Engineering Compliance in Legged Robots via Robust Co-Design

Gabriel Bravo-Palacios[1], *Member, IEEE*, He Li[1] *Member, IEEE*, and Patrick M. Wensing[1], *Senior Member, IEEE*

***Abstract*—This paper presents a design framework for the scalable co-design of hardware and control as applied to improving the energy efficiency of legged robots with parallel compliance. The proposed framework uses the Alternating Direction Method of Multipliers (ADMM) for design synthesis by solving large-scale trajectory optimization (TO) problems. Specifically, we use Stochastic Programming (SP) constructs to model design uncertainty associated with terrain properties, and enforce robustness by co-optimizing the robot morphology, a nominal trajectory, and a feedback control policy. Our framework is applied to tune the design of parallel elastic actuation (PEA) via considering how the PEA can be used to actively tailor compliance to different locomotion scenarios. The design optimization framework is validated with the MIT Mini Cheetah quadruped, where added compliance reduces its cost of transport (CoT) by 58.3% in simulation of optimized planar bounding gaits, and up to 17.4% and 8.3% in experiments when executing trotting and bounding gaits respectively.**

***Index Terms*—Optimization and optimal control, legged robots, mechanism design.**

I. Introduction

THE passive dynamics of biological systems is envisioned as a form of embodied intelligence during locomotion, triggered by physical interactions with the world [1], [2]. This paper aims to contribute toward the design of legged robots embodied with mechanical intelligence via the concurrent design (co-design) of their mechanical and control systems. The complexity of jointly modeling dynamics, hardware, and control systems has led to decoupled design strategies that challenge the systematic engineering of robots. We propose a computational framework, rooted in trajectory optimization (TO), that produces robots with passive dynamics crafted via hardware and feedback control co-design that can be applied across different locomotion scenarios (e.g., multiple terrains).

The proposed co-design framework is an evolution of our previous conference work [3] using the Alternating Direction Method of Multipliers (ADMM) [4] to manage computational scalability when considering multiple scenarios. Fig. 2 presents an updated parallelizable strategy that includes steps to handle the extra complexity that arises when considering passive dynamics from added compliance. To exploit passive dynamics, a global controller is complemented by a local controller that manages the compliance of pneumatic cylinders acting as air springs. We choose linear pneumatic actuators as PEAs due to their light weight, high power density, and versatility operating in active and passive modes, which enabled us to modulate the PEA compliance via control. Despite their advantages, pneumatic actuators are difficult to control due to air compressibility and nonlinear dynamics [5], [6]. However, design decentralization via the ADMM simplifies the treatment of pneumatic models for motion planning via TO.

[1]G. Bravo-Palacios, He Li, and P. M. Wensing are with the Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN, 46556 USA. (e-mail: gbravopa@nd.edu, hli25@nd.edu, pwensing@nd.edu)

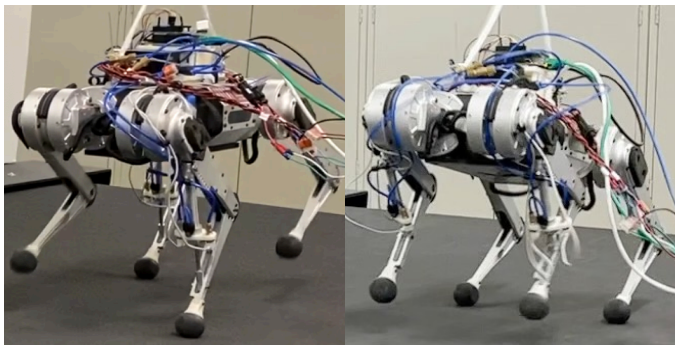

Fig. 1. MIT Mini Cheetah with pneumatic elastic actuators: (Left) Trotting configuration, (Right) Bounding configuration.

Similarly to series elastic actuators (SEAs), which incorporate an elastic element in series between an actuator and the load, PEAs can store energy during the flight and stance phases of a gait. PEAs have a compliant element connected in parallel to another actuator, generally a DC electric motor. As an advantage over SEAs, the PEA torque adds to the torque of the actuator connected in parallel. The additive torque combination can contribute to reduce the thermal losses of electric motors [7]. Nonetheless, on the negative side of PEAs, if the generated torques oppose each other, the electric motor will end up applying more work to counteract the torque of the elastic element. This challenge can lead to PEA designs that are very task-specialized, wherein favorable parallel elasticity that reduces energy use in one situation may instead lead to drastic increases in other operating regimes [8]. Such conflicts make design optimization difficult for PEAs, and makes them an excellent candidate to validate our multi-scenario approach.

A. Related Work

Without explicit co-design, designs that exploited passive dynamics started producing impressive robots in the 80's. Some of the best examples are the running robots of Raibert [9], who used pneumatic actuators to imbue his robots with a springy bouncing behavior modulated by active control. Such a combination yielded fast and agile legged robots capable of performing gymnastic maneuvers [10]. Similar to Raibert's robots, this paper aims to produce robots embodied with mechanical intelligence via exploiting passive dynamics, but we aim to do so through automated methods that ensure the passive dynamics are favorable across a variety of scenarios. Overall, we refer to mechanical intelligence as the ability of a robot to passively execute tasks that are not pre-programmed, as with soft robots [11], or without the sole intervention of a centralized controller, as is intended in this work.

There are no explicit guidelines to produce designs that exploit passive dynamics both in terms of mechanical design

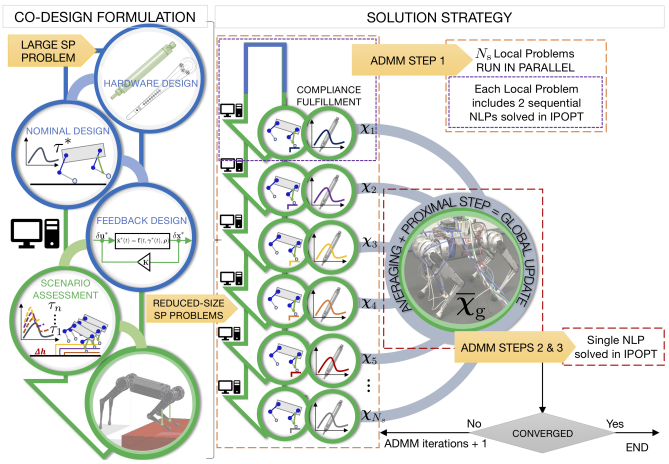

Fig. 2. Evolved ADMM-based Co-Design enforces agreement between local designs $\boldsymbol{\chi}_s$ and a global design $\overline{\boldsymbol{\chi}}_g$. Distributed PEA force fulfillment, and averaging and proximal steps ease complexity due to pneumatic models.

and control. Our work intends to fill the literature gap by presenting the co-design of hardware and control as a formal framework to produce robots that exploit their passive dynamics to facilitate control. In this paper, the connection with passive dynamics comes from applying co-design to optimize compliance from PEAs in legged robots.

Regarding co-design strategies, an important body of work has used TO to produce designs targeting a single scenario [12]–[14]. The main disadvantage is the misrepresentation of performance in other scenarios. More recent work has started incorporating multi-scenario strategies through bi-level schemes that integrate hardware and control design steps as part of outer and inner optimization loops. In a broad sense, such schemes use artificial intelligence (AI) to manage the hardware design step while allowing for multi-scenario assessment. Common such methods applied recently for robot design include covariance matrix adaptation evolution (CMA-E) [15]–[17] and reinforcement learning (RL) [18], [19]. Our implementation uses gradient-based optimization and exploits the distributed structure of ADMM to manage multi-scenario convergence requirements. Table I compares the strategies referred above in terms of their features for robot co-design.

Adding elastic actuation to legged robots has been common in research to improve energy efficiency for locomotion. Among robots with elastic actuators, legged robots using SEAs include the original HyQ [20] and ANYmal [21] quadrupeds, and the Cassie [22] and ATRIAS [23] bipeds. Legged robots built with compliance from PEAs include the SpaceBok [24] and most recently modified ANYmal, AoPs [19], quadrupeds, the BirdBot [25] biped, and the SPEAR [26] hopper. All citations report energy-related cost reductions (e.g., AoPS 33% torque-square efficiency improvement); however, the design methods and metrics are robot specific, which our co-design strategy aims to improve. Table II compares the previous robots in terms of their features and design considerations. In contrast to the AoPS implementation [19], which also co-designs hardware and control and is the most recent adding elastic actuation to a quadruped robot, we show experimental results with bounding gaits in addition to trotting gaits. Bounding gaits are more dynamic and challenging to control than trotting gaits, which justifies considering them herein.

TABLE I
RELATED CO-DESIGN STRATEGIES

Main Features	*References* [12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]	*This Work*
Base Method	SQP		QP	DDP					Collocation
Search Method		GA		CMA-E	CMA-E	CMA-E	RL	RL	
Multi-Scenario							✓	✓	✓
Energy Objective		✓	✓	✓		✓		✓	✓
Feedback Controller			✓	✓	✓				✓
Uncertainty							✓	✓	✓
Robustness				✓	✓	✓			✓
Full Dynamics			✓	✓	✓		✓	✓	✓
Predefined Motion	✓	✓	✓	✓	✓	✓		✓	✓
Compliance								✓	✓

GA: Genetic Algorithm

TABLE II
REPRESENTATIVE LEGGED ROBOTS WITH ELASTIC ACTUATION

Main Features	*References* [20]	[21]	[22]	[24]	[19]	[25]	[26]	*This Work*
Robot	Hy-Q	ANYmal	Cassie	SpaceBok	AoPS	BirdBot	SPEAR	MC
1st Version	✓	✓	✓	✓	✓	✓	✓	✓
Actuation	SEA	SEA	SEA	PEA	PEA	PEA	PEA	PEA
Compliance Source	H-O	M-S	M-S	M-S	M-S	M-S	M-S	P-A
Co-design					✓			✓
Energy Metric	✓	✓	✓	✓	✓	✓	✓	✓
Robustness Metric			✓					✓
Compliance Control	✓	✓					✓	✓
Hardware Proof	✓	✓	✓	✓	✓	✓	✓	✓
Showcased Gaits	1	1	2	3	1	1	1	2

H-O: Hydraulics-Oil, M-S: Mechanical-Spring, P-A: Pneumatics-Air, MC: Mini Cheetah

B. Preliminaries

We investigate whether the compliance in hardware can be shaped and controlled via distributed co-design of hardware and control to exploit passive dynamics for energetic gains. For design exploration, we consider the addition of pneumatic cylinders as PEAs to a pair of legs of the MIT Mini Cheetah [27]. We apply ideas of decentralized control to ease scalability limitations that arise in legged systems with many DoFs. Our framework co-optimizes morphology and control parameters, while accounting for the central or decentralized effect of each of them in completing multiple tasks. In doing so, we intend to embody our designs with mechanical intelligence as a physical property (e.g., PEA compliance) that reduces energy and control effort when facing disturbances.

C. Contribution

The contributions of this work can be summarized as follows. First, our work formalizes the co-design of hardware and control as a framework to produce robots that exploit their passive dynamics to facilitate control. We aim to design complex legged robots (i.e., high number of DoFs), which imposes scalability limitations for simulation and control. Second, we provide a parallelizable pipeline for design of compliant robots. The accounting of compliance accentuates the convergence limitations for design, which our framework handles through distributed computation via the ADMM.

The work herein is organized as follows: Section II covers background material on the co-design formulation and system

dynamics. Section III describes the control policy added for robust co-design. Section IV overviews our co-design solution method using the ADMM, with Section V describing numerical implementation details when applied for robot co-design with PEAs. Sections VI and VII show simulation and experimental results, respectively. Finally, Section VIII provides concluding remarks.

II. Robot Refinement via Co-Design

Our work envisions embodying mechanical intelligence into a robot, by applying co-design strategies to refine an existing design. We account for deployment uncertainty at design time, by considering diverse terrain characteristics in simulation, ensuring that the design is not overfit to a nominal scenario. This section first describes our TO problem for a single scenario before then extending to multi-scenario design, and finally detailing the dynamics considered.

A. Trajectory Optimization Problem for Co-Design

The robot co-design problem for a single scenario takes the form of a TO problem extended with decision variables for the design parameters $\boldsymbol{\rho}$, in addition to the state and control trajectories $\boldsymbol{\gamma}(t) = (\mathbf{x}(t), \mathbf{u}(t)) \in \mathbb{R}^n \times \mathbb{R}^m$, and the final time, t_f. Assuming there exists a vector $\boldsymbol{\xi} \in \mathbb{R}^s$ that parameterizes this single scenario, the TO problem for co-design is given by a baseline problem

$$\underset{t_\mathrm{f}, \boldsymbol{\gamma}(\cdot), \boldsymbol{\rho}}{\text{minimize}} \; V_{\boldsymbol{\xi}}(t_\mathrm{f}, \boldsymbol{\gamma}(\cdot), \boldsymbol{\rho}), \tag{1a}$$

$$\text{subject to} \; \dot{\mathbf{x}}_{\boldsymbol{\xi}}(t) = \mathbf{f}(t, \boldsymbol{\gamma}(\cdot), \boldsymbol{\rho}) \quad \forall\, t, \tag{1b}$$

$$\mathbf{h}_{\boldsymbol{\xi}}(t, \boldsymbol{\gamma}(\cdot), \boldsymbol{\rho}) \leq \mathbf{0} \quad \forall\, t. \tag{1c}$$

where Eq. (1a) denotes the optimization objective using a cost functional $V_{\boldsymbol{\xi}}(\cdot)$, (1b) represents dynamics constraints, and (1c) includes path and boundary constraints. We used direct collocation [28] to transcribe this problem into a finite-dimensional nonlinear programming problem (NLP).

B. Multi-scenario Co-Design via Stochastic Programming

Our SP formulation [29] manages the morphology-control trade-off through probabilistic reasoning, and co-optimization of morphology parameters $\boldsymbol{\rho}$, a control policy $\boldsymbol{\pi}$, and state-control trajectories $\boldsymbol{\gamma}^*(\cdot)$ and $\{\boldsymbol{\gamma}_i(\cdot)\}_{i=1}^{N_\mathrm{s}}$, for a nominal scenario $\boldsymbol{\xi}^*$, and N_s perturbed scenarios $\{\boldsymbol{\xi}_i\}_{i=1}^{N_\mathrm{s}}$, respectively. The multi-scenario co-design problem takes the form:

$$\mathcal{P}(\{\boldsymbol{\xi}_i\}_{i=1}^{N_\mathrm{s}}) := \tag{2a}$$

$$\underset{t_\mathrm{f}, \boldsymbol{\rho}, \boldsymbol{\gamma}^*(\cdot), \boldsymbol{\pi}(\cdot), \{\boldsymbol{\gamma}_i(\cdot)\}}{\text{minimize}} \; V_{\boldsymbol{\xi}^*}(\boldsymbol{\gamma}^*, \boldsymbol{\rho}) + \sum_{i=1}^{N_\mathrm{s}} p(\boldsymbol{\xi}_i) V_{\boldsymbol{\xi}_i}(\boldsymbol{\gamma}_i, \boldsymbol{\rho}),$$

$$\text{subject to} \; \dot{\mathbf{x}}^*(t) = \mathbf{f}(t, \boldsymbol{\gamma}^*(t), \boldsymbol{\xi}^*, \boldsymbol{\rho}) \quad \forall t, \tag{2b}$$

$$\mathbf{h}_{\boldsymbol{\xi}^*}(t, \boldsymbol{\gamma}^*(t), \boldsymbol{\rho}) \leq \mathbf{0} \quad \forall t, \tag{2c}$$

$$\dot{\mathbf{x}}_i(t) = \mathbf{f}(t, \boldsymbol{\gamma}_i(t), \boldsymbol{\xi}_i, \boldsymbol{\rho}) \quad \forall i, t, \tag{2d}$$

$$\mathbf{h}_{\boldsymbol{\xi}_i}(t, \boldsymbol{\gamma}_i(t), \boldsymbol{\rho}) \leq \mathbf{0} \quad \forall i, t, \tag{2e}$$

$$\mathbf{u}_i(t) = \boldsymbol{\pi}(t, \mathbf{x}_i(t), \boldsymbol{\gamma}^*(t)) \quad \forall i, t, \tag{2f}$$

where $p(\boldsymbol{\xi}_i) \in [0, 1]$ is the probability of realization of the scenarios parameterized by $\boldsymbol{\xi}_i$. There exist dynamics and trajectory constraints (2b)-(2e) for the nominal and the disturbed scenarios. Eq. (2f) pre-specifies the control trajectories for the perturbed scenarios based on the control policy $\boldsymbol{\pi}$. Hence, the disturbed scenarios shape the nominal trajectory and the feedback policy, adding design robustness.

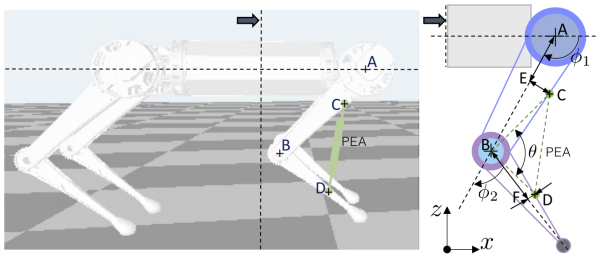

Fig. 3. Refined Robot: (Left) MIT Mini Cheetah with added pneumatic actuators (green) at the front legs. (Right) Schematic of one front leg. The distances to attach the PEAs are: $l_1 = \overline{AE}$, $l_2 = \overline{CE}$, $l_3 = \overline{BF}$, $l_4 = \overline{DF}$.

Using SP constructs to build a TO problem involves conceptualization "stages." The first stage considers deterministic decisions, while the second stages include decisions that are dependent on the realization of scenarios. The SP co-design formulation requires consensus around the first-stage decisions (i.e., design parameters, nominal trajectories, and control policy). The perturbed trajectories $\{\boldsymbol{\gamma}_i(\cdot)\}_{i=1}^{N_\mathrm{s}}$ represent second-stage decisions. The ADMM formulation exploits the SP stage-decision reasoning to gain scalability.

For robot co-design herein, the optimization objectives account for the total task duration $t_f = T$ and the total electrical energy required by the electric actuators, which measure task performance and energy efficiency, respectively. To treat hybrid effects, the problem $\mathcal{P}(\cdot)$ in (2) is further broken into n_ph motion phases. Hence, the total time $T = \sum_{i=1}^{n_\mathrm{ph}} T_i$, where T_i is the duration of phase i. Further formulation details are annexed as supplementary material.

C. Dynamics: Robot Model

Consider a planar floating-base robot with kinematic loops created by a PEA at each front leg ($\triangle BCD$ in Fig. 3). The robot state $\mathbf{x} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$ contains the configuration $\boldsymbol{q} \in \mathbb{R}^{11}$ with two degrees of freedom (DoFs) per leg and three DoFs for the floating base. The dynamics take the form:

$$\boldsymbol{H}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{\tau}_\mathrm{g}(\boldsymbol{q}) = \boldsymbol{S}^\top \boldsymbol{\tau} + \boldsymbol{J}_c(\boldsymbol{q})^\top \boldsymbol{f}_c + \boldsymbol{J}_l(\boldsymbol{q})^\top \boldsymbol{\lambda}_l, \tag{3}$$

where $\boldsymbol{H}$, $\boldsymbol{C}$, $\boldsymbol{\tau}_g$, $\boldsymbol{\tau}$, and $\boldsymbol{S}$ denote the joint-space inertia matrix, Coriolis and centrifugal terms, generalized gravitational and actuator torques, and the selection matrix, respectively. $\boldsymbol{J}_c$ and $\boldsymbol{f}_c$ represent the contact Jacobian and associated external forces. The PEAs are approximated as massless force sources, acting on the thigh and shank, with $\boldsymbol{\lambda}_l$ representing the scalar forces along the PEAs, and $\boldsymbol{J}_l$ the Jacobian matrix that accounts for the action-reaction effect at the attachment points. In TO, we specify $\boldsymbol{\lambda}_l$ as a decision variable. Approximating the PEA as massless thus avoids introducing a rigid-body that would result in loop-closure constraints in the dynamics due to the kinematic loop. We enforce (3) using constraints that replicate the Recursive Newton Euler Algorithm (RNEA) [30] by treating the PEA forces as equal and opposite external forces on the thigh and shank.

An alternative approach would be to add the effect of the PEA force directly to the knee torque:

$$\tau_{kn,eff} = F_{\mathrm{p}} \frac{\ell_5 \ell_6 \sin(\theta)}{\ell_7}, \tag{4}$$

where F_{p} is the piston force, and with respect to Fig. 3, $\ell_5 = \overline{BD}$, $\ell_6 = \overline{BC}$, $\ell_7 = \overline{CD}$, and $\theta = \angle CBD$. This approach would give equivalent dynamics, but does not accurately represent the internal joint loads within the RNEA, and so was not used during whole-body optimization.

D. Actuators

The electrical power and the voltage limits for the gearbox motors follow the actuator models in [29], with regeneration efficiency $\eta_{\mathrm{r}} = 50\%$ and parameters specified in [27]. The pneumatic PEA models follow [5], [31]. The PEA architecture involves a pneumatic system composed mainly of a bi-directional pneumatic cylinder and a five-port proportional control valve. The pneumatic system can be described as a fourth-order control affine nonlinear system, with states:

$$x_1 = x_p, \quad x_2 = \dot{x}_p, \quad x_3 = P_1, \quad x_4 = P_2, \tag{5}$$

where, x_p and $\dot{x}_p$ are the piston position and velocity, respectively, and P_1 and P_2 are the (absolute) pressure at the cap-end and rod-end cylinder chambers. Details are provided in the supplementary material.

III. Control Strategies

This section describes a phase-adaptive control framework for the robot including PEAs. The framework is primarily composed of two components, a global controller and a local (decentralized) PEA controller. The subsequent section details the methods for co-design posed over top of both controllers.

A. Robust Co-Design via Global Feedback Optimization

For co-optimization, a global controller targets the kinematic states of the robot (i.e., not considering the PEA dynamics). The controller co-optimization (rather than fixing a few control policies in advance) allows the full capabilities of the design to be explored and assessed. We embed robustness reasoning into the co-design formulation through co-optimization of the nominal trajectory, the global controller, and the morphology. Accordingly, the control policy $\boldsymbol{\pi}(\cdot)$ in (2a) denotes a control law

$$\mathbf{u}_i(t) = \mathbf{u}^*(t) - \mathbf{K}(t)(\mathbf{x}_i(t) - \mathbf{x}^*(t)), \tag{6}$$

applied for each scenario $\boldsymbol{\xi}_i$, where $\mathbf{K}(t)$ is a time-varying gain matrix included as a decision variable in the optimization. The global states $\mathbf{x}_i(t)$ do not account for the PEA pressure states, which are controlled locally as described below.

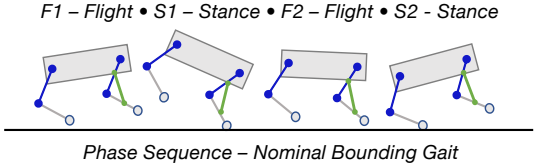

Fig. 4. Quadruped with PEA: Phases in nominal bounding gait.

B. Decentralized Control for Passive Phase Adaptation

When the cylinders are pressurized, the knee motor torque τ_m is modulated according to the following scheme

$$\tau_m = \tau_{m,g} - \tau_{m,l}, \tag{7}$$

where $\tau_{m,g}$ is the global-controller output, and $\tau_{m,l} = \tau_{kn,eff}$ from (4) is an effective knee torque contributed by the PEA, and as targeted by the output of a local controller (See Fig. 5).

The local controller behaves differently across the phases of the bounding gait depicted in Fig. 4: it generates active and passive forces, respectively, during the F1 and S1 phases, and zero forces during the F2 and S2 phases.

1) Active PEA Force: The force $F_{\mathrm{p}} = F_{\mathrm{pf}}$ during the F1 flight phase, prior to contact of the legs carrying PEAs, is an active force because valve control is required to achieve a certain piston force before the S1 stance phase.

We obviate the derivation of the active force model for simplicity of presentation, with the interested reader referred to [32] for further details. The model combines the pressure dynamics of the cylinder chambers, and the result is:

$$\dot{F}_{\mathrm{pf}} = -\left(\frac{k_h A_p \dot{x}_p}{V_1}\right) F_{\mathrm{pf}} - \left(\frac{k_h A_p^3 \dot{x}_p P_2 \ell_s}{V_1 V_2}\right), \tag{8}$$

where k_h is the air specific heat ($k_h = 1.1$ $[-]$[1] assuming the gas expansion process is close to isentropic), A_p is an approximated cross-section piston area (average of the cap-end and rod-end piston areas [31]), and $V_{1,2}$ are the cap-end and rod-end cylinder chambers, respectively. Eq. (8) reduces the number of variables needed to mimic the pressure dynamics, and it does not depend on the nonlinear valve flow dynamics.

The above model allowed us to distribute the complexity of the pneumatic force dynamics across the passes of the co-design algorithm presented in the following sections.

2) Passive PEA Force: During the S1 stance phase, we assume the PEA becomes an air spring triggered by the passive compression of the air in the cylinder chambers. The pneumatic force arises from air compression in the cap-end cylinder chamber, with the valve locked in the final position of the previous flight phase. Modeling this control mode, we also optimize the cylinder cap-end pressure before impact, P_1^*.

The cylinder pressure can be considered as mechanical feedback from the passive dynamics, which then modulate the active motor control required. We modeled the passive force generation in the cap-end cylinder chamber as a compression/expansion process, with air treated as an ideal gas. Assuming that the air compression/expansion are reversible processes, the following expression relates the change in volume to the pressure change:

$$P_1 / P_{01} = (V_{01}/V_1)^{k_h}, \tag{9}$$

where the 01 subscript denotes initial conditions, with $P_{01} = P_1^*$. From (9), the force generated by the piston is given by

$$F_{\mathrm{p}} = F_{\mathrm{ps}} = A_p \left[P_1^* \left(\frac{\ell_s}{\ell_s - \Delta x_p} \right)^{k_h} - P_{\mathrm{atm}} \right], \tag{10}$$

[1] [-] denotes dimensionless quantities.

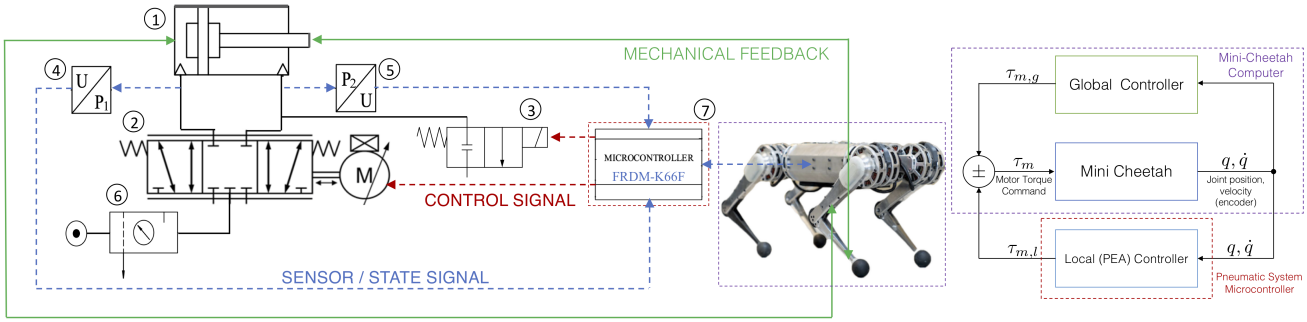

Fig. 5. (Left) Experimental setup: (1) Pneumatic cylinder, (2) five-way proportional control valve, (3) two-port solenoid valve, (4) and (5) analog pressure sensors, (6) compressor with air filtering and regulation unit, (7) local controller board. (Right) Control Strategy: The global controller commands the states of the robot. The local controller commands the pneumatic cylinder. The knee-joint torque is modulated according to the PEA contribution.

where the change in piston position Δx_p is measured from the start of the stance phase. Notice that (9) describes an idealized process; however in reality, losses due to heat transfer, friction, and valve leakage occur, which will be analyzed in Sec. VII.

IV. Scalable Co-Design via ADMM

The sections below describe the underlying co-design mathematical framework based on the ADMM [4]. The co-design problem retains the structure given by the two-stage SP formulation to handle uncertainty at design time, with the ADMM coordinating the solution of reduced-size problems that in total add to a large-size SP problem.

The following sections detail updated versions of steps that we initially reported in the preliminary conference version of this work [3]. We apply a multi-block decomposition that splits the multi-scenario problem $\mathcal{P}(\{\boldsymbol{\xi}_i\}_{i=1}^{N_\mathrm{s}})$ into $n_\mathrm{b} = (N_\mathrm{s}/n_\mathrm{s}) \in \mathbb{Z}$ blocks with $n_\mathrm{s} \leq N_\mathrm{s}$ scenarios each.

A. Global-Design Consensus Problem

For each scenario block $s \in \{1, \dots, n_\mathrm{b}\}$, the variable set $\boldsymbol{\chi}_s := (\boldsymbol{\rho}, T, \boldsymbol{\gamma}^*(\cdot), \mathbf{K}(\cdot))$ collects a local version of the decision variables. The goal using ADMM is to reach consensus between local sets of variables, $\{\boldsymbol{\chi}_s\}_{s=1}^{n_\mathrm{s}}$, and a global set $\overline{\boldsymbol{\chi}}_\mathrm{g} := (\boldsymbol{\rho}_\mathrm{g}, T_\mathrm{g}, \boldsymbol{\gamma}_\mathrm{g}^*(\cdot), \mathbf{K}_\mathrm{g}(\cdot))$. Hence, we define the global-design consensus problem as

$$\underset{\{\boldsymbol{\chi}_s\},\{\boldsymbol{\gamma}_{s,i}(\cdot)\},\overline{\boldsymbol{\chi}}_\mathrm{g}}{\text{minimize}} \ \mathcal{R}(\overline{\boldsymbol{\chi}}_\mathrm{g}) + \sum_{s=1}^{n_\mathrm{b}} \mathcal{V}_s(\boldsymbol{\chi}_s, \{\boldsymbol{\gamma}_{s,i}\}_{i=1}^{n_\mathrm{s}}), \tag{11a}$$

$$\text{subject to } \boldsymbol{\chi}_s = \overline{\boldsymbol{\chi}}_\mathrm{g}, \ s = 1, ..., n_\mathrm{b} \tag{11b}$$

in addition to constraints (2b)-(2f) for each scenario block s. The cost term $\mathcal{R}$ is a regularization function that we use to enforce energy minimization. The second term denotes the cost for the multi-scenario problem $\mathcal{P}(\{\boldsymbol{\xi}_i\}_{i=1}^{N_\mathrm{s}})$ split into n_b blocks, with $\mathcal{V}_s(\cdot)$ as a compact representation of (2a),

$$\mathcal{V}_s(\cdot) := V_{\boldsymbol{\xi}_s^*}(\boldsymbol{\chi}_s) + \sum_{i=1}^{n_\mathrm{s}} p(\boldsymbol{\xi}_{s,i}) V_{\boldsymbol{\xi}_{s,i}}(\boldsymbol{\gamma}_{\boldsymbol{\xi}_{s,i}}(\cdot), \boldsymbol{\rho}), \tag{12}$$

for a reduced scenario set n_s, and with the probability $p(\cdot)$ prescribed from sampling of the total scenario set of size N_s.

The constraints (11b) enforce that the local versions $\{\boldsymbol{\chi}_s\}_{s=1}^{n_\mathrm{b}}$ reach consensus with the global version $\overline{\boldsymbol{\chi}}_\mathrm{g}$. The local and global versions can represent different variations on a single design. However, as the ADMM iteration process evolves, the local versions become approximately equal to the global version. The strategy follows the scheme from [33].

The ADMM accounts for the consistency constraints (11b), via the residuals $\mathbf{r}_{s,j} = \boldsymbol{\nu}_{s,j} - \boldsymbol{\nu}_{\mathrm{g},j}$ for every block s, where $\boldsymbol{\nu}_{s,j}$ and $\boldsymbol{\nu}_{\mathrm{g},j}$ are generic variables that take a value according to $j \in \mathcal{J} := \{\rho, T, \gamma^*, K\}$.

B. ADMM-based Distributed SP Co-Design

We define $\mathcal{P}_\mathcal{L}(\cdot)$ and $\mathcal{P}_G(\cdot)$ as the problems to find the local and global designs, respectively. Then, the ADMM solution of (11) follows the steps below:

$$\mathcal{P}_\mathcal{L}(\boldsymbol{\chi}_s, \{\boldsymbol{\gamma}_{s,i}(\cdot)\}):$$

$$\boldsymbol{\chi}_s^{k+1} := \underset{\boldsymbol{\chi}_s, \{\boldsymbol{\gamma}_{s,i}(\cdot)\}}{\text{argmin}} \ \mathcal{V}_s(\cdot) + \sum_{j \in \mathcal{J}} \frac{\zeta_j}{2} \|\mathbf{r}_{s,j} + \mathbf{w}_{s,j}^k\|_2^2, \tag{13a}$$

$$\mathcal{P}_G(\overline{\boldsymbol{\chi}}_\mathrm{g}): \overline{\boldsymbol{\chi}}_\mathrm{g}^{k+1} := \underset{\overline{\boldsymbol{\chi}}_\mathrm{g}}{\text{argmin}} \ \mathcal{R}(\cdot) + \sum_{j \in \mathcal{J}} \frac{n_\mathrm{b} \zeta_j}{2} \|\overline{\mathbf{r}}_j + \overline{\mathbf{w}}_j^k\|_2^2, \tag{13b}$$

$$\mathbf{w}_{s,j}^{k+1} := \mathbf{w}_{s,j}^k + \mathbf{r}_{s,j}^{k+1}, . \tag{13c}$$

where $\mathbf{w}_{s,j}$ are dual variables, $\zeta_j \geq 0$ are penalty parameters, $s = 1, ..., n_\mathrm{b}$ in (13a) and (13c), and (13b) is a global-update step with average residuals and dual variables, respectively,

$$\overline{\mathbf{r}}_j = \left(\frac{1}{n_\mathrm{b}} \sum_{s=1}^{n_\mathrm{b}} \boldsymbol{\nu}_{s,j}^{k+1} \right) - \boldsymbol{\nu}_{\mathrm{g},j}^{k+1}, \quad \overline{\mathbf{w}}_j^k = \frac{1}{n_\mathrm{b}} \sum_{s=1}^{n_\mathrm{b}} \mathbf{w}_{s,j}^k. \tag{14}$$

Following the three-step sequence of the ADMM in base form [4], (13a) optimizes the local designs $\boldsymbol{\chi}_s$, (13b) optimizes the global design $\overline{\boldsymbol{\chi}}_\mathrm{g}$, and (13c) updates the dual variables $\mathbf{w}_{s,j}$ in an alternating manner.

Compared to [3], (13b) represents an averaging step, as in consensus ADMM [4], followed by a proximal step involving regularization $\mathcal{R}(\cdot)$. For SP co-design, the proximal step denotes the compromise of the global design $\overline{\boldsymbol{\chi}}_\mathrm{g}$ between minimizing $\mathcal{R}(\cdot)$ and being near to the average of the local designs $\boldsymbol{\chi}_s^{k+1}$ across all scenarios N_s. The update in (13b) affects the convergence of the ADMM algorithm, and it was crucial to solve the posed co-design problem. Overall, it is one of the most important distinctions compared to our past work [3]. In particular, this step improves the convergence of the method for co-design of legged robots with added complexity due to elastic actuators, considering up to $N_\mathrm{s} = 30$ perturbed scenarios for robustness assessment.

C. Formulation Details

The following subsections cover strategies to improve the convergence of the ADMM for co-design of legged robots.

1) Penalty Update: To improve convergence, the penalty parameters ζ_j can be updated every $k_\zeta \geq 1$ ADMM iterations following (13c). A scheme reported to work well in [4], [34] showed satisfactory performance for systems of up to five DoFs (e.g., monopod robot in [3]). We applied a strategy that spaces out the use of this standard scheme, and takes a correction action if the global problem $\mathcal{P}_G(\overline{\boldsymbol{\chi}}_{\mathrm{g}})$ (13b) shows limited feasibility. The strategy follows principles of the Bound-Constrained Lagrangian (BCL) approach [35]. The supplementary material details our penalty-update approach.

2) Over-relaxation: Results in [34] have shown that over-relaxation of the consensus constraints (11b) in (13b) can lead to improved convergence. Over-relaxation is achieved by including a parameter $\alpha > 1$ as follows:

$$\bar{\mathbf{r}}_j = \alpha\left(\frac{1}{n_{\mathrm{b}}}\sum_{s=1}^{n_{\mathrm{b}}} \boldsymbol{\nu}_{s,j}^{k+1}\right) - (1-\alpha)\boldsymbol{\nu}_{\mathrm{g},j}^{k} - \boldsymbol{\nu}_{\mathrm{g},j}^{k+1}, \tag{15}$$

which approximates the quadratic term in (13b). We denote over-relaxation in (13b) by $\mathcal{P}_G(\overline{\boldsymbol{\chi}}_{\mathrm{g}}, \alpha)$, and use $\alpha = 1.5$.

3) Convergence: The ADMM iterates approach convergence when $\mathbf{r}^k \to 0$ as $k \to \infty$. Depending on the stopping criteria, modest accuracy has shown to be sufficient for many applications. We specify a maximum number of iterations ($k_{\max}$), and apply the following stopping condition [4]:

$$\|\mathbf{r}_{s,j}\|_2 \leq \epsilon_j^{\mathrm{pri}} \;\; \text{AND} \;\; \|\mathbf{d}_{\mathrm{g},j}\|_2 \leq \epsilon_j^{\mathrm{dual}},\; j \in \mathcal{J} \tag{16}$$

where $\epsilon_j^{\mathrm{pri}}$ and $\epsilon_j^{\mathrm{dual}}$ are user tolerances (positive scalars).

V. Co-Design Implementation

The co-design framework was built in CasADi [36], with IPOPT [37] and the MA97 linear solver [38]. We carried co-design for the quadruped robot with PEAs, requiring it to jump forward at least 0.3 $[m]$ per gait cycle. We consider an 11-DoF model executing bounding gaits in the sagittal plane, as previously defined in Sec. II-C. Each leg is actuated by two gearbox motors, and the front legs include identical pneumatic actuators connecting the thigh and shank links (See Fig. 3). As an exercise of design refinement, the leg geometry and motor gear ratios are left unvaried, and the co-design formulation optimizes the parameters that specify the pneumatic actuator (bore diameter ϑ and stroke length ℓ_s) and its position relative to the links of each leg ($\ell_{1,3}$ in Fig. 3).

Perturbed scenarios were randomly sampled assuming a stair obstacle with terrain friction coefficient μ raised at a time fraction $T_{\%}$ of the F1 flight phase. The stair characteristics were taken with truncated Gaussian probability densities $\sim \mathcal{N}_{\mu}(0.7, 0.05) \in (0.6, 0.8)$ and $\sim \mathcal{N}_{T_{\%}}(0.8, 0.25) \in (0.5, 1.0)$. The probability $p(\boldsymbol{\xi}_i)$ in (2) follows a multivariate distribution combining the stair characteristics as independent values.

As noted before, consensus was enforced relative to the variable set $\boldsymbol{\chi} := (\boldsymbol{\rho}, T, \boldsymbol{\gamma}^*(\cdot), \mathbf{K}(\cdot))$, with $\boldsymbol{\rho}$ including the bore diameter ϑ and stroke length ℓ_s of the pneumatic cylinders, the attachment coordinates $\ell_{1,3}$, and the setup pressure P_1^*. Table III identifies the corresponding hardware constraints.

The local problem $\mathcal{P}_{\mathcal{L}}(\cdot)$ (13a) for each block $n_{\mathrm{b}} = N_{\mathrm{s}}$ (chosen with one scenario each, $n_{\mathrm{s}} = 1$) included nine phases (4 for $\boldsymbol{\xi}^*$, 5 for $\boldsymbol{\xi}_i$), adding up to $50,159$ variables in the NLP (6 finite elements/phase, 4 Radau collocation points/element, approximated with Lagrange polynomials) passed to IPOPT. The perturbed scenarios included a front stance (S1) after the early impact, followed by a full gait at modified terrain height.

A. PEA Force Optimization

In contrast to the quadruped without PEAs, the number of NLP variables increased by 12% due to the modeled piston dynamics. We optimized the piston dynamics across separate solve routines distributed in the ADMM algorithm as follows, which mitigated the complexity of the pneumatic models.

Consider the three ADMM steps in Sec. IV-B. First, we produced local designs solving problem $\mathcal{P}_{\mathcal{L}}(\cdot)$ with the pneumatic force included during the S1 stance phase, only, using Eq. (10). The solver showed intractability instances when also including the force dynamics model (8) during flight.

The local problem $\mathcal{P}_{\mathcal{L}}(\cdot)$ finds the work required from the PEAs during the S1 stance phase, with P_1^* marking the pressure target for the PEA to reach by the end of the F1 flight phase. Accordingly, as a subroutine for each local design $\boldsymbol{\chi}_s^{k+1}$, we re-optimized the state-control trajectory $\boldsymbol{\gamma}^*(\cdot)$ over the F1 phase alone, keeping the morphology parameters and state-control boundaries fixed, and using the force dynamics model (8) to plan for how to reach this pressure target at touchdown. Fig. 2 labels this step as compliance fulfillment.

Subsequently, we solved the global problem $\mathcal{P}_G(\cdot)$ with information from the local designs $\boldsymbol{\chi}_s^{k+1}$ complemented with pneumatic force trajectories for the F1 and S1 phases. Also as a subroutine, we computed a PEA averaged force $\overline{F}_{\mathrm{pf}}$ for the F1 flight phase, similar to the averaging step in (13b). We included this averaged force as a fixed disturbance for optimization of the global design $\overline{\boldsymbol{\chi}}_{\mathrm{g}}$. The overall co-design approach is summarized in Algorithm 1.

Fig. 6 shows optimization results for a global design $\overline{\boldsymbol{\chi}}_{\mathrm{g}}^{k+1}$ as an output of Algorithm 1. The top subplot showcases the force-optimization scheme described in this subsection. The optimization attempts to minimize the PEA force to zero during the F1 flight phase. The bottom subplot shows this effect: the opposite torque directions at the beginning and end of the F1 flight phase represent the price to pay for

TABLE III
ADMM Implementation: Hardware Constraints

Optimization Variables	*Bounds*	*Units*
Phase Duration	$0 \leq \lvert T_{ph}\rvert \leq 300$	[s]
Base-joint angle	$\lvert\phi_{\mathrm{base}}\rvert \leq \pi/2$	[rad]
Hip-joint angle	$\lvert\phi_{\mathrm{hip}}\rvert \leq 3\pi/8$	[rad]
Knee-joint angle	$\lvert\phi_{\mathrm{knee}}\rvert \leq 3\pi/4$	[rad]
Joint Torque	$\lvert\ \tau_{\mathrm{i}}\ \rvert \leq 200$	[Nm]
Motor Voltage	$\lvert V_{m_i}\rvert \leq 24$	[volts]
Stance Setup Pressure	$P_{\mathrm{atm}} \leq P_1^* \leq 10$	[Bars]
Bore Diameter	$8 \leq \vartheta \leq 24$	[mm]
Stroke Length	$10 \leq \ell_s \leq 1000$	[mm]
Length $\overline{AE}$	$61 \leq \ell_1 \leq 112$	[mm]
Length $\overline{BF}$	$89 \leq \ell_3 \leq 137$	[mm]

Algorithm 1 ADMM for SP Co-Design of Compliant Robots

Inputs:
$\overline{\boldsymbol{\chi}}_{\mathrm{g}}^{0}$, $\mathbf{w}_{s,j}^{0}$, ζ_j^0, $k_\zeta \in \mathbb{Z}^+$, $\alpha \in (1,2)$, $p(\boldsymbol{\xi}_i) \in [0,1]$
$k \leftarrow 0$, $j \in \mathcal{J} := \{\rho, T, \boldsymbol{\gamma}, K\}$
repeat
  Update Local Designs:
  for $s = 1$ **to** n_{b} **do**
    Initialize: $\boldsymbol{\chi}_s \leftarrow \overline{\boldsymbol{\chi}}_{\mathrm{g}}^{k}$, $\{\boldsymbol{\gamma}_{s,i}(\cdot)\} \leftarrow \{\boldsymbol{\gamma}_{s,i}(\cdot)\}^k$
    $\boldsymbol{\chi}_s^{k+1} = solve\ \mathcal{P}_{\mathcal{L}}(\boldsymbol{\chi}_s, \{\boldsymbol{\gamma}_{s,i}(\cdot)\})$
    Fix $\boldsymbol{\rho}$, $\boldsymbol{\gamma}(t_0)$ and $\boldsymbol{\gamma}(t_f)$, and include model (8)
    $F_{\mathrm{pf,s}} = solve$ (1) for the F1 phase alone
  Update Global Design:
    $\overline{F}_{\mathrm{pf}} = (\sum_s^{n_{\mathrm{b}}} F_{\mathrm{pf,s}})/n_{\mathrm{b}}$
    Initialize: $\overline{\boldsymbol{\chi}}_{\mathrm{g}} \leftarrow \overline{\boldsymbol{\chi}}_{\mathrm{g}}^{k}$ and fix $\overline{F}_{\mathrm{pf}}$
    $\overline{\boldsymbol{\chi}}_{\mathrm{g}}^{k+1} = solve\ \mathcal{P}_G(\overline{\boldsymbol{\chi}}_{\mathrm{g}}, \alpha)$
  Update Dual Variables:
  for $s = 1$ **to** n_{b} **do**
    $\mathbf{w}_{s,j}^{k+1} = \mathbf{w}_{s,j}^{k} + \mathbf{r}_{s,j}^{k}$,
  Assess Solution and Update Penalties:
  run *Penalty-update Strategy*
  $k \leftarrow k + 1$
until stopping criterion is satisfied

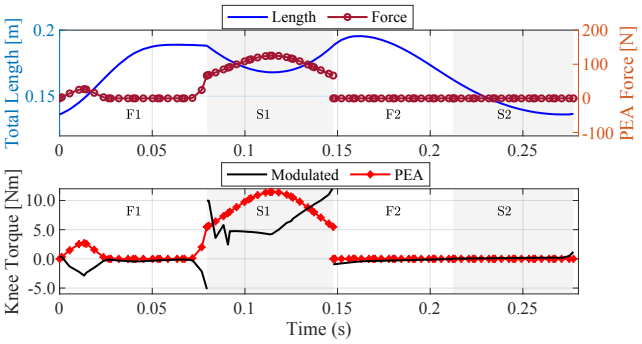

Fig. 6. Quadruped with PEA: (Top) Pneumatic-actuator total length and force. (Bottom) Knee-joint torque as prescribed by control sequence.

loading/unloading the air spring so that its passive dynamics can then offer benefits during the S1 stance phase.

The following section assesses how the distributed structure of the ADMM manages multi-scenario convergence requirements, despite scalability limitations due to the addition of compliance as a mechanical-intelligence carrier.

B. Convergence Results

Our ADMM algorithm uses the total residual $\|\mathbf{r}_{s,j}\|_2$, computed as the norm of all the individual residuals independently of their units, to measure the convergence attained at each iteration. Compared to the case without PEAs in [3], the difficulty in decreasing the residuals grew in connection to the complexity added by the PEA models. The steps to account for the piston force F_{p} impose fixed conditions, which affected the computation time and the consensus rate. Fig. 7 shows the progression of the average normalized error, $\overline{e}$, where we normalized each component of the residual by half the sum of absolute values of the local and global designs.

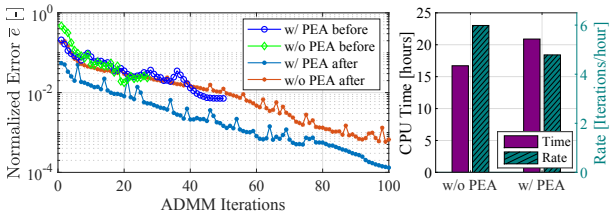

Fig. 7. Co-Design with and without PEA including 30 perturbed scenarios: (Left) Normalized average error before and after algorithm updates, (Right) CPU time in IPOPT and ADMM iteration rate solving $\mathcal{P}_{\mathcal{L}}(\cdot)$ after updates.

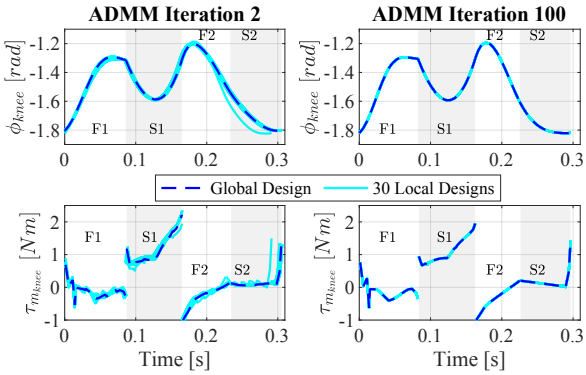

Fig. 8. Quadruped with PEA: Front-leg nominal knee-angle position and control trajectories of the robot jumping over flat terrain after 2 (left) and 100 (right) ADMM iterations. Implementation with 30 disturbance scenarios.

The figure shows the co-design studies with and without PEAs including 30 perturbed scenarios. After 100 ADMM iterations, the residual and error for the case with PEAs are $\|\mathbf{r}_{s,j}\|_2 = 0.078$ and $\overline{e} = 1.32 \times 10^{-4}$ vs. $\|\mathbf{r}_{s,j}\|_2 = 0.097$ and $\overline{e} = 6.67 \times 10^{-4}$ without PEAs. The quality of the error progression is a byproduct of the penalty-update strategy and the recurrent initialization steps in Alg. 1. Fig. 7 contrasts the results before (in [3]) and after the updates included in Alg. 1

Fig. 8 shows the trajectory consensus achieved. The trajectory plots demonstrate the effectiveness of the ADMM pulling the 30 local designs into consensus with the global design.

Closing this section and looking back to Fig. 7, the optimization with 30 perturbed scenarios ran at a rate of five and six iterations per hour, respectively, for the co-design studies with and without PEAs, for 100 ADMM iterations. The added PEA complexity decreased the consensus rate; however, the averaging global-update step (13b) and the penalty-update strategy contributed to preserve the convergence margins close to those observed for the design without PEAs.

VI. Design Validation via Simulation

This section quantifies the robustness and energy efficiency of the optimized designs. We verify how the addition of compliance contributes to reducing the energy and feedback control effort when facing terrain disturbances.

A. Robustness Validation

As a robustness metric, we estimated the probability of failure $\hat{P}(F)$ via Monte-Carlo Simulation (MCS), with failure

denoting scenarios where a design/controller pair results in an infeasible trajectory according to a constraint tolerance of 1×10^{-4} in IPOPT. We ran the extended multi-scenario problem $\mathcal{P}(\{\boldsymbol{\xi}_i\}_{i=1}^{N_s})$ (2) with zero cost, $N_s = 400$ random scenarios, and fixing the global design $\overline{\boldsymbol{\chi}}_g$ found with 10, 20, and 30 perturbed scenarios. Fig. 9 shows the operating space covered with blue and red markers, respectively denoting the successful and failed scenarios. The design co-optimized with 30 scenarios attains zero failure probability, $\hat{P}(F) = 0\%$, with expanded feasibility over the left-side region of the operating space. The same region ($T_{\%} \leq 0.6,\ \ 0.6 \leq \mu \geq 0.8$) is fully infeasible with 10 and 20 scenarios.

A limitation producing Fig. 9 is on the use of IPOPT as a simulator. IPOPT can solve large systems of differential algebraic equations; however, it relies on initialization to enforce convergence. Some scenarios showing adequate constraint satisfaction (i.e., below 1×10^{-4}) required more than one simulation pass to achieve full convergence. We initialized those additional simulation passes with the previous solution or with solutions from neighboring feasible scenarios.

B. Energy Assessment

We use the cost of transport CoT as an efficiency metric. Fig. 10 compares the CoT with and without PEAs. The addition of compliance reduced the CoT by up to 58.3% for the nominal scenario and the design co-optimized with 30 perturbed scenarios. There is a significant decrease in the CoT, even when considering the use of feedback control for the perturbed scenarios, showcasing the advantage over the system without PEAs and the contribution of the passive dynamics via compliance to improve energy efficiency.

Table IV summarizes the optimized results for the quadruped with PEA. As we increased the number of scenarios for co-optimization, the cylinder bore diameter decreased by 11%, while the setup pressure and stroke length increased by 72% and 6%, respectively, compared to the design optimized with 10 scenarios. The morphology combination demonstrates the trade-off achieved between size and force for the PEAs: a small bore diameter requires high pressure to achieve the force required during stance, and vice versa.

VII. Design Validation via Experiments

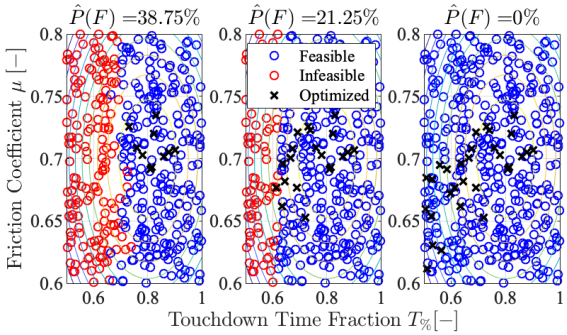

Fig. 9. From left to right, results for global design $\overline{\boldsymbol{\chi}}_g$ co-optimized with 10, 20, and 30 disturbance scenarios (black crosses). Blue and red markers denote feasible and infeasible results, respectively.

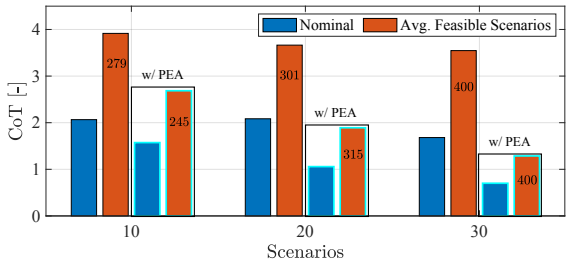

Fig. 10. CoT with and without PEA - Designs co-optimized using 10, 20, and 30 disturbance scenarios. Orange bars inform the number of successful scenarios out of 400 when applying the feedback policy (6).

TABLE IV
Morphology Co-Designed with 10-30 Perturbed Scenarios

Parameters	*Results*			*Units*
	10 Scs.	*20 Scs.*	*30 Scs.*	
Stance Setup Pressure P_1^*	3.3	3.9	5.7	[Bars]
Bore Diameter ϑ	15.99	14.82	14.24	[mm]
Stroke Length ℓ_s	55.94	46.13	59.22	[mm]
Length ℓ_1	70.5	87.7	70.6	[mm]
Length ℓ_3	110	106.3	110	[mm]

We used the MIT Mini Cheetah robot [27] to assess the proposed PEA configuration using pneumatic cylinders. Trotting and bounding gaits were executed using the robot's default convex model predictive controller (cMPC) [39].

A. Experimental Procedure

Fig. 5 shows the experimental setup. A five-way proportional control valve actuated the cylinders with pressurized air from a laboratory outlet. In addition, we used a single two-port solenoid valve to depressurize the cylinders during portions of the gait phases. To measure battery power, we used a secondary setup, including a Coulomb counter connected between the battery and the motors.

In the experiments, the robot executed trotting and bounding gaits with forward motion over a treadmill running at $0.3\ [m/s]$ for 120 seconds. When the robot performed trotting gaits (See Fig. 1), we connected the pneumatic cylinders to the front-left (FL) and hind-right (HR) legs to preserve gait symmetry. For the bounding experiments, we connected the PEAs to the front legs, paralleling the optimization setup.

We present results for three configurations described in Table V, and collected data with supply pressure settings: $P = 3$ to $6\ [Bars]$. Configurations C1 and C2 involve the same cylinder, selected preliminarily from intuition, and based on immediate market availability. Configuration C3 is the best match of the optimized design in Table IV, using 30 scenarios.

The simulation results in Table IV considered bounding gaits with trajectories different than those of the cMPC controller. Hence, the optimized cylinders in Sec.V are likely not optimal for the gaits produced by the cMPC controller. Using the cMPC controller, we aim to showcase that our co-design strategy is not overfitting to any of the optimization scenarios.

TABLE V
MORPHOLOGY CONFIGURATIONS FOR EXPERIMENTS

Parameters	*Configuration*			*Units*
	C1	*C2*	*C3*	
Bore Diameter ϑ	12	12	14.22	[mm]
Stroke Length ℓ_s	50	50	58.4	[mm]
Length ℓ_1	60.73	60.73	68.03	[mm]
Length ℓ_3	88.9	130.9	112.9	[mm]

$\ell_2 = 26.7$ [mm], $\ell_4 = 0.0$ [mm] for the three configurations

B. Results and Analysis

The results herein assess the performance of the configurations in Table V, and they are categorized by the gait applied.

1) Trotting Gait: The trotting experiments considered the three configurations in Table V, and three pressure settings: $P = 3, 4, 5$ $[Bars]$. For control of the pneumatic proportional valve, we applied an open-loop scheme increasing/decreasing the pulse-width modulated -PWM- voltage gradually depending on the gait phase. Figure 11 shows the PWM command sent to the proportional valve depending on the reference signal for the gait phase percent provided by the state estimator.

Regarding Fig. 11, the valve command (upper subplot) is kept constant during the stance phase, aiming to fix the flow for passive compression. In Sec. III-B2, the fixed pressure condition was idealized in having a perfectly sealed cylinder chamber and a leakage-free valve. The middle subplot shows the pressure trajectories for both cylinder chambers. During the swing phase, we open the proportional valve toward the cylinder rod-end chamber and simultaneously activate the two-way solenoid valve to depressurize the system. Due to the tubing length, there exists a flow delay that does not allow for immediate air discharge. The bottom subplot shows the torque modulation achieved. The global and modulated torques remained nearly equal during the swing phase, reducing the motor counteraction. The ability to adjust the cylinder compliance via valve command is one of the main advantages of our PEA actuation scheme. The average torque decrease

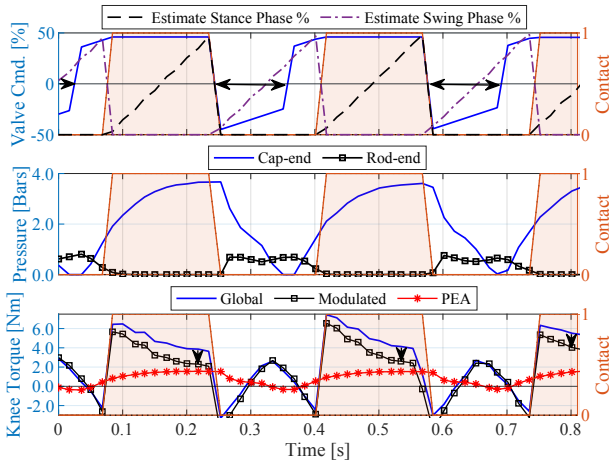

Fig. 11. Trotting experiments (Conf. C3, P = 5 [Bars]): (Top) Valve command with arrows marking the depressurization event. (Middle) Gauge pressure in cylinder chambers. (Bottom) Knee torque of FL leg, with arrows marking the torque reduction. Shaded region corresponds to S1 stance phase.

TABLE VI
RESULTS FROM TROTTING EXPERIMENTS: COT REDUCTION

Conf.	*Baseline CoT*	*CoT [-] | Reduction [%]*			
		P = 3 [bars]	*P = 4 [bars]*	*P = 5 [bars]*	*Average*
C1	3.31	2.96 | 10.6	2.80 | 15.5	2.71 | 17.9	2.82 | 14.7
C2	3.31	2.92 | 11.9	2.81 | 15.0	2.78 | 16.0	2.84 | 14.3
C3	3.31	2.86 | 13.8	2.75 | 16.9	2.73 | 17.4	2.78 | 16.0

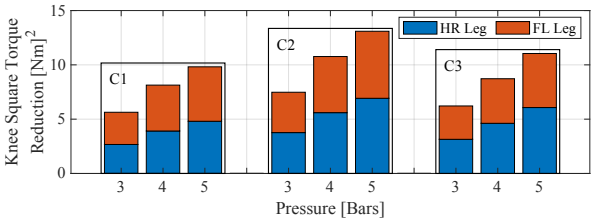

Fig. 12. Trotting experiments: Average knee squared torque reduction for legs with PEA in three morphology configurations and three pressure settings.

with a $P = 5$ [Bars] setting was about 34% (0.86 [Nm]) of the value specified by the cMPC controller. These torque reductions are similar to those reported in [19] of up to 32.8% for the ANYmal with PEAs (AoPS).

To account for energy efficiency, we measured the battery power for the motors of each leg. The results are tabulated in Table VI, showing the CoT reduction with respect to the robot without PEAs (baseline). The computed CoT does not quantify the positive mechanical work alone, but the net energetic losses [40], which should better capture the morphology effect. The optimized C3 configuration produced the overall highest CoT reduction across all setup pressures.

Fig. 12 illustrates the knee squared torque reduction (denoting Joule heating power reduction) for the legs carrying PEAs. The C2 configuration provides the highest squared torque decrease for all the pressure levels, which denotes a potentially misleading output considering the squared joint torques alone as optimization objectives, by neglecting losses due to transmission friction and physical interaction.

2) Bounding Gait: The bounding experiments considered the C3 configuration, only, and two pressure settings: $P = 5, 6$ $[Bars]$. It was challenging to complete the experiments with the C1 and C2 configurations. We observed a higher failure rate due to reaching the PEA kinematic-loop limits and the resulting body trajectories then not being well matched with those optimized by the cMPC controller. The fact that only the optimized design allowed us to complete the bounding experiments supports that our co-design framework does not overfit the design to a specific scenario.

Figure 13 shows the valve PWM command and the pressure trajectories. Opposite to the trotting experiments, the air discharge during the swing phase is faster because of the shorter tubing length having both cylinders at the front legs (See Fig. 1). Likewise, the timing to charge/discharge the cylinder was critical to keep the gait stable, which is trivial for trotting thanks to the continuous support of the opposite pair of legs.

Figure 14 shows that despite the cMPC trajectories not being co-optimized for exploiting passive dynamics, the addition of PEAs is still favorable to the reduction of the knee motor

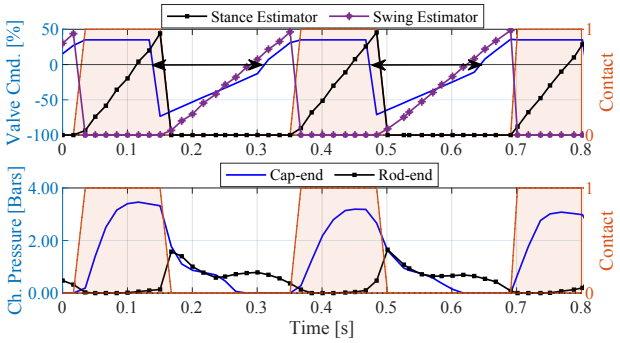

Fig. 13. Bounding experiments (Conf. C3, P = 6 [Bars]): (Top) Valve command with arrows marking the depressurization event. (Bottom) Gauge pressure in cylinder chambers. Orange shaded region corresponds to S1 phase.

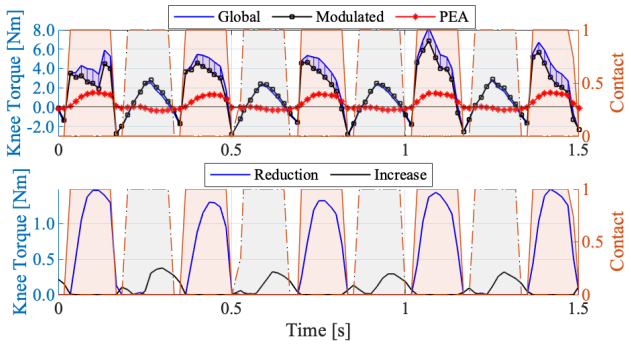

Fig. 14. Bounding experiments (Configuration C3, P = 6 [Bars]): (Top) Total knee torque of the FL leg with PEA. (Bottom) Knee-torque reduction. Shaded orange and gray regions mark the S1 and S2 phases, respectively.

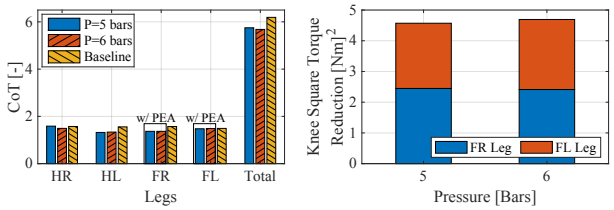

Fig. 15. Bounding experiments: (Left) CoT for each leg and total for the robot without and with PEA. (Right) Average knee squared torque reduction.

torque during stance: 23% on average. During the swing phase, the strategy keeps the torque increase below 5 % (on average).

In energy terms, with the $P = 5, 6$ $[Bars]$ settings, the CoT and % reduction compared to the baseline ($CoT = 6.19$) were respectively: 5.75 and 7.13%, and 5.68 and 8.25%. Fig. 15 extends the details about the CoT and knee torque reduction. The results still show the potential advantage of the added compliance, which can be maximized through more advanced valve control across the swing phase.

C. Limitations

The main limitation of our validation is the lack of sim-to-real transfer of the optimized control policy. The optimized policy helps to emulate closed-loop control conditions during the co-design process, but does not account for the full details of the physical platform (e.g., deviations from the sagittal plane). To further corroborate the experimental results, Fig. 16 shows the CoT simulated with and without PEA for different

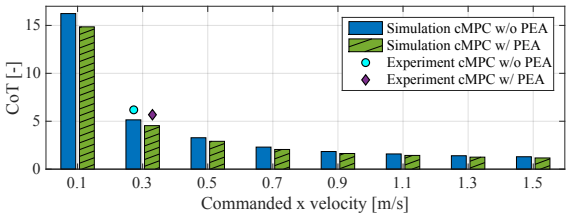

Fig. 16. CoT computed from 3D simulation with cMPC controller: The robot executes bounding gaits at different forward speeds with and without PEAs.

forward speeds and the robot executing nominal bounding gaits in 3D with the cMPC controller. We simulated the ideal PEA action (i.e., ideal stance passive compression and lossless swing motor counteraction) assuming a $P_1^* = 5.7$ $[Bars]$ stance setup pressure (optimal value in Table IV). We concentrate on the results for 0.3 $[m/s]$, the speed commanded for experiments. For the case with PEAs, the CoT in simulation was 20.1% lower than in experiments, while the case without PEAs had a similar discrepancy (16.9%). Comparing the assessed benefits from PEAs, in simulation, the CoT decreased by 11.7% with the addition of the PEA. This simulated CoT reduction deviates from the experimental finding (8.3%) by 3.4%, a small error margin despite the sim-to-real limitation.

VIII. Conclusions

Our main contribution is the ADMM-based co-design framework to engineer compliance in legged robots. Despite the complexity due to the PEA pneumatic models, our framework showed favorable energy reduction and computational convergence regarding the number of scenarios used for co-optimization. The experimental results showed that despite disagreements between the simulation models and the real system, the addition of compliance led to motor energy-use reductions of up to 17.4% with the robot executing trotting gaits. The importance of co-design was demonstrated by the change in performance with different morphology configurations. The experiments showcase the ability of the added compliance to adapt to gaits not optimized for exploiting passive dynamics. This last effect can be attributed as mechanical intelligence, which is better outsourced with the co-designed hardware.

References

[1] R. Pfeifer, F. Iida, and G. Gómez, "Morphological computation for adaptive behavior and cognition," *Int. Congress Series*, vol. 1291, pp. 22 – 29, 2006.

[2] R. Pfeifer and G. Gómez, *Morphological Computation - Connecting Brain, Body, and Environment*. Springer Berlin, 2009, pp. 66–83.

[3] G. Bravo-Palacios and P. M. Wensing, "Large-scale admm-based co-design of legged robots," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Sys.*, 2022, pp. 8842–8849.

[4] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[5] Y. Tassa, T. Wu, J. Movellan, and E. Todorov, "Modeling and identification of pneumatic actuators," in *IEEE Int. Conf. on Mechatronics and Automation*, 2013, pp. 437–443.

[6] C. Mailer, S. Shield, R. Govender, and A. Patel, "Getting air: Modelling and control of a hybrid pneumatic-electric legged robot," in *IEEE Int. Conf. on Robotics and Automation*, 2023, pp. 9973–9979.

[7] Y. Yesilevskiy, W. Xi, and C. D. Remy, "A comparison of series and parallel elasticity in a monoped hopper," in *IEEE Int. Conf. on Robotics and Automation*, May 2015, pp. 1036–1041.

[8] S.-H. Chae, S.-M. Baek, J. Lee, and K.-J. Cho, "Agile and energy-efficient jumping–crawling robot through rapid transition of locomotion and enhanced jumping height adjustment," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 5890–5901, 2022.
[9] M. H. Raibert, *Legged robots that balance*. MIT press, 1986.
[10] R. R. Playter and M. H. Raibert, "Control of a biped somersault in 3D," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Sys.*, vol. 1, 1992, pp. 582–589.
[11] C. Schaff, A. Sedal, and M. R. Walter, "Soft robots learn to crawl: Jointly optimizing design and control with sim-to-real transfer," 2022.
[12] S. Ha, S. Coros, A. Alspach, J. Kim, and K. Yamane, "Computational co-optimization of design parameters and motion trajectories for robotic sys." *The Int. Journal of Robotics Research*, vol. 37, no. 13-14, pp. 1521–1536, 2018.
[13] M. Chadwick, H. Kolvenbach, F. Dubois, H. F. Lau, and M. Hutter, "Vitruvio: An open-source leg design optimization toolbox for walking robots," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6318–6325, 2020.
[14] F. De Vincenti, D. Kang, and S. Coros, "Control-aware design optimization for bio-inspired quadruped robots," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Sys.*, 2021, pp. 1354–1361.
[15] G. Fadini, T. Flayols, A. D. Prete, and P. Souères, "Simulation aided co-design for robust robot optimization," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 306–11 313, 2022.
[16] L. J. Maywald, F. Wiebe, S. Kumar, M. Javadi, and F. Kirchner, "Co-optimization of acrobot design and controller for increased certifiable stability," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Sys.*, 2022, pp. 2636–2641.
[17] A. Sathuluri, A. Vazhapilli Sureshbabu, and M. Zimmermann, "Robust co-design of robots via cascaded optimisation," in *IEEE Int. Conf. on Robotics and Automation*, 2023, pp. 11 280–11 286.
[18] A. Belmonte-Baeza, J. Lee, G. Valsecchi, and M. Hutter, "Meta reinforcement learning for optimal design of legged robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12 134–12 141, 2022.
[19] F. Bjelonic, J. Lee, P. Arm, D. Sako, D. Tateo, J. Peters, and M. Hutter, "Learning-based design and control for quadrupedal robots with parallel-elastic actuators," *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1611–1618, 2023.
[20] C. Semini, N. G. Tsagarakis, B. Vanderborght, Y. Yang, and D. G. Caldwell, "HyQ - hydraulically actuated quadruped robot: Hopping leg prototype," in *2008 2nd IEEE RAS & EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, 2008, pp. 593–599.
[21] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch, R. Diethelm, S. Bachmann, A. Melzer, and M. Hoepflinger, "ANYmal - a highly mobile and dynamic quadrupedal robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Sys.*, 2016, pp. 38–44.
[22] A. M. Abate, "Mechanical design for robot locomotion," Ph.D. dissertation, Oregon State University, 2018.
[23] C. Hubicki, J. Grimes, M. Jones, D. Renjewski, A. Spröwitz, A. Abate, and J. Hurst, "Atrias: Design and validation of a tether-free 3d-capable spring-mass bipedal robot," *The Int. Journal of Robotics Research*, vol. 35, no. 12, pp. 1497–1521, 2016.
[24] P. Arm, R. Zenkl, P. Barton, L. Beglinger, A. Dietsche, L. Ferrazzini, E. Hampp, J. Hinder, C. Huber, D. Schaufelberger, F. Schmitt, B. Sun, B. Stolz, H. Kolvenbach, and M. Hutter, "Spacebok: A dynamic legged robot for space exploration," in *Int. Conf. on Robotics and Automation*, 2019, pp. 6288–6294.
[25] A. Badri-Spröwitz, A. A. Sarvestani, M. Sitti, and M. A. Daley, "Birdbot achieves energy-efficient gait with minimal control using avian-inspired leg clutching," *Science Robotics*, vol. 7, no. 64, p. eabg4055, 2022.
[26] X. Liu, A. Rossi, and I. Poulakakis, "A switchable parallel elastic actuator and its application to leg design for running robots," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 6, pp. 2681–2692, 2018.
[27] B. Katz, J. D. Carlo, and S. Kim, "Mini cheetah: A platform for pushing the limits of dynamic quadruped control," in *Int. Conf. on Robotics and Automation*, 2019, pp. 6295–6301.
[28] L. T. Biegler, *Nonlinear programming: concepts, algorithms, and applications to chemical processes*. Siam, 2010, vol. 10.
[29] G. Bravo-Palacios, G. Grandesso, A. D. Prete, and P. M. Wensing, "Robust co-design: Coupling morphology and feedback design through stochastic programming," *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 2, February 2022.
[30] R. Featherstone and D. Orin, "Chapter 2: Dynamics," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. New York: Springer, 2008.
[31] G. Bravo-Palacios, "Design and simulation of a distortion masking control algorithm for a pneumatic cylinder," Master's thesis, Iowa State University of Science and Technology, 2015.
[32] G. Bravo-Palacios, "Engineering mechanical intelligence in legged robots," Ph.D. dissertation, University of Notre Dame, 2023.
[33] B. O'Donoghue, G. Stathopoulos, and S. Boyd, "A splitting method for optimal control," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 6, pp. 2432–2442, 2013.
[34] Z. Zhou and Y. Zhao, "Accelerated ADMM based trajectory optimization for legged locomotion with coupled rigid body dynamics," in *American Control Conference*, 2020, pp. 5082–5089.
[35] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 1999.
[36] J. Andersson, J. Åkesson, and M. Diehl, "Casadi: A symbolic package for automatic differentiation and optimal control," in *Recent advances in algorithmic differentiation*. Springer, 2012, pp. 297–307.
[37] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.
[38] HSL, "A collection of Fortran codes for large scale scientific computation." http://www.hsl.rl.ac.uk/, 2015.
[39] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, "Dynamic locomotion in the MIT cheetah 3 through convex model-predictive control," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Sys.*, 2018, pp. 1–9.
[40] H.-W. Park, P. M. Wensing, and S. Kim, "High-speed bounding with the MIT cheetah 2: Control design and experiments," *The Int. Journal of Robotics Research*, vol. 36, no. 2, pp. 167–192, 2017.

Gabriel Bravo-Palacios (Member, IEEE/ASME) received the B.S. degree (2012) from the National Polytechnic School, Quito, Ecuador, the M.S. degree (2015) from Iowa State University, Ames, IA, USA, and the Ph.D. degree (2023) from the University of Notre Dame, Notre Dame, IN, USA; all degrees in mechanical engineering. His research interests include the application of design optimization and control theory to produce high-performance and energetically efficient mechatronic systems.

He Li (Member, IEEE) is currently a Ph.D. candidate in the Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN, USA. He received the B.S. degree (2015) from Shandong University of Technology, China, and the M.S. degree (2019) from Missouri University of Science and Technology, Rolla, MO, USA. He is interested in achieving highly dynamic and complex maneuvers on legged robots, leveraging control theory, numerical optimization, and machine learning.

Patrick M. Wensing (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from The Ohio State University, Columbus, OH, USA, in 2009, 2013, and 2014, respectively.

He is currently the Wanzek Family Associate Professor in the Department of Aerospace and Mechanical Engineering, University of Notre Dame, where he directs the Robotics, Optimization, and Assistive Mobility (ROAM) Laboratory. Before joining Notre Dame, he was a Postdoctoral Associate with MIT, working on control system design for the MIT cheetah robots. His current research interests include aspects of dynamics, optimization, and control toward advancing the mobility of legged robots and assistive devices. He received the NSF CAREER Award in 2020. He currently serves as a Co-Chair for the IEEE RAS Technical Committee on Model-Based Optimization for Robotics and as a Senior Editor for the IEEE Transactions on Robotics.