

# Control of Separable Subsystems with Application to Prostheses

Rachel Gehlhar, Jenna Reher, and Aaron D. Ames

**Abstract**—Nonlinear control methodologies have successfully realized stable human-like walking on powered prostheses. However, these methods are typically restricted to model independent controllers due to the unknown human dynamics acting on the prosthesis. This paper overcomes this restriction by introducing the notion of a *separable subsystem control law*, independent of the full system dynamics. By constructing an equivalent subsystem, we calculate the control law with local information. We build a subsystem model of a general open-chain manipulator to demonstrate the control method’s applicability. Employing these methods for an amputee-prosthesis model, we develop a *model dependent* prosthesis controller that relies solely on measurable states and inputs but is equivalent to a controller developed with knowledge of the human dynamics and states.

## I. INTRODUCTION

Powered prostheses offer many advantages over passive prostheses, such as reducing the amputee’s metabolic cost [1], [2] and increasing the amputee’s self-selected walking speed [1]. To control these prostheses, previous work focused on dividing a human gait cycle into phases and controlling each with impedance control [3]. These methods are highly heuristic and require extensive tuning. Recent work determined these impedance parameters through a bipedal robot gait generation optimization method [4]. The researchers later improved tracking and robustness of this method by adding a feedback term, formulated by a model independent quadratic program constructed with output functions [5], inspired by [6]. These successes motivate translating to prosthesis other nonlinear control methods that have proved effective for walking robots, such as feedback linearization [7], control Lyapunov functions [6], control barrier functions [8], and input to state stabilizing controllers [9]. However two problems arise when trying to translate these methods to prostheses. One, the control laws depend on the full system dynamics, but here the *human dynamics are unknown*. Two, the prosthesis dynamics depend on the full system states, but here the *human states are unknown*.

To address these problems, [10] proposed using feedback linearization on a prosthesis in stance by incorporating the interaction forces between the amputee and prosthesis into the prosthesis dynamics. However, this method requires knowledge of the interaction forces at the amputee’s hip, which cannot be measured, meaning we cannot implement this method on an actual device. Other work examined

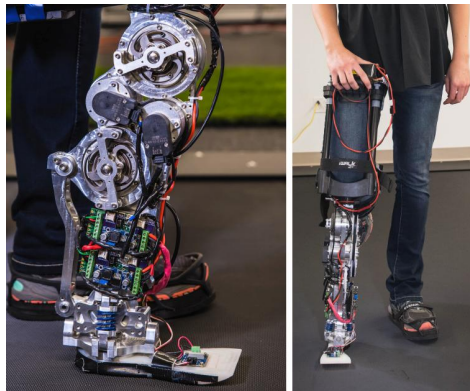


Fig. 1. Powered prosthesis AMPRO3 worn by human.

robotic systems influenced by human behavior by considering the interaction forces [11], [12]. This approach is part of a larger investigation of modeling and control of robots in contact with a dynamic environment, which Vukobratovic examined in many of his works, most comprehensively in [13]. However, these works remain focused on simple models and do not consider incorporating the interaction forces in general nonlinear control methods. Nonlinear control methods were investigated for a multi-body system in the case of multi-machine power system [14]. However, this work formulated the control methods specific to power systems limiting its applicability to other nonlinear systems. This limitation of applying nonlinear control methods to systems coupled to another dynamic system motivates developing a *general framework to control separable subsystems*.

We develop a feedback linearizing control law for a *separable subsystem*, a system independent of its full system dynamics. While this controller solely depends on the subsystem dynamics, we prove it is *equivalent* to one developed with the full system dynamics. Second, we construct the control law using an *equivalent subsystem* with measurable states and inputs. These novel methods hold potential to construct model dependent controllers for nonlinear subsystems where the dynamics of the full system are either unknown or computationally expensive. This ability could allow nonlinear control approaches to give formal guarantees on stability and safety to coupled subsystems. Here we examine the application to robotic systems by outlining construction of a subsystem of a general open-chain manipulator. Following this framework, we model a powered prosthesis, Fig. 1, as a subsystem. By using the measurable interaction forces and global orientation and velocities at the amputee’s socket, we calculate our *separable subsystem control law*, independent of both the human dynamics and states. We demonstrate the application of these ideas with simulation results.

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE1745301 and NSF NRI Grant No. 1724464.

R. Gehlhar, J. Reher, and A. Ames are with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125 USA. Emails: {rgehlhar, jreher, ames}@caltech.edu

## II. SEPARABLE SUBSYSTEM CONTROL

To analyze controller construction for the amputee-prosthesis system, consider a general affine control system,

$$\dot{x} = f(x) + g(x)u, \quad (1)$$

with states  $x \in \mathbb{R}^n$  and control inputs  $u \in \mathbb{R}^m$ , where  $m \leq n$ . The vector fields  $f$  and  $g$  are locally Lipschitz continuous, meaning given an initial condition  $x_0 = x(t_0)$ , there exists a unique solution  $x(t)$  for some time. For simplicity we assume forward completeness, i.e. solutions exist for all time. We can stabilize this system by constructing a *feedback linearizing* control law for  $u$ , which cancels out the nonlinear dynamics of the system and applies a linear controller to stabilize the resultant linear system.

**Feedback Linearization.** To begin construction of  $u$  for the affine control system (1), we define linearly independent outputs  $y : \mathbb{R}^n \rightarrow \mathbb{R}^m$  so that the system has the same number of outputs as inputs. These outputs are of vector relative degree  $\bar{\gamma} = (\gamma_1, \gamma_2, \dots, \gamma_m)$ , chosen so  $\sum_{i=1}^m \gamma_i = n$  such that the system is *full state feedback linearizable* [15]. We define the vector of partial derivatives of the  $\gamma_i - 1$  Lie derivatives [15], [16] of the  $y_i(x)$  outputs with respect to the dynamic drift vector  $f(x)$  for  $i = 1, \dots, m$  as follows:

$$\frac{\partial L_f^{\bar{\gamma}-1} y(x)}{\partial x} \triangleq \begin{bmatrix} \frac{\partial L_f^{\gamma_1-1} y_1(x)}{\partial x} \\ \frac{\partial L_f^{\gamma_2-1} y_2(x)}{\partial x} \\ \vdots \\ \frac{\partial L_f^{\gamma_m-1} y_m(x)}{\partial x} \end{bmatrix}.$$

Then, we define the vector of  $\gamma_i$  Lie derivatives with respect to drift vector  $f(x)$  and control matrix  $g(x)$ , respectively:

$$L_f^{\bar{\gamma}} y(x) \triangleq \frac{\partial L_f^{\bar{\gamma}-1} y(x)}{\partial x} f(x),$$

$$L_g L_f^{\bar{\gamma}-1} y(x) \triangleq \frac{\partial L_f^{\bar{\gamma}-1} y(x)}{\partial x} g(x).$$

Finally, we construct a feedback linearizing controller,

$$u(x) = -\underbrace{(L_g L_f^{\bar{\gamma}-1} y(x))^{-1}}_{A(x)} \underbrace{(L_f^{\bar{\gamma}} y(x) - \mu)}_{L_f^* y(x)} \quad (2)$$

$$= -A^{-1}(x)(L_f^* y(x) - \mu),$$

where  $\mu \in \mathbb{R}^m$  is the auxiliary control input the user defines to render the linearized system stable. See [15] or [16] for details. Note that  $A(x)$  is invertible because the outputs are linearly independent and the system is square [15], since the number of inputs equals the number of outputs.

**Remark 1.** Here, constructing a feedback linearizing controller requires the dynamics of the full system. However, in the case of large dimensional systems, the full dynamics may be unknown or may become computationally expensive, inhibiting feedback linearization.

### A. Control Law for Separable Subsystem

This section eliminates the need to know the full system dynamics for feedback linearization by constructing a *separable subsystem control law* that only depends on subsystem dynamics. We begin by defining a *separable control system*.

**Definition 1:** The affine control system (1) is a *separable control system* if it can be structured as

$$\begin{bmatrix} \dot{x}_r \\ \dot{x}_s \end{bmatrix} = \begin{bmatrix} f^r(x) \\ f^s(x) \end{bmatrix} + \begin{bmatrix} g_1^r(x) & g_2^s(x) \\ 0 & g^s(x) \end{bmatrix} \begin{bmatrix} u_r \\ u_s \end{bmatrix} \quad (3)$$

$$x_r \in \mathbb{R}^{n_r}, \quad x_s \in \mathbb{R}^{n_s}, \quad u_r \in \mathbb{R}^{m_r}, \quad u_s \in \mathbb{R}^{m_s},$$

where  $n_r + n_s = n$  and  $m_r + m_s = m$ .

Because of the structure of  $g(x)$  in (3),  $u_r$  only acts on part of the system. This motivates defining a *separable subsystem* independent of  $u_r$ .

**Definition 2.** For a separable control system (3), its *separable subsystem* is defined as

$$\dot{x}_s = f^s(x) + g^s(x)u_s, \quad (4)$$

which depends on the full system states  $x \in \mathbb{R}^n$ .

Now, to construct a feedback linearizing control law for this separable subsystem, we construct output functions that solely depend on the subsystem states  $x_s \in \mathbb{R}^{n_s}$  and whose Lie derivatives solely depend on the subsystem (4).

**Definition 3.** For a separable subsystem (4) of the separable control system (3), a set of linearly independent output functions with vector relative degree  $\bar{\gamma}^s = (\gamma_1^s, \gamma_2^s, \dots, \gamma_{m_s}^s)$  with respect to (3), where  $\sum_{i=1}^{m_s} \gamma_i^s = n_s$  are *separable subsystem outputs* if they only depend on  $x_s \in \mathbb{R}^{n_s}$ ,

$$y^s(x_s) \in \mathbb{R}^{m_s}, \quad (5)$$

and meet the following cross-term cancellation conditions for  $j = 1, \dots, \gamma_i^s - 1$  and  $i = 1, \dots, m_s$ :

$$\frac{\partial L_{f^s}^j y^s(x)}{\partial x_r} f^r(x) = 0, \quad (D3.1)$$

$$\frac{\partial L_{f^s}^{\gamma_i^s-1} y^s(x)}{\partial x_r} [g_1^r(x) \quad g_2^r(x)] = [0 \quad 0]. \quad (D3.2)$$

We use these outputs to introduce a *separable subsystem control law* in terms of the subsystem (4) alone.

**Definition 4.** For a separable subsystem (4) with separable subsystem outputs (5), a *separable subsystem control law* can be defined as the following feedback linearizing control law:

$$u_{\text{ssc}}(x) \triangleq -\underbrace{(L_{g^s} L_{f^s}^{\bar{\gamma}^s-1} y^s(x))^{-1}}_{A_s(x)} \underbrace{(L_{f^s}^{\bar{\gamma}^s} y^s(x) - \mu_s)}_{L_{f^s}^* y^s(x)} \quad (6)$$

$$= -A_s^{-1}(x)(L_{f^s}^* y^s(x) - \mu_s).$$

This control law is independent of the rest of the system dynamics  $f^r(x)$ ,  $g_1^r(x)$ , and  $g_2^r(x)$ , but still depends on the

full system states  $x$ . We will address this dependence in subsequent results to develop an implementable form of this control law that is solely dependent on *measurable states*.

To compare this control law  $u_{\text{ssc}}(x)$  to  $u_s(x)$ , we construct *separable outputs* for the full system that include the separable subsystem outputs  $y^s(x_s)$  used for (6).

**Definition 5.** For a separable control system, a set of linearly independent output functions with vector relative degree  $\vec{\gamma}$  are **separable outputs** if they are structured as

$$y(x) = \begin{bmatrix} y^r(x) \\ y^s(x_s) \end{bmatrix}, \quad y^r(x) \in \mathbb{R}^{m_r}, \quad y^s(x_s) \in \mathbb{R}^{m_s}, \quad (7)$$

and  $y^s(x_s)$  are separable subsystem outputs with vector relative degree  $\vec{\gamma}^s$ . The remaining outputs  $y^r(x)$  have vector relative degree  $\vec{\gamma}^r$ , where  $\sum_{i=1}^{m_r} \gamma_i^r = n_r$ , and can depend on any of the system states  $x$ . The number of subsystem outputs  $m_s$  and the number of the rest of the outputs  $m_r$  sums to  $m$ , and  $\vec{\gamma} = (\vec{\gamma}^r, \vec{\gamma}^s)$ .

For the following theorem, we define the auxiliary control input  $\mu$  as divided in the following form:

$$\mu = \begin{bmatrix} \mu_r \\ \mu_s \end{bmatrix}, \quad \mu_r \in \mathbb{R}^{m_r}, \quad \mu_s \in \mathbb{R}^{m_s}. \quad (8)$$

**Theorem 1:** For a separable control system (3), if the control law  $u(x) = (u_r(x)^T, u_s(x)^T)^T$  (2) is constructed with separable outputs (7) and auxiliary control input (8), then  $u_s(x) = u_{\text{ssc}}(x)$ .

*Proof:* We begin by relating the 3 components  $(A^{-1}(x), L_f^* y(x), \mu)$  of  $u(x)$  to the components of  $u_{\text{ssc}}(x)$ ,  $(A_s^{-1}(x), L_{f_s}^* y^s(x), \mu_s)$ . We are given by (8) that  $\mu = \begin{bmatrix} \star \\ \mu_s \end{bmatrix}$ . With condition (D3.2), we show:

$$\begin{aligned} A(x) &= \left( \begin{bmatrix} \frac{\partial L_f^{\vec{\gamma}^r - 1} y^r(x)}{\partial x_r} & \frac{\partial L_f^{\vec{\gamma}^r - 1} y^r(x)}{\partial x_s} \\ \frac{\partial L_{f_s}^{\vec{\gamma}^s - 1} y^s(x)}{\partial x_r} & \frac{\partial L_{f_s}^{\vec{\gamma}^s - 1} y^s(x)}{\partial x_s} \end{bmatrix} \begin{bmatrix} g_1^r(x) & g_2^r(x) \\ 0 & g^s(x) \end{bmatrix} \right) \\ &= \begin{bmatrix} \star & \star \\ 0 & \frac{\partial L_{f_s}^{\vec{\gamma}^s - 1} y^s(x)}{\partial x_s} g^s(x) \end{bmatrix}^{-1} = \begin{bmatrix} \star & \star \\ 0 & A_s(x) \end{bmatrix} \\ A(x)^{-1} &= \begin{bmatrix} \star & \star \\ 0 & A_s(x)^{-1} \end{bmatrix}. \end{aligned}$$

Similarly, we show:

$$\begin{aligned} L_f^* y(x) &= \begin{bmatrix} \frac{\partial L_f^{\vec{\gamma}^r - 1} y^r(x)}{\partial x_r} & \frac{\partial L_f^{\vec{\gamma}^r - 1} y^r(x)}{\partial x_s} \\ \frac{\partial L_{f_s}^{\vec{\gamma}^s - 1} y^s(x)}{\partial x_r} & \frac{\partial L_{f_s}^{\vec{\gamma}^s - 1} y^s(x)}{\partial x_s} \end{bmatrix} \begin{bmatrix} f^r(x) \\ f^s(x) \end{bmatrix} \\ &= \begin{bmatrix} \star & \star \\ \frac{\partial L_{f_s}^{\vec{\gamma}^s - 1} y^s(x)}{\partial x_s} f^s(x) \end{bmatrix} = \begin{bmatrix} \star & \star \\ L_{f_s}^* y^s(x) \end{bmatrix}. \end{aligned}$$

Putting these components together to construct  $u(x)$  yields

$$\begin{aligned} u(x) &= - \begin{bmatrix} \star & \star \\ 0 & A_s^{-1}(x) \end{bmatrix} \left( \begin{bmatrix} \star & \star \\ L_{f_s}^* y^s(x) \end{bmatrix} - \begin{bmatrix} \star \\ \mu_s \end{bmatrix} \right) \\ &= \begin{bmatrix} \star & \star \\ -A_s^{-1}(x)(L_{f_s}^* y^s(x) - \mu_s) \end{bmatrix} = \begin{bmatrix} \star \\ u_{\text{ssc}}(x) \end{bmatrix}, \end{aligned}$$

showing that  $u_s(x) = u_{\text{ssc}}(x)$  as defined in (6).

By Theorem 1, we conclude that we can construct a stabilizing controller, namely (6), for the subsystem independent of the rest of the system dynamics that is identical to the portion of the controller constructed with the full system dynamic that acts on the subsystem. This enables *model dependent* controllers to be constructed for separable subsystems without knowledge of the full system dynamics.

## B. Equivalency of Subsystems

Although we now have a subsystem control law independent of the rest of the system dynamics, it still depends on the full system states  $x \in \mathbb{R}^n$ . Consider the case where the states  $x_r \in \mathbb{R}^{n_r}$  cannot be measured. If we could construct an equivalent subsystem whose dynamics are a function of the subsystem states  $x_s \in \mathbb{R}^{n_s}$ , measurable states  $x_r \in \mathbb{R}^{n_r}$ , and measurable inputs  $\mathcal{F} \in \mathbb{R}^{n_f}$ , we could calculate the subsystem control law *independent of the full system states*.

Consider another subsystem,

$$\begin{aligned} \dot{\bar{x}}_s &= \bar{f}^s(\mathcal{X}) + \bar{g}^s(\mathcal{X}) \bar{u}_s \\ \mathcal{X} &= \begin{bmatrix} \bar{x}_r \\ \bar{x}_s \\ \mathcal{F} \end{bmatrix} = \begin{bmatrix} \bar{x}_r \\ x_s \\ \mathcal{F} \end{bmatrix} \in \mathbb{R}^{\bar{n}}, \end{aligned} \quad (9)$$

where  $\mathcal{X}$  is the state vector  $\bar{x} = (\bar{x}_r^T, x_s^T)^T$  augmented with an input  $\mathcal{F}$ . Using the same *separable subsystem outputs* as (5), we can define a control law  $\bar{u}_s(\mathcal{X})$  for the subsystem as the following:

$$\begin{aligned} \bar{u}_s(\mathcal{X}) &\triangleq - \underbrace{(L_{\bar{g}^s} L_{\bar{f}_s}^{\vec{\gamma}^s - 1} y^s(\mathcal{X}))^{-1}}_{\bar{A}_s(\mathcal{X})} \underbrace{(L_{\bar{f}_s}^{\vec{\gamma}^s} y^s(\mathcal{X}) - \mu_s)}_{L_{\bar{f}_s}^* y^s(\mathcal{X})} \\ &= -\bar{A}_s^{-1}(\mathcal{X})(L_{\bar{f}_s}^* y^s(\mathcal{X}) - \mu_s). \end{aligned} \quad (10)$$

**Theorem 2:** For the subsystems (4) and (9), if  $\exists T : \mathbb{R}^n \rightarrow \mathbb{R}^{\bar{n}}$  s.t.  $T(x) = \mathcal{X}$  and the following conditions hold,

$$\begin{aligned} f^s(x) &= \bar{f}^s(\mathcal{X}), \\ g^s(x) &= \bar{g}^s(\mathcal{X}), \end{aligned} \quad (\text{T2})$$

then  $u_s(x) = \bar{u}_s(\mathcal{X})$ . Applying these to (4) and (9), respectively, results in dynamical systems such that given the same initial condition  $\begin{bmatrix} x_{r0} \\ x_{s0} \end{bmatrix} = \begin{bmatrix} x_r(t_0) \\ x_s(t_0) \end{bmatrix}$  yields solutions  $x_s(t) = \bar{x}_s(t) \forall t \geq t_0$ .

*Proof.* Since the subsystems have the same dynamics and outputs, the Lie derivatives that comprise their control laws will be the same as well, hence

$$\begin{aligned} u_s(x) &= -A_s^{-1}(x)(L_{f_s}^* y^s(x) - \mu_s) \\ &= -\bar{A}_s^{-1}(\mathcal{X})(L_{\bar{f}_s}^* y^s(\mathcal{X}) - \mu_s) = \bar{u}_s(\mathcal{X}). \end{aligned}$$

With the same control law and dynamics, the closed-loop dynamics of the subsystems will be the same,;

$$\begin{aligned} \dot{x}_s &= f^s(x) + g^s(x)u_s(x) \\ &= \bar{f}^s(\mathcal{X}) + \bar{g}^s(\mathcal{X})\bar{u}_s(\mathcal{X}) = \dot{\bar{x}}_s. \end{aligned}$$

Hence, given the same initial condition  $\begin{bmatrix} x_{r0} \\ x_{s0} \end{bmatrix} = \begin{bmatrix} x_r(t_0) \\ x_s(t_0) \end{bmatrix}$ , they will have the same solution  $x_s(t) = \bar{x}_s(t) \forall t \geq t_0$ .

By Theorems 1 and 2, we can construct a stabilizing *model dependent* controller, namely (10), for the subsystem (4) independent of the rest of the system dynamics and with measurable states and inputs.

**Zero Dynamics.** We can apply these theorems to partially feedback linearizable systems by considering the zero dynamics' stability. The full system zero dynamics must either project the same value for the subsystem or not occur in the subsystem. A formal proof of this remains for future work. Since we only care about the feedback linearizable part of the system, we restrict our attention to full state feedback linearizable systems.

### III. SEPARABLE ROBOTIC CONTROL SYSTEMS

This method of subsystem control for separable systems can be applied to robotic control systems. While a robotic system may not initially be in the aforementioned form (3), one can construct an equivalent model by dividing the model into 2 subsystems and constraining them to each other through the use of a holonomic constraint [17]. For a model in  $\nu$ -dimensional space, one can consider this holonomic constraint as a  $\delta := \frac{\nu(\nu+1)}{2}$  DOF fixed joint. Now, the control inputs of one subsystem only affect the other subsystem through the *constraint wrench*. This construction hence decouples the dynamics of one subsystem from the control input of the other so the robotic system can be in separable system form (3).

#### A. Robotic System in Separable Form

Consider an  $\eta$  DOF robotic system in  $\nu$ -dimensional space. The coordinates  $\theta \in \mathbb{R}^\eta$  define the robot's configuration space  $\mathcal{Q}$ . Now, consider decomposing this robotic system into 2 subsystems denoted with coordinates  $\theta_l \in \mathbb{R}^{\eta_l}$  and  $\theta_s \in \mathbb{R}^{\eta_s}$ , respectively, and attached with a  $\delta$  DOF fixed joint, as previously described, denoted by coordinates  $\theta_f$ , where  $\eta_l + \eta_s + \delta = \eta$ . An example of this configuration for an amputee-prosthesis system is shown in Fig. 2.

The dynamics of a robotic system are given by the classical Euler-Lagrangian equation [18], [17]:

$$D(\theta)\ddot{\theta} + H(\theta, \dot{\theta}) = Bu + J^T(\theta)F(\theta, \dot{\theta}). \quad (11)$$

Here  $D(\theta) \in \mathbb{R}^{\eta \times \eta}$  is the inertial matrix.  $H(\theta, \dot{\theta}) = C(\theta, \dot{\theta}) + G(\theta) \in \mathbb{R}^\eta$ , a vector of centrifugal and Coriolis forces and a vector containing gravity forces, respectively. The actuation matrix  $B \in \mathbb{R}^{\eta \times m}$  contains the gear-reduction ratio of the actuated joints and is multiplied by the control inputs  $u \in \mathbb{R}^m$ , where  $m$  denotes the number of control inputs. The wrenches  $F(\theta, \dot{\theta}) \in \mathbb{R}^{\eta_h}$  enforce the  $\eta_h$  number of holonomic constraints. The Jacobian matrix of the holonomic constraints  $J(\theta) = \frac{\partial h}{\partial \theta} \in \mathbb{R}^{\eta_h \times \eta}$  enforces the holonomic constraints by the following equation:

$$\dot{J}(\theta, \dot{\theta})\dot{\theta} + J(\theta)\ddot{\theta} = 0. \quad (12)$$

Solving (11) and (12) simultaneously yields the *constrained dynamics*. These terms will now be referred to as  $D$ ,  $H$ ,  $J$ , and  $F$ , respectively, for notational simplicity.

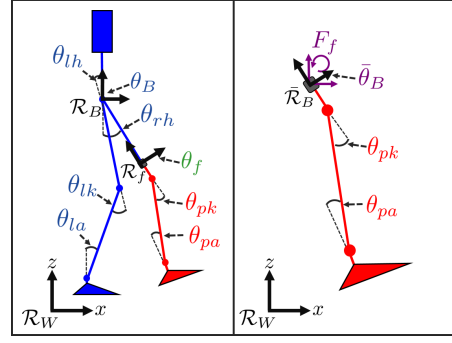


Fig. 2. (Left) Model of the full amputee-prosthesis system labeled with its generalized coordinates, where  $\theta_l$  is notated in blue,  $\theta_f$  in green, and  $\theta_s$  in red. (Right) Model of prosthesis subsystem with its generalized coordinates, where  $\theta_B$  and  $F_f$  are notated in violet.

The number of holonomic constraints  $\eta_h$  is the summation of the number of holonomic constraints for each subsystem,  $\eta_{h,l}$  and  $\eta_{h,s}$  respectively, and the fixed joint  $\delta$ , i.e.  $\eta_h = \eta_{h,l} + \eta_{h,s} + \delta$ . The wrench  $F$  includes the fixed joint wrenches  $F_f \in \mathbb{R}^\delta$ , i.e.  $F_f \subset F$ , and the Jacobian  $J$  includes the Jacobian of the fixed joint's holonomic constraints  $J_f = \frac{\partial h_f}{\partial \theta} \in \mathbb{R}^{\delta \times \eta}$ , i.e.  $J_f \subset J$ .

**Robotic System in Nonlinear Form.** Using the notation from Section II-A,  $x_s = (\theta_s^T, \dot{\theta}_s^T)^T \in \mathbb{R}^{n_s}$  will denote the states of the subsystem under consideration for control, and  $x_r = (\theta_r^T, \dot{\theta}_r^T, \theta_l^T, \dot{\theta}_l^T)^T = (\theta_r^T, \dot{\theta}_r^T)^T \in \mathbb{R}^{n_r}$  will denote the states of the rest of the system. We also define the control input  $u = (u_r^T, u_s^T)^T$ , where  $u_r \in \mathbb{R}^{m_r}$  and  $u_s \in \mathbb{R}^{m_s}$ . By defining the following functions,

$$\begin{aligned} \begin{bmatrix} \hat{f}^r(\theta, \dot{\theta})_{(\eta_l + \delta) \times 1} \\ \hat{f}^s(\theta, \dot{\theta})_{\eta_s \times 1} \end{bmatrix} &\triangleq D^{-1}(-H + J^T F), \\ \begin{bmatrix} \hat{g}_1^r(\theta, \dot{\theta})_{(\eta_l + \delta) \times m_r} & \hat{g}_2^r(\theta, \dot{\theta})_{(\eta_l + \delta) \times m_s} \\ \hat{g}_3^r(\theta, \dot{\theta})_{\eta_s \times m_r} & \hat{g}^s(\theta, \dot{\theta})_{\eta_s \times m_s} \end{bmatrix} &\triangleq D^{-1}B, \end{aligned} \quad (13)$$

we construct the robotic system in nonlinear form with  $x = (x_r^T, x_s^T)^T$ :

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ &= \underbrace{\begin{bmatrix} \dot{\theta}_r \\ \hat{f}^r(\theta, \dot{\theta}) \\ \dot{\theta}_s \\ \hat{f}^s(\theta, \dot{\theta}) \end{bmatrix}}_{f(x)} + \underbrace{\begin{bmatrix} 0 & 0 \\ \hat{g}_1^r(\theta, \dot{\theta}) & \hat{g}_2^r(\theta, \dot{\theta}) \\ 0 & 0 \\ \hat{g}_3^r(\theta, \dot{\theta}) & \hat{g}^s(\theta, \dot{\theta}) \end{bmatrix}}_{g(x)} \begin{bmatrix} u_r \\ u_s \end{bmatrix} \end{aligned} \quad (14)$$

For a coupled robotic system, the control input on one side will only affect the other through the constraint wrench at the connection. The wrench appears in the drift dynamics  $f(x)$ , resulting in  $\hat{g}_3^r(\theta, \dot{\theta}) = 0$ . We numerically verified this result for the amputee-prosthesis system under consideration in this paper. This result is produced from the structure of  $D^{-1}$  and  $B$ . Proving the former structure requires the derivation of the inverse inertia matrix and remains for future work.

**Assumption 1.** An unactuated  $\delta$  DOF joint separating a robotic subsystem from its full system in  $\nu$ -dimensional space will decouple the subsystem dynamics from the control input of the rest of the system  $u_r$ , meaning  $\hat{g}_3^r(\theta, \dot{\theta}) = 0$ .

Given Assumption 1, we write system (14) as a *separable control system* (3),

$$\begin{aligned} \begin{bmatrix} \dot{x}_r \\ \dot{x}_s \end{bmatrix} &= \begin{bmatrix} \dot{\theta}_r \\ \hat{f}^r(\theta, \dot{\theta}) \\ \dot{\theta}_s \\ \hat{f}^s(\theta, \dot{\theta}) \end{bmatrix} + \begin{bmatrix} 0 \\ \hat{g}_1^r(\theta, \dot{\theta}) \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ \hat{g}_1^s(\theta, \dot{\theta}) \\ 0 \\ \hat{g}^s(\theta, \dot{\theta}) \end{bmatrix} \begin{bmatrix} u_r \\ u_s \end{bmatrix} \\ &\triangleq \begin{bmatrix} f^r(x) \\ f^s(x) \end{bmatrix} + \begin{bmatrix} g_1^r(x) & g_2^r(x) \\ 0 & g^s(x) \end{bmatrix} \begin{bmatrix} u_r \\ u_s \end{bmatrix}, \end{aligned} \quad (15)$$

which implies a *separable subsystem* can be written as (4).

**Remark 2.** Now by Theorem 1, for any *separable subsystem outputs*  $y^s(x_s)$ , a feedback linearizing control law  $u_s(x)$  can be constructed with (6). Typically, we calculate  $u(x)$  with  $F$  in terms of  $u$  by solving (11) for  $\theta$  and substituting it into (12). Solving for  $F$  yields

$$\begin{aligned} F(\theta, \dot{\theta}) &= (JD^{-1}J^T)^{-1}(JD^{-1}(H - Bu) - \dot{J}\dot{\theta}) \\ &\triangleq \lambda_f + \lambda_g u. \end{aligned} \quad (16)$$

Then we rearrange the dynamics in (13) as  $D^{-1}(-H + J^T\lambda_f)$  and  $D^{-1}(B + J^T\lambda_g)$ , respectively. While this arrangement does not yield a separable control system, it calculates the same  $u(x)$ , hence Theorem 1 still applies.

### B. Equivalent Robotic Subsystem and Controller

The advantage of constructing a robotic system in the form (3) is applying Theorem 2 to construct an equivalent control law for a robotic subsystem without knowledge of the full system dynamics and states.

**Construction of Equivalent Robotic Subsystem.** We construct an equivalent robotic subsystem, Fig. 2, by modeling the links and joints of the full model denoted by coordinates  $\theta_s$ . We place a  $\delta$  DOF base frame, denoted by  $\bar{\theta}_B$ , at the end of the link that connects to the fixed joint in the full model. At these base coordinates we project the interaction forces and moments  $F_f$  with  $\bar{J}_f^T = (R_{Wf}(\bar{\theta}_B)[I_{\delta \times \delta} \ 0_{\delta \times n_s}])^T$ . Here  $R_{Wf}^T(\bar{\theta}_B)$  rotates the wrench  $F_f$  from the fixed joint reference frame  $\mathcal{R}_f$  into the world frame  $\mathcal{R}_W$ . With configuration coordinates  $\bar{\theta} = (\bar{\theta}_B; \theta_s)$ , the *constrained subsystem dynamics* equations are

$$\begin{aligned} \bar{D}(\bar{\theta})\ddot{\bar{\theta}} + \bar{H}(\bar{\theta}, \dot{\bar{\theta}}) &= \bar{B}\bar{u} + \bar{J}^T(\bar{\theta})\bar{F}(\bar{\theta}, \dot{\bar{\theta}}) + \bar{J}_f^T(\bar{\theta})F_f, \\ \bar{J}\ddot{\bar{\theta}} + \bar{J}\dot{\bar{\theta}} &= 0, \end{aligned} \quad (17)$$

where  $\bar{J}(\bar{\theta}) = \frac{\partial \bar{h}}{\partial \bar{\theta}} \in \mathbb{R}^{\eta_{h,s} \times (\delta + \eta_s)}$  is the Jacobian of the holonomic constraints acting on the subsystem, apart from those for the fixed joint. These terms will now be referred to as  $\bar{D}$ ,  $\bar{H}$ ,  $\bar{J}$ ,  $\bar{J}_f$ , and  $\bar{F}$ , respectively.

**Transformation for Subsystem States and Inputs.** Using the notation from Section II-B, with  $x_s$  from the full system,  $\bar{x}_r = (\bar{\theta}_B^T, \dot{\bar{\theta}}_B^T)^T \in \mathbb{R}^{\bar{n}_r}$ , and  $\mathcal{F} = F_f \in \mathbb{R}^{\eta_f}$ , we construct the robotic subsystem in nonlinear form. We relate the subsystem's augmented state vector  $\mathcal{X}$  to the full system states  $x$ . The base coordinates  $\bar{\theta}_B$  relate to the full system through the forward kinematics of the fixed joint reference frame  $\mathcal{R}_f$  with respect to the world frame  $\mathcal{R}_W$ ,  $\bar{\theta}_B = g_{Wf}(\theta)$  [17]. Their velocities  $\dot{\bar{\theta}}_B$  relate by the *spatial manipulator Jacobian*,  $\dot{\bar{\theta}}_B = J_{Wf}^s(\theta)\dot{\theta}$  [17]. To obtain an

expression for  $F_f$  based on the equation for  $F(\theta, \dot{\theta})$  given in (16), we define the transformation  $\iota : \mathbb{R}^{\eta_h} \rightarrow \mathbb{R}^{\delta}$  where  $\iota_f(F) = \iota_f(F \supset F_f) = F_f$ . Hence the transformation  $T(x) = \mathcal{X}$  is defined as:

$$T(x) \triangleq \begin{bmatrix} g_{Wf}(\theta) \\ J_{Wf}^s(\theta)\dot{\theta} \\ x_s \\ \iota_f(F(\theta, \dot{\theta})) \end{bmatrix} = \begin{bmatrix} \bar{\theta}_B \\ \dot{\bar{\theta}}_B \\ x_s \\ F_f \end{bmatrix} = \mathcal{X}.$$

**Robotic Subsystem in Nonlinear Form.** By defining the following functions,

$$\begin{aligned} \begin{bmatrix} \hat{f}^r(\bar{\theta}, \dot{\bar{\theta}}, F_f)_{\delta \times 1} \\ \hat{f}^s(\bar{\theta}, \dot{\bar{\theta}}, F_f)_{\eta_s \times 1} \end{bmatrix} &\triangleq \bar{D}^{-1}(-\bar{H} + \bar{J}^T\bar{F} + \bar{J}_f^T F_f), \\ \begin{bmatrix} \hat{g}^r(\bar{\theta}, \dot{\bar{\theta}})_{\delta \times m_s} \\ \hat{g}^s(\bar{\theta}, \dot{\bar{\theta}})_{\eta_s \times m_s} \end{bmatrix} &\triangleq \bar{D}^{-1}\bar{B}, \end{aligned} \quad (19)$$

we construct the equivalent robotic subsystem as (9):

$$\begin{aligned} \dot{x}_s &= \begin{bmatrix} \dot{\theta}_s \\ \hat{f}^s(\bar{\theta}, \dot{\bar{\theta}}, F_f) \end{bmatrix} + \begin{bmatrix} 0_{\eta_s \times m_s} \\ \hat{g}^s(\bar{\theta}, \dot{\bar{\theta}}) \end{bmatrix} \bar{u}_s \\ &\triangleq \bar{f}^s(\mathcal{X}) + \bar{g}^s(\mathcal{X})\bar{u}_s. \end{aligned} \quad (20)$$

We numerically verified condition (T2) is met for the amputee-prosthesis system under consideration for this paper.

**Assumption 2.** For the given structure of the robotic subsystem model, condition (T2) is met.

**Remark 3.** With Assumption 2, Theorem 2 applies, enabling users to construct controllers for a robotic subsystem, without knowledge of the full dynamics and states, given the constraint forces and moments  $F_f$  at the fixed joint and its global coordinates  $\bar{\theta}_B$  and velocities  $\dot{\bar{\theta}}_B$ . This control law yields the same evolution of subsystem states  $x_s$  as those of the full system under the control law  $u(x)$ . As explained for the full system, we typically solve  $\bar{F}$  in terms of  $\bar{u}_s$  and calculate  $\bar{u}_s$  with rearranged dynamics in (19). While this form does not meet (T2), Theorem 2 still holds since the method calculates the same  $\bar{u}_s$ .

## IV. AMPUTEE-PROSTHESIS APPLICATION

The control methods presented in II-A and II-B apply to an amputee-prosthesis system by modeling it as a 2D bipedal robot per the methods of [18] with the addition of a fixed joint at the socket per the methods in III-A. The prosthesis subsystem is constructed according to the method in III-B. The parameters for the human limbs are estimated based on the user's total height and mass. The length, mass, and COM are calculated based on Plagenhoef's table of percentages [19]. The inertia of each limb is calculated based on Erdmann's table of radiuses of gyration [20]. The prosthesis parameters are based on a CAD model of AMPRO3 [21], a powered transfemoral prosthesis, Fig. 1.

**Generalized Coordinates.** For the generalized coordinates  $\theta = (\theta_l^T, \theta_f^T, \theta_s^T)^T$  of III, we define  $\theta_l$  as  $(\theta_B^T, \theta_l^T, \theta_{rl}^T)^T$ . Here  $\theta_B \in SE(2)$  are the extended coordinates representing position and rotation of the full system's base frame  $\mathcal{R}_B$  with

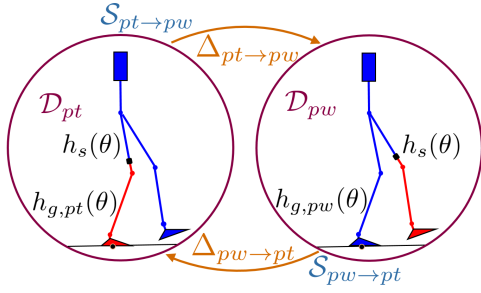


Fig. 3. The walking pattern of an asymmetric amputee-prosthesis system can be represented as a 2-step *directed cycle*.

respect to the world frame  $\mathcal{R}_W$ , and  $\theta_{ll} = (\theta_{lh}^T, \theta_{lk}^T, \theta_{la}^T)^T$  and  $\theta_{rl} = \theta_{rh}$  are the relative joint angles of the human's left and right leg, respectively. The subsystem coordinates  $\theta_s = (\theta_{pk}^T, \theta_{pa}^T)^T$  denote the relative joint angles of the prosthesis knee and ankle, respectively. See Fig. 2.

### A. Hybrid Systems

To model human locomotion which consists of both continuous and discrete dynamics, a hybrid system is employed. This *multi-domain hybrid control system* is defined as a tuple,

$$\mathcal{H}\mathcal{C} = (\Gamma, \mathcal{D}, \mathcal{U}, S, \Delta, FG),$$

where:

- $\Gamma = (V, E)$  is a *directed cycle*, with vertices  $V = \{v_1 = pt, v_2 = pw\}$  and edges  $E = \{e_1 = \{pt \rightarrow pw\}, e_2 = \{pw \rightarrow pt\}\}$ ,
- $\mathcal{D} = \{\mathcal{D}_v\}_{v \in V}$ , set of *domains of admissibility*,
- $\mathcal{U} = \{\mathcal{U}_v\}_{v \in V}$ , set of *admissible control inputs*,
- $S = \{S_e\}_{e \in E}$ , set of *guards* or *switching surfaces*,
- $\Delta = \{\Delta_e\}_{e \in E}$ , set of *reset maps* that define the discrete transitions triggered at  $S_e$ ,
- $FG = \{(f_v, g_v)\}_{v \in V}$  with  $(f_v, g_v)$  a *control system* on  $\mathcal{D}_v$ , that defines the continuous dynamics  $\dot{x} = f_v(x) + g_v(x)u_v$  for each  $x \in \mathcal{D}_v$  and  $u_v \in \mathcal{U}$ .

A detailed explanation of each hybrid system component can be found in [7]. See Fig. 3.

**Continuous Domains of Full System.** The dynamics of the full system are given by (11), and control system by (14). For each domain, holonomic constraints are used to model the foot contact points present in human walking behavior [18]. Above,  $pt$  represents “prosthesis stance” where the prosthetic leg’s foot is constrained to the ground with the 3 DOF constraint  $h_{g,pt}(\theta)$ , and the human leg is swinging. The leg positions are switched for “prosthesis swing” ( $pw$ ), where the 3 DOF constraint  $h_{g,pw}(\theta)$  constrains the human foot to the ground. Both domains have a holonomic constraint  $h_s(\theta)$  for the 3 DOF fixed joint at the socket. See Fig. 3. For this system,  $m_s = 2$  for the number of actuated joints in the prosthesis subsystem  $(\theta_{pk}, \theta_{pa})$ , and  $m_r = 4$  for the rest of the of actuated joints  $(\theta_{lh}, \theta_{lk}, \theta_{la}, \theta_{rh})$ .

**Continuous Domains for Prosthesis Subsystem.** The continuous dynamics for both domains are defined by (17) where  $\bar{J}^T$  is only present for *prosthesis stance* when  $\bar{h}_{g,pt}$  defines the prosthesis foot contact. The control system is defined by (20). As explained in II-B and III-B, the prosthesis subsystem

must measure its states  $\bar{x}_r = (\bar{\theta}_B^T, \dot{\bar{\theta}}_B^T)^T$  and input  $\mathcal{F} = F_f$ . In  $\mathcal{D}_{pt}$ ,  $\bar{\theta}_B$  and  $\dot{\bar{\theta}}_B$  can be calculated using inverse kinematics of its own joints since the foot is assumed to be flat on the ground. In  $\mathcal{D}_{pw}$ , an IMU on the amputee’s amputated thigh can provide the global rotation and velocities. The global positions are not required since they do not appear in the dynamics. Both domains require a load cell at the socket interface to measure the interaction forces  $F_f$ .

**Guards and Reset Maps.** For both domains, the condition for the guard  $S$  is met when the swing leg reaches the ground. The reset map  $\Delta$  dictates the discrete dynamics that occur assuming a perfectly plastic impact. See [22] for details.

### B. Prosthesis Gait Design

To control the prosthesis to yield stable human-like walking for the entire amputee-prosthesis system, we design walking trajectories for outputs of the full system. We define *separable outputs* to enable separable subsystem control:

- hip velocity:  $y_{1,vhip}^a = (r_{sk} + r_{sa})\dot{\theta}_{sk} + r_{sa}\dot{\theta}_{sa}$
- stance calf:  $y_{2,sc}^a = -\theta_{sk} - \theta_{sa}$ ,
- stance hip:  $y_{2,sh}^a = -\theta_{sh}$ ,
- non-stance hip:  $y_{2,nsh}^a = -\theta_{nsh}$ ,
- non-stance knee:  $y_{2,nsk}^a = \theta_{nsk}$ ,
- non-stance ankle:  $y_{2,nsa}^a = \theta_{nsa}$ .

Here  $r_{sk}$  and  $r_{sa}$  are the stance knee and ankle limb lengths, respectively. The angles  $(\theta_{sh}, \theta_{sk}, \theta_{sa}, \theta_{nsh}, \theta_{nsk}, \theta_{nsa})$  are defined as  $(\theta_{rl}, \theta_s, \theta_{ll})$  for  $\mathcal{D}_{pt}$  and  $(\theta_{ll}, \theta_{rl}, \theta_s)$  for  $\mathcal{D}_{pw}$ . The *separable subsystem outputs* are defined as  $y_{s,pt}(x_s) = (y_1, y_{2,sc})$  for  $\mathcal{D}_{pt}$  and  $y_{s,pw}(x_s) = (y_{2,nsk}, y_{2,nsa})$  for  $\mathcal{D}_{pw}$ . The remaining outputs for each domain define  $y_{r,pt}(x)$  and  $y_{r,pw}(x)$ , respectively.

**Virtual Constraints.** To restrict the actual motion  $y^a$  of a robotic system to a desired trajectory  $y^d$ , we drive the difference between these functions to 0. We define these differences as *virtual constraints* [23],

$$\begin{aligned} y_{1,v}(\theta, \dot{\theta}) &= y_{1,v}^a(\theta, \dot{\theta}) - y_{1,v}^d, \\ y_{2,v}(\theta) &= y_{2,v}^a(\theta) - y_{2,v}^d(\tau, \alpha), \end{aligned}$$

for  $v \in V$ , where  $y_{1,v}$  and  $y_{2,v}$  are relative degree 1 and 2 virtual constraints, respectively [15]. Here  $y_{1,v}^d$  is defined as the forward hip velocity  $v_{hip,v}$ , which appeared approximately constant in human locomotion data [24], and  $y_{2,v}^d(\tau, \alpha)$  is defined as a 6<sup>th</sup> order Bézier polynomial. The values  $v_{hip,v}$  and  $\alpha_v$  are determined through an optimization method that will be described. Here  $\tau$  is a phase variable that measures the forward progression of the walking trajectory and enables development of a robust state-feedback controller for the prosthesis [23].

**Phase Variable.** Previous work found the forward progression of the stance hip  $p_{hip}$  to be monotonic during a human step cycle [24], qualifying it to be a phase variable for walking trajectories. We linearize this expression as  $\delta p_{hip}(\theta)$ , a function of  $(\theta_{sk}, \theta_{sa})$ , and formulate it as a phase variable:

$$\tau(x) = \frac{\delta p_{hip}(\theta) - \delta p_{hip}^+}{\delta p_{hip}^- - \delta p_{hip}^+}. \quad (21)$$

Here  $\delta p_{hip}^+$  and  $\delta p_{hip}^-$  are the starting and ending positions in a step, respectively, determined through the optimization method. In  $\mathcal{D}_{pt}$ , the prosthesis can calculate the phase variable, but in  $\mathcal{D}_{pw}$ , it needs an IMU on the human's stance leg to measure the phase variable. It can differentiate this signal to calculate the derivatives  $\dot{\tau}$  and  $\ddot{\tau}$ .

**Partial Hybrid Zero Dynamics and Optimization.** The controller (2) drives the outputs  $y_{1,v}$  and  $y_{2,v}$  to 0 over the continuous dynamics of the domain, rendering invariant zero dynamics. However, the zero dynamics do not necessarily remain invariant through impacts. In fact, enforcing impact invariance on the velocity-modulating output is too restrictive due to the jump of velocities by the impact map. Hence, we only enforce an impact invariance condition on the relative degree 2 outputs, resulting in *partial zero dynamics*:

$$\mathbf{PZ}_{\alpha_v} = \{(\theta, \dot{\theta}) \in \mathcal{D}_v : y_{2,v}(\theta, \alpha_v) = 0, \dot{y}_{2,v}(\theta, \dot{\theta}, \alpha_v) = 0\}.$$

We design the trajectories with *partial hybrid zero dynamics* (PHZD) constraints for each transition,

$$\Delta_{e_i}(S_{e_i} \cap \mathbf{PZ}_{\alpha_{v_i}}) \subseteq \mathbf{PZ}_{\alpha_{v_{i+1}}},$$

in a direct collocation based multi-domain HZD gait optimization approach, as described in [25]. Here  $v_{i+1}$  indicates the next domain in the directed cycle. The solution gives the sets of  $v_{hip,v}$  and  $\alpha_v$  for each domain  $\mathcal{D}_v$ , defining the desired output functions  $y_v^d$  to yield stable amputee-prosthesis human-like walking.

### C. Prosthesis State-based Control

Since  $\tau$  is a function of  $x$  for  $\mathcal{D}_{pt}$ ,  $x$  is the only time dependent variable in the output functions  $y_{pt}(x, \alpha)$ , meaning a control law for the full system  $u_{pt}(x) = (u_{r,pt}(x)^T, u_{s,pt}(x)^T)^T$  can be defined by (2) and for the subsystem  $\bar{u}_{s,pt}(\mathcal{X})$  by (10). By Theorems 1 and 2,  $u_{s,pt}(x) = \bar{u}_{s,pt}(\mathcal{X})$ .

For  $\mathcal{D}_{pw}$ ,  $\tau$  is measured and hence not a function of  $x$ , so we define the control laws slightly differently to account for the time dependency of  $\tau$ ,

$$u_{pw}(x, \mathcal{T}) = \begin{bmatrix} u_{r,pw}(x, \mathcal{T}) \\ u_{s,pw}(x, \mathcal{T}) \end{bmatrix} = -A(x)^{-1}(L_f^* y(x) - \begin{bmatrix} \dot{y}_{1,pw}^d \\ \dot{y}_{2,pw}^d \end{bmatrix} - \mu), \quad (22)$$

where  $\mathcal{T} = (\tau, \dot{\tau}, \ddot{\tau})$ ,  $A(x)$  and  $L_f^* y(x)$  are given in (2), and

$$\begin{bmatrix} \dot{y}_{1,pw}^d \\ \dot{y}_{2,pw}^d \end{bmatrix} = \begin{bmatrix} \frac{\partial y_{1,pw}^d(\tau, \alpha_{pw})}{\partial \tau} \dot{\tau} \\ \frac{\partial^2 y_{2,pw}^d(\tau, \alpha_{pw})}{\partial \tau^2} \dot{\tau}^2 + \frac{\partial y_{2,pw}^d(\tau, \alpha_{pw})}{\partial \tau} \ddot{\tau} \end{bmatrix}.$$

Note the Lie derivatives of  $y_{pw}(x)$  result in only being with respect to  $y_{pw}^a(x)$  since  $y_{pw}^d(\tau, \alpha)$  is not a function of  $x$ . A feedback linearizing controller for the subsystem is given by

$$\bar{u}_{s,pw}(\mathcal{X}, \mathcal{T}) = -A_s^{-1}(\mathcal{X})(L_{f^s}^* y^s(\mathcal{X}) - \ddot{y}_{2,pw}^{s,d} - \mu_s), \quad (23)$$

where  $A_s(\mathcal{X})$  and  $L_{f^s}^* y^s(\mathcal{X})$  are defined as in (10) and

$$\ddot{y}_{2,pw}^{s,d} = \frac{\partial^2 y_{2,pw}^{s,d}(\tau, \alpha_{pw})}{\partial \tau^2} \dot{\tau}^2 + \frac{\partial y_{2,pw}^{s,d}(\tau, \alpha_{pw})}{\partial \tau} \ddot{\tau}.$$

Here also, the Lie derivatives of  $y_{pw}^s(x_s)$  result in only being with respect to  $y_{pw}^{s,a}$  since  $y_{pw}^{s,d}$  is not a function of  $x$ .

**Proposition 1.** For a separable control system (3) with separable outputs (7), if the control input  $u_{pw}(x, \mathcal{T}) = (u_{r,pw}(x, \mathcal{T})^T, u_{s,pw}(x, \mathcal{T})^T)^T$  is constructed as a feedback linearizing controller of the form (22) with  $\mu$  structured as (8), then  $u_{s,pw}(x, \mathcal{T}) = \bar{u}_{s,pw}(\mathcal{X}, \mathcal{T})$ .

*Proof:* Extending the proof for Theorem 1, we need only to show that

$$\begin{bmatrix} \dot{y}_{1,pw}^d \\ \dot{y}_{2,pw}^d \end{bmatrix} = \begin{bmatrix} \dot{y}_{1,pw}^d \\ \ddot{y}_{2,pw}^d \end{bmatrix} = \begin{bmatrix} \star \\ \dot{y}_{2,pw}^{s,d} \end{bmatrix}.$$

Including this component with the 3 components given in the proof of Theorem 1 yields  $u_{pw}(x, \tau) = [\bar{u}_{s,pw}^*]$ , showing showing that  $u_{s,pw}(x, \mathcal{T}) = \bar{u}_{s,pw}(\mathcal{X}, \mathcal{T})$ .

**Main Result.** Feedback linearization can be performed on the prosthesis subsystem with limited information of the human system and yield the same control law as when a controller is constructed for the prosthesis with full knowledge of the human dynamics and states. Although the domains are fully-actuated, zero dynamics exist due to the relative degree 1 output. Since we constructed the outputs to satisfy PHZD conditions, the zero dynamics are stable. Hence, the theorems presented in this paper still apply.

## V. RESULTS AND CONCLUSIONS

**Simulation Results.** An amputee-prosthesis model was built according to the procedure in IV for a 65.8 kg 1.73 m female. Gaits were generated for this model according to IV-B and the hybrid system IV-A was simulated for 100 steps with  $u_{pt}(x)$  and  $u_{pw}(x, \tau)$  acting on the full system for  $\mathcal{D}_{pt}$  and  $\mathcal{D}_{pw}$ , respectively. The gaits were simulated again for 100 steps with  $u_{pt,r}(x)$  and  $\bar{u}_{pt,s}(\mathcal{X})$  acting on the human and prosthesis joints, respectively, for  $\mathcal{D}_{pt}$ , and  $u_{pw,r}(x, \tau)$  and  $\bar{u}_{pw,s}(\mathcal{X}, \tau)$  for  $\mathcal{D}_{pw}$ . Fig. 4 depicts the prosthesis control inputs for each simulation, showing they match identically, demonstrating Theorem 1 and 2. The control laws enforced the trajectories found from the optimization and yielded stable amputee-prosthesis walking in simulation, as can be seen in <https://youtu.be/UGLE4xfT9uY>.

The prosthesis subsystem was simulated for the continuous domains with values of  $\bar{\theta}_B$ ,  $\dot{\theta}_B$ , and  $F_f$  obtained from the full system simulation. Additionally during  $\mathcal{D}_{pw}$ ,  $\tau$ ,  $\dot{\tau}$ , and  $\ddot{\tau}$  from the full system were fed to the prosthesis simulation. Each domain transitioned to the next domain when  $\tau = 1$ . The post-impact states were obtained from the full system simulation. These subsystem states from the prosthesis simulation and full system simulation are shown in Fig. 5, again showing they match, demonstrating Theorem 2. These phase portraits also show the prosthesis followed stable periodic orbits. A second model was modified with human parameters increased by 24.9 kg and simulated using the same trajectories. Since the prosthesis only relies on the force measurement and global position and velocities, it still achieved perfect tracking of the outputs, shown in Fig. 6.

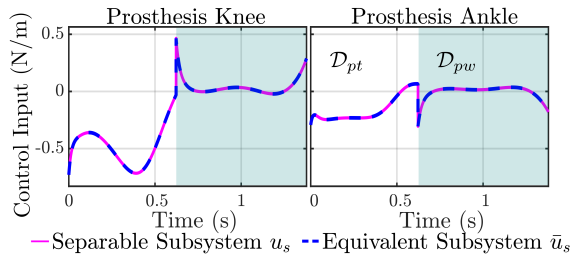


Fig. 4. Control inputs of prosthesis knee (left) and prosthesis ankle (right) for subsystem control law  $u_s$  and  $\bar{u}_s$  over stance and nonstance domains.

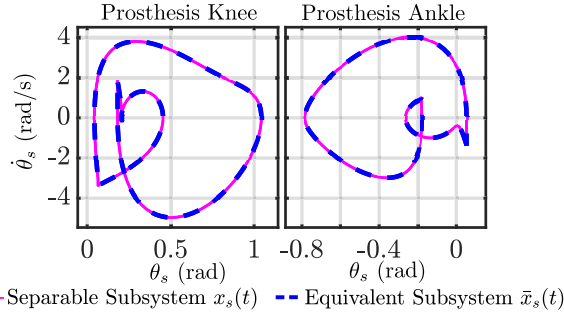


Fig. 5. Phase portraits of prosthesis knee (left) and prosthesis ankle (right) for stance and nonstance domains.

**Conclusions.** This paper presented a novel framework for controlling systems that are separable, resulting in a controller for the subsystem, equivalent to a controller with knowledge of the full states and dynamics. This formulation enables users to develop model dependent controllers for subsystems with limited information about the full system. Further, we outline how to isolate a subsystem from an open-chain manipulator. This decomposition allows robotic modules to be controlled without knowledge of its full system and provides a modeling method for robots that interact with another dynamic system. By applying these methods to a prosthesis, we developed the first model-dependent prosthesis controller, solely dependent on measurable quantities. These results could, therefore, yield new nonlinear control approaches to prostheses that give formal guarantees on stability and safety while being applicable to real-world devices.

## REFERENCES

- [1] E. C. Martinez-Villalpando, L. Mooney, G. Elliott, and H. Herr, "Antagonistic active knee prosthesis. a metabolic cost of walking comparison with a variable-damping prosthetic knee," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 8519–8522, Aug 2011.
- [2] S. K. Au, J. Weber, and H. Herr, "Powered ankle-foot prosthesis improves walking metabolic economy," *IEEE Transactions on Robotics*, vol. 25, pp. 51–66, Feb 2009.
- [3] F. Sup, A. Bohara, and M. Goldfarb, "Design and control of a powered transfemoral prosthesis," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 263–273, 2008. PMID: 19898683.
- [4] N. Aghasadeghi, H. Zhao, L. J. Hargrove, A. D. Ames, E. J. Perreault, and T. Bretl, "Learning impedance controller parameters for lower-limb prostheses," in *Intelligent robots and systems (IROS), 2013 IEEE/RSJ international conference on*, pp. 4268–4274, IEEE, 2013.
- [5] H. Zhao and A. D. Ames, "Quadratic program based control of fully-actuated transfemoral prosthesis for flat-ground and up-slope locomotion," in *American Control Conference (ACC), 2014*, pp. 4101–4107, IEEE, 2014.
- [6] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle, "Rapidly exponentially stabilizing control Lyapunov functions and hybrid zero

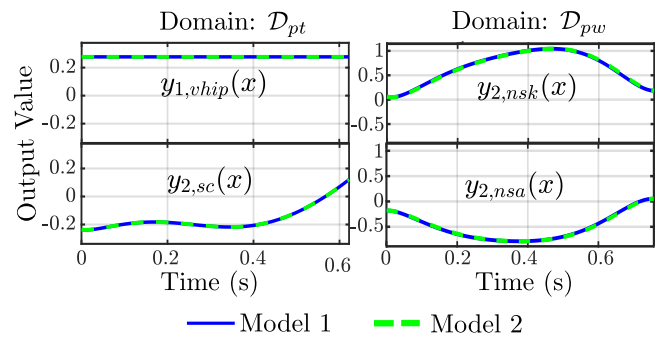


Fig. 6. Output functions of prosthesis knee and ankle for  $\mathcal{D}_{pt}$  (right) and  $\mathcal{D}_{pw}$  (left) for 65.8 kg Model 1 and 90.7 kg Model 2.

- dynamics," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 876–891, 2014.
- [7] A. D. Ames, "Human-inspired control of bipedal walking robots," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1115–1130, 2014.
- [8] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pp. 6271–6278, IEEE, 2014.
- [9] S. Kolathaya, J. Reher, and A. D. Ames, "Input to state stability of bipedal walking robots: Application to DURUS," *arXiv preprint arXiv:1801.00618*, 2018.
- [10] A. E. Martin and R. D. Gregg, "Hybrid invariance and stability of a feedback linearizing controller for powered prostheses," in *2015 American Control Conference (ACC)*, pp. 4670–4676, July 2015.
- [11] G. B. Chung, B. Yi, and W. Kim, "Modeling and control of interaction forces in dynamically-coupled robotic systems," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1476–1483, Oct 2006.
- [12] K. Khalaf, D. Gan, and H. Hemami, "Dynamics and control of separable coupled rigid body systems," *Robotics and Biomimetics*, vol. 4, p. 14, Nov 2017.
- [13] M. Vukobratovic, V. Matijevic, and V. Potkonjak, *Dynamics of Robots with Contact Tasks*.
- [14] Y. Wang, G. Guo, and D. J. Hill, "Robust decentralized nonlinear controller design for multimachine power systems," *Automatica*, vol. 33, no. 9, pp. 1725 – 1733, 1997.
- [15] S. Sastry, *Nonlinear Systems: Analysis, Stability, and Control*. Interdisciplinary Applied Mathematics, Springer New York, 2013.
- [16] H. Khalil, *Nonlinear Systems*. Pearson Education, Prentice Hall, 2002.
- [17] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL, USA: CRC Press, Inc., 1st ed., 1994.
- [18] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames, "Models, feedback control, and open problems of 3d bipedal robotic walking," *Automatica*, vol. 50, no. 8, pp. 1955 – 1988, 2014.
- [19] S. Plagenhoef, F. G. Evans, and T. Abdelnour, "Anatomical data for analyzing human motion," *Research Quarterly for Exercise and Sport*, vol. 54, no. 2, pp. 169–178, 1983.
- [20] W. Erdmann, "Geometry and inertia of the human body - review of research," vol. 1, no. 1, pp. 23–35, 1999.
- [21] H. Zhao, E. Ambrose, and A. D. Ames, "Preliminary results on energy efficient 3D prosthetic walking with a powered compliant transfemoral prosthesis," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pp. 1140–1147, IEEE, 2017. BestMedical Robotics Paper Award Finalist of ICRA 2017.
- [22] Y. Hurmuzlu and D. B. Marghitu, "Rigid body collisions of planar kinematic chains with multiple contact points," *The International Journal of Robotics Research*, vol. 13, no. 1, pp. 82–92, 1994.
- [23] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, and et al., "Feedback control of dynamic bipedal robot locomotion," 2007.
- [24] S. Jiang, S. Partrick, H. Zhao, and A. D. Ames, "Outputs of human walking for bipedal robotic controller design," in *American Control Conference (ACC), 2012*, pp. 4843–4848, IEEE, 2012.
- [25] A. Hereid, E. A. Cousineau, C. M. Hubicki, and A. D. Ames, "3D dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pp. 1447–1454, IEEE, 2016.