Optimization-Based Control for Dynamic Legged Robots

Patrick M. Wensing, *Senior Member, IEEE*, Michael Posa, *Member, IEEE*, Yue Hu, *Member, IEEE*, Adrien Escande, *Member, IEEE*, Nicolas Mansard, and Andrea Del Prete, *Member, IEEE*

***Abstract*—In a world designed for legs, quadrupeds, bipeds, and humanoids have the opportunity to impact emerging robotics applications from logistics, to agriculture, to home assistance. The goal of this survey is to cover the recent progress toward these applications that have been driven by model-based optimization for the real-time generation and control of movement. The majority of the research community has converged on the idea of generating locomotion control laws by solving an optimal control problem (OCP) in either a model-based or data-driven manner. However, solving the most general of these problems online remains intractable due to complexities from intermittent unidirectional contacts with the environment, and from the many degrees of freedom of legged robots. This survey covers methods that have been pursued to make these OCPs computationally tractable, with a specific focus on how environmental contacts are treated, how the model can be simplified, and how these choices affect the numerical solution methods employed. The survey focuses on model-based optimization while paving its way for broader combination with learning-based formulations to accelerate progress in this growing field.**

***Index Terms*—Contact modeling, legged locomotion, motion control, optimal control, whole-body control.**

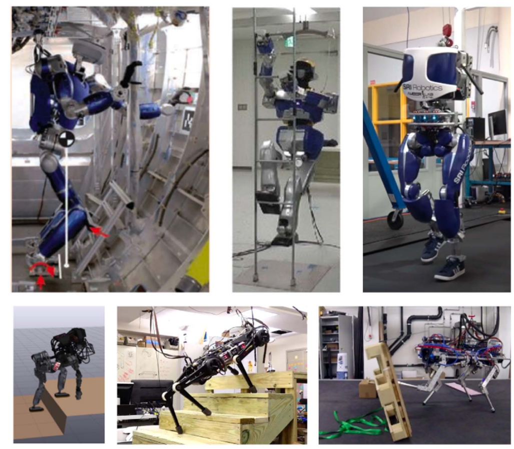


Fig. 1. Optimization-based control has been a common enabler for legged systems to handle complex environments [3], [10], [34], [58], [59], [60].

I. Introduction

Over the past decade, we have witnessed rapid growth in the capabilities of legged robots, transitioning from a state of the art where only a few research groups had access to capable platforms, to one where robust locomotion is now common in industry and academic laboratories. This rapid progress has been enabled by a combination of advances across design and control, with optimization-based control strategies playing a central role in many of the milestone demonstrations during this period (Fig. 1). Through these advances, current bipeds, quadrupeds, and humanoids can now walk reliably in nominal environments, and these improved capabilities have led to the first practical deployments of legged systems (e.g., the robot spot from Boston Dynamics). In a world built for legs, the growing scope of these deployments offers a broad opportunity for impact on applications spanning logistics (e.g., delivery), agriculture, and home assistance, among many others.

Across these applications, optimization-based control strategies offer many advantages as a pathway to (or component of) capable autonomy. Two prevailing types of optimization-based control have been pursued: predictive and reactive. Predictive controllers consider an explicit system model to "reason" (in a strictly mathematical sense) about the consequences of their actions, iteratively devising and improving motion plans in response to the situation at hand. Reactive controllers, by contrast, only consider their actions for the current instant. The execution of these controllers may depend on the system model or may be model-free (e.g., the result of offline policy optimization). The benefits of the model-based approach are that it allows one to naturally consider safety constraints, which will be critical

Manuscript received 18 November 2022; revised 12 May 2023; accepted 4 September 2023. Date of publication 16 October 2023; date of current version 15 December 2023. This paper was recommended for publication by Associate Editor J. Carpentier and Editor P. Robuffo Giordano upon evaluation of the reviewers' comments. This work was supported in part by the NSF under Grant CMMI-2220924 (Wensing), in part by the NSF under Grant CMMI-1830218 (Posa), in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN-2022-03857 (Hu), in part by the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie under Grant 101066915 (Escande), and in part by the PRIN Project DOCEAT under Grant CUP E63C22000410001 (Del Prete). *(Corresponding author: Patrick M. Wensing.)*

Patrick M. Wensing is with the University of Notre Dame, Notre Dame, IN 46556-5637 USA (e-mail: pwensing@nd.edu).

Michael Posa is with the University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: posa@seas.upenn.edu).

Yue Hu is with the University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: yue.hu@uwaterloo.ca).

Adrien Escande is with the Inria centre at the University Grenoble Alpes, 38334 Saint Ismier, France (e-mail: adrien.escande@inria.fr).

Nicolas Mansard is with the LAAS-CNRS, 31400 Toulouse, France (e-mail: nmansard@laas.fr).

Andrea Del Prete is with the University of Trento, 38122 Trento, Italy (e-mail: andrea.delprete@unitn.it).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TRO.2023.3324580.

Digital Object Identifier 10.1109/TRO.2023.3324580

for many of the envisioned applications for legged robots. An additional motivation is that first-principles models generalize well to unforeseen situations, which will be an asset for flexible autonomous deployments. There is also a growing interest in extending model-based methods by augmenting or complementing them with learning-based approaches, e.g., learning hard-to-model aspects of physics, accounting for perception, or addressing higher level aspects of decision making that are not amendable to physics-based reasoning. With this big picture in mind, a central goal of the review is to synthesize the past decade of model-based advances so that they can be further improved in the future and also incorporated with learning-based strategies—ultimately accelerating the practical deployment of legged robots.

A. *Historical Context for the Survey*

Much of the recent progress in optimization-based control for robots rests on decades of progress from the mathematical programming community and its early translation to our field. The idea of using optimization for the specification and control of robot movement traces back to decades before its current popularity (e.g., [61]). The 1990s saw the use of quadratic programming for torque control of redundant manipulators [62], with the 2000s seeing generalizations of these ideas for the control of humanoids in simulation [63], [64], [65]. The advent of polynomial-time solutions to important classes of convex optimization problems in the mid-1990s [66] and subsequent commercial solvers (e.g., [67]) has contributed to the rise in maturity and accessibility of optimization solutions for control. Building on these advances, work in the late 2000s and in the wave of research before the 2015 DARPA Robotics Challenge (DRC) focused on convex optimization for maintaining balance via reactive control (e.g., [43], [68], and [69]). The role of optimization in these accomplishments has since led to recent work on pushing the boundaries of predictive control for quadrupeds and humanoids executing broader dynamic locomotion in more challenging environments.

B. *Survey Goals*

The overarching goal of this survey is to provide a synthesized entry point into this recent body of work. As such, our intention is for the survey to serve as the most effective resource for early-stage graduate students, while also providing new perspectives and targeted follow-on reading for established experts. Work since the DRC has led to many new methods for optimization through contacts, for real-time optimization using new simplified models, and through advances to numerical methods for handling ever more complicated robot models. These advances are the primary focus of the review.

Beyond covering this past work, a secondary motivation for this survey is to provide the background for the next steps in legged locomotion research, enabling these systems to move beyond nominal environments and unlock mobility in unstructured terrains. A great deal of current research interest focuses on how to combine machine learning strategies with previously developed model-based ones. In this sense, the survey will provide valuable background for machine learning practitioners to come up to speed on existing model-based approaches. Other research continues to push forward numerical methods and algorithms for legged robots to make informed model-based decisions. The survey should serve as a valuable overview of recent work for these groups as well.

II. Problem Statement and Overview

During the last decade, most of the research community working on legged robots (humanoids and quadrupeds in particular) has converged on the idea of generating motion based on the formulation of an optimal control problem (OCP). In a high-level form, such an OCP can be written as follows:

$$\underset{x(\cdot),u(\cdot),\lambda(\cdot)}{\text{minimize}} \quad \text{Cost}(x(\cdot), u(\cdot), \lambda(\cdot)) \tag{1a}$$

$$\text{subject to} \quad M(q)\,\dot{\nu} + C(q,\nu)\,\nu + \tau_g(q) = S^T\tau + J_c(q)^T\lambda \tag{1b}$$

$$\text{ContactConstraints}(x(t), \lambda(t), u(t), \text{Env}) \tag{1c}$$

$$\text{KinematicsConstraints}(x(t)) \tag{1d}$$

$$\text{InputConstraints}(u(t), x(t)) \tag{1e}$$

$$\text{TaskConstraints}(x(t), u(t), \lambda(t)) \qquad \forall t\,. \tag{1f}$$

The (infinite-dimensional) decision variables of the OCP are the *trajectories* of state $x = (q, \nu)$, control $u \triangleq \tau$, and contact forces λ exchanged between the robot and the environment Env. The robot configuration is represented as q, whereas its velocity as ν. Equation (1a) is a user-defined cost function representing a metric to minimize, such as energy consumption or distance to a desired target. Equation (1b) represents the nonlinear whole-body robot dynamics [70], where $M(q)$ is the mass matrix, $C(q,\nu)\nu$ accounts for Coriolis and centrifugal forces, $\tau_g(g)$ contains the gravity forces, S is typically a matrix that selects the actuated degrees of freedom (DoFs), and $J_c(q)$ is the contact Jacobian. Equation (1c) encodes the contact-related constraints, such as nonpenetration of rigid objects and friction-cone force constraints. Equation (1d) includes state constraints resulting from the kinematics, such as joint position and velocity limits. Equation (1e) typically includes the motor torque limits, but it could in general represent any input constraint. Finally, (1f) can be used to include any task-specific constraint in the OCP, such as fixed initial/final state, or field-of-view limits for a camera.

Clearly, this OCP is very general and therefore allows one to encode a wide variety of movements. For instance, point-to-point locomotion can be generated by specifying the initial and final robot states [29]. An object manipulation behavior could instead be computed by including the state of the object in x and specifying its desired value with an appropriate cost function [32], [71], [72]. While appreciating its versatility, we should acknowledge that some intrinsic limitations do exist in the OCP approach. For instance, it is hard to ensure stability [29] or robustness for the computed trajectories [39]. However, this review focuses on another limitation and how to address it: the computational complexity of the problem. Indeed, problem (1) hides several challenges, the main ones being: the nonsmoothness/stiffness of the dynamics, nonconvexity, and dimensionality. The rest of this section acts as an executive summary of the rest of this article by briefly discussing different approaches to tackle these challenges. Each aspect is analyzed more thoroughly in Sections III–VI.

A. Contact Models

The first issue (nonsmoothness/stiffness), which is discussed in detail in Section III, arises from physical contacts modeled in (1c). Contacts can be modeled as either *rigid* or *visco–elastic*. With appropriate choices, visco–elastic models ensure continuous dynamics, and smoothing techniques can be used to make them differentiable. The downside of visco–elastic models is that large stiffness values are necessary to generate realistic behaviors. This feature leads to stiff differential equations with corresponding numerical challenges for simulation and optimization.

Alternatively, contacts can be modeled as rigid (i.e., no penetration is allowed). When two points make contact, their relative velocity must immediately become zero to avoid penetration. Therefore, the robot dynamics must be described by a mix of continuous-time and discrete-time equations, i.e., as a hybrid dynamic system. The resulting OCP can then be tackled either as a linear complementarity program (LCP) [10], [13], [73] or as a mixed integer program (MIP) [21], both of which require customized optimization techniques that are typically much less efficient than classic smooth optimization strategies.

A common way around this nonsmoothness is to let the user fix the order in which contacts are made and broken. This makes the dynamics time-switched (a special case of hybrid dynamics) and the OCP differentiable, therefore efficient smooth optimization can be used. The obvious downside is that it may be hard to guess the contact phases.

B. Dynamic Models

Besides the contact dynamics, the other major source of complexity is the robot multibody dynamics (1b), which are high-dimensional and nonlinear. In turn, this makes the resulting optimization problem high-dimensional and nonconvex. The high dimension is especially concerning in the context of online optimization, where fast computation times are mandatory. The nonconvexity instead is always concerning, as it makes the solver sensitive to the initial guess that is used. As a partial or total remedy to these issues, several *simplified models* have been proposed in the literature (also known as reduced-order models, or template models), and they are discussed in Section IV. These models should capture the most important part of the robot dynamics with a reduced state size. For example, for locomotion, the widely used linear inverted pendulum (LIP) model [47] considers only the contact locations and the center of mass (CoM) of the robot, neglecting the details of joint angles and velocities. While simplified models are key enablers for fast online computation, their simplifying assumptions (e.g., constant CoM height), or neglected constraints [74] (e.g., joint

TABLE I
OVERVIEW OF THE STATE OF THE ART

Papers	Contact model	Gait	Dynamics	Transcription
[1]–[5]	Soft/Smoothed	Optimized	Full	DDP
[6]	Soft/Smoothed	Optimized	Full	Collocation
[7]	Soft/Smoothed	Optimized	Simplified	Collocation
[8], [9]	Rigid	Optimized	Full	DDP
[10]–[15]	Rigid	Optimized	Full	Collocation
[16]–[24]	Rigid	Optimized	Simplified	Collocation
[25]–[28]	Rigid	Fixed	Full	DDP
[29]–[31]	Rigid	Fixed	Full	Multiple Shooting
[32]–[34]	Rigid	Fixed	Full	Collocation
[26], [35]	Rigid	Fixed	Simplified	Multiple Shooting
[36]–[38]	Rigid	Fixed	Simplified	DDP
[39]–[46]	Rigid	Fixed	Simplified	Collocation
[47]–[57]	Rigid	Fixed	Simplified	Single Shooting

position and torque bounds) can severely limit the generated motions.

C. Optimal Control Solution Methods

After the OCP is mathematically formulated, there are a range of design choices that remain for solving it. General optimal control methods can be categorized as direct methods, indirect methods, or those based on dynamic programming. Within robotics, direct optimization methods prevail almost universally. However, even after choosing to use a direct method, the optimization formulation needs to be approximated as a finite-size nonlinear program (NLP) to be solved by a numerical solver. This approximation process is called *transcription* and it crucially affects the results in terms of accuracy, numerical stability, and computational complexity.

Two families of transcription methods have been used in robotics (shooting and collocation) and we discuss them in Section V. A shooting method known as differential dynamic programming (DDP) has been a topic of recent interest in the community due to its favorable computational properties and provision of a locally optimal feedback policy. More generally, but specific to direct optimization for locomotion, we examine how contact modeling choices affect the choice of transcription tools and algorithms available, which becomes most critical when optimization is left to freely choose contact sequences.

Table I serves to categorize the choices that state-of-the-art papers have used in adopting different approaches for modeling the contacts and their sequencing, for simplifying the dynamics, and, ultimately, for numerically solving variants of (1). In practice, the highlighted approaches play a key role in breaking problem (1) into many smaller subproblems that focus on building an overall solution in a hierarchical fashion (e.g., solving for footsteps first, then optimizing motions with fixed footsteps).

D. Trajectory Stabilization Via Whole-Body Control

Depending on the complexity of the formulated OCP, the computation time may be too large for it to be solved inside a fast control loop for model predictive control (MPC). In these cases, a reactive stabilizing controller is required to execute the computed motion on real hardware, or to provide control of aspects of the system that were ignored during trajectory optimization (TO) (e.g., due to modeling simplifications). In the last decade, the legged robotics community has converged on a certain class of reactive optimization-based whole-body control techniques, which mainly rely on the fast solution of small convex quadratic programs (QPs) to compute motor commands as a function of state feedback.

These QP techniques have represented an evolution of previous operational-space control paradigms for the control of manipulators. However, for legged systems, new problems related to contact constraints, impacts, self-collisions, etc., have motivated a broader perspective on this classical problem. The most general extensions have focused on treating constraints via task-space inequalities. In spite of many extensions, the formulation of reactive control as an instantaneous control strategy has enabled the method to retain a convex formulation that is lost in more general transcription strategies. The convex-optimization perspective on reactive control has also enabled other paradigms [43], [75], [76], [77], conventionally separate from operational-space control, to be incorporated as well. We describe these techniques in Section VI, while also reflecting on recent advances in whole-body MPC.

III. Contact

Computationally tractable treatment of physical and frictional contact with the environment, which leads to stiff and/or discontinuous equations of motion, is a fundamental challenge in optimization-based control for robotics and a primary distinction between (1) and standard OCPs. As a result, design decisions for 1) how to model the effects of contact, and 2) the sequencing or scheduling of contact events play a significant role in distinguishing different approaches to solving (1). Specifically, the details in this section center on how the choice of contact model affects parameterizations of the force λ and the form of the contact constraints in (1c).

This section is comprised of two parts. In Section III-A, we outline common techniques and associated numerical challenges for numerical models of contact dynamics. In Section III-B, we describe how the need to sequence contact events affects the algorithmic and modeling choices.

A. Modeling Contact

Highly accurate mechanics models define contact as *visco–elastic*, representing local deformations of surfaces and the corresponding stress–strain relationships. Some approaches in robotics, particularly in soft robotics (e.g., [78]), do attempt to accurately represent deformation, which prevents interpenetration between bodies. Rigid-body approximations, by contrast, do not consider deformation but do permit some penetration between contacting bodies. The resulting contact forces are modeled as a function of penetration depth, relative normal and tangential velocities, and material properties. Rigid-body models are most commonly used and will be the focus of this review. Noting critical definitions here, we write $\phi(q)$ to be the vector-valued signed distance function, for all possible contacts. $J_n(q)$ and $J_t(q)$ are the normal and tangential Jacobians, such that $J_n(q)\nu$ and $J_t(q)\nu$ are the contact frame velocities. Normal and tangential forces are similarly decomposed into λ_n and λ_t, and, where convenient, we use $J_c(q)$ and λ to represent stacked contact Jacobians and forces. The contact force, therefore, can be expressed as

$$\lambda = F_{\text{contact}}(\phi, J_c\,\nu).$$

Common methods treat this function as a nonlinear spring and damper (e.g., [79] and [80]), where unilateral constraints determine when the spring releases and typically, contact forces cannot "pull," constraining $\lambda_n \geq 0$, and so separation occurs if $\lambda_n < 0$. However, to accurately represent the interaction between rigid bodies, F_{contact} is numerically stiff [81], inheriting the mechanical stiffness of the material properties in the robot. When making and breaking contact ($\phi = 0$), F_{contact} may also be nondifferentiable. Frictional effects exhibit similar issues: in dry Coulomb friction, λ_t depends on the sign of the tangential velocity, which is discontinuous.

Simply introducing these stiff or discontinuous dynamics to (1) leads to a poorly conditioned optimization problem and is generally avoided. Algorithmic approaches, therefore, tend to focus on one of two choices: 1) *infinitely* stiff representations, as hybrid models, or 2) smoothing or softening F_{contact} to improve numerical performance. An illustrative example of a single contact is shown in Fig. 2, demonstrating the resulting

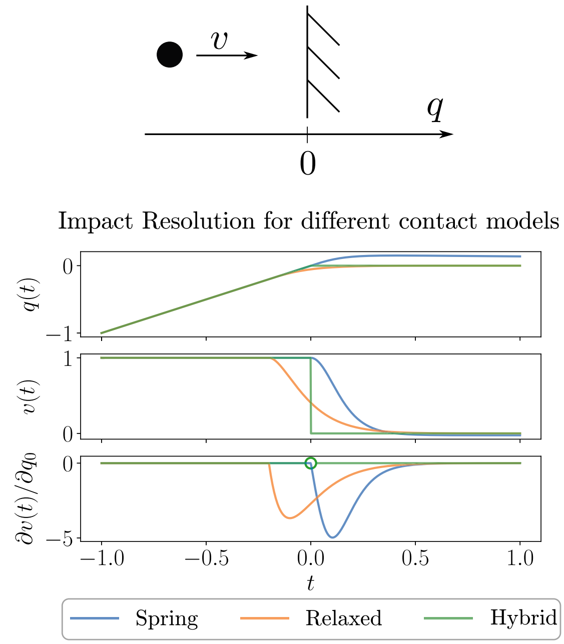


Fig. 2. In an illustrative conceptual example, a particle heads toward a wall, with inelastic impact when $q = 0$. Three common contact models are compared, showing (top-to-bottom) position, velocity, and sensitivity of velocity to initial conditions. (Blue) A Hunt–Crossley spring model [82] activates when contact occurs at $q(t) = 0$, allowing some penetration. The sensitivity $\partial v(t)/\partial q_0$ is well-defined, although not smooth (C^∞) at the impact. (Orange) A relaxed (softened) complementarity model, such as MuJoCo, is smooth, with contact forces occurring prior to contact [83]. (Green) A hybrid model with inelastic impact captures rigid behavior with an instantaneous jump in velocity, noting that $\partial v(t)/\partial q_0$ is undefined at impact. All models have well-defined gradients *away* from the impact event, enabling effective differentiation in these regimes. For the hybrid and spring model, these gradients are zero prior to contact, and for the relaxed model, as it is smooth, these gradients are nonzero everywhere.

state trajectory and, critical for optimization-based control, the sensitivity of the solution with respect to initial conditions.

1) Hybrid Dynamics: In the limit of infinite stiffness, when contact is initiated, the resulting forces become *impulsive*, impacts, therefore, cause an instantaneous jump in velocity. While a full description of hybrid systems is outside the scope of this review (see [84] and [85] for an overview), we briefly note the principles here. Hybrid systems are defined by *modes*, *guards*, and *resets*. In this context, the mode is the contact state: identifying which objects are touching, and whether they are sticking or sliding. Guards determine when mode transitions occur, for instance, making or breaking contact, and the reset map R specifies the result of the transition (e.g., impact) $x^+ = R(x^-)$. Within a given mode, the dynamics are well defined and differentiable, and thus, this formulation compresses the complexity of contact with the guards and transition events. Referring to the root OCP (1), a hybrid formulation typically explicitly splits the decision variables $x(\cdot), u(\cdot)$, and $\lambda(\cdot)$ by mode, introducing boundary constraints at the resets when transcribing the dynamics in Section V.

2) Complementarity Models: A mathematically equivalent formulation of hybrid contact dynamics relies upon *complementarity constraints* [86], [87], where a relationship between λ and x implicitly defines the hybrid dynamics. For example, a nonpenetration condition $\phi(q) \geq 0$, combined with $\lambda_n \geq 0$, and the property that forces can only be nonzero when in contact leads to the complementarity constraint

$$0 \leq \lambda_n \perp \phi(q) \geq 0$$

where $\lambda_n \perp \phi(q)$ means that the vectors λ_n and $\phi(q)$ must be orthogonal, which is equivalent to $\lambda_n^T \phi(q) = 0$. This formulation can be expressed in time-stepping [73] or continuous [88] models, with a similar set of complementarity constraints used to capture Coulomb friction.

3) Pathologies: Hybrid formulations of multicontact robotics, while effective for simulation and control, can exhibit certain pathologies where solutions do not exist or where infinite solutions are possible [86], [89]. These challenges are the limiting case of the high sensitivity to initial conditions seen in the stiff differential equations. From the perspective of optimization-based control, stiff dynamics lead directly to poorly conditioned optimization problems. Alternatively, for instance, when using hybrid or complementarity models, planning algorithms that assume uniqueness, therefore implicitly (perhaps even unbeknownst to the algorithm designer) select from the set of possible solutions (a property, for instance, of [10]). In many scenarios, this selection is benign and in others, it indicates that the resulting trajectory is practically impossible to track, due to the nonuniqueness. Nonuniqueness can occur in many scenarios, and most commonly, if $J_c(q)$ is rank deficient. Think, for example, of static indeterminacy with a four-legged table, where the normal force distribution is indeterminate.

4) Hybrid Differentiability: In the hybrid formulation, while both the resulting trajectories $x(t)$ and the equations of motion may be discontinuous, under certain circumstances it is possible to generate well-conditioned derivatives of $x(t)$ with respect to initial conditions and control actions [90], [91] (see Fig. 2 for an example). Typically, these derivatives exist if the sequence of modes is constant, and the event times themselves might change, but the order does not and all transitions are inevitable. When deviations might change the mode sequence, resulting trajectories are often *not* differentiable with respect to initial conditions or parameterizations, but may admit more general sensitivity characterizations [90]. Practically, if a trajectory $x(t)$ does not make contact, then no amount of local differentiation provides insight into the effects of initiating contact, and put more explicitly, solutions $x(t)$ are nonanalytic in their initial conditions and control inputs. This property poses natural challenges for optimizers to discover new contacts through local information alone, and this motivates many relaxations or other tailored strategies for scheduling contacts.

To mitigate this issue, certain simulators ensure global differentiability or even smoothness (termed *differentiable simulation*, e.g., MuJoCo [83] or TDS [92]). While these methods do consistently generate local gradients, this inevitably ties the accuracy (stiffness) of the underlying model to stiffness in the resulting optimization problem. The precise nature of this tradeoff and the relevance of the different modeling inaccuracies (see, e.g., [93] and [94]) remain unknown.

B. Scheduling Contact

As the discussion of differentiability above implies, there are clear distinctions in computational tractability between versions of (1) with a fixed, known mode sequence and variations where the optimization problem must also determine the ordering. In many robotics problems, this sequence may be clear, for example, bipedal walking over flat terrain typically follows a "left foot and right foot" ordering with minimal deviation. In others, for instance, movements using whole-body contact or locomotion over varied surfaces with multiple potential foothold locations, the challenge of finding an optimal ordering may dominate the control problem.

1) Known Modes Sequences: In this setting, once discretized (see Section V), the hybrid OCP is differentiable with respect to the transcribed decision variables. In some settings, particularly when deploying simplified models or when seeking stability proofs, it is common to use *minimal coordinates* to represent the configuration of a robot in contact (e.g., [95] and [96]). Contact points are transformed into pin joints, removing DoFs and simplifying the resulting OCP. When contact modes change, however, the hybrid jump must also capture the change in the dimension of the state space. The use of minimal coordinates is computationally efficient, although it is difficult to impose constraints (e.g., friction) on λ, as inverse dynamics are necessary to reconstruct the constraint forces. Furthermore, in multicontact settings, where the contacts generate a closed chain, global definitions of minimal coordinates may not exist.

The more general formulation utilizes the same excess, or *floating-base coordinates* for all contact modes. The constrained dynamics then enforce that active contacts remain touching [29], [97]. Some approaches directly compute the force, e.g., $\lambda(x, u)$ [97]. This approach has the advantage of relative simplicity, although heuristics are necessary when the rigid-body force is nonunique. Alternate methods include $\lambda(t)$ as a decision variable [see (1)], along with corresponding constraints to ensure physical accuracy. Imposing constraints *differentially* in this fashion comes at the risk of constraint drift due to integration error, typically addressed via implicit integration schemes [33], Baumgarte stabilization [27], or a mixture of approaches [98].

2) Hybrid Sequence Optimization: As an immediate extension of the known-sequence scenario, one could jointly optimize over the discrete states of the hybrid sequence and the corresponding robot motion. This can naturally be expressed via *mixed-integer optimization* [17], [21], or as a bilevel optimization problem [16]. MIPs have been used to compute contact sequences accounting for obstacle avoidance and step-to-step reachability [99] and approximated as computationally efficient L1-norm minimization [100]. Other strategies ensure "quasi-static" feasibility by limiting the search to "quasi-flat" contact surfaces (i.e., surfaces where the friction cone contains the gravity direction) [100]. In the worst case, these methods must explore every possible mode sequence, and so are most effective when there are relatively few potential sequences or when effective heuristics are available to guide the search.

Sampling-based techniques are an alternative to MIPs. For instance, rapidly exploring random trees have been used to plan footsteps on flat ground, avoiding obstacles [101]. Similarly, probabilistic road maps have been used to plan a collision-free path for the robot's base, keeping it close enough to the environment to allow for contact creation [102].

3) Contact-Implicit Planning: Contact-implicit (alternatively, contact-invariant) methods are variations on hybrid optimization that embed the relationship between state and force into an NLP, implicitly representing the hybrid mode without discrete variables. Approaches here are most often based on either the complementarity formulation [10], [103] or on smooth approximations of the contact dynamics [3], [7]. Prior to convergence, these methods typically violate strict complementarity, thus simultaneously exploring multiple contact sequences. Contact–implicit optimization, however, can suffer from poor numerical conditioning and can require high-quality initial guesses. Contact-implicit methods have found applications in many robotic domains, including MPC [1] and gait optimization for microrobots [14]. Recent work has focused on improving numerical performance, for example by using simplified models [18], [20] or improving the accuracy of the numerical integration schemes [12], [13]. While these methods are typically limited to offline computation, progress has been made in real-time scheduling of contact [15], [23]

These previous contact-implicit approaches explicitly account for the hybrid nature of planning through contact. Alternatively, gradient-based methods, such as the iterative linear quadratic regulator (iLQR) or DDP (see Section V-C), can be applied. This is most common when coupled with a differentiable contact model, enabling the optimization method to "discover" new modes (when it happens to bump into them, perhaps guided by the smoothing in the differentiable model) [1], [5]. Other methods maintain a rigid model, and consider gradient analyses around the contact sequence found during forward simulation (e.g., [8] and [9]), leveraging the almost everywhere differentiability discussed above. Broadly speaking, these methods are typically highly sensitive to their initializations and struggle to discover contact sequences that vary dramatically from that of the initial guess.

C. Summary

When the sequence of environmental contacts can be known a priori, then the choice of contact model should focus purely on physical realism, as nearly any transcription method can be applied without disruption. It is substantially more difficult, however, to simultaneously plan motion and contact schedule. While significant progress has been made over the last decade, contact planning remains the main challenge for generating arbitrary locomotion behaviors in complex environments. Ultimately, this challenge arises from our reliance on gradient-based optimization, which despite being the workhorse behind the progress discussed in this survey, is fundamentally unsuited for nonsmooth contact-implicit problems.

1) Relationship to Learning: While this review focuses on model-based optimization, there is an implicit requirement that such a model must first be identified or learned. Methods for learning and identification of contact align closely with the modeling choices detailed in this section. When contact modes are known or can be easily identified for the training data, hybrid approaches are commonly used [93]. Otherwise, machine learning can be directly applied to a smoothed or differentiable contact model (e.g., [104]). Alternatively, contact-implicit methods have led to data-efficient mechanisms for contact model learning [105]. Aside from identifying *models*, machine learning has also shown promise in identifying (or providing high-quality guesses for) potential mode sequences for control [106], [107], where an effective presolve for the mode schedule substantially reduces the computational difficulty of the subsequent TO. We note, as well, that *model-free* reinforcement learning (RL), which has been widely applied to similar problems in control (e.g., legged locomotion [108]), also commonly leverages smoothed (or stochastic [109]) contact models.

IV. Simplified Models

A second source of complexity for the efficient solution of (1) arises from the whole-body dynamics model (1b). As noted earlier, the dynamics of legged systems are high dimensional and nonlinear, and these features correspondingly make the optimization problem high dimensional and nonconvex. These challenges motivate simplified models in place of (1b) that capture its most salient features in a reduced set of differential equations. A central question is then how to select such a simplified model. We use the evolution in Fig. 3 to walk through the most common modeling simplifications adopted.

The role that contacts play within the whole-body dynamics (1b) has motivated the majority of simplified models employed to date. For a floating-base system, it is common to partition the generalized velocity as $\nu = (\nu_b, \nu_j)$ where $\nu_b = (\omega_b, v_b) \in \mathbb{R}^6$ gives the angular and linear velocity of the floating base and ν_j

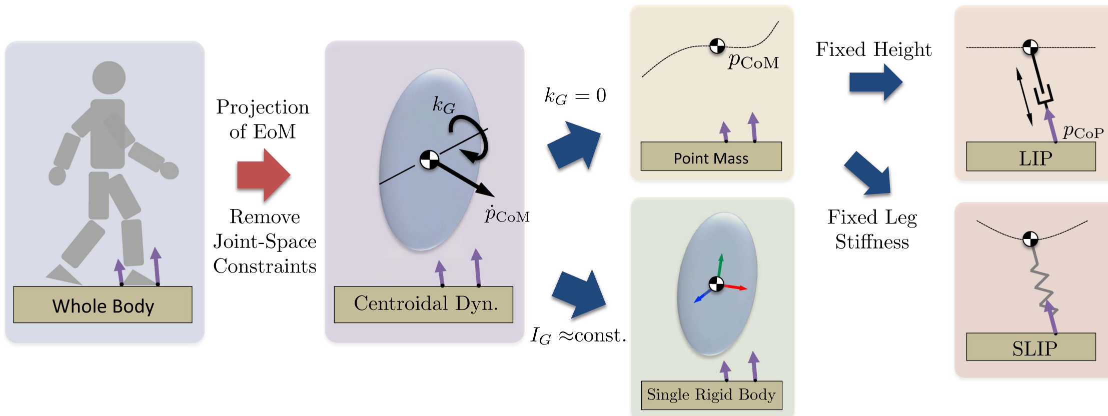


Fig. 3. The centroidal dynamics model is a common modeling simplification that focuses on the CoM and net momentum. It is a special modeling simplification in that it exactly projects the whole-body dynamics, while relaxing joint-space constraints. Restrictions placed on this model then lead to a tree of other models.

the generalized velocity of the joints. With this partitioning in ν, the dynamics are partitioned as

$$\begin{bmatrix} M_{bb} & M_{bj} \\ M_{jb} & M_{jj} \end{bmatrix} \begin{bmatrix} \dot{\nu}_b \\ \dot{\nu}_j \end{bmatrix} + C(q,\nu)\nu + \tau_g(q) = \begin{bmatrix} 0 \\ \tau \end{bmatrix} + J_c(q)^T \lambda \,. \tag{2}$$

Suppose now that the robot is given a desired motion $(q_d(t), \nu_d(t))$. Considering (2), the left-hand side of the equation is fixed along the desired motion, and so the motion is possible only when the equation can be satisfied through proper choice of joint torques τ and contact forces λ. If the robot has sufficiently strong actuators, then τ can be chosen to satisfy the bottom set of equations. However, since joint torques do not affect the first six rows, the main dynamic limitations on movement are determined by the contact force constraints (i.e., friction and unilaterality).

A. Centroidal Dynamics—Modeling and Optimization

It can be shown that the first six rows of (2) describe the evolution of the net linear and angular momentum of the system as a whole [110], [111]. It is common to consider the angular momentum about the CoM denoted $k_G \in \mathbb{R}^3$, as this quantity is both conserved during free flight and empirically remains close to zero during human walking. Denoting the total linear momentum as $l_G \in \mathbb{R}^3$, these momenta compose the *centroidal momentum* $h_G = (k_G, l_G) \in \mathbb{R}^6$ [112]. This quantity is related to the generalized velocities via $h_G = A_G(q)\,\nu$, where $A_G(q)$ is called the centroidal momentum matrix [112]. Consider a case with n_c contact points at locations $\{p_i\}_{i=1}^{n_c}$. It can be shown that the top rows of (2) are equivalent to

$$\dot{h}_G = \begin{bmatrix} \dot{k}_G \\ \dot{l}_G \end{bmatrix} = \begin{bmatrix} 0 \\ M a_g \end{bmatrix} + \sum_{i=1}^{n_c} \begin{bmatrix} (p_i - p_{\text{CoM}}) \times \lambda_i \\ \lambda_i \end{bmatrix} \tag{3}$$

where $p_{\text{CoM}} = [c_x, c_y, c_z]^T$ gives the CoM position, M is the total mass, and a_g is the gravitational acceleration [110], [111]. It is important to note that $M \frac{\mathrm{d}}{\mathrm{d}t} p_{\text{CoM}} = l_G$. These *centroidal momentum dynamics* (3) (also called the *centroidal dynamics*) motivate solving a reduced problem posed over trajectories for the CoM p_{CoM}, angular momentum k_G, contact locations p_i, and contact forces λ_i. Such a reduction loses detail about contacting limbs, with forces no longer restricted according to the contact models of the previous section.

While the reduction to a centroidal dynamics model addresses the high dimensionality of the original problem, it remains nontrivial to ensure that centroidal solutions are whole-body feasible, since they neglect the geometric constraints and actuation bounds. This omission in the model has motivated planning with whole-body kinematics and centroidal dynamics [18], or by alternating between centroidal dynamics and whole-body solves [26], [55]. Despite these advances, heuristic constraints on the CoM and footholds (e.g., to ensure reachability) remain prevalent, and simple bounds on the angular momentum (or fixing it to zero) are often used to simplify the problem.

Even with these simplifications, the centroidal dynamics (3) contain nonlinearities in the angular momentum equation due to bilinear terms from the cross product $(p_i - p_{\text{CoM}}) \times \lambda_i$. Tailored methods to address these terms have been a topic of great focus. Dai and Tedrake [41] consider fixing a priori a polytopic or ellipsoidal region for the CoM, enabling optimization of an upper bound on the magnitude of the angular momentum. Valenzuela [113] considers relaxing the bilinear equalities with McCormick envelopes and employing mixed-integer convex optimization (see also, [21] and [22]). Fernbach et al. [46] presented a development where contact locations are fixed and the CoM trajectory is parameterized via a Bézier curve with one free knot point, which avoids nonlinear effects. Overall, many choices exist for optimizing centroidal trajectories, with tradeoffs between accuracy and computational efficiency.

It is possible to further simplify the optimization of contact forces by considering their net effect. Given frictional limitations, each individual contact force λ_i is constrained to a friction cone $\mathcal{C}_i$. Then, the net effect of these forces is a 6-D cone known as the *contact wrench cone* (CWC)

$$\text{CWC} = \left\{ \sum_{i=1}^{n_c} \begin{bmatrix} p_{i/\text{CoM}} \times \lambda_i \\ \lambda_i \end{bmatrix} \middle|\, \lambda_i \in \mathcal{C}_i \quad \forall i \in \{1 \ldots, n_c\} \right\}.$$

When each friction cone is replaced with a polygonal approximation (e.g., friction pyramid), then the CWC is a polyhedral convex cone, and many tools from computational geometry are available to support its use in planning (cf., [114]). The CWC plays an integral role in checking for static stability in multicontact scenarios [115], [116] and more recently has seen applicability for multicontact CoM motion generation [42]. Feasibility constraints based on the CWC are natural generalizations of the zero-moment point criteria [117] to multicontact. Since the computation of the CWC is itself costly, it is most useful when the CWC can be precomputed offline (e.g., when contact locations are known in advance).

B. Other Simplified Models

The centroidal dynamics (3) are a special modeling simplification in that they represent a projection of the equations of motion without adding artificial restrictions on possible movements (i.e., any whole-body feasible motion can be projected to a feasible solution of the centroidal dynamics). Many other simple models follow by adopting artificial motion restrictions (or approximations) to further simplify analysis and optimization. The suitability of these approximations/restrictions is often dictated by the robot morphology being modeled.

A common simple model for quadruped locomotion is the single-rigid-body (SRB) model (e.g., [57], [118], and [119]). This simplification is motivated by the light leg designs of many quadruped robots, which makes the total rotational inertia of the system I_G about the CoM approximately invariant with configuration [120]. Thus, the SRB can be viewed as a restricted version of the centroidal momentum model in the case when the mass distribution takes a constant shape. Variations on the model accommodate inertia shaping [119], [121], however, there is an important subtlety that must be kept in mind. While the linear momentum of a system is related to its CoM velocity (i.e., the rate of change in some average position), the angular momentum cannot, in general, be equated to the rate of change in any meaningful average orientation of the system [122]. This property is intimately related to the fact that the conservation of angular momentum, in general, defines a nonholonomic constraint [110].

The most common simple model for humanoid gait planning considers the centroidal dynamics with $k_G = 0$ (i.e., thus removing consideration of orientation dynamics) and the additional restriction that the CoM height c_z takes a fixed value denoted h. Rather than considering a collection of contact points, when walking on level ground, the overall center of pressure (CoP) position $p_{\text{CoP}} = [p_x, p_y, p_z]^T$ can be considered with $p_z = 0$ an assumed ground height. These restrictions lead to the LIP Model [47] with dynamics: $\ddot{c}_{x,y} = \omega^2(c_{x,y} - p_{x,y})$ where $\omega = \sqrt{g/h}$ represents its natural frequency. The beauty and power of the LIP reside in that these dynamics are linear, enabling convex optimization for planning CoM and CoP trajectories [49] or linear systems tools (e.g., LQR) for trajectory tracking [43].

More recently, work has focused on discharging assumptions on the CoM height for added flexibility [123]. One approach is to model height variations as a perturbation to the LIP [52] along with constraint tightening to ensure robust feasibility. Rather than treating height variations as a disturbance, they can be introduced into the dynamics model [36], [123]. A common simple form is $\ddot{p}_{\text{CoM}} = a_g + k(p_{\text{CoM}} - p_{\text{CoP}})$ where k represents a variable stiffness-like parameter, constrained to be positive. Strategies for running (e.g., based upon the spring-loaded inverted pendulum model) [124], [125] can be seen as placing additional restrictions on the form of k to mimic a Hookean spring. Overall, these point mass models remain most applicable to bipeds and humanoids where orientation dynamics can be minimized by ensuring all GRFs pass close to the CoM. This is also the case with a subset of quadruped gaits (e.g., trot walking), while others (e.g., bounding) cannot satisfy $k_G = 0$, and are more amenable to SRB models.

C. Summary

Modeling simplifications are commonly employed to reduce the computational burden for dynamic planning in legged robots. The most successful of these models (e.g., the centroidal model, SRB model, and LIP model) focus their modeling detail on addressing motion limitations imposed by contact interactions. The centroidal model does not impose any restrictions on motions (i.e., it represents a projection of the whole-body dynamics), whereas the other simplified models result from adding motion restrictions. These simplifications can unlock new structure (e.g., linearity) that can be leveraged for motion optimization problems over them (e.g., using the methods in the next section). In all cases, many whole-body details remain to be planned (e.g., swing foot trajectories). This omission represents a main downside to simple model planning, motivating more advanced numerical methods (see Section V) for the efficient optimization of whole-body plans.

1) Relationship With Learning: There are many ways emerging learning strategies have been used to advance the application of simplified models for locomotion planning. With regard to the models themselves, this review has covered existing strategies based on considerations of physics and expert intuition. Moving forward, the automatic discovery of simple models remains an important open problem (cf., [126]). Other work has looked at how to close the gap between simplified models and their whole-body counterparts, for example, to learn constraints on simplified models that address kinematics constraints at the whole-body level [74] or that address the gap via learning robust MPC strategies [127]. Yet other strategies have wrapped full RL pipelines around low-level control based on simple models [128] to increase the sample efficiency of learning. It should be acknowledged that even though RL strategies are often architected (e.g., via autoencoders) to learn low-dimensional representations within their layers (cf., [129]), these latent representations remain difficult to interpret. It is an open problem whether the existing simplified models may be used to increase the interpretability of learned controllers as well.

V. Numerical Methods for Solving OCPs

While the decomposition of our general problem formulation in Section II (e.g., via modeling simplifications) remains a bit of an art, the technical details of solving a TO problem represent a field unto its own. This field has benefited from contributions across engineering applications ranging from chemical process control [130] to flight planning [131]. We begin this section by making a few simplifications to our general problem (1), so that we may review the most common methods for solving OCPs at large, which provides context for the most common approaches taken for robotics in particular. We then gradually discharge our original simplifications and explain nuances that arise in OCPs for locomotion. The treatment of these nuances has recently fueled a rapid increase in whole-body optimization, where the need for online replanning has led to the development of fast structure-exploiting solvers tailored for robotics.

To begin, let us consider (1) in the case of a single contact phase (e.g., optimizing a trajectory for a humanoid doing an in-place dance on two feet). For better alignment with the dynamic optimization literature, we rewrite the dynamics (1b) as a system of first-order ordinary differential equations (ODEs) $\dot{x}(t) = f(x(t), u(t), p(t))$, with contact forces $\lambda(t)$ treated (mathematically) as time-varying parameters $p(t) = \lambda(t)$. The formulation can also be readily extended to consider time-invariant parameters. In the presence of contacts, the contact points must not move relative to the ground, which can be written as an algebraic constraint $g(t, x(t), u(t), p(t)) = 0$. These ODEs, combined with the algebraic constraints, lead to a system of differential–algebraic equations (DAEs) [130]. A generic nonlinear OCP for such a system can then be considered as follows:

$$\underset{x(\cdot),u(\cdot),p(\cdot)}{\text{minimize}} \int_{t_0}^{t_f} \ell(x(t), u(t), p(t))\mathrm{d}t + L(x(t_f)) \tag{4a}$$

$$\text{subject to } \dot{x}(t) = f(x(t), u(t), p(t)) \tag{4b}$$

$$0 = g(x(t), u(t), p(t)) \qquad \forall t \in [t_0, t_f] \tag{4c}$$

where the objective function, with running cost ℓ and terminal cost L, is minimized over the time interval $[t_0, t_f]$ subject to the system dynamics defined by the DAEs. While the variables above match those in our original formulation (1), the problem could equally apply to a whole-body or simplified model.

For locomotion problems, formulation (4) requires an extension to consider how the dynamics change in different contact modes. Here, we assume that the mode sequence is fixed a priori, and consider the multiphase problem as follows:

$$\underset{\substack{x(\cdot),u(\cdot),p(\cdot),\\ s_1,\ldots,s_{n_{\mathrm{ph}}}}}{\text{minimize}} \sum_{j=1}^{n_{\mathrm{ph}}} \left[\int_{s_{j-1}}^{s_j} \ell_j(x(t), u(t), \lambda(t))\mathrm{d}t + L_j(x(s_j)) \right] \tag{5a}$$

$$\text{subject to } \dot{x}(t) = f_j(x(t), u(t), \lambda(t)) \tag{5b}$$

$$g_j(x(t), u(t), \lambda(t)) = 0 \tag{5c}$$

$$h_j(x(s_j^-)) = 0 \tag{5d}$$

$$x(s_j^+) = R_j(x(s_j^-)) \tag{5e}$$

$$\forall t \in [s_{j-1}, s_j],\ j = 1, \ldots, n_{\mathrm{ph}} \tag{5f}$$

$$s_0 = t_0,\ s_{n_{\mathrm{ph}}} = t_f \tag{5g}$$

where $j = 1, \ldots, n_{\mathrm{ph}}$ denotes the phase index, with the time interval of the jth phase given by $[s_{j-1}, s_j]$, and n_{ph} the number of phases. At the mode switch, a guard constraint (5d) is typically used to ensure the active contact points are in contact with the environment. Depending on the contact model, the reset maps (5e) may correspond to the impact dynamics or to switching between coordinate representations.

Generally speaking, many methods for solving an OCP only apply directly to (4) in the ODE case, but often admit extensions to the DAE and multiphase settings. These methods can be categorized into three groups: global methods based on the Hamilton–Jacobi–Bellman (HJB) equation (or dynamic programming in discrete time), local indirect methods, and local direct methods (see Table II).

Methods based on dynamic programming exploit Bellman's principle of optimality [132] to solve a discrete-time version of the OCP (4). They do so via finding the optimal cost-to-go $V(t_0, x_0)$ (also called the *value function*). In continuous time, the value function must satisfy a partial differential equation (PDE) known as the HJB. Its solution can be approximated by discretizing time and space and applying dynamic programming. However, it is well known that the complexity of this strategy increases exponentially with the number of states and controls. Therefore, it is not directly applicable to most legged robots, requiring clever approximate decomposition strategies for application to high-DoF models [133], [134].

Indirect methods transform the original OCP into a boundary value problem by using Pontryagin's maximum principle [135] to formulate the so-called costate equations. This approach enables preoptimizing the control as a function of state and costate if the dynamics are control affine—which is the case for most legged robots. Indirect methods turn the OCP into a root-finding problem, which can provide fast and accurate solutions. It can be challenging to initialize the costate for these methods, and additional mechanisms (e.g., homotopy) are needed to enforce state inequalities [136].

Direct methods have been more broadly adopted by the legged robotics community, and are generally compatible with DAEs. In direct methods, the OCP is transcribed into a finite-dimensional NLP by discretizing the controls and states with respect to time. Therefore, direct methods directly find the minimum of (1a), where the NLP can be solved with well-established optimization techniques, e.g., sequential quadratic programming [137]. The differences between various direct methods lie in how the discretization is carried out. In the following, we focus exclusively on the three prominent direct methods in the community, namely, *direct multiple shooting*, *direct collocation*, and *DDP*. A more comprehensive review can be found in [138].

A. Multiple Shooting

In direct shooting methods [139], the controls $u(t)$ are discretized over the time horizon, while the state trajectories are obtained via forward integration. In contrast to *single shooting*, where a single integration is performed over the whole time horizon, in multiple shooting, the time horizon is discretized, then integration is performed on each of the segments (see

TABLE II
HIGH-LEVEL RELATIONSHIP BETWEEN OPTIMAL CONTROL SOLUTION METHODS

		Supporting Theory	***Computation***
Global		Hamilton-Jacobi-Bellman (HJB)	Numerical PDE Solvers or Dynamic Programming
Local	***Direct*** (*discretize then optimize*)	Numerical Integration	**Direct Shooting/Collocation & NLP**
	Indirect (*optimize then discretize*)	Pontryagin Maximum Principle (PMP)	Indirect Shooting/Collocation & Root Find

This survey focuses mostly on direct methods for continuous-time problems due to their prominence in robotics applications.

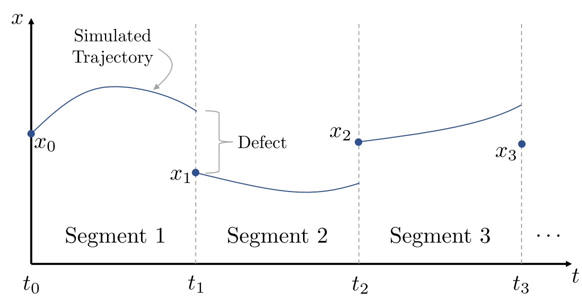


Fig. 4. Depiction of multiple shooting. Each shooting segment is associated with a numerical simulation (e.g., via a numerical integration method). Optimization variables include the states x_i at the beginning of each segment and variables describing the controls applied in each segment. Constraints enforce continuity, i.e., that the defect between segments is zero.

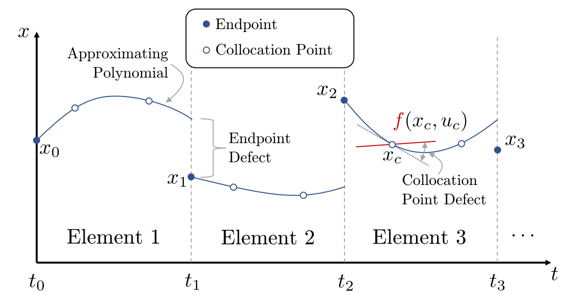


Fig. 5. Depiction of collocation. Each finite element is associated with a polynomial that approximates the trajectory. Optimization variables include the states x_i at each endpoint, and states/controls at each collocation point. Constraints enforce that the endpoint defect is zero, and that the slope of the approximating polynomial matches the evaluation of the dynamics function f at each collocation point.

Fig. 4). The initial state of each segment is added to the decision variables, and *continuity constraints* are introduced to guarantee that the final state of each segment i (computed with integration) matches the initial state of the next segment $i + 1$. In essence, as in Fig. 4, the initial optimization problem is divided into smaller optimization problems over the discretized grid, whose initial conditions can be set separately. By parameterizing the time horizon with a series of initial value problems, the time interval over which integration is performed is shortened, reducing the high sensitivity issue of single shooting and preventing states from diverging. This makes multiple shooting deliver rather robust solutions, as shown in [140], which motivated many researchers to use it for motion generation with complex robots.

In particular, the use of multiple shooting has been popular for generating motions considering whole-body dynamics where contact modes are set a priori, i.e., predefined contact sequences. This is the case of [29] and [31], where contacts are assumed to be rigid and impacts are instantaneous and inelastic. The resulting OCP is then a multiphase problem with discontinuous phase transitions, as in (4). Hereid et al. [30], used multiple shooting to optimize virtual constraints as part of gait design for a hybrid zero dynamics (HZD) controller with a compliant bipedal robot.

B. Collocation

In direct collocation (also known as direct transcription [138]) the original OCP is transformed into an NLP having control and state trajectories as decision variables. Controls and states are discretized over a time grid, where the intervals are called *finite elements* (see Fig. 5). Controls are approximated in each finite element by a finite-dimensional representation, and the states are approximated by polynomials. The system dynamics are then enforced by imposing *continuity constraints* to eliminate endpoint defects at the end of each element and through internal *collocation constraints* imposed at each of the *collocation points* (see Fig. 5). These collocation constraints enforce that the slope of the approximating polynomial matches the system dynamics $f(x(t), u(t), p(t))$ at each collocation point. The choice of the number and location of these collocation points has been thoroughly investigated [141] because it affects the accuracy of the method. More details can be found in [142] and [143].

Collocation has been a popular method to compute walking motions for legged robots using whole-body models. In [34], a periodic motion was generated for a whole-body bipedal robot, where the convergence of the periodic cycle was guaranteed through the implementation of HZD virtual constraints. Pardo et al. [144], used collocation to generate motions for a quadruped robot, where the contact sequences were predefined. Collocation constraints were modified to address contact constraints in [33], providing third-order integration accuracy with demonstrations on a full-body humanoid.

Most robotics methods to date have considered collocation where the system dynamics are expressed as a system of first-order ODEs. Vanilla collocation methods apply separate approximating polynomials for each component of the state (e.g., separate polynomials for q and $\dot{q}$), with their consistency only enforced at collocation points [145]. Recent work [145], [146] has shown the benefit of using a unified interpolating polynomial for the components of q and $\dot{q}$, which ensures their consistency across the trajectory and improves accuracy [145].

C. Differential Dynamic Programming

In recent years a high interest has grown for DDP and its variants. DDP [147] is an age-old method for solving discrete-time unconstrained OCPs. The method is based on dynamic programming, but it overcomes the curse of dimensionality by working with a local quadratic estimate of the value function. This makes DDP different from dynamic programming because the method does not attempt to exhaustively explore the state space of nonconvex problems. DDP can be roughly seen as an efficient iterative algorithm for solving the banded system of linear equations associated with the Karush–Kuhn–Tucker (KKT) conditions of an unconstrained OCP transcribed with collocation. Indeed, DDP closely resembles Newton's method [148], but it is not completely equivalent to it. The main difference is that to solve the KKT system, DDP expresses the control inputs as linear functions of the state, but then, instead of computing control perturbations by using linearized dynamics from the KKT system, it forward simulates with the nonlinear dynamics. This makes DDP a direct single-shooting strategy (since the state is computed by integrating the dynamics), but it is better suited to handle unstable dynamics compared with vanilla shooting since its use of state feedback helps prevent divergence.

DDP requires the second derivatives of the dynamics to achieve quadratic convergence. While this is a desirable property, it may be challenging to compute these terms for complex systems. Variants of DDP that make use of Hessian approximations have been more popular, such as iLQR and iLQG [1], [149], [150], with other recent work focused on efficiently computing the full Hessian [151], [152].

Many extensions to the original DDP algorithm have been considered to address constraints beyond those from the system dynamics. Box DDP [153] accounts for box constraints on the control inputs. Hierarchical DDP [150] allows for the optimization of a hierarchy of cost functions in a lexicographic order. DDP extensions that can treat arbitrary nonlinear inequality state-control constraints have been proposed based either on interior-point techniques [154], active-set methods [155], augmented Lagrangian [156], [157], or relaxed barrier [38] approaches. Recently, DDP has also been extended to handle implicit dynamics [5], multiphase dynamics [27], [97], mode switching constraints [158], and switch timing [28], [158], all of which are useful for handling contacts. Even though DDP was born as a single-shooting method, it has been extended to multiple shooting [159], [160], [161] to further improve its handling of sensitive dynamics.

D. Contact-Implicit Considerations

As a departure from the main assumptions of this section, contact-implicit (or contact-invariant) formulations (e.g., as first discussed in Section III-B2) instead seek to optimize the mode sequence. The choice of transcription strategy is highly coupled with the contact model adopted. When using shooting methods, the key need is for gradients that relate changes in states/controls to trajectory outcomes. For example, when adopting a relaxed contact model, the outcomes of simulation are always differentiable, enabling shooting solvers and/or DDP [1], [5]. When adopting an event-driven hybrid model, these methods can still be used when paired with suitable hybrid sensitivity analysis, as in [9]. Likewise, discrete-time shooting can be applied with time-stepping LCP solvers [8], although both this and the hybrid case exhibit pathologies that prevent differentiability in corner cases (cf., Section III-A4).

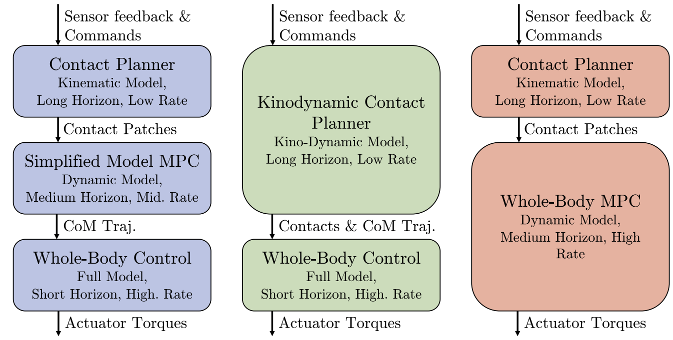


Fig. 6. Different potential strategies for breaking Problem 1 into smaller pieces to ease computational challenges.

When using direct collocation methods, existing contact-implicit methods rely on complementary constraints in the optimization. First-order methods [10] have been extended to high-order schemes, assuming the mode is not changing during an element [11], [12], [13]. These methods either require impacts at the element boundaries [13], [162] or sacrifice accuracy to allow impacts during the element [12], [13]. There is a tradeoff, since enforcing impacts at the element boundaries requires additional complementarity constraints. The alternative is to consider relaxations of the LCP constraints [23], which again present tradeoffs for accuracy and computational complexity.

E. Summary

Within robotics, direct methods remain the most attractive numerical approaches for TO. While both direct shooting and direct collocation can be used in cases when the mode sequence is fixed a priori, contact-implicit strategies require more careful consideration regarding the contact modeling choices adopted and how those affect transcription. Recent years have seen progress in accelerating the solution of shooting problems through the use of DDP, with recent sparse QP solvers (e.g., [163]) opening the door for similar accelerations to closely related collocation formulations.

1) Relationship to Learning: There have been many motivating examples of learning being used to support TO. A common strategy is to employ learned networks to warm start optimization (see, e.g., [164]). Another common way for learning to support TO is by learning the value function, which enables online TO over shorter horizons [165]. Other strategies simplify the optimization by learning footholds [166], e.g., akin to a higher level contact planner.

VI. Online Whole-Body Planning and Control

There are many ways in which the state-of-the-art locomotion architectures combine planning with optimal control solutions for deployment online, depending on computational hardware, robot complexity, and task complexity, as depicted in Fig. 6. The most hierarchical strategy (left column) decomposes locomotion control into footstep planning, CoM planning (via MPC with a simplified model), and whole-body reactive control (e.g., [43]). Recent trends have been flattening this hierarchy, in particular, as optimization solvers have matured. Several recent demonstrations are relying on whole-body MPC (right column). The current state of the art still breaks the contact planning into a high-level module, as contact-implicit schemes do not yet run at the rates necessary for online control. Owing to its connection with the previous section, we review whole-body MPC before returning to the more classical methods of reactive whole-body control.

A. Online Whole-Body MPC

Within the past few years, the deployment of MPC with whole-body models of legged robots has gone from an aspirational target to one with broad experimental support. While we provide a short overview here, the interested reader is referred to an excellent and complementary recent review on the topic [167]. Works by Katayama and Ohtsuka [28] and Mastalli et al. [168] have shown how to accelerate DDP methods by leveraging inverse dynamics modeling of full rigid-body dynamics for quadrupeds. As a representative data point, Mastalli et al. [168] ran MPC at 50 Hz using a 1 s horizon with 100 timesteps of 10 ms. Other quadruped work has shown the viability of using whole-body kinematics and centroidal/SRB dynamics [169], [170], [171] for MPC. Contrary to the previous DDP results, Grandia et al. [171] employed sparse QP solvers [163] for MPC (100 Hz update, 1 s horizon, 15 ms timesteps). A commonality in these online strategies is the willingness to live with the suboptimal. Real-time iteration schemes [172] are applied where TO only runs a few iterations before resampling the initial state and proceeding to solve again [171]. Beyond quadrupeds, a recent milestone [173] demonstrated online MPC with a 22 DoF model of the TALOS humanoid operating at 100 Hz (0.5 s planning horizon, 10 ms timesteps) and employing DDP's local feedback between updates [38].

Across these demonstrations, a few commonalities emerge. The first is the use of a long planning horizon. The work in [169] nicely characterizes the effects of the horizon on performance and notes that planning for less than two gait cycles leads to a loss of stability. Within the literature on MPC theory, a proper design of terminal constraints and costs allows the use of shorter horizons without sacrificing recursive feasibility and stability—however, such theoretical certificates remain out of reach for the robot models currently in use. Instead, significant effort has gone into warm starting solvers [173], which might be unneeded if the theoretical problem was more thoroughly addressed. Given the computation challenges with high-DoF humanoid robots, and the limited applicability of SRB models, we see the most opportunity for growth in whole-body MPC for humanoids moving forward.

B. Instaneous Whole-Body Control

In cases when computational hardware or model/task complexity prevents the use of whole-body MPC, hierarchical schemes, such as the one on the left of Fig. 6 are necessary. For example, consider the setup in Fig. 7. When performing TO

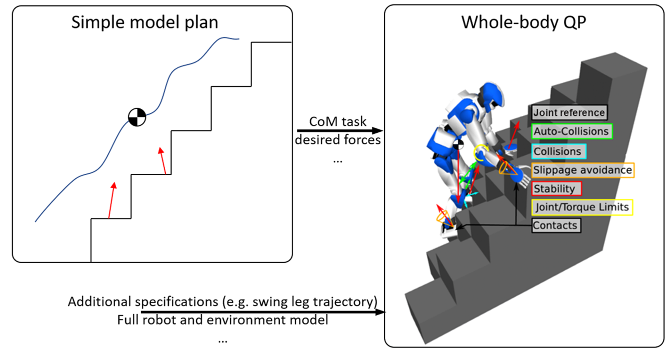


Fig. 7. Classical use of a whole-body QP in a pipeline to execute a simple-model plan obtained from the methods of the previous sections. A typical list of tasks and constraints is also depicted.

over simplified models (e.g., the SRB or a point-mass model), the OCP may be able to be solved quickly, but there are still many details that need to be defined (e.g., swing leg motion). Instantaneous control schemes are widely used in practice as a means to accomplish this goal, or more generally to track desired trajectories. The popularity of these methods owes to their ease of implementation and the fact that, with well-chosen objectives/constraints, the problem can be written as a QP, or a close variant, and solved quickly (from hundreds of μs to a few ms).

1) Formulation: In instantaneous settings, one common way to define the motion is by specifying quantities e_i that we want to regulate to 0, or keep above 0. For example, let us suppose we want the robot to follow a walking gait that was generated using the LIP model. For the robot CoM to track the LIP trajectory, we want to achieve $e_i(q,t) = 0$ where $e_i \in \mathbb{R}^3$ gives the error between the target position from the LIP and the true CoM position computed from q via forward kinematics. Other tasks, such as collision avoidance, are naturally captured by inequality constraints $e_i(q) \geq 0$. More generally, e_i can be a function of any subset of (q, ν, t). Such a definition is often called a *task error function* by reference to the task function [174] or operational-space [175] formalism.

A key point is that the dynamics (1b) are linear in $\dot{\nu}$, τ, and λ. This points to writing the regulation at the acceleration level where τ and λ can also be used directly to affect motions.

Task Dynamics: Consider a general task in the case where $\dot{q} = \nu$ and $e_i \in \mathbb{R}^m$. Differentiating $e_i(q,t)$ twice w.r.t. time

$$\ddot{e}_i(q, \nu, \dot{\nu}, t) = J_i(q)\dot{\nu} + \dot{J}_i(q, \nu)\nu + a_i(q, \nu, t) \tag{6}$$

where J_i is the Jacobian matrix of e_i and $a_i = \partial^2 e_i / \partial t^2$. To bring or keep e_i to a desired value, one then writes $\ddot{e}_i = \ddot{e}_i^d$ or $\ddot{e}_i \geq \ddot{e}_i^{\min}$, where $\ddot{e}_i^d / \ddot{e}_i^{\min}$ is a function of any subset of (q, ν, t) that indicates how we want to regulate e_i. This gives an equality/inequality that is linear in $\dot{\nu}$ as follows:

$$J_i(q)\dot{\nu} \geqq \ddot{e}_i^d - \dot{J}_i(q, \nu)\nu - a_i(q, t)\,. \tag{7}$$

If e_i also depends on ν, a single differentiation leads to

$$\frac{\partial e_i}{\partial \nu}\dot{\nu} \geqq \dot{e}_i^d - J_i(q, \nu)\nu - \frac{\partial e_i}{\partial t}(q, \nu, t)\,. \tag{8}$$

Both (7) and (8) can be written as

$$A_i(q,\nu)\dot{\nu} \geqq b_i(q,\nu,t)\,. \tag{9}$$

Task expressions: The most common tasks involve geometric features (e.g., body or CoM position) that are direct functions f_i of q. Typical equality tasks have the form $e_i(q,t) = f_i(q) - f_i^d(q,t)$, where f_i^d is the target value. For Lie group tasks (e.g., orientation), this error can be generalized as $e_i(q,t) = f_i(q) \ominus f_i^d(q,t)$, where $\ominus$ denotes a meaningful difference in the Lie group [176]. This general definition can capture tasks for the CoM, body positions, body orientations, and joint posture. Other equality tasks include a gaze task [177], visual servoing [178], or any subpart of one of the above [59], [179]. Inequality tasks commonly include joint bounds, collision avoidance, and balance [177]. Few tasks depend directly on ν. This is the case of joint speed limits, but also the centroidal momentum task [180], [181], [182] $e_i = A_G(q)\nu$.

We can also call tasks, by extension, constraints or motion requirements that are directly written on $\dot{\nu}$, τ, and λ and do not need regulation. These include, for example, limits on $\dot{\nu}$, τ, or friction constraints on the forces (to prevent sliding), that are usually approximated using a linear form $C\lambda \geq 0$ (e.g., as was also considered to simplify the CWC in Section IV-A). Direct references can also be given for these variables. For example, $\lambda - \lambda^d = 0$ can be used for force control.

Regulation of Equality Tasks: The task error e_i is often driven to 0 by first specifying a desired value for the derivative $e_i^{(l)}$ where l is the number of differentiations (1 or 2) such that $e_i^{(l)}$ is an affine function of $\dot{\nu}$, τ, or λ. For geometric equality tasks ($l = 2$), the most common approach follows a proportional-derivative (PD) construction: $\ddot{e}_i^d = -K_d\dot{e}_i - K_p e_i$, where K_d and K_p are positive definite matrices, usually taken as diagonal matrices with $K_d = 2K_p^{1/2}$ to obtain critical damping. A common (and essential) equality task relates to maintaining contacts. Most often, hard contacts are assumed and $\ddot{e}_i^d$ is set to 0 or a damping term is used to stabilize any slippage [59].

Within the framework of HZD, a critical task comes from the construction of virtual constraints that specify actuated joints as a function of a state-dependent phase variable [96]. Historically, HZD controllers were first implemented via analytical feedback linearization [96], while later being connected to constrained optimization methods herein [75].

Born out of HZD implementation advances, an alternative for making e_i converge to zero is to design a control Lyapunov function (CLF) V_i for e_i. Then, the inequality $\dot{V}_i \leq -\gamma_i V$ forces the exponential convergence of e_i to 0, where γ_i sets the convergence rate. This is the idea behind the CLF-QP [75], which guarantees convergence when the inequalities are always feasible. However, ensuring persistent feasibility with practical robot models remains an open challenge.

Regulation of Inequality Tasks: Inequality tasks at the acceleration level, such as joint torque limits or force friction cones, can be directly imposed. Inequality tasks at the velocity level, such as joint velocity limits, can be regulated by setting $\dot{e}_i^{\min}$ such that e_i remains positive after a controller time step Δt: $e_i^+ = e_i + \Delta t\,\dot{e}_i \geq 0$. This approach works well for small Δt, and implies setting $\dot{e}_i^{\min} = -e_i/\Delta t$. Inequality tasks at the position level, such as joint position limits or obstacle avoidance, are much harder to enforce. The most common approach is to compute $\ddot{e}_i^{\min}$ so that e_i^+ is positive, but this can easily lead to conflicts with acceleration-level constraints [183]. Alternative approaches range from a trick of increasing Δt in the computation of e_i^+ [177], [183], to bounding velocities based on the distance to the position limit [184], to control barrier functions (CBFs) [76]. However, none of these approaches ensure compatibility between position-level and acceleration-level inequalities. The only methods that can guarantee them are those that account for position-level and acceleration-level inequalities at once [185], [186], albeit under the assumption of constant acceleration bounds. Enforcing position constraints remains precarious with instantaneous control, since it is the evolution of the system over an extended time period that determines whether the constraint can be respected in the future. This observation motivates predictive strategies (e.g., MPC) to address position inequality constraints via lookahead.

2) Resolution: Instantaneous control schemes make the assumption that the state x is known and constant during the control period Δt. This is supported by the fact that Δt is small (typically between 1 and 10 ms), and relies on the ability to solve the problem in that time. The contact state (i.e., which bodies are in contact) is assumed fixed at each control cycle.

Under these conditions, and if the friction cones are approximated as pyramids, aggregating all the desired behaviors of the tasks leads to a set of linear constraints as follows:

$$M\dot{\nu} - J_c^T\lambda - S^T\tau = -C\nu - \tau_g \tag{10a}$$

$$J_c\dot{\nu} = a_c \tag{10b}$$

$$A_{\dot{\nu},i}\dot{\nu} + A_{\lambda,i}\lambda + A_{\tau,i}\tau \geqq b_i \quad \text{for non-contact tasks } i \tag{10c}$$

where (10a) is a reorganization of (1b), (10b) gives contact constraints, and (10c) is a generalization of (9) to account for tasks directly written on τ or λ. Problem (10) represents what we would like the robot to achieve. The idea is to solve (10) for $\dot{\nu}$, τ, and λ as best as possible (as discussed below). On position- or velocity-controlled robots, $\dot{\nu}$ is integrated and the resulting configuration [59] or velocity [178] is sent to the robot. For a torque-controlled robot, τ is used as a command.

Presolving variants: Problem (10) is written over all 3 variables $\dot{\nu}$, τ, and λ. This gives great flexibility to express new tasks/constraints and offers a clear vision of what we want to achieve. However, the specific structure, of (10a) and (10b) can be exploited to presolve some variables and constraints before passing the problem to a solver.

The form of $S = [0 \quad I]$ for legged robots allows expressing τ as a linear function of $\dot{\nu}$ and λ, and eliminating it from the problem. Only the nonactuated rows of (10a) are kept [59], [182], leading to a near-universal decrease in computation time. Alternatively, joint accelerations $\dot{\nu}$ can be eliminated via (10a) with the inversion of M [187], or forces λ can be eliminated via projection on the nullspace of the contact constraints [188], [189]. Finally, both $\dot{\nu}$ and λ can be eliminated via (10a) and (10b), leaving τ as the only variable. Regardless, the optimal presolving variant is highly coupled with the numerical solver used.

Arbitrating Conflict: For most task sets considered in practice, task requirements will conflict, meaning (10) has no solution. In this case, one can measure violations with an $L2$ norm, and specify how important it is to minimize each violation. Two main approaches exist that can be combined: weighing each violation (also known as *soft priorities*) or defining a hierarchy between them (*strict priorities*).

Numerous works use a QP approach: all inequality constraints and some equality constraints in (10) are kept in the QP, making them the top priority. The weighted $L2$ norms of all other constraint violations form the objective of the QP. For example, let us consider the problem of tracking an LIP-based reference trajectory, while trying to use minimal joint torques within bounds. In this case, a QP could be formulated as follows:

$$\begin{aligned}\underset{\dot{\nu},\tau,\lambda}{\text{minimize}}\ & w_1\|J_{\text{CoM}}\dot{\nu} + \dot{J}_{\text{CoM}}\nu - (\ddot{p}^d_{\text{CoM}} + \ddot{e}^d_{\text{CoM}})\|^2 \\ & + w_2\|\tau\|^2 \\ \text{subject to}\ & M\dot{\nu} + C\nu + \tau_g = S^T\tau + J_c^T\lambda \text{ (dynamics)} \\ & J_c\dot{\nu} = a_c \text{ (fixed contacts)} \\ & C\lambda \leq 0 \text{ (friction cones)} \\ & \ldots \text{(other top-priority constraints)} \ldots\end{aligned}$$

with J_{CoM} being the CoM Jacobian, $\ddot{p}^d_{\text{CoM}}$ the target CoM acceleration, and $\ddot{e}^d_{\text{CoM}}$ the desired acceleration of e_{CoM}.

The use of QP formulations for reactive control is now nearly universal but follows a long historical development. After an early work motivated by the inclusion of unilateral contact forces [190], the QP-based inverse dynamics approach started with [63], [65] and closely related convex formulations [64], continuing with [59], [179], [191], [192] among many others. An advantage of this approach is that it relies on mature off-the-shelf solvers, which are often free (e.g., [193] and [194]). Its main limitations are the inability to handle inequalities at lower priority levels, and its need for detailed tuning.

One of the benefits of the QP formalism is its flexibility to incorporate other control paradigms into reactive whole-body control. For example [43] and [195] embed long-horizon optimal control for simple-model tasks via cost function terms for descending a task-space value function. Other work has explored the incorporation of passivity-based control approaches that improve robustness to unmodeled effects [77]. To improve robustness through other means, recent QP strategies have explored methods for controls to be invariant to velocity jumps from impact events [196], which are hard to detect in practice. Alternative strategies that consider soft contact can also be incorporated into whole-body QPs [197], [198], [199].

In contrast to using a weighting approach for tasks, a hierarchical approach assigns an explicit priority level for each task. Tasks at the same level can be combined with weights, and top-priority constraints need not be feasible. The resulting problem is called *lexicographic least-squares* or *hierarchical QP* (HQP) [177], [180], [182]. Kanoun et al. [200] presented the first solver to tackle inequality constraints at any priority level, followed by computational improvements in [180] and [182]. Dedicated solvers [201], [202] have been introduced to solve the problem efficiently in one pass. The hierarchical formulation is a strict superset of the QP-based formulation and is the limit case for when the ratio between task weights goes to infinity [203]. An HQP can be solved faster using a dedicated solver [202], both in theory and practice. However, computational costs and the complexity of handling singularities [204] have limited broad use.

C. Summary

This section has focused on common methods for reactive control that coordinate the selection of joint torques and contact forces at the current instant. These methods complement predictive control (i.e., over a horizon). By focusing on an instantaneous control problem, the formulation inherits a desirable structure (e.g., nonlinear constraints become linear ones). This enables solutions at rapid rates, which are amenable to computation using off-the-shelf QP solvers. There are many different variants to such whole-body problems that consider priorities between tasks, or that strategically presolve for some decision variables to accelerate the QP solution. While these methods have represented a natural outgrowth of operational-space control to legged robotics applications, the shift from solving these problems analytically to solving them via QPs has enabled many other control paradigms to be considered, including CLFs, CBFs, passivity-based strategies, and others guided by value functions. Overall, while there remain opportunities to further improve these methods, the current state of the art represents a reliable technology for future applications.

VII. Outlook and Prospects

We conclude this survey by reflecting on some of the main trends that were identified through the literature review, and also some of the notable gaps that remain to be addressed.

A. Main Trends

1) Main Trends for Optimization with Contacts: The past five years have seen an increase in contact implicit approaches that avoid specifying gait/contact sequences a priori. These methods have considered either compliant contact or MIP/LCP strategies. Solution speed remains a challenge for these methods, as does the nonconvexity of the problem. Strategies that combine planning and/or learning with contact implicit methods appear as a necessary next step. In the near term, fixing contact sequences/timing remains the most viable option for practical deployment in applications that require online optimization (e.g., in MPC). In the end, while much more recent work has concentrated on addressing rigid contacts, other approaches end up being similar since rigid complementarity constraints are often relaxed and smoothed, leading to similar operations as with compliant models.

2) Main Trends in the Dynamic Models Adopted for Model-Based Optimization: The past five years have observed an increase in model complexity used for MPC (with several examples using the full model). This advance has occurred from an increased understanding of gradient-based numerical methods and the availability of fast open-source software libraries to compute robot dynamics and their derivatives [205]. With Moore's law ending, subsequent advances may center on parallelization. However, even with increases in computational speed, the nonconvexity of problems is an unavoidable challenge with complex models. This observation suggests that simple models will likely remain pertinent in the near future. The review points out that a combination of learning strategies that include exploration (as in conventional RL) presents promise for optimization-based strategies to avoid poor local minima when adopting complex models.

3) Main Trends for Transcription: The past five years have included an emphasis on DDP methods as well as extensions of collocation schemes to contact-constrained/implicit settings. The adoption of DDP has been motivated in part by its demonstrated scalability to high-DoF systems. However, recent work has also shown specialized Riccati-like solvers [163], which have the promise to bring many of the advantages of shooting via DDP to other transcription approaches. While the numerical methods are improving, convergence to infeasible solutions is a challenge for complex systems, and this holds with any of the transcription methods. The review points that addressing these challenges is an important open point for future work. Further, simultaneous methods (collocation and multiple shooting) will present additional opportunities for algorithm-level parallelism to take advantage of how computational resources are expected to grow in the coming decades.

4) Main Trends for Learning for Optimal Control: Learning has emerged as a valuable ally for the application of TO on challenging systems, such as legged robots. Learning can be combined with TO in different ways. Recent work has proposed learning the robot dynamics model, e.g., using deep neural networks [206] or Gaussian processes [207]. Supervised learning can be used to exploit motion-capture data, for instance, to obtain human-like walking trajectories [208]. Cost functions, instead of being defined by expert users, can be learned, e.g., using active learning techniques [209]. Another promising approach that has been recently investigated is to learn good initial guesses to speed up the convergence of OCP solvers [164], [210]. Finally, another method to help OCP solvers is to learn an approximation of the Value function, which can then be used as a terminal cost to *guide* the solver [106], [165].

B. Overview of Open Problems

1) Formal Analysis: Despite a great deal of recent progress, there is a broad lack of formal analysis regarding the properties and operation of existing optimization-based robot control paradigms. Existing theoretical results regarding the stability and recursive feasibility of nonlinear MPC are mostly focused on the simplified problem of regulation to an equilibrium or tracking a feasible reference state trajectory [211], which leaves a theoretical gap for the classes of problems considered in the locomotion community. Another barrier to meaningful analysis is a relative lack of work addressing considerations of robustness to bounded or stochastic uncertainty (e.g., [25], [52], and [212]). Such a treatment would, for example, be critical to understanding how our control frameworks interact with state estimation. While there are clear academic open questions, it remains an open question of its own as to how much additional performance such analysis would unlock.

2) Reducing the Expertise Required: Beyond the technical expertise required for model-based control, there remains a great deal of domain-specific expertise required for crafting cost functions and constraints. It remains a challenge to remove the human as an outer loop in the tuning of problem formulations. A part of the issue is that it is often not apparent how to write down cost functions for even simple tasks. For example, the problem of opening a door has binary success, and such cost structures are largely incompatible with current tools. Whether through automated methods or through new tools that enable a broader set of cost designs, the best path for reducing required domain-specific expertise remains open.

Given the complexity of the numerical methods involved for online optimal control, an additional consideration is the availability of well-maintained open-source software for newcomers to rapidly deploy OCP solutions. In this regard, the recent review of Carpentier and Wieber [213] provided an excellent resource on various dynamics and optimal control packages available for formulating and solving OCPs for robotics. While we encourage engagement with these tools, we likewise call on everyone to contribute to their advancement and maintenance so that we might continue to accelerate the impact of OCP methods on robotics practice.

3) Optimal Control Versus or With RL?: With the recent progress in RL for legged locomotion (e.g., [129], and [214]) much debate has surrounded whether real-time optimization or reinforcement learned policies are the path forward. RL methods have some clear benefits, the main one probably being their generality. Indeed, since gradients of the model dynamics are not required, these methods have no trouble with nonsmooth contact-implicit problems, which are instead hard to solve with conventional gradient-based OCP solvers. An interesting explanation of why that is the case has been recently provided in [215], drawing a connection between *policy gradient* and the stochastic optimization technique *randomized smoothing*. Although our community has grown attached to gradient-based optimization, in order to overcome the challenges discussed in this survey it may be worth exploring gradient-free approaches, with RL being among the most promising. Another positive feature of RL methods (especially those based on dynamic programming [216]) is their ability to avoid local minima, which is fundamental to solve nonconvex problems. Indeed, while standard TO algorithms simply follow the gradient, some RL methods can exploit the learned action–value function to get out of local minima.

Still, there remain exciting opportunities for how model-based optimization can help accelerate learning. Strategies such as the popular guided policy search method [217] and relatives (see, e.g., [218], [219]) can leverage TO to cut down on needed exploration and guide learning. Another important opportunity for future work is to use the efficient constraint-handling nature of TO to help make learning *safe* [220]. Overall, since RL and OCP methods try to solve similar problems [221] (especially in offline simulations where models are unavoidable), the history of their synergy is still quite early.

C. Closing Remarks

Reflecting on the trajectory of the field, we see a great deal of commonality between the transformation that is taking place now and the one that occurred in the mid-2010s . In 2007, convex optimization strategies for reactive control were beginning to appear, with no shortage of concern regarding both 1) their online computation requirements or 2) the abandonment of previously available closed-form control solutions. By 2015, these methods were commonplace.

From where we are today, early demonstrations of whole-body MPC on hardware have greatly benefited from the structural understanding of physics-based models from previous reactive control developments, while also adopting reactive methods for low-level control. Yet, open challenges remain regarding 1) online computation and 2) the abandonment of convexity properties that would give solution assurances.

While we believe these state-of-the-art methods will become technologies in the near future, we also look forward to the way the community will build upon them via combination with the next generation of learning-based controllers, where shared optimal control fundamentals will allow model-based methodologies to inform and accelerate learning. Challenges regarding 1) offline computation requirements and 2) a lack of generalization guarantees with RL seem to both be well addressed by structure exploiting methods from model-based optimization, and we hope this survey will play a small role in helping those solutions come to light. Overall, the evolutionary progress of the field shows a steady maturation toward applications in logistics, agriculture, and construction. With external commercial funding at unprecedented levels, we look forward to the next decade of progress from legged robots, which promises to be greater than the previous.

References

[1] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 4906–4913.

[2] J. Koenemann et al., "Whole-body model-predictive control applied to the HRP-2 humanoid," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 3346–3351.

[3] M. Neunert, F. Farshidian, A. W. Winkler, and J. Buchli, "Trajectory optimization through contacts and automatic gait discovery for quadrupeds," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1502–1509, Jul. 2017.

[4] M. Neunert et al., "Whole-body nonlinear model predictive control through contacts for quadrupeds," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1458–1465, Jul. 2018.

[5] I. Chatzinikolaidis and Z. Li, "Trajectory optimization of contact-rich motions using implicit differential dynamic programming," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 2626–2633, Apr. 2021.

[6] Y. Zhu, Z. Pan, and K. Hauser, "Contact-implicit trajectory optimization with learned deformable contacts using bilevel optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 9921–9927.

[7] I. Mordatch, E. Todorov, and Z. Popović, "Discovery of complex behaviors through contact-invariant optimization," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–8, 2012.

[8] J. Carius, R. Ranftl, V. Koltun, and M. Hutter, "Trajectory optimization with implicit hard contacts," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3316–3323, Oct. 2018.

[9] N. J. Kong, G. Council, and A. M. Johnson, "iLQR for piecewise-smooth hybrid dynamical systems," in *Proc. IEEE Conf. Decis. Control*, 2021, pp. 5374–5381.

[10] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *Int. J. Robot. Res.*, vol. 33, no. 1, pp. 69–81,2014.

[11] W. Xi, Y. Yesilevskiy, and C. D. Remy, "Selecting gaits for economical locomotion of legged robots," *Int. J. Robot. Res.*, vol. 35, no. 9, pp. 1140–1154, 2016.

[12] Z. Manchester, N. Doshi, R. J. Wood, and S. Kuindersma, "Contact-implicit trajectory optimization using variational integrators," *Int. J. Robot. Res.*, vol. 38, no. 12–13, pp. 1463–1476, 2019.

[13] A. Patel, S. L. Shield, S. Kazi, A. M. Johnson, and L. T. Biegler, "Contact-implicit trajectory optimization using orthogonal collocation," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 2242–2249, Apr. 2019.

[14] N. Doshi, K. Jayaram, B. Goldberg, Z. Manchester, R. J. Wood, and S. Kuindersma, "Contact-implicit optimization of locomotion trajectories for a quadrupedal microrobot," in *Proc. Robot. Sci. Syst.*, 2018, pp. 1–10.

[15] A. Aydinoglu and M. Posa, "Real-time multi-contact model predictive control via ADMM," in *Proc. Int. Conf. Robot. Autom.*, 2022, pp. 3414–3421.

[16] K. Wampler and Z. Popović, "Optimal gait and form for animal locomotion," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 1–8, 2009.

[17] A. Ibanez, P. Bidaud, and V. Padois, "Emergence of humanoid walking behaviors from mixed-integer model predictive control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 4014–4021.

[18] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 295–302.

[19] B. Ponton, A. Herzog, S. Schaal, and L. Righetti, "A convex model of humanoid momentum dynamics for multi-contact motion generation," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2016, pp. 842–849.

[20] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli, "Gait and trajectory optimization for legged system through phase-based end-effector parameterization," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1560–1567, Jul. 2018.

[21] B. Aceituno-Cabezas et al., "Simultaneous contact, gait, and motion planning for robust multilegged locomotion via mixed-integer convex optimization," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2531–2538, Jul. 2018.

[22] Y. Ding, C. Li, and H. W. Park, "Single leg dynamic motion planning with mixed-integer convex optimization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1–6.

[23] S. L. Cleach, T. Howell, M. Schwager, and Z. Manchester, "Fast contact-implicit model-predictive control," 2023, *arXiv:2107.05616*.

[24] T. A. Howell, S. Le Cleach, S. Singh, P. Florence, Z. Manchester, and V. Sindhwani, "Trajectory optimization with optimization-based dynamics," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 6750–6757, Jul. 2022.

[25] J. Morimoto and C. G. Atkeson, "Minimax differential dynamic programming: An application to robust biped walking," *Adv. Neural Inf. Process. Syst.*, 2002.

[26] R. Budhiraja, J. Carpentier, and N. Mansard, "Dynamics consensus between centroidal and whole-body models for locomotion of legged robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 6727–6733.

[27] C. Mastalli et al., "Crocoddyl: An efficient and versatile framework for multi-contact optimal control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 2536–2542.

[28] S. Katayama and T. Ohtsuka, "Whole-body model predictive control with rigid contacts via online switching time optimization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 8858–8865.

[29] K. Mombaur, "Using optimization to create self-stable human-like running," *Robotica*, vol. 27, no. 3, pp. 321–330, 2009.

[30] A. Hereid, C. M. Hubicki, E. A. Cousineau, J. W. Hurst, and A. D. Ames, "Hybrid zero dynamics based multiple shooting optimization with application to robotic walking," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 5734–5740.

[31] Y. Hu and K. Mombaur, "Bio-inspired optimal control framework to generate walking motions for the humanoid robot iCub using whole body models," *Appl. Sci.*, vol. 8, no. 2, 2018, Art. no. 278.

[32] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "CHOMP: Gradient optimization techniques for efficient motion planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 489–494.

[33] M. Posa, S. Kuindersma, and R. Tedrake, "Optimization and stabilization of trajectories for constrained dynamical systems," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1366–1373.

[34] A. Hereid, E. A. Cousineau, C. M. Hubicki, and A. D. Ames, "3D dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1447–1454.

[35] J. Carpentier and N. Mansard, "Multicontact locomotion of legged robots," *IEEE Trans. Robot.*, vol. 34, no. 6, pp. 1441–1460, Dec. 2018.

[36] S. Feng, X. Xinjilefu, W. Huang, and C. G. Atkeson, "3D walking based on online optimization," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2013, pp. 21–27.

[37] F. Farshidian, M. Neunert, A. W. Winkler, G. Rey, and J. Buchli, "An efficient optimal planning and control framework for quadrupedal locomotion," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 93–100.

[38] R. Grandia, F. Farshidian, R. Ranftl, and M. Hutter, "Feedback MPC for torque-controlled legged robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4730–4737.

[39] H. Dai and R. Tedrake, "Optimizing robust limit cycles for legged locomotion on unknown terrain," in *Proc. IEEE Conf. Decis. Control*, 2012, pp. 1207–1213.

[40] H. Audren, J. Vaillant, A. Kheddar, A. Escande, K. Kaneko, and E. Yoshida, "Model preview control in multi-contact motion-application to a humanoid robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 4030–4035.

[41] H. Dai and R. Tedrake, "Planning robust walking motion on uneven terrain via convex optimization," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2016, pp. 579–586.

[42] S. Caron and A. Kheddar, "Multi-contact walking pattern generation based on model preview control of 3D COM accelerations," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2016, pp. 550–557.

[43] S. Kuindersma et al., "Optimization-based locomotion planning, estimation, and control design for the Atlas humanoid robot," *Auton. Robots*, vol. 40, pp. 429–455, 2016.

[44] X. Xiong and A. D. Ames, "Dynamic and versatile humanoid walking via embedding 3D actuated SLIP model with hybrid LIP based stepping," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 6286–6293, Oct. 2020.

[45] M. Khadiv, A. Herzog, S. A. A. Moosavian, and L. Righetti, "Walking control based on step timing adaptation," in *Proc. IEEE Trans. Robot.*, 2020, pp. 629–643.

[46] P. Fernbach, S. Tonneau, O. Stasse, J. Carpentier, and M. Taïx, "C-CROC: Continuous and convex resolution of centroidal dynamic trajectories for legged robots in multicontact scenarios," *IEEE Trans. Robot.*, vol. 36, no. 3, pp. 676–691, Jun. 2020.

[47] S. Kajita and F. Kanehiro, "Biped walking pattern generation by using preview control of zero-moment point," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, pp. 1620–1626.

[48] P. B. Wieber, "Viability and predictive control for safe locomotion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 1103–1108.

[49] A. Herdt, H. Diedam, P. B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, "Online walking motion generation with automatic footstep placement," *Adv. Robot.*, vol. 24, no. 5–6, pp. 719–737, 2010.

[50] I. Mordatch, M. D. Lasa, and A. Hertzmann, "Robust physics-based locomotion using low-dimensional planning," in *Proc. ACM SIGGRAPH*, 2010, pp. 1–8.

[51] A. Sherikov, D. Dimitrov, and P.-B. Wieber, "Whole body motion controller with long-term balance constraints," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 444–450.

[52] C. Brasseur, A. Sherikov, C. Collette, D. Dimitrov, and P. B. Wieber, "A robust linear MPC approach to online generation of 3D biped walking motion," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2015, pp. 595–601.

[53] M. A. Hopkins, D. W. Hong, and A. Leonessa, "Compliant locomotion using whole-body control and divergent component of motion tracking," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 5726–5733.

[54] Y. Liu, P. M. Wensing, J. P. Schmiedeler, and D. E. Orin, "Terrain-blind humanoid walking based on a 3-D actuated Dual-SLIP model," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 1073–1080, Jul. 2016.

[55] A. Herzog, S. Schaal, and L. Righetti, "Structured contact force optimization for kino-dynamic motion generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 2703–2710.

[56] C. Mastalli et al., "Trajectory and foothold optimization using low-dimensional models for rough terrain locomotion," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 1096–1103.

[57] 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 *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1–9.

[58] A. Kheddar et al., "Humanoid robots in aircraft manufacturing: The airbus use cases," *IEEE Robot. Autom. Mag.*, vol. 26, no. 4, pp. 30–45, Dec. 2019.

[59] J. Vaillant et al., "Multi-contact vertical ladder climbing with an HRP-2 humanoid," *Auton. Robots*, vol. 40, no. 3, pp. 561–580, 2016.

[60] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "MIT Cheetah 3: Design and control of a robust, dynamic quadruped robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 2245–2252.

[61] J. E. Slotine and M. W. Spong, "Robust robot control with bounded input torques," *J. Robot. Syst.*, vol. 2, no. 4, pp. 329–352, 1985.

[62] F. T. Cheng, R. J. Sheu, and T. H. Chen, "The improved compact QP method for resolving manipulator redundancy," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, no. 11, pp. 1521–1530, Nov. 1995.

[63] C. Collette, A. Micaelli, C. Andriot, and P. Lemerle, "Dynamic balance control of humanoids for multiple grasps and non coplanar frictional contacts," in *Proc. IEEE-RAS 7th Int. Conf. Humanoid Robots*, 2007, pp. 81–88.

[64] J. Park, J. Haan, and F. C. Park, "Convex optimization algorithms for active balancing of humanoid robots," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 817–822, Aug. 2007.

[65] Y. Abe, M. Da Silva, and J. Popović, "Multiobjective control with frictional contacts," in *Proc. ACM SIGGRAPH/Eurograph.*, 2007, pp. 249–258.

[66] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia, PA, USA: SIAM, 1994.

[67] E. D. Andersen, C. Roos, and T. Terlaky, "On implementing a primal-dual interior-point method for conic quadratic optimization," *Math. Program., Ser. B*, vol. 95, pp. 249–277, 2003.

[68] S.-H. Hyon, J. G. Hale, and G. Cheng, "Full-body compliant human–humanoid interaction: Balancing in the presence of unknown external forces," *IEEE Trans. Robot.*, vol. 23, no. 5, pp. 884–898, Oct. 2007.

[69] B. J. Stephens and C. G. Atkeson, "Dynamic balance force control for compliant humanoid robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 1248–1255.

[70] R. Featherstone, *Rigid Body Dynamics Algorithms*,. Berlin, Germany: Springer, 2008.

[71] M. Toussaint, K. R. Allen, K. A. Smith, and J. B. Tenenbaum, "Differentiable physics and stable modes for tool-use and manipulation planning," in *Proc. Robot. Sci. Syst.*, pp. 1–9, 2018.

[72] F. R. Hogan, E. R. Grau, and A. Rodriguez, "Reactive planar manipulation with convex hybrid MPC," in *Proc. Int. Conf. Robot. Autom.*, 2018, pp. 247–253.

[73] D. Stewart and J. C. Trinkle, "An implicit time-stepping scheme for rigid body dynamics with Coulomb friction," *Int. J. Numer. Methods Eng.*, vol. 39, no. 15, pp. 2673–2691, 1996.

[74] J. Carpentier, R. Budhiraja, and N. Mansard, "Learning feasibility constraints for multicontact locomotion of legged robots," in *Proc. Robot. Sci. Syst.*, 2017, pp. 1–9.

[75] A. D. Ames and M. Powell, "Towards the unification of locomotion and manipulation through control Lyapunov functions and quadratic programs," in *Proc. Lecture Notes Control Inf. Sci.*, 2013, pp. 219–240.

[76] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *Proc. Amer. Control Conf.*, 2016, pp. 322–328.

[77] J. Englsberger, A. Dietrich, G. Mesesan, G. Garofalo, C. Ott, and A. Albu-Schaeffer, "MPTC–Modular Passive Tracking Controller for stack of tasks based control frameworks," in *Proc. Robot. Sci. Syst.*, 2020, pp. 1–9.

[78] C. Duriez, "Control of elastic soft robots based on real-time finite element method," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 3982–3987.

[79] D. W. Marhefka and D. E. Orin, "A compliant contact model with nonlinear damping for simulation of robotic systems," *IEEE Trans. Syst., Man, Cybern. ASyst. Humans*, vol. 29, no. 6, pp. 566–572, Nov. 1999.

[80] R. Tedrake et al., "Drake: A planning, control, and analysis toolbox for nonlinear dynamical systems,", 2016.

[81] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Philadelphia, PA, USA: SIAM, 1998.

[82] K. H. Hunt and F. R. E. Crossley, "Coefficient of restitution interpreted as damping in vibroimpact," *J. Appl. Mechanics*, vol. 42, no. 2, pp. 440–445, 1975.

[83] E. Todorov, "Convex and analytically-invertible dynamics with contacts and constraints: Theory and implementation in MuJoCo," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 6054–6061.

[84] A. J. Van Der Schaft and J. M. Schumacher, *An Introduction to Hybrid Dynamical Systems*, vol. 251. London, U.K.: Springer, 2000.

[85] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical System: Modeling, Stability, and Robustness*. Princeton, NJ, USA: Princeton Univ. Press, 2012.

[86] D. E. Stewart, "Rigid-body dynamics with friction and impact," *SIAM Rev.*, vol. 42, no. 1, pp. 3–39, 2000.

[87] B. Brogliato, *Nonsmooth Mechanics : Models, Dynamics, and Control*. Berlin, Germany: Springer, 1999.

[88] M. Anitescu and F. A. Potra, "Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems," *Nonlinear Dyn.*, vol. 14, no. 3, pp. 231–247, 1997.

[89] M. Halm and M. Posa, "Modeling and analysis of non-unique behaviors in multiple frictional impacts," in *Proc. Robot. Sci. Syst.*, 2019, pp. 1–9.

[90] A. Saccon, N. Van De Wouw, and H. Nijmeijer, "Sensitivity analysis of hybrid system with state jumps with application to trajectory tracking," in *Proc. IEEE Conf. Decis. Control*, 2014, pp. 3065–3070.

[91] S. A. Burden, S. S. Sastry, D. E. Koditschek, and S. Revzen, "Event-selected vector field discontinuities yield piecewise-differentiable flows," *SIAM J. Appl. Dynamical Syst.*, vol. 15, no. 2, pp. 1227–1267, 2016.

[92] E. Heiden, D. Millard, E. Coumans, Y. Sheng, and G. S. Sukhatme, "NeuralSim: Augmenting differentiable simulators with neural networks," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 9474–9481.

[93] N. Fazeli, S. Zapolsky, E. Drumwright, and A. Rodriguez, "Fundamental limitations in performance and interpretability of common planar rigid-body contact models," *Int. Symp. Robot. Res.*, vol. 15, no. 2, pp. 1227–1267, 2017.

[94] B. Acosta, W. Yang, and M. Posa, "Validating robot. simulators on real-world impacts," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 6471–6478, Jul. 2022.

[95] K. Byl and R. Tedrake, "Approximate optimal control of the compass gait on rough terrain," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 1258–1263.

[96] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Trans. Autom. Control*, vol. 48, no. 1, pp. 42–56, Jan. 2003.

[97] R. Budhiraja, J. Carpentier, C. Mastalli, and N. Mansard, "Differential dynamic programming for multi-phase rigid contact dynamics," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2018, pp. 1–9.

[98] R. Bordalba, T. Schoels, L. Ros, J. M. Porta, and M. Diehl, "Direct collocation methods for trajectory optimization in constrained robotic systems," *IEEE Trans. Robot.*, vol. 39, no. 1, pp. 183–202, Feb. 2023.

[99] R. Deits and R. Tedrake, "Footstep planning on uneven terrain with mixed-integer convex optimization," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 279–286.

[100] D. Song et al., "Solving footstep planning as a feasibility problem using L1-Norm minimization," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 5961–5968, Jul. 2021.

[101] N. Perrin, O. Stasse, L. Baudouin, F. Lamiraux, and E. Yoshida, "Fast humanoid robot collision-free footstep planning using swept volume approximations," *IEEE Trans. Robot.*, vol. 28, no. 2, pp. 427–439, Apr. 2012.

[102] S. Tonneau, A. Del Prete, J. Pettre, C. Park, D. Manocha, and N. Mansard, "An efficient acyclic contact planner for multiped robots," *IEEE Trans. Robot.*, vol. 34, no. 3, pp. 586–601, Jun. 2018.

[103] S. Dafarra, G. Romualdi, and D. Pucci, "Dynamic complementarity conditions and whole-body trajectory optimization for humanoid robot locomotion," *IEEE Trans. Robot.*, vol. 38, no. 6, pp. 3414–3433, Dec. 2022.

[104] F. de Avila Belbute-Peres, K. Smith, K. Allen, J. Tenenbaum, and J. Z. Kolter, "End-to-end differentiable physics for learning and control," *Adv. Neural Inf. Process. Syst.*, vol. 31, pp. 1–12, 2018.

[105] S. Pfrommer, M. Halm, and M. Posa, "ContactNets: Learning of discontinuous contact dynamics with smooth, implicit reps," in *Proc. Conf. Robot Learn.*, 2021, pp. 2279–2291.

[106] R. Deits, T. Koolen, and R. Tedrake, "LVIS: Learning from value function intervals for contact-aware robot controllers," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 7762–7768.

[107] A. Cauligi, P. Culbertson, E. Schmerling, M. Schwager, B. Stellato, and M. Pavone, "COCO: Online mixed-integer control via supervised learning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1447–1454, Apr. 2022.

[108] J. Siekmann, Y. Godse, A. Fern, and J. Hurst, "Sim-to-real learning of all common bipedal gaits via periodic reward composition," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 7309–7315.

[109] H. J. T. Suh, T. Pang, and R. Tedrake, "Bundled gradients through contact via randomized smoothing," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4000–4007, Apr. 2022.

[110] P. B. Wieber, "Holonomy and nonholonomy in the dynamics of articulated motion," in *Fast Motions in Biomechanics and Robotics: Optimization and Feedback Control*. Berlin, Heidelberg, Germany: Springer, 2006, pp. 411–425.

[111] P. M. Wensing and D. E. Orin, "Improved computation of the humanoid centroidal dynamics and application for whole-body control," *Int. J. Humanoid Robot.*, vol. 13, no. 01, 2016, Art. no. 1550039.

[112] D. E. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Auton. Robots*, vol. 35, no. 2, pp. 161–176, 2013.

[113] A. K. Valenzuela, "Mixed-integer convex optimization for planning aggressive motions of legged robots over rough terrain," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, USA, 2016.

[114] S. Caron, Q. C. Pham, and Y. Nakamura, "Leveraging cone double description for multi-contact stability of humanoids with applications to statics and dynamics," in *Proc. Robot. Sci. Syst.*, 2015, pp. 1–9.

[115] T. Bretl and S. Lall, "Testing static equilibrium for legged robots," *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 794–807, Aug. 2008.

[116] H. Audren and A. Kheddar, "3-D robust stability polyhedron in multi-contact," *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 388–403, Apr. 2018.

[117] H. Hirukawa et al., "A universal stability criterion of the foot contact of legged robots - adios ZMP," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 1976–1983.

[118] Y. Ding, A. Pandala, C. Li, Y.-H. Shin, and H.-W. Park, "Representation-free model predictive control for dynamic motions in quadrupeds," *IEEE Trans. Robot.*, vol. 37, no. 4, pp. 1154–1171, Aug. 2021.

[119] Z. Zhou, B. Wingo, N. Boyd, S. Hutchinson, and Y. Zhao, "Momentum-aware trajectory optimization and control for agile quadrupedal locomotion," *IEEE Robot. Automat. Letters*, vol. 7, no. 3, pp. 7755–7762, 2022.

[120] Y. Sim and J. Ramos, "Tello leg: The study of design principles and metrics for dynamic humanoid robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 9318–9325, Oct. 2022.

[121] S.-H. Lee and A. Goswami, "Reaction mass pendulum (RMP): An explicit model for centroidal angular momentum of humanoid robots," in *Proc. IEEE RAS Int. Conf. Robot. Autom.*, 2007, pp. 4667–4672.

[122] A. Saccon, S. Traversaro, F. Nori, and H. Nijmeijer, "On centroidal dynamics and integrability of average angular velocity," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 943–950, Apr. 2017.

[123] T. Koolen, M. Posa, and R. Tedrake, "Balance control using center of mass height variation: Limitations imposed by unilateral contact," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2016, pp. 8–15.

[124] P. M. Wensing and D. E. Orin, "High-speed humanoid running through control with a 3D-SLIP model," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 5134–5140.

[125] G. Piovan and K. Byl, "Approximation and control of the SLIP model dynamics via partial feedback linearization and two-element leg actuation strategy," *IEEE Trans. Robot.*, vol. 32, no. 2, pp. 399–412, Apr. 2016.

[126] Y. M. Chen and M. Posa, "Optimal reduced-order modeling of bipedal locomotion," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 8753–8760.

[127] A. Pandala, R. T. Fawcett, U. Rosolia, A. D. Ames, and K. A. Hamed, "Robust predictive control for quadrupedal locomotion: Learning to close the gap between reduced-and full-order models," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 6622–6629, Jul. 2022.

[128] Z. Xie, X. Da, B. Babich, A. Garg, and M. van de Panne, "Glide: Generalizable quadrupedal locomotion in diverse environments with a centroidal model," in *Proc. Workshop Alg. Foundations Robot.*, 2022, pp. 523–539.

[129] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Sci. Robot.*, vol. 5, no. 47, 2020, Art. no. eabc5986.

[130] L. T. Biegler and V. M. Zavala, "Large-scale nonlinear programming using IPOPT: An integrating framework for enterprise-wide dynamic optimization," *Comp. Chem. Eng.*, vol. 33, no. 3, pp. 575–582, 2009.

[131] D. Malyuta et al., "Convex optimization for trajectory generation," *IEEE Control Syst. Mag.*, vol. 42, no. 5, pp. 40–113, Oct. 2022.

[132] R. Bellman, "The theory of dynamic programming," *Bull. Amer. Math Soc.*, vol. 60, no. 6, pp. 503–515, 1954.
[133] E. C. Whitman and C. G. Atkeson, "Control of instantaneously coupled systems applied to humanoid walking," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2010, pp. 210–217.
[134] M. Chen, S. L. Herbert, M. S. Vashishtha, S. Bansal, and C. J. Tomlin, "Decomposition of reachable sets and tubes for a class of nonlinear systems," *IEEE Trans. Autom. Control*, vol. 63, no. 11, pp. 3675–3688, Nov. 2018.
[135] D. Liberzon, *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton, NJ, USA: Princeton Univ. Press, 2011.
[136] R. Bonalli, B. Hérissé, and E. Trélat, "Optimal control of endoatmospheric launch vehicle systems: Geometric and computational issues," *IEEE Trans. Autom. Control*, vol. 65, no. 6, pp. 2418–2433, Jun. 2020.
[137] S. Wright and J. Nocedal, *Numerical Optimization*. Berlin, Germany: Springer, 1999.
[138] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. Philadelphia, PA, USA: SIAM, 2010.
[139] M. Diehl, H. G. Bock, H. Diedam, and P.-B. Wieber, "Fast direct multiple shooting algorithms for optimal robot control," in *Fast Motions in Biomechanics and Robot*. Berlin, Germany: Springer, 2006, pp. 65–93.
[140] H. G. Bock and K.-J. Plitt, "A multiple shooting algorithm for direct solution of optimal control problems," in *IFAC Proc. Volumes*, vol. 17, no. 2, pp. 1603–1608, 1984.
[141] W. W. Hager, H. Hou, S. Mohapatra, and A. V. Rao, "Convergence rate for an hp collocation method applied to unconstrained optimal control," *J. Optim. Theory Appl.*, vol. 169, no. 3, pp. 801–824, 2016.
[142] O. Von Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Ann. Operations Res.*, vol. 37, no. 1, pp. 357–373, 1992.
[143] M. Kelly, "An introduction to trajectory optimization: How to do your own direct collocation," *SIAM Rev.*, vol. 59, no. 4, pp. 849–904, 2017.
[144] D. Pardo, M. Neunert, A. W. Winkler, R. Grandia, and J. Buchli, "Hybrid direct collocation and control in the constraint-consistent subspace for dynamic legged robot locomotion," in *Proc. Robot. Sci. Syst.*, 2017, pp. 1–9.
[145] S. Moreno-Martin, L. Ros, and E. Celaya, "Collocation methods for second order system," in *Proc. Robot. Sci. Syst.*, 2022, pp. 1–11.
[146] N. S. Hoang, "Collocation Runge–Kutta–Nyström methods for solving second-order initial value problems," *Int. J. Comput. Math.*, vol. 98, no. 12, pp. 2423–2444, 2021.
[147] D. Mayne, "A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems," *Int. J. Control*, vol. 3, no. 1, pp. 85–95, 1966.
[148] D. M. Murray and S. J. Yakowitz, "Differential dynamic programming and Newton's method for discrete optimal control problems," *J. Optim. Theory Appl.*, vol. 43, no. 3, pp. 395–414, Jul. 1984.
[149] E. Todorov and W. Li, "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *Proc. Amer. Control Conf.*, 2005, pp. 300–306.
[150] M. Geisert, A. Del Prete, N. Mansard, F. Romano, and F. Nori, "Regularized hierarchical differential dynamic programming," *IEEE Trans. Robot.*, vol. 33, no. 4, pp. 819–833, Aug. 2017.
[151] J. N. Nganga and P. M. Wensing, "Accelerating second-order differential dynamic programming for rigid-body systems," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7659–7666, Oct. 2021.
[152] S. Singh, R. P. Russell, and P. M. Wensing, "Analytical second-order partial derivatives of rigid-body inverse dynamics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 11781–11788.
[153] Y. Tassa, N. Mansard, and E. Todorov, "Control-limited differential dynamic programming," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 1168–1175.
[154] A. Pavlov, I. Shames, and C. Manzie, "Interior point differential dynamic programming," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 6, pp. 2720–2727, Nov. 2021.
[155] Z. Xie, C. K. Liu, and K. Hauser, "Differential dynamic programming with nonlinear constraints," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 695–702.
[156] T. A. Howell, B. E. Jackson, and Z. Manchester, "ALTRO: A fast solver for constrained trajectory optimization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 7674–7679.
[157] W. Jallet, A. Bambade, N. Mansard, and J. Carpentier, "Constrained differential dynamic programming: A primal-dual augmented lagrangian approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 13371–13378.
[158] H. Li and P. M. Wensing, "Hybrid systems differential dynamic programming for whole-body motion planning of legged robots," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5448–5455, Oct. 2020.
[159] E. Pellegrini and R. P. Russell, "A multiple-shooting differential dynamic programming algorithm," in *Proc. AAS/AIAA Space Flight Mechanics Meeting*, 2017, pp. 759–778.
[160] B. Plancher and S. Kuindersma, "A performance analysis of parallel differential dynamic programming on a GPU," in *Proc. Int. Workshop Algorithmic Found. Robot.*, 2018, pp. 656–672.
[161] M. Giftthaler, M. Neunert, M. Stäuble, J. Buchli, and M. Diehl, "A family of iterative Gauss-Newton shooting methods for nonlinear optimal control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1–9.
[162] S. Shield, A. M. Johnson, and A. Patel, "Contact-implicit direct collocation with a discontinuous velocity state," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 5779–5786, Apr. 2022.
[163] G. Frison and M. Diehl, "HPIPM: A high-performance quadratic programming framework for model predictive control," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6563–6569, 2020.
[164] N. Mansard, A. Del Prete, M. Geisert, S. Tonneau, and O. Stasse, "Using a memory of motion to efficiently warm-start a nonlinear predictive controller," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 2986–2993.
[165] K. Lowrey, A. Rajeswaran, S. Kakade, E. Todorov, and I. Mordatch, "Plan online, learn offline: Efficient learning and exploration via model-based control," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–15.
[166] S. Gangapurwala, M. Geisert, R. Orsolino, M. Fallon, and I. Havoutis, "RLOC: Terrain-aware legged locomotion using reinforcement learning and optimal control," *IEEE Trans. Robot.*, vol. 38, no. 5, pp. 2908–2927, Oct. 2022.
[167] S. Katayama, M. Murooka, and Y. Tazaki, "Model predictive control of legged and humanoid robots: Models and algorithms," *Adv. Robot.*, vol. 37, no. 5, pp. 298–315, 2023.
[168] C. Mastalli, S. P. Chhatoi, T. Corbéres, S. Tonneau, and S. Vijayakumar, "Inverse-dynamics MPC via nullspace resolution," *IEEE Trans. Robot.*, vol. 39, no. 4, pp. 3222–3241, Aug. 2023.
[169] A. Meduri, P. Shah, J. Viereck, M. Khadiv, I. Havoutis, and L. Righetti, "BiConMP: A nonlinear model predictive control framework for whole body motion planning," *IEEE Trans. Robot.*, vol. 39, no. 2, pp. 905–922, Apr. 2023.
[170] J.-P. Sleiman, F. Farshidian, M. V. Minniti, and M. Hutter, "A unified MPC framework for whole-body dynamic locomotion and manipulation," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4688–4695, Jul. 2021.
[171] R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter, "Perceptive locomotion through nonlinear model-predictive control," *IEEE Trans. Robot.*, vol. 39, no. 5, pp. 3402–3421, Oct. 2023.
[172] M. Diehl, R. Findeisen, F. Allgöwer, H. G. Bock, and J. P. Schlöder, "Nominal stability of real-time iteration scheme for nonlinear model predictive control," in *Proc. IEE Control Theory Appl.*, vol. 152, no. 3, pp. 296–308, 2005.
[173] E. Dantec et al., "Whole body model predictive control with a memory of motion: Experiments on a torque-controlled Talos," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 8202–8208.
[174] C. Samson, B. Espiau, and M. L. Borgne, *Robot Control: The Task Function Approach*. London, U.K.: Oxford Univ. Press, 1991.
[175] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE J. Robot. Autom.*, vol. 3, no. 1, pp. 43–53, Feb. 1987.
[176] J. Sola, J. Deray, and D. Atchuthan, "A micro Lie theory for state estimation in Robot," 2018, *arXiv:1812.01537*.
[177] L. Saab, O. E. Ramos, F. Keith, N. Mansard, P. Soueres, and J. Y. Fourquet, "Dynamic whole-body motion generation under rigid contacts and other unilateral constraints," *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 346–362, Apr. 2013.
[178] D. J. Agravante, G. Claudio, F. Spindler, and F. Chaumette, "Visual servoing in an optimization framework for the whole-body control of humanoid robots," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 608–615, Apr. 2017.
[179] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, "Optimization-based full body control for the DARPA robotics challenge," *J. Field Robot.*, vol. 32, no. 2, pp. 293–312, 2015.
[180] M. de Lasa, I. Mordatch, and A. Hertzmann, "Feature-based locomotion controllers," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–10, 2010.
[181] P. M. Wensing and D. E. Orin, "Generation of dynamic humanoid behaviors through task-space control with conic optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 3103–3109.
[182] A. Herzog, N. Rotella, S. Mason, F. Grimminger, S. Schaal, and L. Righetti, "Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid," *Auton. Robots*, vol. 40, pp. 473–491, 2016.

[183] K. C. Park, P. H. Chang, and S. H. Kim, "The enhanced compact QP method for redundant manipulators using practical inequality constraints," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1998, pp. 107–114.
[184] F. Kanehiro, M. Morisawa, W. Suleiman, K. Kaneko, and E. Yoshida, "Integrating geometric constraints into reactive leg motion generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 4069–4076.
[185] W. Decré, R. Smits, H. Bruyninckx, and J. De Schutter, "Extending iTaSC to support inequality constraints and non-instantaneous task specification," *IEEE Int. Conf. Robot. Autom.*, 2009, pp. 964–971.
[186] A. Del Prete, "Joint position and velocity bounds in discrete-time acceleration / torque control of robot manipulators," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 281–288, Jan. 2018.
[187] N. Mansard, "A dedicated solver for fast operational-space inverse dynamics," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 4943–4949.
[188] J. Park and O. Khatib, "Contact consistent control framework for humanoid robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 1963–1969.
[189] L. Sentis, J. Park, and O. Khatib, "Compliant control of multicontact and center-of-mass behaviors in humanoid robots," *IEEE Trans. Robot.*, vol. 26, no. 3, pp. 483–501, Jun. 2010.
[190] Y. Fujimoto, S. Obata, and A. Kawamura, "Robust biped walking with active interaction control between foot and ground," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1998, pp. 2030–2035.
[191] K. Bouyarmane and A. Kheddar, "Using a multi-objective controller to synthesize simulated humanoid robot motion with changing contact configurations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2011, pp. 4414–4419.
[192] T. Koolen et al., "Summary of team IHMC's virtual robotics challenge entry," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2013, pp. 307–314.
[193] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: A parametric active-set algorithm for quadratic programming," *Math. Program. Comput.*, vol. 6, no. 4, pp. 327–363, 2014.
[194] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "OSQP: An operator splitting solver for quadratic programs," *Math. Program. Comput.*, vol. 12, no. 4, pp. 637–672, 2020.
[195] M. Chignoli and P. M. Wensing, "Variational-based optimal control of underactuated balancing for dynamic quadrupeds," *IEEE Access*, vol. 8, pp. 49785–49797, 2020.
[196] W. Yang and M. Posa, "Impact invariant control with application to bipedal locomotion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 5151–5158.
[197] S. Fahmi, M. Focchi, A. Radulescu, G. Fink, V. Barasuol, and C. Semini, "Stance: Locomotion adaptation over soft terrain," *IEEE Trans. Robot.*, vol. 36, no. 2, pp. 443–457, Apr. 2020.
[198] G. Romualdi, S. Dafarra, and D. Pucci, "Modeling of visco-elastic environments for humanoid robot motion control," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4289–4296, Jul. 2021.
[199] T. Flayols, A. Del Prete, M. Khadiv, N. Mansard, and L. Righetti, "Reactive balance control for legged robots under visco-elastic contacts," *Appl. Sci.*, vol. 11, no. 353, 2021.
[200] O. Kanoun, F. Lamiraux, P.-B. Wieber, F. Kanehiro, E. Yoshida, and J.-P. Laumond, "Prioritizing linear equality and inequality systems: Application to local motion planning for redundant robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 2939–2944.
[201] A. Escande, N. Mansard, and P. B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *Int. J. Robot. Res.*, vol. 33, no. 7, pp. 1006–1028, 2014.
[202] D. Dimitrov, A. Sherikov, and P.-B. Wieber, "Efficient resolution of potentially conflicting linear constraints in robotics,", 2015.
[203] C. Van Loan, "On the method of weighting for equality-constrained least-squares problems," *SIAM J. Numer. Anal.*, vol. 22, no. 5, pp. 851–864, 1985.
[204] K. Pfeiffer, A. Escande, and A. Kheddar, "Singularity resolution in equality and inequality constrained hierarchical task-space control by adaptive nonlinear least squares," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3630–3637, Oct. 2018.
[205] J. Carpentier et al., "The Pinocchio C++ library: A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *Proc. IEEE/SICE Int. Symp. Syst. Integration*, 2019, pp. 614–619.
[206] M. Lutter, C. Ritter, and J. Peters, "Deep Lagrangian networks: Using physics as model prior for deep learning," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–17.
[207] R. Calandra, S. Ivaldi, M. P. Deisenroth, E. Rueckert, and J. Peters, "Learning inverse dynamics models with contacts," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 3186–3191.
[208] P. M. Viceconte et al., "ADHERENT: Learning human-like trajectory generators for whole-body control of humanoid robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2779–2786, Apr. 2022.
[209] C. Daniel, M. Viering, J. Metz, O. Kroemer, and J. Peters, "Active reward learning," in *Proc. Robot. Sci. Syst.*, 2014, pp. 1–10.
[210] W. X. Merkt, V. Ivan, T. Dinev, I. Havoutis, and S. Vijayakumar, "Memory clustering using persistent homology for multimodality- and discontinuity-sensitive learning of optimal control warm-starts," *IEEE Trans. Robot.*, vol. 37, no. 5, pp. 1649–1660, Oct. 2021.
[211] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*. Santa Barbara, CA, USA: Nob Hill Publishing, 2017.
[212] A. Gazar, M. Khadiv, A. Del Prete, and L. Righetti, "Stochastic and robust MPC for bipedal locomotion: A comparative study on robustness and performance," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2021, pp. 61–68.
[213] J. Carpentier and P.-B. Wieber, "Recent progress in legged robots locomotion control," *Curr. Robot. Rep.*, vol. 2, no. 3, pp. 231–238, 2021.
[214] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Sci. Robot.*, vol. 7, no. 62, 2022, Art. no. eabk2822.
[215] Q. L. Lidec, L. Montaut, C. Schmid, I. Laptev, and J. Carpentier, "Augmenting differentiable physics with randomized smoothing," 2022, *arXiv:2206.11884*.
[216] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," in *Proc. Int. Conf. Learn. Representations*, 2016, pp. 1–14.
[217] S. Levine and V. Koltun, "Guided policy search," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1–9.
[218] Z. Zhao, S. Zuo, T. Zhao, and Y. Zhao, "Adversarially regularized policy learning guided by trajectory optimization," in *Proc. Learn. Dyn. Control Conf.*, 2022, pp. 844–857.
[219] K. Chatzilygeroudis, V. Vassiliades, F. Stulp, S. Calinon, and J.-B. Mouret, "A survey on policy search algorithms for learning robot controllers in a handful of trials," *IEEE Trans. Robot.*, vol. 36, no. 2, pp. 328–347, Apr. 2020.
[220] M. Zanon and S. Gros, "Safe reinforcement learning using robust MPC," *IEEE Trans. Autom. Control*, vol. 66, no. 8, pp. 3638–3652, Aug. 2021.
[221] B. Recht, "A tour of reinforcement learning: The view from continuous control," *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 2, pp. 253–279, 2019.

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.

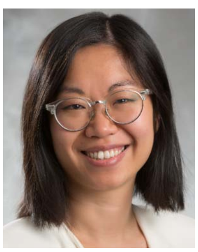
He is currently an Associate Professor with the Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN, USA, where he directs the Robotics, Optimization, and Assistive Mobility (ROAM) Laboratory. Before joining Notre Dame, he was a Postdoctoral Associate with MIT, 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.

Dr. Wensing was the recipient of the NSF CAREER Award in 2020. He is a Co-Chair for the IEEE RAS Technical Committee on Model-Based Optimization for Robotics and an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS.

Michael Posa (Member, IEEE) received the B.S. and M.S. degrees in mechanical engineering from Stanford University, Stanford, CA, USA, in 2007 and 2008, respectively. He received the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2017.

He is currently an Assistant Professor of Mechanical Engineering and Applied Mechanics with the University of Pennsylvania, Philadelphia, PA, USA, where he is a Member of the General Robotics, Automation, Sensing and Perception (GRASP) Lab. He holds secondary appointments in electrical and systems engineering and in computer and information science. He leads to Dynamic Autonomy and Intelligent Robotics Lab, University of Pennsylvania, which focuses on developing computationally tractable algorithms to enable robots to operate both dynamically and safely as they maneuver through and interact with their environments, with applications including legged locomotion and manipulation.

Dr. Posa was a recipient of multiple awards, including the NSF CAREER Award, RSS Early Career Spotlight Award, and best paper awards. He is an Associate Editor for IEEE TRANSACTIONS ON ROBOTICS.



Yue Hu (Member, IEEE) received the Doctorate degree in computer science (Dr. rer. nat.) from Heidelberg University, Heidelberg, Germany, in 2017.

From 2018 to 2020, she was a JSPS Fellow with the CNRS-AIST Joint Robotics Laboratory, Tsukuba, Japan. In 2021, she was an Assistant Professor with the Tokyo University of Agriculture and Technology, Fuchu, Japan. She is currently an Assistant Professor with the Department of Mechanical and Mechatronics Engineering, the University of Waterloo, Waterloo, ON, Canada, where she is also the head of the Active & Interactive Robotics Lab. Her research interests include human-robot interaction, collaborative robots, human factors, optimal control, and humanoid robots.

Dr. Hu is a Co-Chair for IEEE RAS Technical Committee on Model-Based Optimization for Robotics and is an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS.

Adrien Escande (Member, IEEE) received the M.S. degree in science and executive engineering from École des Mines de Paris, Paris, France, in 2005, and the Ph.D. degree in robotics from Université d'Évry Val-d'Essonne, Essonne, France, in 2008.

He worked with CEA–LIST in Fontenay-aux-Roses, Paris, France, and with the CNRS-AIST Joint Robotics Laboratory, Tsukuba, Japan. He is now a Researcher with the Inria Centre, University Grenoble Alpes, Grenoble, France. His research interests include whole-body planning and control for multibody robots, numerical optimization for robotics, and inexact computations to reduce control energy consumption.

Dr. Escande currently is a Co-Chair for IEEE RAS Technical Committee on Model-Based Optimization for Robotics and was an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS.

Nicolas Mansard received the M.Sc. degree in computer science from the University Grenoble Alpes, Grenoble, France, in 2003 and the Ph.D. degree in robotics from the University of Rennes, Rennes, France, in 2006.

He was a post-doctoral Researcher with Stanford University, Stanford, CA, USA, in 2007 and with the Joint Robotics Lab, Tsukuba, Japan, in 2008. He was an invited Researcher with the University of Washington, Seattle, WA, USA, in 2014. He is currently a CNRS Researcher (*"chargé de recherche CNRS"*) with CNRS, Paris, France, since 2009. His main research interests include the motion generation and planning and control of complex robots, with a special regard in humanoid robotics. His expertise covers sensor-based (vision and force) control, numerical mathematics for control, bipedal locomotion, and locomotion planning. He published more than 70 papers in international journals and conferences and supervised ten Ph.D. thesis.

Dr. Mansard was the recipient of the CNRS Bronze Medal in 2015 (one medal is awarded in France in automatic/robotic/signal-processing every year). He was previously an Associate Editor of IEEE TRANSACTIONS ON ROBOTICS.

Andrea Del Prete (Member, IEEE) received the degree (Hons.) in computer engineering from the University of Bologna, Bologna, Italy, in 2009, and the Ph.D. degree in robotics from the Cognitive Humanoids Laboratory, Italian Institute of Technology, Genova, Italy, in 2013.

From 2014 to 2017, he was a Postdoctoral Researcher with the Laboratoire d'Analyse et d'Architecture des Systemes, CNRS, Toulouse, France. In 2018, he worked as a Senior Researcher with the Max Planck Institute for Intelligent Systems, Tubingen, Germany. He is currently an Associate Professor in the Industrial Engineering Department with the University of Trento, Trento, Italy. His research interest focuses on the use of optimization and learning algorithms for the control, planning, and estimation of autonomous robots, with a special focus on legged locomotion. He has published more than 50 papers in international journals and conferences.

Dr. Prete is an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS.