REFERENCES

[1] AMBER robustness tests.

http://youtu.be/RgQ8atV1NWO.

[2] Human-Inspired Proportional Control with AMBER in simulation.

http://www.youtube.com/watch?v=z3oPzlStzRo.

[3] Motion transitions up the slope and on a sinusoidal terrain.

http://youtu.be/aUiEXt8otrY.

[4] Robust walking of amber on flat-ground and up-slope.

http://www.youtube.com/watch?v=aBfszNZs5mQ.

[5] Simulated walking on random and sinusoidal terrains.

http://youtu.be/OtW1I4kRWj8

http://youtu.be/p7bdqmYhzcc.

[6] 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.

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

[8] A. D. Ames, E. A. Cousineau, and M. J. Powell. Dynamically stable robotic walking with NAO via human-inspired hybrid zero dynamics. In *Hybrid Systems: Computation and Control, 2012*, Beijing, China, 2012.

44

[9] A. D. Ames, R. Vasudevan, and R. Bajcsy. Human-data based cost of bipedal robotic walking. In *Hybrid Systems: Computation and Control*, Chicago, IL, 2011.

[10] T. Burg, D. Dawson, J. Hu, and M. de Queiroz. An adaptive partial state-feedback controller for RLED robot manipulators. *IEEE Transactions on Automatic Control*, 41(7):1024 –1030, 1996.

[11] K. Byl and R. Tedrake. Metastable walking machines. *International Journal of Robotics Research*, 28(8):1040 – 1064, Aug 2009.

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

[13] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):263 –273, 2010.

[14] J. W. Grizzle, J. Hurst, B. Morris, H. Park, and K. Sreenath. MABEL, a new robotic bipedal walker and runner. In *American Control Conference*, pages 2030–2036, St. Louis, MO, USA, 2009.

[15] 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.

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

[17] C. Liu, C. C. Cheah, and J. E. Slotine. Adaptive jacobian tracking control of rigid-link electrically driven robots based on visual task-space information. *Automatica*, 42(9):1491 – 1501, 2006.

[18] I. R. Manchester, U. Mettin, F. Iida, and R. Tedrake. Stable dynamic walking over uneven terrain. *The International Journal of Robotics Research*, 30(3):265–279, 2011.

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

[20] R. M. Murray, Zexiang Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, 1994.

[21] J. Bo. Nielsen. How we walk : Central control of muscle activity during human walking. *The Neuroscientist*, 9(3):195–204, 2003.

[22] H. Park, K. Sreenath, J. Hurst, and J. W. Grizzle. Identification of a bipedal robot with a compliant drivetrain: Parameter estimation for control design. *IEEE Control Systems Magazine*, 31(2):63–88, 2011.

[23] I. Poulakakis and J. W. Grizzle. The Spring Loaded Inverted Pendulum as the Hybrid Zero Dynamics of an Asymmetric Hopper. *Transaction on Automatic Control*, 54(8):1779–1793, 2009.

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

[25] S. S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer, New York, 1999.

[26] R. Sinnet, M. Powell, R. Shah, and A. D. Ames. A human-inspired hybrid control approach to bipedal robotic walking. In *18th IFAC World Congress*, Milano, Italy, 2011.

[27] M. W. Spong. Passivity based control of the compass gait biped. In *IFAC World Congress*, Beijing, 1999.

[28] S. Srinivasan, I. A. Raptis, and E. R. Westervelt. Low-dimensional sagittal plane model of normal human walking. *ASME J. of Biomechanical Eng.*, 130(5), 2008.

[29] 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.

[30] 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.

[31] 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)*, Algarve, Portugal, 2012.