

# Hierarchical Feedback Control for Complex Hybrid Models of Multiagent Legged Robotic Systems

Kaveh Akbari Hamed<sup>1</sup>, Wen-Loong Ma<sup>2</sup>, Vinay R. Kamidi<sup>1</sup>, and Aaron D. Ames<sup>2</sup>

**Abstract**—This paper presents a hierarchical feedback control strategy for complex hybrid systems that represent collaborative multiagent legged robotic systems with arms for manipulating an object. We develop high-dimensional hybrid models, including continuous- and discrete-time dynamics, for multiagent legged systems. Each agent has its own baseline controller to produce an exponentially stable legged locomotion pattern. For the manipulating purpose of the object, a two-level control strategy is proposed. At the higher level, a model predictive control based convex quadratic programming (QP) computes the required forces to be generated by the end effectors (EEs). The lower-level controllers are then set up based on convex QPs to refine the baseline controllers while generating the prescribed forces at the EEs. Two different strategies for the deployment of the lower-level controllers, including centralized and decentralized algorithms, are proposed. The power of the control algorithms is finally illustrated on an extensive numerical simulation of two quadruped agents (Vision 60 robots) with Kinova arms to steer an object for which the complex hybrid model has 64 continuous-time domains, 132 state variables, and 36 control inputs.

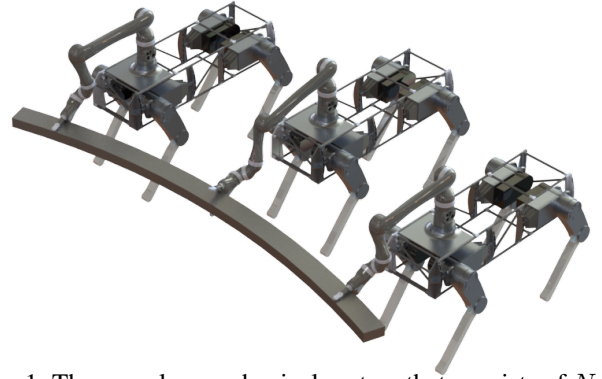


Fig. 1: The complex mechanical system that consists of  $N \geq 2$  agents (Vision 60s) with arms to steer an object.

## I. INTRODUCTION

The *overarching goal* of this paper is to present an analytical foundation, based on hybrid systems theory, hierarchical centralized and decentralized control algorithms and convex optimization for enabling collaborative legged robots with arms to manipulate objects and, specifically, steer an object (see Fig. 1). Models of legged machines are described by hybrid systems. This paper addresses complex and high-dimensional hybrid models that describe multiagent legged robotic systems. It further develops hierarchical centralized and decentralized control algorithms, based on model predictive control (MPC) and convex quadratic programming (QP), to manipulate an object while having stable locomotion patterns. The proposed framework can ameliorate specific challenges in the design of controllers for multiagent legged robots arising from high dimensionality as well as hybrid nature of models.

### A. Related Work

Hybrid systems theory has become a powerful approach for modeling legged locomotion [19, 15, 35, 36, 27, 17, 11, 33,

The work of K. Akbari Hamed is supported by the National Science Foundation (NSF) under Grant Number 1637704. The work of A. D. Ames is supported by the NSF under Grant Numbers 1544332, 1724457, and 1724464 as well as Disney Research LA. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NSF.

<sup>1</sup>K. Akbari Hamed and V. R. Kamidi are with the Department of Mechanical Engineering, Virginia Tech, Blacksburg, VA 24061 USA kaveh.akbarihamed@vt.edu and vinay28@vt.edu

<sup>2</sup>W. Ma and A. D. Ames are with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125 USA wma@caltech.edu and ames@cds.caltech.edu

37, 16, 10, 39, 31, 23, 32, 26, 34]. State-of-the-art controller design methods that address the hybrid nature of legged locomotion models are developed based on hybrid reduction [6], controlled symmetries [35], transverse linearization [27], and hybrid zero dynamics (HZD) [40, 8]. HZD-based controllers have been validated numerically and experimentally for bipedal robots [14, 28, 23, 37, 5, 8], powered prosthetic legs [18, 41], exoskeletons [3, 21], monopedal [33], and quadrupedal robots [20, 12]. State-of-the-art nonlinear control approaches for legged robots are tailored to path planning and stabilization of dynamic gaits for these sophisticated machines. However, they do *not* address path planning and control of complex hybrid dynamical models that describe the evolution of multiagent legged robotic systems working collaboratively.

### B. Motivation, Objectives, and Contributions

More than half the Earth's continent is unreachable to wheeled vehicles which motivates the deployment of multiagent legged robotic systems to enable the accessibility of these environments. Although enormous theoretical and technological advances have occurred for constructing hierarchical controllers for complex robot systems, existing approaches are tailored to the modeling and control of multiagent systems composed of collaborative robotic arms [29], multifingered robot hands [30], aerial vehicles [38], and ground vehicles [13] but *not* collaborative legged agents. The *objective* of this paper is to establish an analytical foundation to 1) develop complex hybrid models of collaborative multiagent legged robotics systems, and 2) create a hierarchical control scheme for the network of these agents to steer an object. This objective will be achieved through the following *contributions*: 1) Continuous-time dynamics for complex robotic systems composed of  $N \geq 2$  multiagent legged robots are addressed

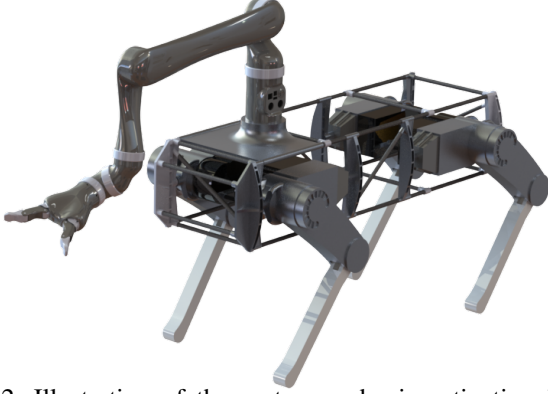


Fig. 2: Illustration of the system under investigation in this work. A commercially available six-DOF arm built by Kinova is affixed on a quadrupedal platform, Vision 60, manufactured by Ghost Robotics [1].

(see Fig. 1); 2) Discrete-time dynamics of these complex systems are studied; 3) The higher-level control is developed based on MPC and convex QPs to compute the required forces to be generated by the end effectors (EEs) for steering the object; 4) The lower-level controller is then developed based on QPs to refine the HZD baseline controllers of agents while generating the prescribed forces at the EEs; 5) Two different strategies, including centralized and decentralized algorithms, for the deployment of lower-level controllers are proposed; and 6) To demonstrate the power of the analytical foundation and control algorithms, an extensive numerical simulation of two quadrupedal agents (Vision 60 robots) with Kinova arms steering an object is finally presented (see Fig. 2). In this simulation, the complex hybrid model has 64 continuous-time domains, 132 state variables, and 36 control inputs.

## II. HYBRID MODEL OF LOCOMOTION FOR ONE AGENT

### A. Vision 60

Vision 60 is a mid-sized tele-op and autonomous all-terrain ground drone manufactured by Ghost Robotics [1] (see Fig. 2). It weighs approximately 20 kg and supports a total payload of 14 kg. Vision 60 has 18 DOFs of which 12 leg DOFs are actuated. In particular, each leg of the robot consists of a 1 DOF actuated knee joint with pitch motion and a 2 DOF actuated hip joint with pitch and roll motions. The remaining 6 DOFs are associated with the translational and rotational movements of the torso.

### B. Robot Model

To describe the configuration variables of Vision 60, we make use of the floating base model. For this purpose, we rigidly attach a body coordinate frame to the torso of the robot. The Cartesian coordinates of the origin of this frame with respect to the ground can be given by  $p_b \in \mathbb{R}^3$ . The orientation of this frame with respect to the ground (i.e., inertial world frame) is also described by  $\phi_b \in \text{SO}(3)$ . Next, let us suppose that  $q_{\text{body}} \in \mathcal{Q}_{\text{body}} \subset \mathbb{R}^{12}$  denotes the body angles that describe the shape of Vision 60. The generalized coordinates of the robot are then expressed as  $q := \text{col}(p_b, \phi_b, q_{\text{body}}) \in \mathcal{Q} := \mathbb{R}^3 \times \text{SO}(3) \times \mathcal{Q}_{\text{body}}$ . The state vector of Vision 60 is also

taken as  $x := \text{col}(q, \dot{q}) \in \text{T}\mathcal{Q} \subset \mathbb{R}^{2n_q}$ , where  $\text{T}\mathcal{Q}$  denotes the tangent bundle of  $\mathcal{Q}$  and  $n_q := \dim(q) = 18$ .

### C. Hybrid Systems Formulation

The open-loop model of quadruped locomotion can be expressed as a hybrid control system given by the following tuple [7]

$$\mathcal{H}\mathcal{L}^{\text{ol}} = (\Lambda, \mathcal{X}, \mathcal{U}, \mathcal{D}, \mathcal{S}, \Delta, FG), \quad (1)$$

where  $\Lambda := (\mathcal{V}, \mathcal{E})$  is a *directed cycle* with the vertices set  $\mathcal{V}$  and the edges set  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ . The vertices represent the continuous-time dynamics of locomotion, referred to as *domains* or *phases*, whereas edges denote the discrete-time transitions among continuous-time dynamics. The evolution of the system during continuous-time domains is described by ordinary differential equations (ODEs) arising from the Euler-Lagrange equations. The evolution of the system during discrete-time transitions is expressed by possible abrupt changes in the state vector arising from the changes in physical constraints (e.g., new contact is added to the existing set of contact points with the ground or an existing contact breaks). In our notation,  $\mathcal{X}$  and  $\mathcal{U}$  denote the *set of state manifolds* and *set of admissible controls*, respectively, and are shown by  $\mathcal{X} := \{\mathcal{X}_v\}_{v \in \mathcal{V}}$  and  $\mathcal{U} := \{\mathcal{U}_v\}_{v \in \mathcal{V}}$ . Here,  $\mathcal{X}_v \subseteq \mathbb{R}^{2n_q}$  and  $\mathcal{U}_v \subseteq \mathbb{R}^m$  represent the *state manifold* and *set of admissible control inputs* for the domain  $v \in \mathcal{V}$ , where  $m = 12$ .  $\mathcal{D} := \{\mathcal{D}_v\}_{v \in \mathcal{V}}$  denotes the *set of domains of admissibility*, in which  $\mathcal{D}_v \subseteq \mathcal{X}_v \times \mathcal{U}_v$  is a smooth submanifold of  $\mathbb{R}^{2n_q} \times \mathbb{R}^m$ . The continuous-time dynamics are given by  $FG := \{(f_v, g_v)\}_{v \in \mathcal{V}}$ , where  $(f_v, g_v)$  is a control system on  $\mathcal{D}_v$ , that is, in coordinates  $\dot{x} = f_v(x) + g_v(x)u$ . The *set of guards* for the hybrid system is shown by  $\mathcal{S} := \{\mathcal{S}_e\}_{e \in \mathcal{E}}$  on which the discrete transitions  $e \in \mathcal{E}$  occur when the state and control trajectories  $(x(t), u(t))$  cross  $\mathcal{S}_e$ . Finally,  $\Delta := \{\Delta_e\}_{e \in \mathcal{E}}$  represents the set of discrete-time transitions, in which  $\Delta_e$  denotes the discrete-time dynamics for the edge  $e \in \mathcal{E}$ . In particular,  $x^+ = \Delta_e(x^-)$ , where  $x^-(t) := \lim_{\tau \nearrow t} x(\tau)$  and  $x^+(t) := \lim_{\tau \searrow t} x(\tau)$  represent the state of the system right before and after the discrete transition, respectively.

### D. Continuous-Time Dynamics

During the continuous-time domain  $v \in \mathcal{V}$ , we consider a set of physical holonomic constraints to represent the contact of the stance legs with the ground as  $\eta_v(q) = 0$ . The associated velocity constraints are given by  $J_v(q)\dot{q} = 0$ , in which  $J_v(q) := \frac{\partial \eta_v}{\partial q}(q)$ . The equations of motion for the domain  $v$  can then be described by the Euler-Lagrange equations and principle of virtual work as follows

$$\begin{aligned} D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) &= Bu + J_v^\top(q)\lambda \\ J_v(q)\ddot{q} + \frac{\partial}{\partial q}(J_v(q)\dot{q})\dot{q} &= 0. \end{aligned} \quad (2)$$

Here,  $D(q) \in \mathbb{R}^{n_q \times n_q}$  represents the positive definite mass-inertia matrix,  $H(q, \dot{q}) := C(q, \dot{q})\dot{q} + G(q) \in \mathbb{R}^{n_q}$  denotes the Coriolis, centrifugal and gravitational terms,  $B \in \mathbb{R}^{n_q \times m}$  is the input distribution matrix with the property  $\text{rank}(B) = m$ , and  $\lambda$  represents the Lagrange multipliers (i.e., ground reaction

forces). If  $J_v$  is full-rank, one can eliminate the Lagrange multipliers from (2) to express the state equation as  $\dot{x} = f_v(x) + g_v(x)u$ .

### E. Discrete-Time Dynamics

In this paper, we assume that the  $\mu : \mathcal{V} \rightarrow \mathcal{V}$  denotes the *index of the next domain function* for the studied locomotion pattern. Using this function, the set of edges can be expressed as  $\mathcal{E} = \{e = (v \rightarrow \mu(v))\}_{v \in \mathcal{V}}$ . If one of the existing contacts breaks during the transition, the discrete-time dynamics are taken as the identity map to preserve the continuity of position and velocity, i.e.,  $x^+ = \Delta_e(x^-) := x^-$ . However, if a new contact point is added to the existing set of contact points, the velocity components of the state vector would undergo an abrupt change according to the instantaneous and rigid impact model [24]. In particular, if  $\delta\lambda$  represents the intensity of the impulsive Lagrange multipliers, the evolution of the system during the infinitesimal period of the impact can be given by

$$D(q)\dot{q}^+ - D(q)\dot{q}^- = J_{\mu(v)}^\top \delta\lambda, \quad J_{\mu(v)}(q)\dot{q}^+ = 0, \quad (3)$$

where  $\dot{q}^-$  and  $\dot{q}^+$  represent the generalized velocity vector right before and after the impact, respectively. By assuming the continuity of position (i.e.,  $q^+ = q^-$ ) and eliminating  $\delta\lambda$  from (3), one can obtain a closed-form expression for the discrete-time dynamics as  $x^+ = \Delta_e(x^-)$ .

### F. Closed-Loop System and Periodic Orbit

We consider a family of nonlinear and HZD-based state feedback laws for the hybrid model of locomotion in (1) as  $\Gamma := \{\Gamma_v\}_{v \in \mathcal{V}}$ . In our notation,  $u = \Gamma_v(x)$  represents the state feedback law during the continuous-time domain  $v \in \mathcal{V}$ , where  $\Gamma_v : \mathcal{X}_v \rightarrow \mathcal{U}_v$  is a smooth mapping (i.e.,  $\mathcal{C}^\infty$ ). In the HZD method, we consider a set of output functions  $y_v(x)$ , referred to as *virtual constraints*, to be regulated during each domain. Then,  $\Gamma_v(x)$  is designed using the standard input-output (I-O) linearization technique [25] to asymptotically derive the outputs to zero (i.e.,  $\lim_{t \rightarrow \infty} y_v(t) = 0$ ). Virtual constraints represent holonomic or nonholonomic kinematic constraints to coordinate the links of the robot within a stride and are satisfied through the action of the state feedback  $\Gamma_v(x)$ . We assume that by employing the state feedback laws, there is a *periodic orbit* for the hybrid model of locomotion that describes steady-state locomotion. The periodic orbit can be given by  $\mathcal{O} := \{\mathcal{O}_v\}_{v \in \mathcal{V}}$ , in which  $\mathcal{O}_v$  denotes the projection of the orbit onto the state manifold of the domain  $v$ . To design the orbit  $\mathcal{O}$ , we make use of FROST (Fast Robot Optimization and Simulation Toolkit) – an open source toolkit for planning of dynamic legged locomotion [22, 20, 23]. FROST provides an efficient trajectory optimization framework for nonlinear and hybrid models of legged machines. It uses the Hermite Simpson collocation method to translate a trajectory planning problem into a constrained nonlinear programming. We further suppose that the periodic orbit  $\mathcal{O}$  is locally orbitally exponentially stable for the closed-loop system, i.e.,

$$\text{dist}(x(t), \mathcal{O}) \leq \kappa \text{dist}(x(0), \mathcal{O}) \exp(-\gamma t), \quad \forall t \geq 0, \quad (4)$$

for some  $\kappa \geq 1$  and  $\gamma > 0$  and every initial state  $x(0)$  in an open neighborhood of  $\mathcal{O}$ . Here,  $\text{dist}(x, \mathcal{O}) := \inf_{z \in \mathcal{O}} \|x - z\|$

denotes the distance of the point  $x$  to the set  $\mathcal{O}$ . A systematic approach for synthesizing exponentially stabilizing virtual constraint controllers  $\Gamma_v(x)$ , based on an iterative sequence of optimization problems involving linear and bilinear matrix inequalities (LMIs and BMIs), has been developed in [20, 5, 4].

## III. COMPLEX HYBRID MODEL FOR MULTIAGENT LEGGED ROBOTIC SYSTEMS WITH ARMS AND AN OBJECT

In this section, we address the complex hybrid model that describes the equations of motion for  $N$  agents manipulating an object (see Fig. 1). The agents are augmented by robotic arms for the steering purpose. To simplify the analysis, we assume that the models of agents/arms are identical. In this paper, we consider a 6 DOF Kionva manipulator for the arms. To augment the agent model with that of the arm, we consider a floating base model for the arm. This model has 12 DOFs that include the internal 6 DOFs of the arm together with additional 6 DOFs that describe the translational and rotational motions of its base. The configuration variables of the floating base model can then be given by  $q_a \in \mathcal{Q}_a \subset \mathbb{R}^{n_a}$ , where  $n_a = 12$ . The equations of motion for the floating base model is also described by

$$D_a(q_a)\ddot{q}_a + C_a(q_a, \dot{q}_a)\dot{q}_a + G_a(q_a) = B_a u_a, \quad (5)$$

where  $D_a(q_a) \in \mathbb{R}^{n_a \times n_a}$  is the positive definite mass-inertia matrix,  $H_a(q_a, \dot{q}_a) := C_a(q_a, \dot{q}_a)\dot{q}_a + G_a(q_a) \in \mathbb{R}^{n_a}$  denotes the Coriolis, centrifugal, and gravitational terms,  $u_a \in \mathbb{R}^{m_a}$  represents the control inputs, and  $B_a \in \mathbb{R}^{n_a \times m_a}$  is the input distribution matrix with the property  $\text{rank}(B_a) = m_a = 6$ . The object is also assumed to be rigid with  $n_o = 6$  DOFs. Its free motion can be described as follows

$$D_o(q_o)\ddot{q}_o + C_o(q_o, \dot{q}_o)\dot{q}_o + G_o(q_o) = 0, \quad (6)$$

in which  $q_o \in \mathcal{Q}_o \subset \mathbb{R}^{n_o}$  represents the translational and rotational variables of the object with respect to the ground frame,  $D_o(q_o) \in \mathbb{R}^{n_o \times n_o}$  is the mass-inertia matrix and  $H_o(q_o, \dot{q}_o) := C_o(q_o, \dot{q}_o)\dot{q}_o + G_o(q_o) \in \mathbb{R}^{n_o}$  denotes the Coriolis, centrifugal and gravitational terms. In what follows, we will address the *complex hybrid model* that describes the evolution of the augmented mechanical system consisting of  $N$  agents with the arms while steering the object through their end effectors (EEs).

### A. Augmented Continuous-Time Dynamics

In order to rigidly connect the base of the arm to a specific point on Vision 60's torso, we consider the following physical holonomic constraint (see Fig. 3)

$$p_t(q) - p_a(q_a) = 0, \quad (7)$$

where  $p_t(q) \in \mathbb{R}^6$  denotes the Cartesian coordinates and orientation (e.g., roll-pitch-yaw angles) of the aforementioned point on the robot torso and  $p_a(q_a) \in \mathbb{R}^6$  represents the equivalent quantities for the base of the arm. For later purposes, we assume that  $u_a = \Gamma_a(x_a)$  is a state feedback controller that regulates the arm joints to a desired configuration (e.g., as shown in Fig. 2), where  $x_a := \text{col}(q_a, \dot{q}_a)$ . This feedback law can be a simple PD controller with gravitational compensation.

Next, we study an alternative holonomic constraint for the EE and object connection. In this paper, it is assumed that the EE and object connection is through a ball joint to simplify the analysis. Consequently, we only consider holonomic constraints on the translational motions of the EE and object as follows

$$p_A(q_o) - p_e(q_a) = 0, \quad (8)$$

where  $p_A(q_o) \in \mathbb{R}^3$  represents the Cartesian coordinates of the point  $A$  on the object and  $p_e(q_a) \in \mathbb{R}^3$  denotes the Cartesian coordinates of the EE.

Now let us consider a team of collaborative  $N$  agents, where  $i \in \{1, 2, \dots, N\}$  represents the agent number. The state variables for the agent  $i$  can be given by  $\bar{x}_i := \text{col}(\bar{q}_i, \dot{\bar{q}}_i)$ , in which  $\bar{q}_i := \text{col}(q_i, q_{a,i})$ , and  $q_i$  and  $q_{a,i}$  represent the configuration variables of the  $i$ th Vision 60 and corresponding arm, respectively. The control input for the agent  $i$  with the arm is also denoted by  $\bar{u}_i := \text{col}(u_i, u_{a,i})$ . The continuous-time domain index for the agent  $i$  is also shown by  $v_i \in \mathcal{V}$ .

*Assumption 1 (Baseline Controllers):* The *baseline controller* for the agent  $i$  is defined as  $(\Gamma_{v_i}(\bar{x}_i), \Gamma_a(\bar{x}_i))$ , where  $\Gamma_{v_i}(\bar{x}_i)$  and  $\Gamma_a(\bar{x}_i)$  denote the virtual constraint controller and PD feedback law for the agent and arm, respectively. We suppose that each agent with the arm can have an exponentially stable locomotion pattern when employing the baseline controller. The new orbit can be shown by  $\bar{\mathcal{O}}$  that is close to  $\mathcal{O}$  (i.e., the orbit without the arm) (see Fig. 4). This is not a restrictive assumption as Vision 60 is much heavier than the arm.

To develop the continuous-time dynamics of the complex model, we assume that the EE of the agent  $i$  is rigidly connected to the point  $A_i$  on the object. Using the principle of virtual work, the equations of motion can then be described as follows

$$\begin{aligned} D \ddot{q}_i + H &= B u_i + J_t^\top \lambda_{t,i} + J_{v_i}^\top \lambda_i \\ D_a \ddot{q}_{a,i} + H_a &= B_a u_{a,i} - J_a^\top \lambda_{t,i} - J_e^\top \lambda_{e,i} \\ J_t \ddot{q}_i + \frac{\partial}{\partial q_i} (J_t \dot{q}_i) \dot{q}_i &= J_a \ddot{q}_{a,i} + \frac{\partial}{\partial q_{a,i}} (J_{a,i} \dot{q}_{a,i}) \dot{q}_{a,i} \\ J_{A,i} \ddot{q}_o + \frac{\partial}{\partial q_o} (J_{A,i} \dot{q}_o) \dot{q}_o &= J_e \ddot{q}_{a,i} + \frac{\partial}{\partial q_{a,i}} (J_e \dot{q}_{a,i}) \dot{q}_{a,i} \\ J_{v_i} \ddot{q}_i + \frac{\partial}{\partial q_i} (J_{v_i} \dot{q}_i) \dot{q}_i &= 0, \quad 1 \leq i \leq N \\ D_o \ddot{q}_o + H_o &= \sum_{j=1}^N J_{A,j}^\top \lambda_{e,j}, \end{aligned} \quad (9)$$

where  $\lambda_i, \lambda_{t,i} \in \mathbb{R}^6$ , and  $\lambda_{e,i} \in \mathbb{R}^3$  for  $1 \leq i \leq N$  are the ground reaction forces, contact wrench between the robot and arm, and forces between the EE and object, respectively (see Fig. 3). Moreover,  $J_t(q) := \frac{\partial p_t}{\partial q}(q) \in \mathbb{R}^{6 \times n_q}$ ,  $J_a(q_a) := \frac{\partial p_a}{\partial q_a}(q_a) \in \mathbb{R}^{6 \times n_a}$ ,  $J_e(q_a) := \frac{\partial p_e}{\partial q_a}(q_a) \in \mathbb{R}^{3 \times n_a}$ , and  $J_{A,i}(q_o) := \frac{\partial p_{A,i}}{\partial q_o}(q_o) \in \mathbb{R}^{3 \times n_o}$ . By defining

$$\begin{aligned} \bar{D}_i(\bar{q}_i) &:= \text{block diag} \{D(q_i), D_a(q_{a,i})\} \\ \bar{J}_i^\top(\bar{q}_i) &:= \begin{bmatrix} -J_{v_i}^\top(q_i) & -J_t^\top(q_i) & 0 \\ 0 & J_a^\top(q_{a,i}) & J_e^\top(q_{a,i}) \end{bmatrix} \\ \bar{J}_{A,i}^\top(q_o) &:= [0 \quad 0 \quad -J_{A,i}^\top(q_o)] \end{aligned} \quad (10)$$

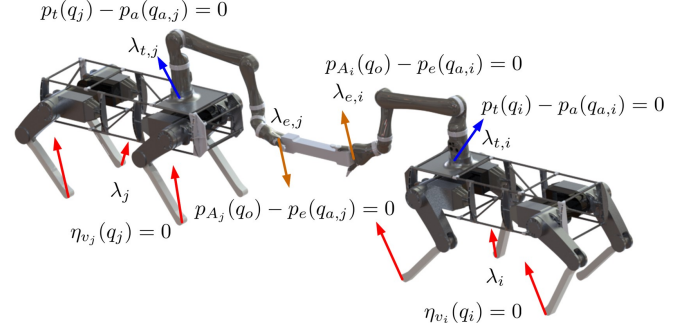


Fig. 3: Illustration of the ground reaction forces  $\lambda_i$ , wrench between the arm and Vision 60  $\lambda_{t,i}$ , and EE forces  $\lambda_{e,i}$  for two agents with the corresponding physical holonomic constraints.

for  $1 \leq i \leq N$ , the equations of motion in (9) can be rewritten in the following compact form

$$\begin{bmatrix} D_{\text{aug}}(q_{\text{aug}}) & J_{\text{aug}}^\top(q_{\text{aug}}) \\ J_{\text{aug}}(q_{\text{aug}}) & 0 \end{bmatrix} \begin{bmatrix} \ddot{q}_{\text{aug}} \\ \lambda_{\text{aug}} \end{bmatrix} = H_{\text{aug}}(q_{\text{aug}}, \dot{q}_{\text{aug}}, u_{\text{aug}}), \quad (11)$$

in which  $q_{\text{aug}} := \text{col}(\bar{q}_1, \dots, \bar{q}_N, q_o)$ ,  $u_{\text{aug}} := \text{col}(\bar{u}_1, \dots, \bar{u}_N)$ ,  $\lambda_{\text{aug}} := \text{col}(\lambda_1, \dots, \lambda_N)$ ,  $\lambda_i := \text{col}(\lambda_i, \lambda_{t,i}, \lambda_{e,i})$  for  $1 \leq i \leq N$ ,

$$\begin{aligned} D_{\text{aug}} &:= \text{block diag} \{ \bar{D}_1, \dots, \bar{D}_N, D_o \} \in \mathbb{R}^{n_{\text{aug}} \times n_{\text{aug}}} \\ J_{\text{aug}}^\top &:= \begin{bmatrix} \bar{J}_1^\top & 0 & \dots & 0 \\ 0 & \bar{J}_2^\top & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \bar{J}_N^\top \\ \bar{J}_{A,1}^\top & \bar{J}_{A,2}^\top & \dots & \bar{J}_{A,N}^\top \end{bmatrix}, \end{aligned} \quad (12)$$

$n_{\text{aug}} := N(n_q + n_a) + n_o$ , and  $H_{\text{aug}}$  is an appropriate vector to include the remaining terms of (9) that are not included in the left hand side of (11). For future purposes, it is worth mentioning that  $H_{\text{aug}}$  is an affine function in terms of  $u_{\text{aug}}$ . This property would help us to design the lower-level controllers for steering the object in Section IV-B through convex QPs.

*Remark 1:* Note that if the agent  $i$  is in the continuous-time domain  $v_i \in \mathcal{V}$ , the agent  $j \neq i \in \{1, 2, \dots, N\}$  can take any domain  $v_j \in \mathcal{V}$ . As a consequence, there would be  $M^N$  different continuous-time domains for the complex hybrid model of  $N$  agents, where  $M := |\mathcal{V}|$  denotes the cardinality of the vertices set  $\mathcal{V}$ .

## B. Augmented Discrete-Time Dynamics

In this section, we address the impact dynamics for the complex hybrid system. If one of the agents has an impact with the ground, the velocity components of the complex mechanical system would undergo a possible abrupt change. To make this concept more precise, we first define the extended Jacobin matrix for the agent  $i$  as follows

$$\hat{J}_i(q_i) := \begin{cases} J_{v_i}(q_i) & \text{if the agent is in domain } v_i \\ J_{\mu(v_i)}(q_i) & \text{if the agent is in transition } v_i \rightarrow \mu(v_i). \end{cases}$$

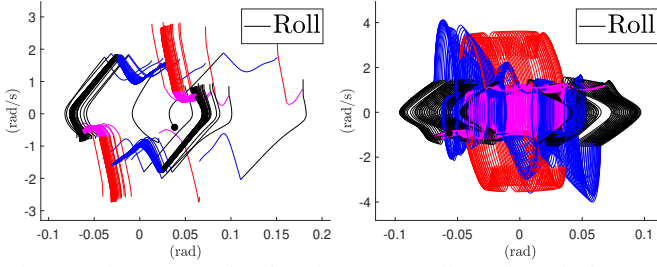


Fig. 4: Phase portraits for the torso roll motion during 150 consecutive steps of an amble gait of Vision 60 without (left) and with (right) the Kinova arm. The figures illustrate the convergence to exponentially stable periodic orbits.

Then, the evolution of the augmented system during the infinitesimal period of the impact can be given by

$$\begin{aligned}
 D(q_i) \dot{q}_i^+ - D(q_i) \dot{q}_i^- &= J_t^\top(q_i) \delta \lambda_{t,i} + \hat{J}_i^\top(q_i) \delta \lambda_i \\
 D_a(q_a) \dot{q}_a^+ - D_a(q_a) \dot{q}_a^- &= -J_a^\top(q_a) \delta \lambda_{t,i} - J_e^\top(q_a) \delta \lambda_{e,i} \\
 J_t(q_i) \dot{q}_i^+ - J_a(q_{a,i}) \dot{q}_{a,i}^+ &= 0 \\
 J_{A,i}(q_o) \dot{q}_o^+ - J_e(q_{e,i}) \dot{q}_{e,i}^+ &= 0 \\
 \hat{J}_i(q_i) \dot{q}_i^+ &= 0, \quad 1 \leq i \leq N,
 \end{aligned} \quad (13)$$

where  $\delta \lambda_i, \delta \lambda_{t,i} \in \mathbb{R}^6$ , and  $\delta \lambda_{e,i} \in \mathbb{R}^3$  for  $1 \leq i \leq N$  denote the impulsive ground reaction forces, contact wrench between the arm and robot, and forces between the EE and object. Analogous to the analysis provided in Section III-A, one can present a compact equation for the impact dynamics.

#### IV. HIERARCHICAL CONTROL SCHEME

In order to coordinate the action of the agents while steering the object, we make use of a hierarchical control strategy. In the proposed approach, the higher-level controller considers the dynamics of the object and then utilizing MPC, it computes the required forces to be generated by the EEs. The low-level controllers that can be either centralized or decentralized are then developed based on QPs to refine the baseline controllers while generating the prescribed forces at the EEs.

##### A. Higher-Level MPC Control

In this section, we develop the higher-level MPC control for steering the object. We consider the translational motion of the object to simplify the analysis. In particular, if  $r_o \in \mathbb{R}^3$  represents the Cartesian coordinates of a specific point on the object (e.g., center of mass), the equation of translational motion can be given by

$$m_o \ddot{r}_o + m_o \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \sum_{i=1}^N \lambda_{e,i} =: \lambda_{\text{net}}, \quad (14)$$

where  $m_o$  and  $g$  denote the known mass of the object and gravitational constant, respectively. In addition,  $\lambda_{\text{net}} = \sum_{i=1}^N \lambda_{e,i} \in \mathbb{R}^3$  is the net force applied by the EEs. By taking the state vector as  $x_o := \text{col}(r_o, \dot{r}_o) \in \mathbb{R}^6$  and control input as  $\lambda_{\text{net}}$  while utilizing the zero-order hold (ZOH) approach, (14) can be discretized as follows

$$x_o[k+1] = \Phi x_o[k] + \Omega \lambda_{\text{net}}[k] + \Psi, \quad k = 0, 1, \dots, \quad (15)$$

where  $\Phi \in \mathbb{R}^{6 \times 6}$ ,  $\Omega \in \mathbb{R}^{6 \times 3}$  and  $\Psi \in \mathbb{R}^6$  are appropriate matrices/vectors, and  $k$  denotes the discrete time. We remark that  $\Psi$  represents the gravitational force. Next, we set up the following receding horizon control problem for every time sample  $k$

$$\begin{aligned}
 \min_U \mathcal{J}(x_o[0], U) &:= (x_o[N_h] - x_{o,\text{ref}})^\top P (x_o[N_h] - x_{o,\text{ref}}) \\
 &+ \sum_{\ell=0}^{N_h-1} (x_o[\ell] - x_{o,\text{ref}})^\top Q (x_o[\ell] - x_{o,\text{ref}}) \\
 &+ \sum_{\ell=0}^{N_h-1} \lambda_{\text{net}}^\top[\ell] R \lambda_{\text{net}}[\ell] \\
 \text{s.t.} \quad x_o[\ell+1] &= \Phi x_o[\ell] + \Omega \lambda_{\text{net}}[\ell] + \Psi \\
 x_{o,\text{min}} &\leq x_o[\ell] \leq x_{o,\text{max}} \\
 \lambda_{\text{net},\text{min}} &\leq \lambda_{\text{net}}[\ell] \leq \lambda_{\text{net},\text{max}} \\
 x_o[0] &= x_o[k],
 \end{aligned} \quad (16)$$

in which  $N_h \geq 1$  is the control horizon,  $x_{o,\text{ref}} \in \mathbb{R}^6$  represents the reference state for the object,  $P \succeq 0$ ,  $Q \succeq 0$ ,  $R \succ 0$ , and  $U := \text{col}(\lambda_{\text{net}}[0], \dots, \lambda_{\text{net}}[N_h-1]) \in \mathbb{R}^{3N_h}$ . Furthermore,  $x_{o,\text{min}}$ ,  $x_{o,\text{max}}$ ,  $\lambda_{\text{net},\text{min}}$ , and  $\lambda_{\text{net},\text{max}}$  are some bounds on the state variables and control inputs. The higher-level MPC control is then taken as the global solution of the QP (16) for each time sample  $k$ , i.e.,  $\lambda_{\text{net}}(x[k]) := \lambda_{\text{net}}[0]$ . It can be shown that the problem (16) is indeed a QP in the presence of  $\Psi$ , e.g., one can extend the Batch Approach of [9, Chap. 8] to (16) to show this fact.

##### B. Lower-Level Centralized/Decentralized QP Controllers

This section develops the lower-level controllers to produce the prescribed forces computed by the higher-level MPC at EEs while being close to the baseline controllers for the desired locomotion pattern. We develop these controllers through centralized as well as decentralized algorithms.

1) *Centralized Algorithm:* In the centralized scheme, we assume that there is one lower-level controller unit that has access to all of the agents' state variables  $\bar{x}_i, 1 \leq i \leq N$  to make the decision for them. To make the idea more precise, let us define the desired (i.e., baseline) torques for the agent  $i$  as  $\bar{u}_i^{\text{des}}(\bar{x}_i)$  (see Assumption 1). Next from (11), one can solve for  $\lambda_{\text{aug}}$  in terms of  $(q_{\text{aug}}, \dot{q}_{\text{aug}}, u_{\text{aug}})$ . We remark that  $\lambda_{\text{aug}}$  is affine in terms of  $u_{\text{aug}}$ . In particular, the EE forces can be computed as follows

$$\begin{bmatrix} \lambda_{e,1} \\ \vdots \\ \lambda_{e,N} \end{bmatrix} = \underbrace{\begin{bmatrix} \Pi_{11}(x_{\text{aug}}) & \cdots & \Pi_{1N}(x_{\text{aug}}) \\ \vdots & \ddots & \vdots \\ \Pi_{N1}(x_{\text{aug}}) & \cdots & \Pi_{NN}(x_{\text{aug}}) \end{bmatrix}}_{\Pi(x_{\text{aug}})} \underbrace{\begin{bmatrix} \bar{u}_1 \\ \vdots \\ \bar{u}_N \end{bmatrix}}_{\bar{u}(x_{\text{aug}})} + \underbrace{\begin{bmatrix} \Upsilon_1(x_{\text{aug}}) \\ \vdots \\ \Upsilon_N(x_{\text{aug}}) \end{bmatrix}}_{\Upsilon(x_{\text{aug}})}.$$

The net force applied to the object is then given by

$$\begin{aligned}
 \lambda_{\text{net}} &= \sum_{i=1}^N \lambda_{e,i} = \sum_{i=1}^N \sum_{j=1}^N \Pi_{ij}(x_{\text{aug}}) \bar{u}_i + \sum_{i=1}^N \Upsilon_i(x_{\text{aug}}) \\
 &=: \bar{\Pi}(x_{\text{aug}}) u_{\text{aug}} + \tilde{\Upsilon}(x_{\text{aug}}).
 \end{aligned} \quad (17)$$

Next, we set up the following centralized and lower-level QP algorithm to solve for  $u_{\text{aug}}$  at every time sample

$$\begin{aligned} \min_{u_{\text{aug}}} & \left\| \hat{\Pi}(x_{\text{aug}}) u_{\text{aug}} + \hat{\Upsilon}(x_{\text{aug}}) - \lambda_{\text{net}}^{\text{MPC}} \right\|^2 + w \|u_{\text{aug}} - u_{\text{aug}}^{\text{des}}\|^2, \\ \text{s.t.} & u_{\text{aug},\min} \leq u_{\text{aug}} \leq u_{\text{aug},\max}, \end{aligned} \quad (18)$$

where  $\|\cdot\|$  is the Euclidean norm,  $u_{\text{aug}}^{\text{des}} := \text{col}(\bar{u}_1^{\text{des}}, \dots, \bar{u}_N^{\text{des}})$ , and  $\lambda_{\text{net}}^{\text{MPC}}$  represents the net force generated by the higher-level MPC controller. In addition,  $w > 0$  denotes the weighting factor as a trade-off between 1) making  $\lambda_{\text{net}}$  being close to  $\lambda_{\text{net}}^{\text{MPC}}$  and 2) making  $u_{\text{aug}}$  being close to the baseline controllers of the agents  $u_{\text{aug}}^{\text{des}}$ . Finally,  $u_{\text{aug},\min}$  and  $u_{\text{aug},\max}$  represent lower and upper limits to include the torque limitations.

2) *Decentralized Algorithm*: In the decentralized scheme, we set up local QPs for each individual agent. The local QP for the agent  $i$  has access to its own state variables  $\bar{x}_i$  as well as those of the object to make local decisions. In addition, for each agent, we would need to approximate the state variables of the other agents to be able to compute the net force through (17). For this purpose, we make the following assumptions.

*Assumption 2 (Phasing Variable)*: For every agent  $i \in \{1, \dots, N\}$ , there is a smooth, scalar, and strictly increasing function of time during each domain  $v_i \in \mathcal{V}$ , referred to as the *phasing variable*, that represents the progress of the machine on the gait  $\bar{O}$ . The phasing variable can be time- or state-based. For the purpose of this paper, we shall make use of a state-based phasing variable and denote it by  $\tau_i(\bar{x}_i, v_i)$ .

Using Assumption 2, one can represent the desired evolution of the state variables  $\bar{x}_i$  on the orbit  $\bar{O}$  in terms of the phasing variable  $\tau_i$  rather than the time  $t$  [5, Assumption 3]. In particular, the desired evolution of  $\bar{x}_i$  on the orbit  $\bar{O}_{v_i}$  can be give by  $\bar{x}_{i,d}(\tau_i, v_i)$ , where the subscript “d” stands for the desired evolution. One can obtain  $\bar{x}_{i,d}(\cdot)$  through polynomial regression techniques.

*Assumption 3 (Coordination Mechanism)*: We assume that agent  $i$  does not have access to the state measurements of other agents  $j \neq i \in \{1, \dots, N\}$ . However, it has access to their phasing variables  $\tau_j$  as well as their vertices index  $v_j \in \mathcal{V}$ .

From Assumptions 2 and 3, the agent  $i$  can approximate the state variables of the other agents  $j \neq i$  through the knowledge of their desired state trajectories in terms of the phasing variable  $\tau_j$  as well as the domain index  $v_j$ . In particular, the estimate of  $\bar{x}_j$  for the agent  $i$ , denoted by  $\hat{x}_{j|i}$ , is given by

$$\hat{x}_{j|i} = \bar{x}_{j,d}(\tau_j, v_j). \quad (19)$$

We further suppose that agent  $i$  approximates the control inputs for the other agents as its own, that is  $\bar{u}_i$ . Using these assumptions, we can now approximate the net force in (17) based on the agent  $i$ 's information, i.e.,

$$\hat{\lambda}_{\text{net}} = \hat{\Pi}_i(\bar{x}_i, \hat{x}_{j|i}) \bar{u}_i + \hat{\Upsilon}_i(\bar{x}_i, \hat{x}_{j|i}), \quad j \neq i. \quad (20)$$

We then set up the following decentralized QP for the agent  $i$

$$\begin{aligned} \min_{\bar{u}_i} & \left\| \hat{\Pi}_i \bar{u}_i + \hat{\Upsilon}_i - \lambda_{\text{net}}^{\text{MPC}} \right\|^2 + w \|\bar{u}_i - \bar{u}_i^{\text{des}}\|^2, \\ \text{s.t.} & \bar{u}_{\min} \leq \bar{u}_i \leq \bar{u}_{\max}, \end{aligned} \quad (21)$$

in which  $\bar{u}_{\min}$  and  $\bar{u}_{\max}$  represent the torque limitations for the agent. We remark that in this scheme, each agent has its own decentralized QP algorithm for local decision making.

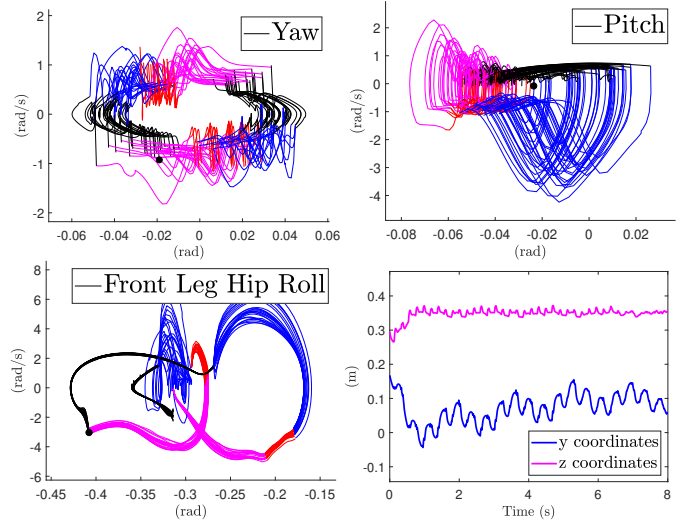


Fig. 5: Phase portraits for the closed-loop complex system composed of two agents with arms that steer an object using the proposed centralized hierarchical control. The figure depicts the time evolution of the object's CoM Cartesian coordinates with respect to the world frame.

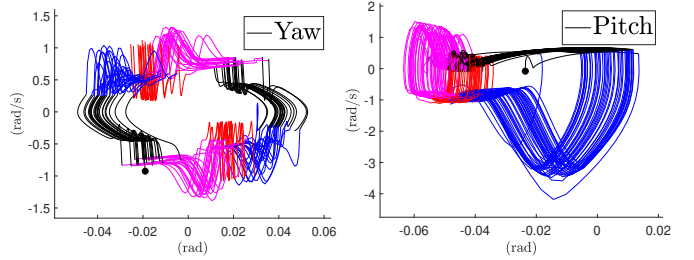


Fig. 6: Phase portraits for the closed-loop complex system using the proposed decentralized hierarchical control.

## V. NUMERICAL SIMULATIONS

The objective of this section is to numerically validate the analytical foundation of the paper on a simulation model of  $N = 2$  Vision 60 robots with Kinova arms for steering an object. We consider an amble gait for Vision 60 robots that is designed through FROST [22] in [20]. The hybrid model of locomotion for each agent has  $M = |\mathcal{V}| = 8$  continuous-time domains. The complex hybrid model of multi-agent systems has consequently  $M^N = 64$  continuous-time domains with  $N(2n_q + 2n_a) + 2n_o = 132$  state variables and  $N(m + m_a) = 36$  control inputs. The baseline controller for Vision 60 is designed through virtual constraints that are optimized for stability through LMIs and BMIs [20, 5, 4]. The stable limit cycles for the roll motion of the robot without and with having the arm can be seen in Fig. 4. The sampling time for the hierarchical control action is taken as 1 ms, where the control horizon for the higher-level MPC is assumed to be  $N_h = 3$  with  $P = 10^8 I_{6 \times 6}$ ,  $Q = 100 I_{6 \times 6}$ , and  $R = I_{3 \times 3}$ . Furthermore, the weighting factor for the lower-level centralized and decentralized QPs is chosen as  $w = 0.01$ . Figure 5 depicts the phase portraits for one of the agents of the complex hybrid system while steering an object using the centralized hierarchical control strategy. It also illustrates the time evolution of the object's center of mass (CoM) Cartesian coordinates. Here, the agents walk along the  $x$ -axis of the

world frame and the reference command for the  $y$  and  $z$  coordinates of the object are 0.1 (m) and 0.35 (m), respectively. We remark that the oscillation in the object coordinates is due to agents' legged locomotion. However as seen in Fig. 5, the oscillation pattern (e.g., in the  $y$  coordinates) reaches the steady state and occurs around the desired command. The phase portraits using the proposed decentralized hierarchical strategy are also depicted in Fig. 6. Figure 7 illustrates the snapshots of locomotion of two agents generated in RViz. The animation of this simulation can be found at [2].

## VI. CONCLUSION

This paper introduced a hierarchical feedback control algorithm for complex models of multiagent legged robotic systems with arms that steer an object. We addressed the continuous- and discrete-time dynamics for high-dimensional and complex hybrid models of  $N$  legged agents. We considered HZD baseline controllers for each agent to have an exponentially stable locomotion pattern. The higher-level controller of the proposed strategy was developed based on MPC and convex QPs to prescribe the required forces to be generated by EEs for the steering purpose. The lower-level controllers were developed based on QPs to refine the baseline controllers while generating the prescribed forces. We also presented two different strategies, including centralized and decentralized QP algorithms, for the deployment of lower-level controllers. The analytical results of the paper were successfully validated through an extensive and full-order simulation model of two Vision 60 agents with Kinova arms that collaboratively steer an object. The simulated complex hybrid model has 64 continuous-time domains, 132 state variables, and 36 control inputs. Both centralized and decentralized hierarchical control strategies resulted in stable locomotion as well as steering. Their application on real robots is dependent on the availability of global state measurements. For future research, we will experimentally investigate the hierarchical control algorithms on two Vision 60 robots with Kinova arms. Furthermore, we will study control and optimization algorithms for autonomous multiagent legged systems that steer an object while considering safety critical conditions and avoiding obstacles in complex environments.

## REFERENCES

- [1] Ghost Robotics, <https://www.ghostrobotics.io/>.
- [2] Hierarchical Feedback Control for Multiagent Legged Robotic Systems, <https://youtu.be/VD6oUjGx3iE>, Jan 2019.
- [3] A. Agrawal, O. Harib, A. Hereid, S. Finet, M. Masselin, L. Praly, A. Ames, K. Sreenath, and J. Grizzle. First steps towards translating HZD control of bipedal robots to decentralized control of exoskeletons. *IEEE Access*, 5:9919–9934, 2017.
- [4] K. Akbari Hamed and R. D. Gregg. Decentralized feedback controllers for robust stabilization of periodic orbits of hybrid systems: Application to bipedal walking. *Control Systems Technology, IEEE Transactions on*, 25(4):1153–1167, July 2017.
- [5] K. Akbari Hamed, B.G. Buss, and J.W. Grizzle. Exponentially stabilizing continuous-time controllers for periodic orbits of hybrid systems: Application to bipedal locomotion with ground height variations. *The International Journal of Robotics Research*, 35(8):977–999, 2016.
- [6] A. D. Ames, R. D. Gregg, E. D. B. Wendel, and S. Sastry. On the geometric reduction of controlled three-dimensional bipedal robotic walkers. In *Lagrangian and Hamiltonian Methods for Nonlinear Control 2006*, pages 183–196, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.
- [7] A. D. Ames, R. Vasudevan, and R. Bajcsy. Human-data based cost of bipedal robotic walking. In *Proceedings of the 14th international conference on Hybrid systems: computation and control*, pages 153–162. ACM, 2011.
- [8] A.D. Ames, K. Galloway, K. Sreenath, and J.W. Grizzle. Rapidly exponentially stabilizing control Lyapunov functions and hybrid zero dynamics. *Automatic Control, IEEE Transactions on*, 59(4):876–891, April 2014.
- [9] F. Borrelli, A. Bemporad, and Morari M. *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.
- [10] S. A. Burden, S. S. Sastry, D. E. Koditschek, and S. Revzen. Event-selected vector field discontinuities yield piecewise-differentiable flows. *SIAM Journal on Applied Dynamical Systems*, 15(2):1227–1267, 2016.
- [11] K. Byl and R. Tedrake. Approximate optimal control of the compass gait on rough terrain. In *Robotics and Automation. IEEE International Conference on*, pages 1258–1263, May 2008.
- [12] Q. Cao and I. Poulakakis. Quadrupedal running with a flexible torso: control and speed transitions with sums-of-squares verification. *Artificial Life and Robotics*, 21(4):384–392, Dec 2016.
- [13] J. Carius, M. Wermelinger, B. Rajasekaran, K. Holtmann, and M. Hutter. Deployment of an autonomous mobile manipulator at MBZIRC. *Journal of Field Robotics*, 35(8):1342–1357, 2018.
- [14] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E.R. Westervelt, Carlos Canudas-de Wit, and J.W. Grizzle. RABBIT: A testbed for advanced control theory. *Control Systems Magazine, IEEE*, 23(5):57–79, Oct 2003.
- [15] C. Chevallereau, J.W. Grizzle, and C.-L. Shih. Asymptotically stable walking of a five-link underactuated 3-D bipedal robot. *Robotics, IEEE Transactions on*, 25(1):37–50, Feb 2009.
- [16] S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307(5712):1082–1085, 2005.
- [17] H. Dai and R. Tedrake.  $\mathcal{L}_2$ -gain optimization for robust bipedal walking on unknown terrain. In *Robotics and Automation, IEEE International Conference on*, pages 3116–3123, May 2013.
- [18] R.D. Gregg, T. Lenzi, L.J. Hargrove, and J.W. Sensinger. Virtual constraint control of a powered prosthetic leg: From simulation to experiments with transfemoral amputees. *Robotics, IEEE Transactions on*, 30(6):1455–

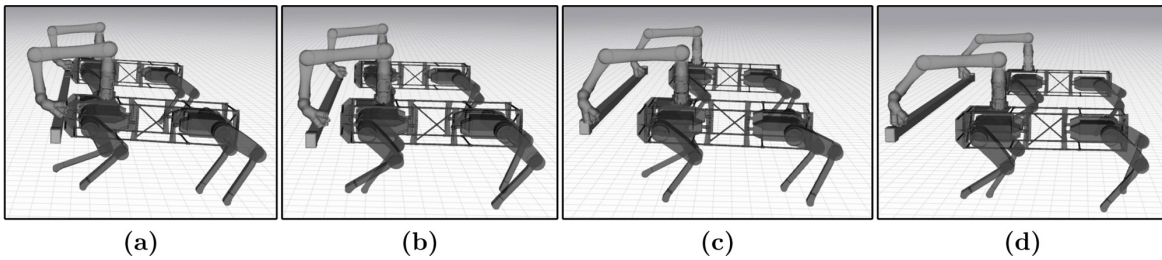


Fig. 7: Simulation snapshots of the closed-loop complex hybrid system that describes the evolution of two agents. Time increases from  $t = 0$  in (a) to  $t = 0.6$  (s) in (d) with the increment  $\Delta t = 0.2$  (s). Here, the robots stably walk along straight lines while changing the location of the object's CoM from the one shown in (a) to its desired location shown in (d).

- 1471, Dec 2014.
- [19] J.W. Grizzle, G. Abba, and F. Plestan. Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. *Automatic Control, IEEE Transactions on*, 46(1):51–64, Jan 2001.
- [20] K. Akbari Hamed, W. Ma, and A. D. Ames. Dynamically stable 3D quadrupedal walking with multi-domain hybrid system models and virtual constraint controllers. In *American Control Conference*, Accepted to appear, 2019. URL <https://arxiv.org/abs/1810.06697>.
- [21] O. Harib, A. Hereid, A. Agrawal, T. Gurriet, S. Finet, G. Boeris, A. Duburcq, M. E. Mungai, M. Masselin, A. D. Ames, K. Sreenath, and J. W. Grizzle. Feedback control of an exoskeleton for paraplegics: Toward robustly stable hands-free dynamic walking. *IEEE Control Systems Magazine, arXiv preprint arXiv:1802.08322*, 2019, Accepted to Appear.
- [22] A. Hereid and A. D. Ames. FROST: Fast robot optimization and simulation toolkit. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 719–726, Sept 2017.
- [23] A. Hereid, C. M. Hubicki, E. A. Cousineau, and A. D. Ames. Dynamic humanoid locomotion: A scalable formulation for HZD gait optimization. *IEEE Transactions on Robotics*, pages 1–18, 2018.
- [24] Y. Hurmuzlu and D. B. Marghitu. Rigid body collisions of planar kinematic chains with multiple contact points. *The International Journal of Robotics Research*, 13(1): 82–92, 1994.
- [25] A. Isidori. *Nonlinear Control Systems*. Springer; 3rd edition, 1995.
- [26] W. Ma, S. Kolathaya, E. R. Ambrose, C. M. Hubicki, and A. D. Ames. Bipedal robotic running with DURUS-2D: Bridging the gap between theory and experiment. In *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control*, pages 265–274. ACM, 2017.
- [27] I.R. Manchester, U. Mettin, F. Iida, and R. Tedrake. Stable dynamic walking over uneven terrain. *The International Journal of Robotics Research*, 30(3):265–279, 2011.
- [28] A. E. Martin, D. C. Post, and J. P. Schmiedeler. The effects of foot geometric properties on the gait of planar bipeds walking under HZD-based control. *The International Journal of Robotics Research*, 33(12):1530–1543, 2014.
- [29] R. M. Murray, D. C. Deno, K. S. J. Pister, and S. S. Sastry. Control primitives for robot systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(1): 183–193, Jan 1992.
- [30] R.M. Murray, Z. Li, and Sastry S.S. *A Mathematical Introduction to Robotic Manipulation*. Taylor & Francis/CRC, 1994.
- [31] H.-W. Park, P. M. Wensing, and S. Kim. High-speed bounding with the MIT Cheetah 2: Control design and experiments. *The International Journal of Robotics Research*, 36(2):167–192, 2017.
- [32] M. Posa, M. Tobenkin, and R. Tedrake. Stability analysis and control of rigid-body systems with impacts and friction. *IEEE Transactions on Automatic Control*, 61 (6):1423–1437, June 2016.
- [33] I. Poulakakis and J.W. Grizzle. The spring loaded inverted pendulum as the hybrid zero dynamics of an asymmetric hopper. *Automatic Control, IEEE Transactions on*, 54(8):1779–1793, Aug 2009.
- [34] A. Ramezani, J.W. Hurst, K. Akbari Hamed, and J.W. Grizzle. Performance analysis and feedback control of ATRIAS, a three-dimensional bipedal robot. *Journal of Dynamic Systems, Measurement, and Control December, ASME*, 136(2), December 2013.
- [35] M.W. Spong and F. Bullo. Controlled symmetries and passive walking. *Automatic Control, IEEE Transactions on*, 50(7):1025–1031, July 2005.
- [36] M.W. Spong, J.K. Holm, and D. Lee. Passivity-based control of bipedal locomotion. *Robotics Automation Magazine, IEEE*, 14(2):30–40, June 2007.
- [37] K. Sreenath, H.-W. Park, I. Poulakakis, and J. W. Grizzle. Compliant hybrid zero dynamics controller for achieving stable, efficient and fast bipedal walking on MABEL. *The International Journal of Robotics Research*, 30(9):1170–1193, August 2011.
- [38] J. Thomas, J. Polin, K. Sreenath, and V. Kumar. Avian-Inspired Grasping for Quadrotor Micro UAVs. In *Volume 6A: 37th Mechanisms and Robotics Conference*, page V06AT07A014. ASME, 8 2013.
- [39] R. Vasudevan. *Hybrid System Identification via Switched System Optimal Control for Bipedal Robotic Walking*, pages 635–650. Springer International Publishing, Cham, 2017.
- [40] E.R. Westervelt, J.W. Grizzle, and D.E. Koditschek. Hybrid zero dynamics of planar biped walkers. *Automatic Control, IEEE Transactions on*, 48(1):42–56, Jan 2003.



- [41] H. Zhao, J. Horn, J. Reher, V. Paredes, and A.D. Ames. Multicontact locomotion on transfemoral prostheses via hybrid system models and optimization-based control. *IEEE Transactions on Automation Science and Engineering*, 13(2):502–513, April 2016.