

2D Bipedal Walking with Knees and Feet: A Hybrid Control Approach

Ryan W. Sinnet and Aaron D. Ames

Abstract—In this paper, we consider an anthropomorphically-inspired hybrid model of a bipedal robot with locking knees and feet in order to develop a control law that results in human-like walking. The presence of feet results in periods of full actuation and periods of underactuation during the course of a step. Properties of each of these phases of walking are utilized in order to achieve a stable walking gait. In particular, we will show that using *controlled symmetries* in the fully-actuated domains coupled with “partial” controlled symmetries and *local ankle control laws* in the underactuated domains yields stable walking; this result is possible due to the amount of time which the biped spends in the fully-actuated domains. The paper concludes with simulation results along with a comparison of these results to human walking data.

I. INTRODUCTION

Motivated by the desire to obtain anthropomorphic walking, this paper considers a bipedal robot with locking knees and feet. Both knees and feet have been studied before but have not yet been combined into a comprehensive model. Walking on point feet with knees has been studied from a variety of perspectives (cf. [3], [4], [11], [14]). As far as control, the models of [3], [11] either uses underactuated controllers, which results in unnatural-looking walking, or assumes full actuation (which allows for the use of controlled symmetries) which is not realistic as a point-footed model is necessarily underactuated. Existing research including feet (e.g., [19]) has shown impressive results; however, the current models make significant simplifying assumptions; to name a few: instantaneous double-support phase, foot lands flat, ignore effects of scuffing. These assumptions simplify the model but unfortunately reduce the realism.

Given a bipedal robot with knees and feet, there are discrete events that occur throughout the course of a step (cf. [20]), e.g., knee-lock, heel-lift, etc. In the paper, we consider five events of this form resulting in a hybrid model for the bipedal robot with five discrete domains, each representing a specific portion of the total gait (cf. [6]); see Fig. 1. For example, in one domain, we have the stance foot flat on the ground and the non-stance knee unlocked. The guard for this domain is knee-lock, which occurs when the non-stance leg straightens. After this transition, the dynamics change due to the knee being locked and we enter a new discrete domain. Simulation results will suggest that these events play an important role in achieving natural-looking walking.

The main goal of this paper is to show that controlled symmetries (which takes the passive gait of a biped going

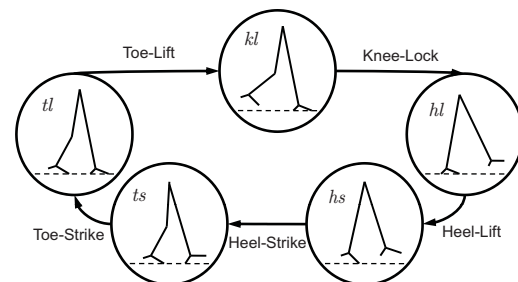


Fig. 1: Graph of system domains

down a slope and “rotates” gravity to obtain walking on flat ground) can be used to achieve stable walking even in the presence of underactuation. Of the five domains in our model, two are fully-actuated and three are underactuated. We utilize the fact that the biped spends the majority of its time having full actuation. The idea is that we can achieve a stable gait by applying controlled symmetries only in the domains with full actuation; however, the short period of time spent with underactuation is enough to disrupt the stability of the system and so we design additional control laws, such as “partial” controlled symmetries, to carry the system through the underactuated domains.

Motivated by the desire to achieve an efficient and anthropomorphic gait, we consider a hybrid model with underactuated domains (e.g., heel-lift, or *hl* in Fig. 1). Transitions into these domains will not occur with only controlled symmetries and so we introduce local control laws to effect the necessary transitions. A natural guard indicating transition in some cases is a Lagrange multiplier, which specifies the restraining force preventing a point on the biped from moving (through the ground). Accordingly, we construct a controller to drive the Lagrange multiplier to zero, thus driving the system to the guard. Another side effect of the model is scuffing, which occurs when the non-stance foot strikes the ground at the wrong time. Clearly, this would be a problem in a realistic model and so we introduce a controller to prevent this phenomenon. Combining the “local” controllers in the underactuated domains with controlled symmetries in the fully actuated domains results in stable walking.

To conclude the paper, we discuss the simulation results of the bipedal model considered. These simulations provides insight into the gait of the model, and further support the anthropomorphic nature of the walking through a brief comparison with human walking. In particular, we describe the effects of the implemented controllers and explore the interplay between the dynamics of walking and the control necessary to achieve a stable gait. Finally, we show numerically that we are able to attain a stable gait, i.e., a locally

R. W. Sinnet and A. D. Ames are with the Mechanical Engineering Department, Texas A&M University, College Station, TX 77843 {rsinnet, ames}@tamu.edu

exponentially stable periodic orbit.

Note that the motivation for the techniques introduced in this paper draw from many sources in the literature, and especially work related to passive dynamic walking, controlled symmetries, and bipeds with feet. Passive dynamic walking has been studied in [4], [8], [10], [13], [14] and [18], where it was shown that a biped can walk down a slope without actuation. Controlled symmetries has been well-studied (e.g. [3]), where gaits obtained from passive models were made to walk on flat ground using actuation. Walking with feet has not been extensively studied yet work has been done in [5], [7], [9] and [19]. This work has shown that using feet can increase the efficiency of walking and can yield a more-natural-looking gait.

II. HYBRID SYSTEMS AND CONSTRAINTS

Hybrid systems are systems that display both continuous and discrete behavior and so bipedal walkers are naturally modeled by systems of this form; the continuous component consists of the dynamics dictated by Lagrangians modeling mechanical systems in different domains and the discrete component consists of the impact equations which instantaneously change the velocity of the system when a knee locks or when a toe or heel contacts the ground. This section introduces the basic terminology of hybrid systems and introduces the hybrid model of the biped considered in this paper.

Definition 1: A non-autonomous hybrid control system is a tuple

$$\mathcal{H}\mathcal{C} = (\Gamma, D, U, G, R, FG),$$

where

- $\Gamma = (V, E)$ is an oriented graph, i.e., V and E are a set of vertices and edges, respectively, and there exists a source function $\text{sor} : E \rightarrow V$ and a target function $\text{tar} : E \rightarrow V$ which associate to an edge its source and target, respectively.
- $D = \{D_v\}_{v \in V}$ is a set of domains, where $D_v \subseteq \mathbb{R}^{n_v} \times \mathbb{R}^{k_v}$ is a smooth submanifold of $\mathbb{R}^{n_v} \times \mathbb{R}^{k_v}$ (with \mathbb{R}^{k_v} representing control inputs and/or time),
- $U = \{U_v\}_{v \in V}$, where $U_v \subset \mathbb{R}^{k_v}$ is a set of admissible controls,
- $G = \{G_e\}_{e \in E}$ is a set of guards, where $G_e \subseteq D_{\text{sor}(e)}$,
- $R = \{R_e\}_{e \in E}$ is a set of reset maps, where $R_e : G_e \rightarrow D_{\text{tar}(e)}$ is a smooth map,
- $FG = \{(f_v, g_v)\}_{v \in E}$, where (f_v, g_v) is a control system on D_v , i.e., $\dot{x} = f_v(x, t) + g_v(x)u$ for $x \in D_v$ and $u \in U_v$.

A hybrid system $\mathcal{H} = (\Gamma, D, G, R, F)$ is a hybrid control system with $U = \{0\}$, in which case $F = \{f_v\}_{v \in E}$ with $\dot{x} = f_v(x, t)$. An autonomous hybrid system is a hybrid system with $D_v \subseteq \mathbb{R}^n$ and $\dot{x} = f_v(x)$.

Remark 1: We allow for non-autonomous hybrid systems because some of our control laws will depend on time.

Poincaré Map. Solutions to hybrid systems, or *hybrid flows* or *hybrid executions*, are defined in the traditional manner

(see [12]). A solution to a hybrid system is k -periodic if it returns to the same point after passing through the domain in which it is contained k times. (In the process it may pass through an arbitrary number of other domains of the hybrid system.) One can consider the local exponential stability of k -periodic solutions in the obvious way (see [2] for this definition in the case of a hybrid system with one domain). One can associate to a k -periodic solution of a hybrid system a Poincaré map, and the stability of the k -periodic solution can be determined by considering the stability of the Poincaré map. Finally, the stability can be determined numerically using approximations of the Jacobian of the Poincaré map (see [15] and [17]). This is how we will determine that the periodic orbit for the 2D biped in this paper is stable.

Lagrangians. Let $Q \subseteq \mathbb{R}^n$ be the *configuration space* of a mechanical system. We will consider Lagrangians $L : TQ \rightarrow \mathbb{R}$ modeling mechanical systems; that is, Lagrangians given in coordinates by

$$L(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} - V(q), \quad (1)$$

where $\frac{1}{2} \dot{q}^T M(q) \dot{q}$ is the kinetic energy and $V(q)$ is the potential energy. We also consider external forces in the forms of non-conservative forces (due to viscous rotational damping at the ankles), $F_{\text{nc}} = F \dot{q}$, and control inputs, which sometimes take the form $u = K(q, \dot{q}, t)$, both of which are smooth through a given domain.

With these forces in mind and motivated by the derivation in [16] (§6.1.2), the Euler-Lagrange equations can be written

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + N(q) = F\dot{q} + Bu, \quad (2)$$

with $M(q)$ the manipulator inertia matrix, $C(q, \dot{q})$ the matrix of centripetal and Coriolis terms, $N(q) = \frac{\partial V(q)}{\partial q}$ containing terms related to gravitational potential, F a constant matrix of damping coefficients, and B a linear transformation with which converts control inputs from relative to absolute coordinates. Using (2), we define the control system:

$$\begin{aligned} f_L(q, \dot{q}) &= \begin{pmatrix} \dot{q} \\ M^{-1}(q)(F\dot{q} - C(q, \dot{q})\dot{q} - N(q)) \end{pmatrix} \\ g_L(q) &= \begin{pmatrix} \mathbf{0}_{n \times k} \\ M^{-1}(q)B \end{pmatrix}, \end{aligned} \quad (3)$$

where $\mathbf{0}_{n \times k}$ is a zero matrix with k the number of inputs.

Unilateral Constraints. For the hybrid model that will be considered, certain domains and guards are obtained from unilateral constraints.

Definition 2: A *unilateral constraint* is defined to be a tuple $\mathbf{h} = (Q, L, h)$, where

- Q is the configuration space (usually assumed to be identical to \mathbb{R}^n),
- $L : TQ \rightarrow \mathbb{R}$ is a hyperregular Lagrangian,
- $h : Q \rightarrow \mathbb{R}$ provides a unilateral constraint on the configuration space; we assume that the zero level set $h^{-1}(0)$ is a smooth manifold.

Given a unilateral constraint \mathbf{h} there is an associated guard, reset map and vector field. Using $h(q)$, we define the domain

and guard respectively as

$$\begin{aligned} D_h &= \{(q, \dot{q}) \in TQ : h(q) \geq 0\}, \\ G_h &= \{(q, \dot{q}) \in TQ : h(q) = 0 \text{ and } d_q h(q) \dot{q} < 0\}, \end{aligned} \quad (4)$$

where $d_q h(q) = \left(\frac{\partial h(q)}{\partial q}\right)^T$.

Holonomic constraints. On certain domains of the model we will consider domains, guards and vector fields obtained from holonomic constraints.

Definition 3: A *holonomic constraint* is defined to be a tuple $\eta = (Q, L, \eta)$, where

- Q is the configuration space (usually assumed to be identical to \mathbb{R}^n),
- $L : TQ \rightarrow \mathbb{R}$ is a hyperregular Lagrangian,
- $\eta : Q \rightarrow \mathbb{R}^c$ provides holonomic constraints on the configuration space; we assume that the zero level set $\eta^{-1}(0)$ is a smooth manifold.

We impose the holonomic constraint using a Lagrange multiplier. Differentiating the constraints $\eta(q)$ we obtain $A(q)\dot{q} = 0$ with $A(q) = d_q \eta(q)$. Following the derivation in [16] (§6.1.2), we write the Euler-Lagrange equations as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + N(q) + A(q)^T \lambda_\eta(q, \dot{q}, u) = F\dot{q} + Bu \quad (5)$$

where $\lambda_\eta : TQ \times \mathbb{R}^c \rightarrow \mathbb{R}^c$ is the Lagrange multiplier.

To find an explicit definition for $\lambda_\eta(q, \dot{q}, u)$, we first differentiate the constraint equation: $A(q)\ddot{q} + \dot{A}(q)\dot{q} = 0$. Using this with (5), we can write the Lagrange multiplier as

$$\begin{aligned} \lambda_\eta(q, \dot{q}, u) = & \quad (6) \\ & (A(q)M^{-1}(q)A^T(q))^{-1}(A(q)M^{-1}(q)(F\dot{q} + Bu \\ & \quad - C(q, \dot{q})\dot{q} - N(q)) + \dot{A}(q, \dot{q})\dot{q}). \end{aligned}$$

The value of the Lagrange multiplier is the force required to impose the holonomic constraint. With this in mind and having derived an expression for $\lambda_\eta(q, \dot{q}, u)$, we can now define the domain and guard respectively as

$$\begin{aligned} D_\eta &= \{(q, \dot{q}, u) \in TQ \times \mathbb{R}^k : \lambda_\eta(q, \dot{q}, u) \geq 0\}, \\ G_\eta &= \{(q, \dot{q}, u) \in TQ \times \mathbb{R}^k : \lambda_\eta(q, \dot{q}, u) = 0 \\ & \quad \text{and } d_q \lambda_\eta(q, \dot{q}, u) \dot{q} < 0\}, \end{aligned} \quad (7)$$

where $d_q \lambda_\eta(q, \dot{q}, u)$ is the Jacobian. We then use (5) and (6) to define the control system:

$$\begin{aligned} f_\lambda(q, \dot{q}) &= (\dot{q}^T, \\ & [M^{-1}(q)((I - A^T(q)\Xi(q)A(q)M^{-1}(q))(F\dot{q} - \\ & \quad C(q, \dot{q})\dot{q} - N(q)) - A^T(q)\Xi(q)\dot{A}(q)\dot{q})^T]^T)^T \\ g_\lambda(q) &= \\ & (\mathbf{0}_{m \times n}, [M^{-1}(q)(I - A^T(q)\Xi(q)A(q)M^{-1}(q))B]^T)^T \end{aligned} \quad (8)$$

with $\Xi(q) = (A(q)M^{-1}(q)A^T(q))^{-1}$. This vector field has augmented dynamics which constrain the system by enforcing the holonomic constraints, $\frac{dh(q)}{dt} = A(q)\dot{q} = 0$.

Kinematic constraints. In order to derive the impact equations, we consider:

Definition 4: A *kinematic constraint* is defined to be a tuple $\kappa = (Q_s, Q_t, M, \kappa, \iota, J)$, where

- Q_s is the configuration space of the source domain (usually assumed to be $\mathbb{R}^{\dim Q_s}$),
- Q_t is the configuration space of the target domain (usually assumed to be $\mathbb{R}^{\dim Q_t}$),
- $M : Q_s \cup J^{-1}(Q_t) \rightarrow \mathbb{R}^{s \times s}$ is a manipulator inertia matrix (with $s = \dim(Q_s \cup J^{-1}(Q_t))$),
- $\kappa : Q_s \cup J^{-1}(Q_t) \rightarrow \mathbb{R}^d$ is a smooth function which typically describes the position of the end-effector of a kinematic chain,
- $\iota : Q_s \rightarrow Q_s \cup J^{-1}(Q_t)$ is an embedding with pushforward $\iota^* : TQ_s \rightarrow T(Q_s \cup J^{-1}(Q_t))$,
- J is an invertible binary matrix to relabel coordinates.

Note that κ is a vector valued function of d constraints which will contain the unilateral constraint, h , if present, and may contain certain holonomic constraints from η , if present. Also note, there is a canonical projection $\pi : Q_s \cup J^{-1}(Q_t) \rightarrow J^{-1}(Q_t)$ that induces a map $\pi^* : T(Q_s \cup J^{-1}(Q_t)) \rightarrow T(J^{-1}(Q_t))$, which is the pushforward.

Impact equations. The discrete jumps in the considered hybrid system result from rigid plastic impacts. These impacts can be modeled as impulses at specific locations (cf. [11]). Let $\kappa = (Q_s, Q_t, M, \kappa, \iota, J)$ be a kinematic constraint and let the coordinates of the embedded space $Q_s \cup J^{-1}(Q_t)$ be represented by q . Then define a map $P : T(Q_s \cup J^{-1}(Q_t)) \rightarrow T_q(Q_s \cup J^{-1}(Q_t))$ describing the velocities after impact given by¹

$$P(q, \dot{q}) = \dot{q} - M^{-1}(q)E^T(q)(E(q)M^{-1}(q)E^T(q))^{-1}E(q)\dot{q} \quad (9)$$

where $E(q) = \frac{\partial \kappa(q)}{\partial q}$. Finally, define the reset map $R : TQ_s \rightarrow TQ_t$ given by

$$R(q_s, \dot{q}_s) = \begin{pmatrix} J & 0 \\ 0 & J \end{pmatrix} \pi^* \begin{pmatrix} \iota(q_s) \\ P(\iota^*(q_s, \dot{q}_s)) \end{pmatrix}. \quad (10)$$

III. BIPEDAL MODEL

This section introduces the hybrid model of bipedal robot with knees and feet. Due to space constraints, the M, C, N matrices have been omitted but can be found at [21].

Discrete structure. We define the hybrid control system:

$$\mathcal{HC} = (\Gamma, D, U, G, R, FG)$$

with Γ the oriented graph of the system given as

$$\begin{aligned} \Gamma &= (\{kl, hl, hs, ts, tl\}, \{e_{kl} = (kl, hl), e_{hl} = (hl, hs), \\ & \quad e_{hs} = (hs, ts), e_{ts} = (ts, tl), e_{tl} = (tl, kl)\}) \end{aligned}$$

and D, U, G, R, FG as defined in Definition 1. The specific elements of these sets will be given throughout this section. There are five discrete states which loop back on each other as shown in Fig. 1. Note that for this hybrid system a temporal order of events is assumed.

¹The velocities before and after impact are sometimes written as q^- and q^+ . For notational conciseness, we choose to write the post-impact velocities as a map.

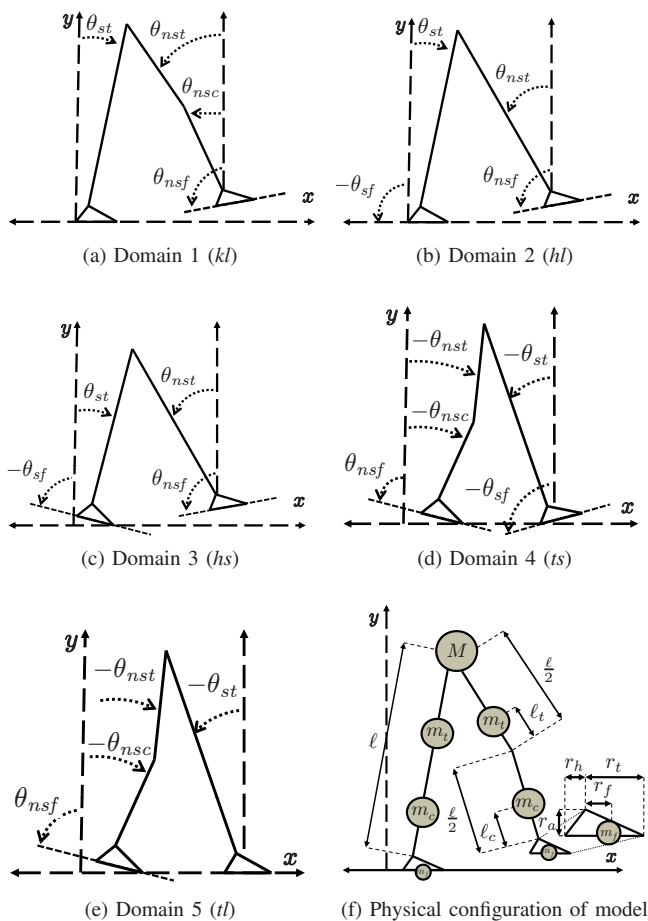


Fig. 2: Biped configurations

Domain 1 (kl): The stance foot is flat on the ground and the non-stance knee is unlocked. The configuration is shown in Fig. 2(a). The admissible control is $U_{kl} = \mathbb{R}^4$ and the guard is *knee-lock*, which occurs when the knee becomes straight and locks. (This would be realized as a physical lock.) We define the unilateral constraint

$$h_{kl} = (Q_{kl}, L_{kl}, h_{kl})$$

where $Q_{kl} = \mathbb{R}^4$ is the configuration space with coordinates $q_{kl} = (\theta_{st}, \theta_{nst}, \theta_{nsc}, \theta_{nsf})^T$, $L_{kl}(q_{kl}) = \frac{1}{2} \dot{q}_{kl}^T M_{kl}(q_{kl}) \dot{q}_{kl} - V_{kl}(q_{kl})$ is the Lagrangian modeling the system, and $h_{kl}(q_{kl}) = \theta_{nsc} - \theta_{nst}$ is the relative angle of the non-stance knee. Using h_{kl} , the domain and guard are then given in (4). We define the kinematic constraint

$$\kappa_{kl} = (Q_{kl}, Q_{hl}, M_{kl}, \kappa_{kl}, \iota_{kl}, J_{kl})$$

with Q_{hl} introduced in the next domain, $\kappa_{kl}(q_{kl}) = h_{kl}(q_{kl})$, $\iota_{kl} : Q_{kl} \rightarrow Q_{kl} \cup J_{kl}^{-1}(Q_{hl})$ the embedding given by

$$(\theta_{st}, \theta_{nst}, \theta_{nsc}, \theta_{nsf}) \mapsto \left(-\frac{\pi}{2}, \theta_{st}, \theta_{nst}, \theta_{nsc}, \theta_{nsf}\right)$$

and $J_{kl} = I$. The reset map is then given by (10). Finally, the control system is given in (3).

Domain 2 (hl): Both knees are locked and the stance foot is flat on the ground. The configuration is shown in Fig. 2(b).

The admissible control is $U_{hl} = \mathbb{R}^3$ and the guard is *heel-lift*. Therefore, we define the holonomic constraint

$$\eta_{hl} = (Q_{hl}, L_{hl}, \eta_{hl})$$

where $Q_{hl} = \mathbb{R}^4$ is the configuration space with coordinates $q_{hl} = (\theta_{sf}, \theta_{st}, \theta_{nst}, \theta_{nsf})^T$, $L_{hl}(q_{hl}) = \frac{1}{2} \dot{q}_{hl}^T M_{hl}(q_{hl}) \dot{q}_{hl} - V_{hl}(q_{hl})$ is the Lagrangian modeling the system, and $\eta_{hl} : Q_{hl} \rightarrow \mathbb{R}$ is the height of the stance heel. Note that we include θ_{sf} even though it is constant throughout the domain. This will be necessary in order to calculate the Lagrange multiplier, $\lambda_{\eta_{hl}}(q_{hl}, \dot{q}_{hl}, u_{hl})$, which we will use as our guard. Using η_{hl} , the domain and guard are then given in (7).

The reset map is simply the identity map. Finally, the control system is given in (8) using the Lagrange multiplier $\lambda_{\eta_{hl}}(q_{hl}, \dot{q}_{hl}, u_{hl})$ obtained from the holonomic constraint. By imposing this constraint, we are forcing the stance heel to have constant zero velocity.

Domain 3 (hs): Both knees are locked and the system is rotating about the stance toe. The configuration is shown in Fig. 2(c). The admissible control is $U_{hs} = \mathbb{R}^3$ and the guard is *heel-strike*. Therefore, we define the unilateral constraint

$$h_{hs} = (Q_{hs}, L_{hs}, h_{hs})$$

where $Q_{hs} = \mathbb{R}^4$ is the configuration space with coordinates $q_{hs} = (\theta_{sf}, \theta_{st}, \theta_{nst}, \theta_{nsf})^T$, $L_{hs}(q_{hs}) = \frac{1}{2} \dot{q}_{hs}^T M_{hs}(q_{hs}) \dot{q}_{hs} - V_{hs}(q_{hs})$ is the Lagrangian modeling the system, and $h_{hs}(q)$ is the height of the stance heel. Using h_{hs} , the domain and guard are given in (4). In order to define the reset map, we need the kinematic constraint,

$$\kappa_{hs} = (Q_{hs}, Q_{ts}, M_{ts}, \kappa_{hs}, \iota_{hs}, J_{hs})$$

with Q_{ts} and M_{ts} introduced in the next domain, κ_{hs} the x and y positions of the non-stance heel, and $\iota_{hs} : Q_{hs} \rightarrow Q_{hs} \cup J_{hs}^{-1}(Q_{ts})$ the embedding given by

$$(\theta_{sf}, \theta_{st}, \theta_{nst}, \theta_{nsf}) \mapsto (\theta_{sf}, \theta_{st}, \theta_{st}, \theta_{nst}, \theta_{nsf}).$$

The reset map requires that the non-stance and stance legs be switched so we use the relabeling matrix $J_{hs} = \text{antidiag}(1, \dots, 1)$. The reset map is then given in (10). Finally, the control system is given in (3).

Domain 4 (ts): The non-stance knee is unlocked and the system is rotating about the stance heel. The non-stance toe is on the ground. The configuration is shown in Fig. 2(d). The admissible control is $U_{ts} = \mathbb{R}^4$ and the guard is *toe-strike*, which occurs when the stance toe rolls into the ground. Therefore, we define the unilateral constraint

$$h_{ts} = (Q_{ts}, L_{ts}, h_{ts})$$

where $Q_{ts} = \mathbb{R}^5$ is the configuration space with coordinates $q_{ts} = (\theta_{sf}, \theta_{st}, \theta_{nst}, \theta_{nsc}, \theta_{nsf})^T$, $L_{ts}(q_{ts}) = \frac{1}{2} \dot{q}_{ts}^T M_{ts}(q_{ts}) \dot{q}_{ts} - V_{ts}(q_{ts})$ is the Lagrangian modeling the system, and h_{ts} is the height of the stance toe. Using h_{ts} , the domain and guard are given in (4).

In order to define the reset map and control system, we will need to define a holonomic constraint,

$$\eta_{ts} = (Q_{ts}, L_{ts}, \eta_{ts}),$$

with $\eta_{ts} : Q_{ts} \rightarrow \mathbb{R}^2$ the x and y position of the non-stance toe which will be enforced by a vector of Lagrange multipliers, $\lambda_{\eta_{ts}} : TQ_{ts} \times U_{ts} \rightarrow \mathbb{R}^2$, of the form $\lambda_{\eta_{ts}}(q_{ts}, \dot{q}_{ts}, u_{ts})$, which will prevent the non-stance toe from moving.

We define the kinematic constraint

$$\kappa_{ts} = (Q_{ts}, Q_{tl}, M_{ts}, \kappa_{ts}, \iota_{ts}, J_{ts})$$

with Q_{tl} introduced in the next domain, $\kappa_{ts} : Q_{ts} \rightarrow \mathbb{R}^3$ given by $\kappa_{ts} = (h_{ts}(q_{ts}), \eta_{ts}^T(q_{ts}))^T$, and ι_{ts} the identity map. Including the holonomic constraints causes the non-stance toe to have zero velocity after impact. The reset map is then given by (10) with relabeling matrix $J_{ts} = I$. Finally, the control system is given in (8).

Domain 5 (tl): The stance foot is flat on the ground and the non-stance knee is unlocked. The non-stance toe is on the ground. The configuration is shown in Fig. 2(e). The admissible control is $U_{tl} = \mathbb{R}^4$ and the guard is *toe-lift*, which occurs when the non-stance toe lifts from the ground. Therefore, we define the holonomic constraint

$$\eta_{tl} = (Q_{tl}, L_{tl}, \eta_{tl})$$

where $Q_{tl} = \mathbb{R}^4$ is the configuration space with coordinates $q_{tl} = (\theta_{st}, \theta_{nst}, \theta_{nsc}, \theta_{nsf})^T$, $L_{tl}(q_{tl}) = \frac{1}{2}\dot{q}_{tl}^T M_{tl}(q_{tl})\dot{q}_{tl} - V_{tl}(q_{tl})$ is the Lagrangian modeling the system, and $\eta_{tl} : Q_{tl} \rightarrow \mathbb{R}^2$ is a vector specifying the x and y positions of the non-stance toe. From the holonomic constraints, we calculate a vector of Lagrange multipliers. We use $(\lambda_{\eta_{tl}})_2$, (i.e., the constraining force in the y direction) to specify the domain and guard as given in (7).

The reset map is simply the identity map. Finally, the control system is given in (8) using the Lagrange multiplier derived from $\eta_{tl}(q_{tl})$, which prevents the non-stance toe from moving.

IV. FUNDAMENTAL CONTROL LAWS

In each domain, we implement a combination of five basic control laws which are given in this section; in the next section, we will discuss how these control laws are utilized on the bipedal robot being considered.

Controlled symmetries. The first law considered is controlled symmetries, motivated by [13]. This controller works by shaping the potential energy of the associated Lagrangian to that of a passive biped walking down a slope. We effectively “rotate the world” via a group action which operates on the potential energy. It was shown in [4] that a kneed walker can walk passively down a slope and further shown in [3] that controlled symmetries gives a stable gait for a kneed walker on flat ground.

Consider the group action $\Psi : \mathbb{S} \times Q \rightarrow Q$ given by

$$\Psi_\gamma(q) := q + \gamma \mathbf{1}_n,$$

for slope angle $\gamma \in \mathbb{S}$, $q \in Q$ and $\mathbf{1}_n$ a vector of 1’s of length n . From this, we define the feedback control law

$$K^\gamma(q) = B^{-1}(N(q) - N(\Psi_\gamma(q))), \quad (11)$$

where B is a linear transformation which converts from relative to absolute coordinates. Note that this control law requires full actuation (i.e., B must be full rank as we require that it be invertible). Application of this control law to the control system (f, g) yields the modified vector field:

$$f^\gamma(q, \dot{q}) = f(q, \dot{q}) + g(q)K^\gamma(q)$$

which is the vector field associated to the shaped Lagrangian:

$$L_\gamma(q, \dot{q}) = \frac{1}{2}\dot{q}^T M(q)\dot{q} - V(\Psi_\gamma(q)).$$

Underactuated controlled symmetries. Consider the case of a system in which we have $n - k$ degrees of underactuation. Motivated by our desire to apply controlled symmetries, we introduce the concept of *underactuated controlled symmetries*. Assume our coordinates are ordered such that the first k coordinates have full actuation and let $q_a = (q_1, \dots, q_k)^T$. Define the following feedback control law:

$$K^\gamma(q) = \begin{pmatrix} B_a^{-1}(N_a(q) - N_a(\Psi(q_a))) \\ \mathbf{0}_{n-k} \end{pmatrix} \quad (12)$$

where B_a represents a linear transformation, found by taking the top-left submatrix of size $k \times k$ from B and $N_a : Q \rightarrow \mathbb{R}^k$ is given by

$$N_a(q) = \begin{pmatrix} \frac{\partial V(q)}{\partial q_1} & \dots & \frac{\partial V(q)}{\partial q_{n-k}} \end{pmatrix}^T.$$

Applying this control law yields the modified vector field:

$$f^\gamma(q, \dot{q}) := f(q, \dot{q}) + g(q)K^\gamma(q).$$

Thus we have partially shaped the system by affecting only those coordinates which have full actuation.

Spring-damper controller. Motivated by the elasticity of the human ankle and the need to keep the foot from spinning freely when not on the ground, we introduce a spring-damper controller which creates forces on the system equivalent to those of a linear spring-damper system. Consider j relative angles $\Theta : Q \rightarrow \mathbb{R}^j$ of the system with angular velocities $\dot{\Theta} : TQ \rightarrow \mathbb{R}^j$ and define the feedback control law:

$$K^\Theta(q, \dot{q}) = (-k_\Theta(\Theta(q) - \Theta_0) - c_\Theta \dot{\Theta}(q, \dot{q}))B_\Theta, \quad (13)$$

with $k_\Theta > 0$ a diagonal matrix of corresponding spring constants, $c_\Theta > 0$ a diagonal matrix of viscous damping coefficients, Θ_0 the undeflected angles of the springs, and $B_\Theta = \frac{\partial \Theta}{\partial q}(0)$. Applying this yields the vector field

$$f^\Theta(q, \dot{q}) = f(q, \dot{q}) + g(q)K^\Theta(q, \dot{q}).$$

λ -zeroing controller. Recall that on certain domains, the guard is defined as a Lagrange multiplier; this multiplier must be zeroed to pass into the next domain. Note from (6) that the Lagrange multiplier depends directly on the control input u ; thus, we can affect the Lagrange multiplier by choosing the control input at the ankle for which the Lagrange multiplier is calculated. For simplicity’s sake, we choose to have the Lagrange multiplier diminish linearly over time as follows:

$$\lambda_r(t) = \lambda_0 - a(t - t_0),$$

where λ_0 and t_0 are respectively the value of the Lagrange multiplier and the time at the beginning of the domain and $a \in \mathbb{R}$ is a positive constant which affects how fast $\lambda_r(t)$ is driven to zero. Consider the Lagrange multiplier given by

$$\lambda_\eta(q, \dot{q}, u) = (A(q)M^{-1}(q)A^T(q))^{-1}(A(q)M^{-1}(q)(F(q, \dot{q}) + Bu - C(q, \dot{q})\dot{q} - N(q)) + \dot{A}(q, \dot{q})\dot{q})$$

with $F : TQ \rightarrow \mathbb{R}^n$ a vector of external forces acting on the system. Solving for the control input gives our control law:

$$K^a(q, \dot{q}, t) = (A(q)M^{-1}(q)B)^{-1}(A(q)M^{-1}(q)(C(q, \dot{q})\dot{q} + N(q) - F(q, \dot{q}) - A^T(q)\lambda_r(t)) - \dot{A}(q)\dot{q})\vec{b}$$

with \vec{b} the basis vector corresponding to the control input being used. Applying this controller gives the vector field:

$$f^a(q, \dot{q}, t) = f(q, \dot{q}) + g(q)K^a(q, \dot{q}, t)$$

Scuffing prevention controller. The final control we consider is designed to prevent scuffing. This control law is a simple feedback control law introducing an effect similar to gravity, but with a repulsive force, given by

$$K^g(q) = -\alpha e^{-\rho h_{toe}(q)} \quad (15)$$

where $\alpha, \rho \in \mathbb{R}$ are positive constants and $h_{toe} : Q \rightarrow \mathbb{R}$ is the height of the toe. α represents the strength of repulsion and ρ represents the spatial dissipation rate. Applying this controller yields the vector field:

$$f^g(q, \dot{q}) = f(q, \dot{q}) + g(q)K^g(q).$$

V. CONTROL LAW CONSTRUCTION

In this section, we describe how the control laws of Sec. IV are implemented in each domain. In all domains, we implement the *spring-damper controller* which would be replaced with a spring-damper system in a physical construct. This control law is implemented in both ankles and thus we have the control input $K^\Theta(q, \dot{q})$ as given in (13).

Domain 1 (kl). First, we implement controlled symmetries since we have full actuation. The majority of the limit cycle is spent in this domain so this controller is responsible for the majority of the energy added to the system during a limit cycle. This gives us the control input $K_{kl}^\gamma(q_{kl})$ as in (11).

Next, we implement the scuffing prevention controller to prevent the non-stance toe from hitting the ground. This gives us the control input $K_{kl}^g(q_{kl})$ as in (15). Combining these controllers with the spring-damper controller, $K_{kl}^\Theta(q_{kl}, \dot{q}_{kl})$, gives us the final control law for this domain,

$$K_{kl}^{\Theta, g, \gamma}(q_{kl}, \dot{q}_{kl}) = K_{kl}^\Theta(q_{kl}, \dot{q}_{kl}) + K_{kl}^g(q_{kl}) + K_{kl}^\gamma(q_{kl}),$$

which yields the closed-loop vector field

$$f_{kl}^{\Theta, g, \gamma}(q_{kl}, \dot{q}_{kl}) = f_{kl}(q_{kl}, \dot{q}_{kl}) + g_{kl}(q_{kl})K_{kl}^{\Theta, g, \gamma}(q_{kl}, \dot{q}_{kl}).$$

Domain 2 (hl). First, we implement controlled symmetries, which is simply an extension of the controlled symmetries

implemented in the previous domain. This gives us the control input $K_{hl}^\gamma(q_{hl})$ as in (11).

To enter the next domain, we must zero the Lagrange multiplier corresponding to the height of the stance heel, causing heel-lift. Therefore, we implement the λ -zeroing controller which gives us the control input $K_{hl}^a(q_{hl}, \dot{q}_{hl}, t)$ as in (14). Adding these controllers and the spring-damper controller, $K_{kl}^\Theta(q_{kl}, \dot{q}_{kl})$, gives the final control law

$$K_{hl}^{\Theta, a, \gamma}(q_{hl}, \dot{q}_{hl}, t) = K_{hl}^\Theta(q_{hl}, \dot{q}_{hl}) + K_{hl}^a(q_{hl}, \dot{q}_{hl}, t) + K_{hl}^\gamma(q_{hl}),$$

which yields the closed-loop vector field

$$f_{hl}^{\Theta, a, \gamma}(q_{hl}, \dot{q}_{hl}, t) = f_{hl}(q_{hl}, \dot{q}_{hl}) + g_{hl}(q_{hl})K_{hl}^{\Theta, a, \gamma}(q_{hl}, \dot{q}_{hl}, t).$$

Domain 3 (hs). In this domain, we use the λ -zeroing control law from the previous domain, $K_{hl}^a(q_{hl}, \dot{q}_{hl}, t)$, as given in (14) modifying only the control gain $a = a_{hs}$. Thus, we continue to drive λ past zero, which keeps the system rotating forward about the stance heel.

Combining this controller with the spring-damper controller, $K_{hs}^\Theta(q_{hs}, \dot{q}_{hs})$, gives us the final control law

$$K_{hs}^{\Theta, a}(q_{hs}, \dot{q}_{hs}, t) = K_{hs}^\Theta(q_{hs}, \dot{q}_{hs}) + K_{hl}^a(q_{hl}, \dot{q}_{hl}, t),$$

which yields the closed-loop vector field

$$f_{hs}^{\Theta, a}(q_{hs}, \dot{q}_{hs}, t) = f_{hs}(q_{hs}, \dot{q}_{hs}) + g_{hs}(q_{hs})K_{hs}^{\Theta, a}(q_{hs}, \dot{q}_{hs}, t).$$

Domain 4 (ts). In this domain, we implement a λ -zeroing controller using the Lagrange multiplier corresponding to the height of the non-stance toe. The control law implemented in this domain is almost identical to that described in Sec. IV except the gain a is negative which drives the value of λ in the opposite direction, causing the toe to push into the floor with greater force. This allows for a harder toe push-off, thus adding more energy to the system. The control input from this control law is $K_{ts}^a(q_{ts}, \dot{q}_{ts}, t)$ as given in (14). Adding this controller to the spring-damper controller, $K_{ts}^\Theta(q_{ts}, \dot{q}_{ts})$, gives us the final control law,

$$K_{ts}^{\Theta, a}(q_{ts}, \dot{q}_{ts}, t) = K_{ts}^\Theta(q_{ts}, \dot{q}_{ts}) + K_{ts}^a(q_{ts}, \dot{q}_{ts}, t),$$

which yields the closed-loop vector field

$$f_{ts}^{\Theta, a}(q_{ts}, \dot{q}_{ts}, t) = f_{ts}(q_{ts}, \dot{q}_{ts}) + g_{ts}(q_{ts})K_{ts}^{\Theta, a}(q_{ts}, \dot{q}_{ts}, t).$$

Domain 5 (tl). In this domain, we have 3 degrees of actuation as a result of having 1 degree of underactuation. Thus, we implement partial controlled symmetries which gives us the control input $K_{tl}^\gamma(q_{tl})$ as in (12). This allows us to add extra energy to the system, increasing the robustness.

In order to enter the next domain, the non-stance toe must lift off the ground. We effect this through the use of a λ -zeroing controller using the Lagrange multiplier specifying the height of the non-stance toe. By choosing the coefficient a , we can affect the strength with which the toe pushes off the ground and thus how much energy is added to the system. This gives us the control input $K_{tl}^a(q_{tl}, \dot{q}_{tl}, t)$ as in (14).

Combining these controllers with the spring-damper controller, $K_{tl}^\Theta(q_{tl}, \dot{q}_{tl})$, gives us the final control law,

$$K_{tl}^{\Theta, a, \gamma}(q_{tl}, \dot{q}_{tl}, t) = K_{ts}^\Theta(q_{ts}, \dot{q}_{ts}) + K_{tl}^a(q_{tl}, \dot{q}_{tl}, t) + K_{tl}^\gamma(q_{tl}),$$

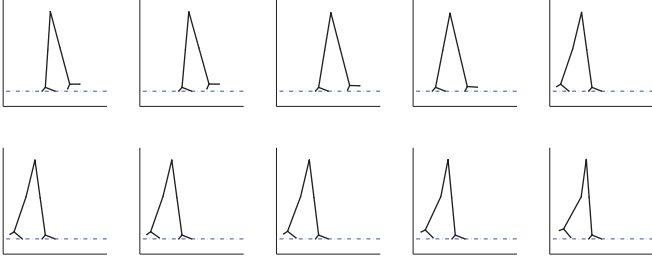


Fig. 3: Example of stable walking gait

which yields the closed-loop vector field

$$f_{tl}^{\Theta, a, \gamma}(q_{tl}, \dot{q}_{tl}, t) = f_{tl}(q_{tl}, \dot{q}_{tl}) + g_{tl}(q_{tl})K_{tl}^{\Theta, a, \gamma}(q_{tl}, \dot{q}_{tl}, t).$$

VI. SIMULATION RESULTS AND CONCLUDING REMARKS

In this section, we present simulation results for the biped and corresponding control laws being considered. We describe the simulation setup and then show that the simulation resulted in stable walking. We choose the following model parameters:

$$\begin{aligned} M &= .5 \text{ kg}, & m_t &= .5 \text{ kg}, & m_c &= 50 \text{ mg}, \\ m_f &= 2.5 \text{ mg}, & \ell &= 1 \text{ m}, & \ell_t &= .175 \text{ m}, \\ \ell_c &= 37.5 \text{ cm}, & r_h &= 5 \text{ cm}, & r_t &= 15 \text{ cm}, \\ r_f &= 5 \text{ cm}, & r_a &= 5 \text{ cm}, & \gamma &= .08 \text{ rads}, \\ k_{\Theta} &= 2 \text{ Nm/rad}, & c_{\Theta} &= .01 \text{ Nms/rad}, & \Theta_0 &= 90^\circ. \end{aligned}$$

For the λ -zeroing controller with gain a_i on domain $i \in V$, we use the following coefficients: $a_{hl} = 200$, $a_{hs} = 60$, $a_{ts} = -150$, $a_{tl} = 65$. For the scuffing prevention controller of domain kl , we use $\alpha = 1$, $\rho = 100$. We perform our simulation starting in domain kl , using as an initial condition the fixed point on the guard G_{kl} :

$$\begin{aligned} (q_{kl})_0 &= (0.0712 \quad -0.2986 \quad -0.2942 \quad -1.8629)^T, \\ (\dot{q}_{kl})_0 &= (0.8787 \quad -0.0412 \quad -7.3957 \quad -7.3118)^T. \end{aligned}$$

The gait is shown in Fig. 3. We can observe from this figure (and from Fig. 4) that the majority of the gait is spent in domain kl . This is beneficial because we implement controlled symmetries in this domain. In domains hs and ts , we do not implement controlled symmetries but rather only a simple controller to achieve the desired feet behavior. We can also see from Fig. 3 that the biped pushes off with the back foot which allows us to add more energy to the system to help make up for energy lost to impact.

Fig. 4 shows a temporal breakdown of the gait and a comparison between the model gait and a sample human gait from [1]. Note the similarity between gaits: this can be seen as an objective measure of the anthropomorphism of the model gait. This metric could be very useful in the future for measuring the extent of the anthropomorphic nature of control laws for bipedal robots.

The phase portraits of the various angles are shown in Figs. 5(b) and 5(c). Note specifically the large angular velocity of the non-stance foot. This occurs for only a short period of time after toe-lift as can be seen from Fig. 5(a) and is caused by the spring-damper controller and the scuffing prevention controller. Thus, the qualitative behavior of the phase portraits of the feet are heavily dependent upon

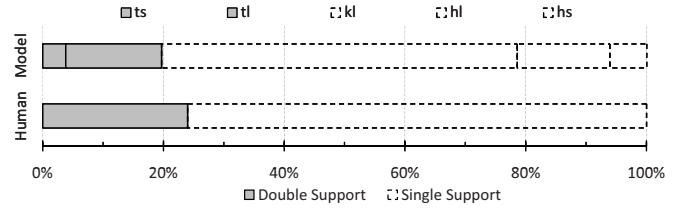


Fig. 4: Temporal comparison of model and human gaits

the parameters chosen for these two controllers. Namely, k_{Θ} , c_{Θ} , Θ_0 , α , ρ affect these phase portraits.

The primary motivation behind adding feet is to allow for the use of controlled symmetries in a realistic model. A system with point feet is necessarily underactuated and is thus incompatible with controlled symmetries. Accordingly, the behavior of the feet through most of the limit cycle does not play a significant role in the gait of the biped. It is also interesting to note that the phase portrait of the legs shown in Fig. 5(b) is qualitatively similar to the phase portrait of a 3D kneed walker without feet as in [3].

The addition of feet introduces new phenomena: toe roll and heel roll. Toe roll occurs when the stance heel lifts off the ground. Heel roll occurs if the heel of the non-stance foot strikes the ground before the toe. In both cases, the system is underactuated during these phases. These phenomena, therefore, seem undesirable, yet having them present seems to yield a more-natural-looking and energy-efficient gait as indicated by Fig. 4.

Another concern addressed by this paper is scuffing. Scuffing occurs when the non-stance foot strikes the ground at an unexpected time. Previous work has largely ignored this problem, yet it seems to play an important role in anthropomorphic gaits. That is, a biped must avoid the ground when walking and doing so has an effect on the overall gait. The scuffing prevention controller implemented in this model results in a gait that appears to be more anthropomorphic than previous gaits obtained, such as that in [3].

We verify numerically that the limit cycle is locally exponentially stable by examining the eigenvalues of the linearized Poincaré map. We calculate the Jacobian at a fixed point by perturbing along the guard. We choose our Poincaré section to be the guard of domain kl , G_{kl} , which is knee-lock. Because we are perturbing along the guard, we will always have $n - 1$ eigenvalues, where n is the dimension of the domain. Since domain kl is of dimension $n = 8$, we will have seven eigenvalues. The fixed point at which we calculate the Jacobian is

$$x^* = \begin{pmatrix} 0.07222, & -0.29912, & -0.29912, & -1.86777, \\ 0.88512, & -0.03234, & -7.41687, & -7.35685 \end{pmatrix}^T.$$

We find that the eigenvalues have magnitudes .9147, .9147, .1306, .1306, 1.382×10^{-6} , 1.382×10^{-6} , 1.104×10^{-6} . Note that all these eigenvalues have magnitude less than 1, indicating stability. In fact, most of the eigenvalues are much less than one which indicates the system is generally robust to perturbations; however, note that the first complex pair of eigenvalues has magnitude near 1,

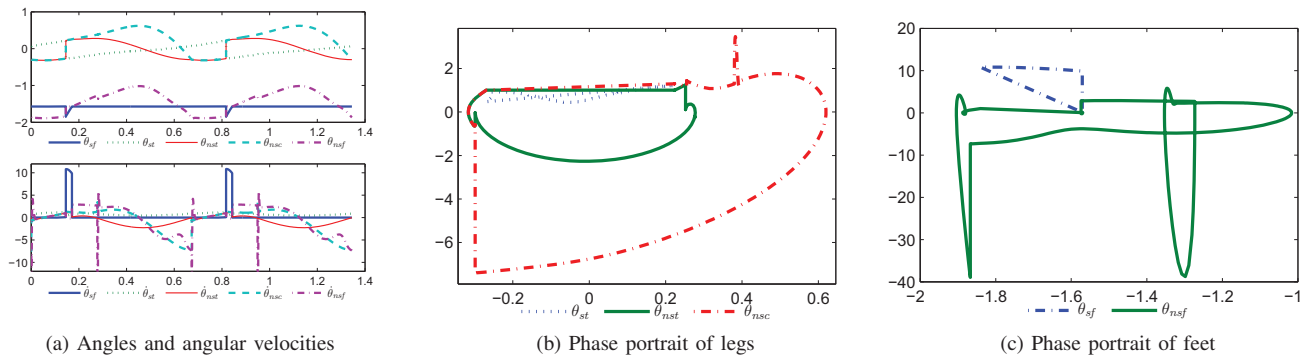


Fig. 5: Angles and phase portraits

which indicates the system is not robust to perturbations with respect to these eigenvalues. This is possibly a result of some of the controllers implemented (esp. the time-dependent controllers) and would be an interesting topic to pursue further.

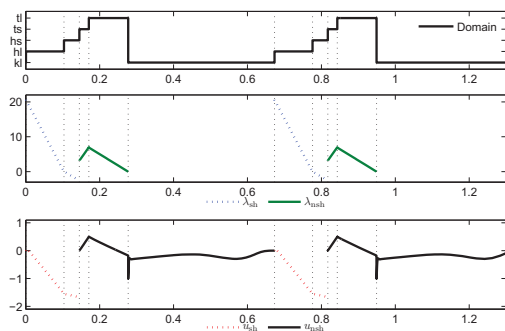


Fig. 6: Lagrange multipliers and controls

Now consider Fig. 6. The top graph shows the progression of domains over time and the middle graph shows the Lagrange multipliers at the stance heel and non-stance toe in the domains in which they are considered. Note that both Lagrange multipliers are linear—a result of the λ -zeroing controller. Also, note the slopes change depending on the value of the control gain. The bottom graph shows the control input necessary on the stance ankle or non-stance ankle (depending on the domain) to have these Lagrange multipliers be linear. Thus, we can see that our λ -zeroing controller does exactly what we expect. The domain transitions are shown as vertical dotted lines. Additionally, these graphs show the amount of time spent in each domain. Note that the value of the control input in domain *kl* results from the scuffing prevention controller and not from the λ -zeroing controller.

The bipedal model we considered, and the corresponding control laws, have certain shortcomings. Namely, the λ -zeroing controller is time-dependent, which apparently reduces the robustness of the system. Possible future research should therefore include replacing this controller with a time-independent state feedback controller. We would also like to optimize not only control gains but also physical model parameters. Finally, it would be interesting to study the robustness of the system in an objective way, e.g., by determining the domain of attraction of the limit cycle.

REFERENCES

- [1] M. Ackermann, “Dynamics and energetics of walking with prostheses,” Ph.D. dissertation, University of Stuttgart, 2007.
- [2] A. D. Ames, R. D. Gregg, and M. W. Spong, “A geometric approach to three-dimensional hipped bipedal robotic walking,” in *45th Conference on Decision and Control*, San Diego, CA, 2007.
- [3] A. D. Ames, R. W. Sinnott, and E. D. B. Wendel, “Three-dimensional knee bipedal walking: A hybrid geometric approach,” in *12th International Conference on Hybrid Systems: Computation and Control*, San Francisco, CA, 2009.
- [4] V. F. H. Chen, “Passive dynamic walking with knees: A point foot model,” Master’s thesis, MIT, 2007.
- [5] C. Chevallereau, D. Djoudi, and J. W. Grizzle, “Stable bipedal walking with foot rotation through direct regulation of the zero moment point,” *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 390–401, 2008.
- [6] J. Choi and J. W. Grizzle, “Planar bipedal walking with anthropomorphic foot action,” *IEEE Transactions on Robotics*, 2005, submitted.
- [7] —, “Planar bipedal walking with foot rotation,” *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 390–401, 2008.
- [8] S. H. Collins, M. Wisse, and A. Ruina, “A 3-d passive dynamic walking robot with two legs and knees,” *International Journal of Robotics Research*, vol. 20, pp. 607–615, 2001.
- [9] A. Goswami, “Postural stability of biped robots and the foot rotation indicator (fri) point,” *International Journal of Robotics Research*, vol. 18, no. 6, pp. 523–533, 1999.
- [10] A. Goswami, B. Thuilot, and B. Espiau, “Compass-like biped robot part I : Stability and bifurcation of passive gaits,” 1996, rapport de recherche de l’INRIA.
- [11] J. W. Grizzle, G. Abba, and F. Plestan, “Asymptotically stable walking for biped robots: Analysis via systems with impulse effects,” *IEEE Transactions on Automatic Control*, vol. 46, no. 1, pp. 51–64, 2001.
- [12] J. Lygeros, K. H. Johansson, S. Simic, J. Zhang, and S. Sastry, “Dynamical properties of hybrid automata,” *IEEE Transactions on Automatic Control*, vol. 48, pp. 2–17, 2003.
- [13] T. McGeer, “Passive dynamic walking,” *International Journal of Robotics Research*, vol. 9, no. 2, pp. 62–82, 1990.
- [14] —, “Passive walking with knees,” in *IEEE International Conference on Robotics and Automation*, Cincinnati, OH, 1990.
- [15] B. Morris and J. W. Grizzle, “A restricted Poincaré map for determining exponentially stable periodic orbits in systems with impulse effects: Application to bipedal robots,” in *44th IEEE Conference on Decision and Control and European Control Conference*, Seville, Spain, 2005.
- [16] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Taylor & Francis/CRC, 1994.
- [17] T. S. Parker and L. O. Chua, *Practical Numerical Algorithms for Chaotic Systems*. Springer-Verlag, 1989.
- [18] M. W. Spong and F. Bullo, “Controlled symmetries and passive walking,” *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 1025–1031, 2005.
- [19] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*. Taylor & Francis/CRC, 2007.
- [20] D. A. Winter, *Biomechanics and Motor Control of Human Movement*, 2nd ed. New York, NY: Wiley-Interscience, 1990.
- [21] Online content <http://www1.mengr.tamu.edu/aames/code>.