Task-based Limb Optimization for Legged Robots

Sehoon Ha[1], Stelian Coros[2], Alexander Alspach[1], Joohyung Kim[1], Katsu Yamane[1]

Abstract— **The design of legged robots is often inspired by animals evolved to excel at different tasks. However, while mimicking morphological features seen in nature can be very powerful, robots may need to perform motor tasks that their living counterparts do not. In the absence of designs that can be mimicked, an alternative is to resort to mathematical models that allow the relationship between a robot's form and function to be explored. In this paper, we propose such a model to co-design the motion and leg configurations of a robot such that a measure of performance is optimized.**

The framework begins by planning trajectories for a simplified model consisting of the center of mass and feet. The framework then optimizes the length of each leg link while solving for associated full-body motions. Our model was successfully used to find optimized designs for legged robots performing tasks that include jumping, walking, and climbing up a step. Although our results are preliminary and our analysis makes a number of simplifying assumptions, our findings indicate that the cost function, the sum of squared joint torques over the duration of a task, varies substantially as the design parameters change.

I. Introduction

The diversity of morphologies seen in the animal kingdom has been a source of inspiration for roboticists since the field's very beginnings. Indeed, a wide variety of existing robotic systems aim to closely mimic real-life creatures. Examples include salamanders[1], cheetahs[2], kangaroos[3], chimpanzees[4], among many others. The process of creating bio-inspired robots is typically guided by observations and measurements coming from real creatures. Rather than mimicking morphological designs evolved by nature, our goal is to develop computational models that allow the intricate relationship between a robot's form and function to be efficiently explored.

The process of designing legged robots is notoriously challenging, due in part to the complex way in which morphological features shape motor capabilities. Current design processes rely on meticulous, time-consuming and largely manual design efforts led by experienced engineers. Once a design is finished and the robot built, control engineers implement locomotion strategies and attempt to push the hardware to its limits. If the robot's performance is unsatisfactory, the design process needs to repeat. However, it is often unclear how to best change the robot's design such that its performance improves. To address this challenge, our goal is to develop models that concurrently reason about a robot's morphology and control policies.

[1] Sehoon Ha, Alexander Alspach, Joohyung Kim, and Katsu Yamane are with Disney Research, Pittsburgh. {`sehoon.ha, alex.alspach, joohyung.kim, kyamane`}`@disneyresearch.com`

[2] Stelian Coros is with the Robotics Institute at Carnegie Mellon University. `scoros@andrew.cmu.edu`

For this preliminary study, we make a number of simplifying assumptions. First, the locomotion tasks that we consider are described by trajectories for a robot's center of mass (COM) and its feet. These motion plans are computed using trajectory optimization, they are independent of morphological features (e.g. limb configuration), and they are treated as constraints that the robots' motions must satisfy. For this work we restrict our attention on analyzing leg configurations, as opposed to other morphological features such as flexible spines. Last, we focus our discussion on planar robots. With these modeling simplifications in place, we address the following question: to what extent does the morphological design of a legged robot affect its ability to perform locomotion-based tasks?

We use an optimization-based approach to co-design the limb configuration (lengths of limb segments) and motions of a legged robot such that joint torques are minimized. Performing the optimization process for different tasks (e.g. walking vs bounding vs stair climbing) results in robot designs that are quite different from each other. While these findings are preliminary, they suggest that significant gains in performance can be expected if a robot's morphological features are appropriately designed.

A. Related Work

The task-based optimal design paradigm has recevied considerable attention in the field of manipulator design. A few papers [5], [6] discussed the optimization of angles and lengths of links to cover the prescribed workspace in 2D and 3D space. Van Henten and his colleagues [7] optimized a cucumber harvesting robot for minimal path lengths and maximal dexterity. Yang *et al.* [8] proposed an optimization formulation to find minimal degree-of-freedom configurations of modular robots for a given task. There is also a large body of work on finding the optimal design of parallel manipulators [9], [10], [11] that avoid singular configurations by checking the inverse condition number of Jacobians. However, Olds [12] pointed out that the inverse condition number is not sufficient for filtering out the worst cases, such as the end-effector velocity becomes slowest or the end-effector error become largest.

Researchers also have investigated task-based optimal design on non-manipulator robots. Jung *et al.* [13] improved the initial manual design of in-pipe cleaning robots to increase the factor of safety and reduce the mass. Kim *et al.* [14] designed a stair-climbing mobile robot with a rocker-bogie mechanism by optimizing the collision-free COM trajectory. However, our design problem for legged robots requires
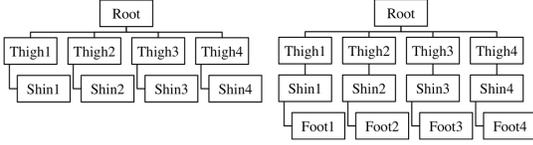
Fig. 1. The input structure of quadrupeds with two-link legs (left) and three-link legs (right).
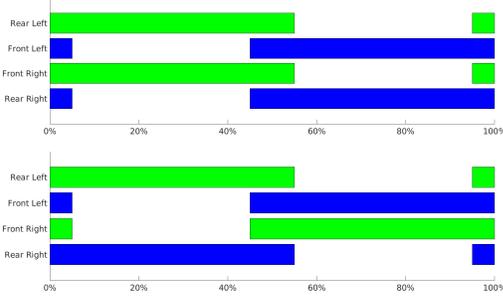

Fig. 2. The input gait graphs of trotting (top) and bounding (bottom) gaits. The x-axis represents one gait cycle, and solid bars represent footfall patterns.

the resolution of additional issues, such as contact and momentum planning.

II. OVERVIEW

Our goal is the development of mathematical models that can automatically design morphological features for legged robots such that they can efficiently perform specific locomotion-based tasks. The input to our system consists of a high-level description of a desired robot (e.g., how many legs the robot has, and how many rigid links are in each leg, (Fig. 1)). Further, the task is specified using a footfall pattern (Fig. 2) and the initial and final states of the robot's COM (Fig. 3). For example, Fig. 1-3 describe the walking task of two or three-link legged quadrupeds. For the given inputs, the framework optimizes design parameters that are required to fully define the shape of the robot, such as the lengths of limbs and links. The framework does not explicitly optimize mass distribution because we assume that the weights of servos cannot be freely adjusted. Instead, masses are treated as dependent variables of link lengths by assigning heavier weights for longer links.

The framework optimizes design parameters in two stages: the motion optimization stage and the design optimization stage. In the motion optimization stage, the framework optimizes the motion of the simplified model to minimize contact forces. The simplified model includes the center of mass trajectory, momentum trajectory, contact positions, and contact forces. In the design optimization stage, the framework optimizes leg link lengths and the associated full-body motions to achieve the given task while minimizing torque consumption.

In the following sections, we will describe the framework for optimizing motion and design parameters. The algorithms
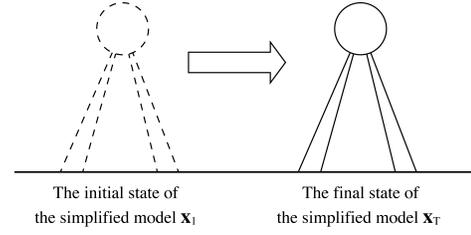

Fig. 3. The input task is described by the initial and final states of the simplified model which includes COM positions $\mathbf{p}^C$, orientations $\mathbf{r}^C$, and contact positions $\mathbf{p}^i$. The above example describes the walking task for quadrupeds.

are generally described for both 2D and 3D cases, although our experiments are limited to planar simulation.

III. MOTION OPTIMIZATION

In this stage, our framework optimizes the motion of the simplified robot model for executing the user specified task. We define our simplified model as a single rigid body with variable-length legs which describes centroidal dynamics and contact behaviors (Fig. 3). Let T be a number of frames and N be a number of legs. At each discretized frame t ($1 \leq t \leq T$), the state $\mathbf{x}_t = [\mathbf{p}_t^C, \mathbf{r}_t^C, \mathbf{p}_t^1, \cdots, \mathbf{p}_t^N, \mathbf{f}_t^1, \cdots, \mathbf{f}_t^N]$ is described by a center of mass position $\mathbf{p}^C$, orientation $\mathbf{r}^C$, contact position $\mathbf{p}^i$, and contact forces $\mathbf{f}^i$ where i ($1 \leq i \leq N$) is a leg index. Note that the motion of the simplified model does not require joint-level information for the leg structures.

The user defines the task with the initial state $\mathbf{x}_1$, the final state $\mathbf{x}_T$, and the gait graph $\mathbf{G} = \{c_t^i\} \; \forall t \leq T, i \leq N$. The contact variable c_t^i is 1 when the ith leg is in contact at the frame t, and 0 otherwise. The free variables of the optimization $\mathbf{x}$ are the states between the initial and final states, $\mathbf{x} = [\mathbf{x}_2, \cdots, \mathbf{x}_{T-1}]$. The objective of the optimization f_{design} is to minimize a weighted sum of velocity, acceleration, and contact forces for all frames.

$$
\begin{aligned}
f_{design}(\mathbf{x}) = \sum_t \Big(& w_v |[\dot{\mathbf{p}}_t^C, \dot{\mathbf{r}}_t^C]|^2 \\
& + w_a |[\ddot{\mathbf{p}}_t^C, \ddot{\mathbf{r}}_t^C]|^2 + w_f \sum_i |\mathbf{f}_t^i|^2 \Big)
\end{aligned}
\tag{1}
$$

where w_v, w_a, w_f are the weights for the velocity, acceleration, and contact force terms, which are set as $100.0, 0.01, 0.0001$ for all experiments. Note that the weights for acceleration and force terms are much smaller than the weight for the velocity term to normalize physical quantities in different units. Derivatives and second derivatives are calculated using second order finite differences. While minimizing the given objective function, the motion is constrained by physics laws:

$$m\ddot{\mathbf{p}}_t^C = \sum_{i=1}^N f_t^i$$

$$I\ddot{\mathbf{r}}_t^C = \sum_{i=1}^N ((\mathbf{p}_t^i - \mathbf{p}_t^C) \times \mathbf{f}_t^i)$$

$$c_t^i z(\mathbf{p}_t^i) = 0 \qquad \forall i \le N \qquad (2)$$

$$(1 - c_t^i)|\mathbf{f}_t^i| = 0 \qquad \forall i \le N$$

$$c_t^i \dot{\mathbf{p}}_t^i = 0 \qquad \forall i \le N$$

$$\mathbf{f}_t^i \in \mathcal{F}_\mu \qquad \forall i \le N$$

$$|\mathbf{p}_t^C - \mathbf{p}_t^i| \le l_{max} \ \forall i \le N$$

where m and I are the mass and inertia of a robot and are provided as user inputs. $l_{max}(= 0.8m)$ is the maximum effective limb length (the distance from the COM to the toes), which is shorter than the maximum cumulative limb length (the sum of all limb link lengths). The function $z(\mathbf{p})$ extracts the vertical z component of the given position. $\mathcal{F}_\mu$ indicates the friction cone. We use the constant friction parameter $\mu = 1.0$ for all test cases. From the top of Eq. (2), constraints enforce the conservation of linear and angular momentum, ensure that contacts occur at the ground, no external forces are allowed without contact, no foot slipping is allowed during contact, and enforce friction cones, and maximum leg lengths. Constraints are implemented as soft-constraints using the penalty method. The formulated motion optimization problem in this stage is solved using Sequential Quadratic Programming (SQP).

IV. DESIGN OPTIMIZATION

In this stage, the framework optimizes the design parameters to efficiently execute the given motion of the simplified model from the previous stage. The main goal of the design optimization stage is to find the optimal lengths of the various limb links $\mathbf{d} = \{d_1, d_2, \cdots, d_M\}$ that minimize a sum of joint torques. The number of links M is $2N$ for two-link legged robots and $3N$ for three-link legged robots.

For locomotion tasks, we include additional parameters $\mathbf{s}$ to modify swing foot trajectories $\bar{\mathbf{p}}_\mathbf{s}^i(t)$, because it is
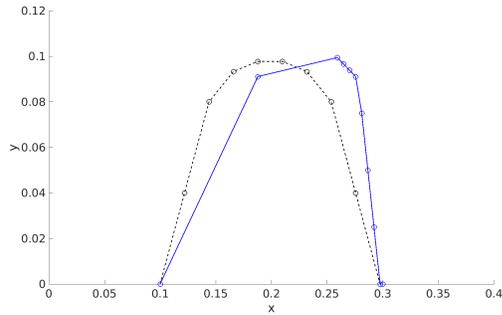
Fig. 4.   Two examples of swing foot trajectories with the same clearance height(0.2m), take-off position(0.1m), and landing position(0.3m). Each dot represent the foot position at each frame. The blue trajectory ($x_{peak} = 0.27$m, $t_{peak} = 0.05$s) takes only two frames to reach the peak, while the black trajectory ($x_{peak} = 0.2$m, $t_{peak} = 0.1$s) takes four frames.
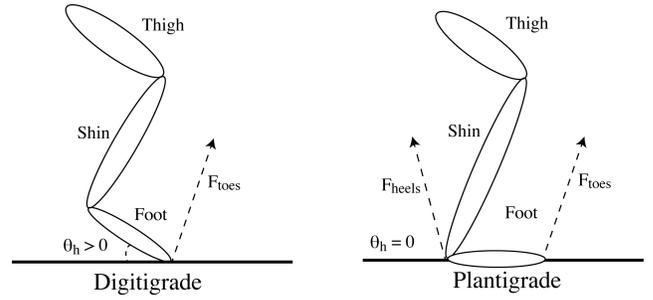
Fig. 5.   Two types of three-links legs. (Left) Digitigrade (the foot angle $\theta_h > 0$) that can apply forces only from toes (Right) Plantigrade(the foot angle $\theta_h = 0$) that can apply forces from both toes and heels.

important to model the passive dynamics of the swing legs, but this cannot be optimized in the previous stage. Some components of the swing foot trajectories are already determined: the desired foot clearance height h_{max} is given, and the contact timings and locations for foot take-off and landing come from the optimized motion plan. Within these constraints, we allow the framework to change the horizontal location of the peak x_{peak} and the time of the peak t_{peak} (Fig. 4) for each ith foot. Therefore, swing foot parameters $\mathbf{s}$ are defined as $\{x_{peak}^1, t_{peak}^1, \cdots x_{peak}^N, t_{peak}^N\}$.

For three-link legs, we have additional parameters $\mathbf{h}$ to define foot angle trajectories $\bar{\theta}_h^i(t)$ as linear splines with knobs $\mathbf{h} = \{h_1^{knob}, \cdots h_K^{knob}\}$ (K = number of knobs). The three-link leg is digitigrade when the foot angle is greater than zero, but is otherwise plantigrade and can apply contact force at its heel (Fig. 5). Therefore, we will treat the fullbody contact forces $\hat{\mathbf{f}}_t^j$ as additional free variables, rather than directly using the contact forces $\mathbf{f}_t^j$ from the motion optimization stage. Because one foot can potentially apply forces at two points, the toes and the heel, the fullbody contact forces $\hat{\mathbf{f}}_t^j$ has index $j \le 2N$.

We optimized the design parameters $\mathbf{d}$ and the fullbody motion parameters $\mathbf{s}$ and $\mathbf{h}$ using CMA-ES (Covariance Matrix Adaptation - Evolution Strategy) [15]. The objective function is described in Algorithm 1.

For the given parameters $\mathbf{d}$, $\mathbf{s}$ and $\mathbf{h}$, the cost function (Algorithm 1) sequentially solves joint positions $\mathbf{q}_t$, joint velocities $\dot{\mathbf{q}}_t$, joint accelerations $\ddot{\mathbf{q}}_t$, fullbody contact forces $\hat{\mathbf{f}}_t^j$, and torques $\boldsymbol{\tau}_t$.

First, the algorithm solves the inverse kinematics (IK) for each time frame to find the joint positions $\mathbf{q}_t$ including global position, orientation, and joint angles at the time t to match the desired trajectory from the motion optimization stage. The desired trajectory includes the desired COM positions $\mathbf{p}_t^C$, the desired COM orientation $\mathbf{r}_t^C$, and foot positions $\mathbf{p}_t^1 \cdots \mathbf{p}_t^N$. If the task is locomotion, the desired swing foot trajectories $\bar{\mathbf{p}}^i(t, \mathbf{s})$ are added. If the robot has three links per leg, the desired foot angle $\bar{\theta}^i(t, \mathbf{h})$ is also included. The

Algorithm 1 Objective function in the design optimization

Require: parameters for lengths $\mathbf{d}$, swing trajectory $\mathbf{s}$, foot angle $\mathbf{h}$.
1: **for** $t \in [1 \cdots T]$ **do**
2:    solve $\mathbf{q}_t$ to match the center of mass $\mathbf{p}_t^C$, the orientation $\mathbf{r}_t^C$, stance foot positions $\mathbf{p}_t^1 \cdots \mathbf{p}_t^N$, swing foot trajectories $\bar{\mathbf{p}}^{1 \cdots N}(t, \mathbf{s})$, and foot angles $\bar{\theta}^{1 \cdots N}(t, \mathbf{h})$
3: **end for**
4: **for** $t \in [1 \cdots T]$ **do**
5:    solve $\dot{\mathbf{q}}_t$ for contact velocity constraints.
6: **end for**
7: **for** $t \in [1 \cdots T]$ **do**
8:    solve $\ddot{\mathbf{q}}_t$ for contact acceleration constraints.
9: **end for**
10: **for** $t \in [1 \cdots T]$ **do**
11:    solve $\boldsymbol{\tau}_t$, $\hat{\mathbf{f}}_t^j$ for equations of motions.
12: **end for**
13: **return** $\sum_t |\boldsymbol{\tau}_t|^2$

IK problem is formulated as:

$$\mathbf{q}_t = \underset{\mathbf{q}_t}{\operatorname{argmin}} |\mathbf{p}^C(\mathbf{q}_t, \mathbf{d}) - \mathbf{p}_t^C|^2 + |\mathbf{r}^C(\mathbf{q}_t, \mathbf{d}) - \mathbf{r}_t^C|^2$$

$$\text{s.t. } \mathbf{p}^i(\mathbf{q}_t, \mathbf{d}) = \mathbf{p}_t^i \qquad \forall i \ \text{ if } c_t^i = 1$$
$$\mathbf{p}^i(\mathbf{q}_t, \mathbf{d}) = \bar{\mathbf{p}}^i(t, \mathbf{s}) \ \forall i \ \text{ if } c_t^i = 0 \qquad (3)$$
$$\theta^i(\mathbf{q}_t, \mathbf{d}) = \bar{\theta}^i(t, \mathbf{h}) \ \forall i \leq N$$

where $\mathbf{p}^C$, $\mathbf{r}^C$, $\mathbf{p}^i$, and θ^i define the position of center of mass, the global orientation, the foot position of the ith leg, and the foot angle of the ith leg for the given lengths $\mathbf{d}$ using forward kinematics. For two-link legs, joint angles have a unique solution under the assumption that the knee must be bent in a particular direction. For three-links, the foot angle trajectory $\bar{\theta}^i$ will remove the ambiguity of the solution.

Then we solve for the joint velocity $\dot{\mathbf{q}}$ to make sure that the foot does not penetrate the ground.

$$\dot{\mathbf{q}}_t = \underset{\dot{\mathbf{q}}_t}{\operatorname{argmin}} |\dot{\mathbf{q}}_t - \hat{\dot{\mathbf{q}}}_t|$$
$$\text{s.t. } \mathbf{J}_t^i \dot{\mathbf{q}}_t = 0 \ \forall i \ \text{ if } c_t^i = 1 \qquad (4)$$

where $\hat{\dot{\mathbf{q}}}$ is the target joint velocity calculated using finite difference and $\mathbf{J}^i$ is the Jacobian matrix of ith leg.

Similarly, we solve the joint acceleration $\ddot{\mathbf{q}}_t$ to hold contact non-penetration conditions.

$$\ddot{\mathbf{q}}_t = \underset{\ddot{\mathbf{q}}_t}{\operatorname{argmin}} |\ddot{\mathbf{q}}_t - \hat{\ddot{\mathbf{q}}}_t|$$
$$\text{s.t. } \mathbf{J}_t^i \ddot{\mathbf{q}}_t + \dot{\mathbf{J}}_t^i \dot{\mathbf{q}}_t = 0 \ \forall i \ \text{ if } c_t^i = 1 \qquad (5)$$

where $\hat{\ddot{\mathbf{q}}}$ is the joint acceleration calculated using finite differences.

Finally, we find the joint torques $\boldsymbol{\tau}_t$ and fullbody contact forces $\hat{\mathbf{f}}_j$ that minimize the squared sum of joint torques and satisfy the equations of motion.

$$\boldsymbol{\tau}_t, \hat{\mathbf{f}}_t^1, \cdots \hat{\mathbf{f}}_t^N = \underset{\boldsymbol{\tau}_t, \hat{\mathbf{f}}_t^1, \cdots \hat{\mathbf{f}}_t^N}{\operatorname{argmin}} |\boldsymbol{\tau}_t|^2$$

$$\text{s.t. } \mathbf{M}(\mathbf{q}_t)\ddot{\mathbf{q}}_t + \mathbf{C}(\mathbf{q}_t, \dot{\mathbf{q}}_t) + \sum_j \mathbf{J}_t^{j^T} \hat{\mathbf{f}}_t^j = \begin{bmatrix} 0 \\ \boldsymbol{\tau}_t \end{bmatrix} \qquad (6)$$

where j is the index of the link in contact, which can be in contact at both the toes and heel. Note that the above equations do not explicitly have terms to match the desired contact forces $\mathbf{f}_t^i$ that are calculated from the motion optimization stage with fullbody contact forces $\hat{\mathbf{f}}_t^j$. However, this stage will result in similar contact forces because we try to realize the desired COM trajectory given by the motion optimization stage.

After solving the entire motion for the given parameters $\mathbf{d}$, $\mathbf{s}$, and $\mathbf{h}$, Algorithm 1 returns the squared sum of the joint torques for the given design parameters.

$$f_{design}(\mathbf{d}, \mathbf{s}, \mathbf{h}) = \sum_t |\boldsymbol{\tau}_t|^2 \qquad (7)$$

Equations (4) - (6) are solved using Quadratic Programming (QP). If one of the equations fails to find a feasible solution, the function returns a high penalty.

V. RESULTS

We tested the proposed framework to design optimized monopod and quadruped robots simulated in the 2D sagittal plane. We render these optimized robots in 3D in figures and videos only for visualization purposes. In the motion optimization stage, the number of discretized frames T is set to 20 with a 0.025s time step. We set the maximum number of iterations for the SQP solver to 500, and solving takes about 15 minutes. In the design optimization stage, we set the number of spline knobs K for the foot angle trajectories as 4. For the CMA-ES algorithm, we set the number of parents μ and offspring λ as 16 and 32, respectively, and it takes around 2 hours for 100 iterations. All the results were produced on a single core of Intel Core i7-3770 3.40GHz CPU.

A. Monopod Robot

The input task for monopod robots is jumping. The goal of jumping is to reach a COM height of 1.0m. The total mass of the robot is set to 1.0kg, and the mass of a single motor is 0.05kg. The motors are located at joints and connected by aluminum bars (density: 2.7g/cm^3) with a 1cm^2 cross section. The length of these links can range from 9cm to 90cm. The resulting designs and motions are illustrated in Fig. 6, and the data can be found in Table I.

The optimal design of the two-link legged jumping robot has a shorter thigh and a longer shin, while the three-link legged robot has three links of almost equal length. Both designs try to maintain short moment arms created during motion. The folding structure of the three-link leg allows the robot to have shorter moment arms than the two-link legged robot, and results in a motion which is 46% more torque efficient.
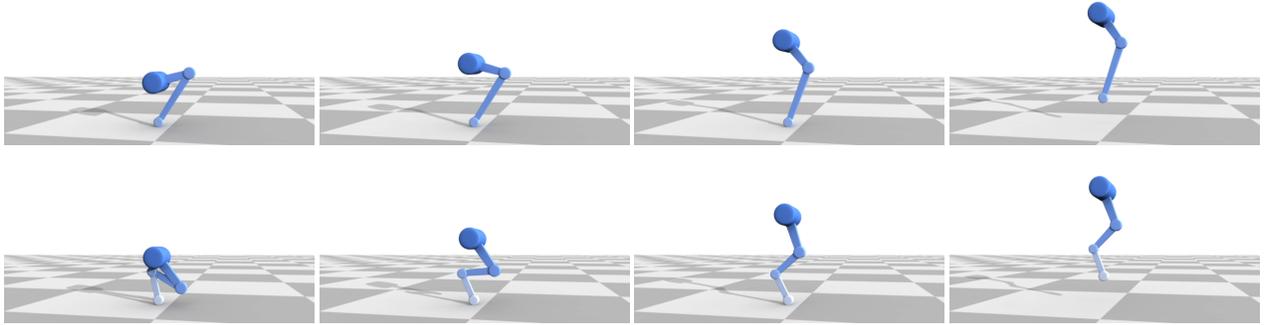
Fig. 6.   Optimized designs and motions for jumping monopod robots: (Top) A two-link legged monopod (Bottom) A three-link legged monopod

TABLE I

TASK AND OPTIMAL LINK LENGTHS FOR MONOPODS

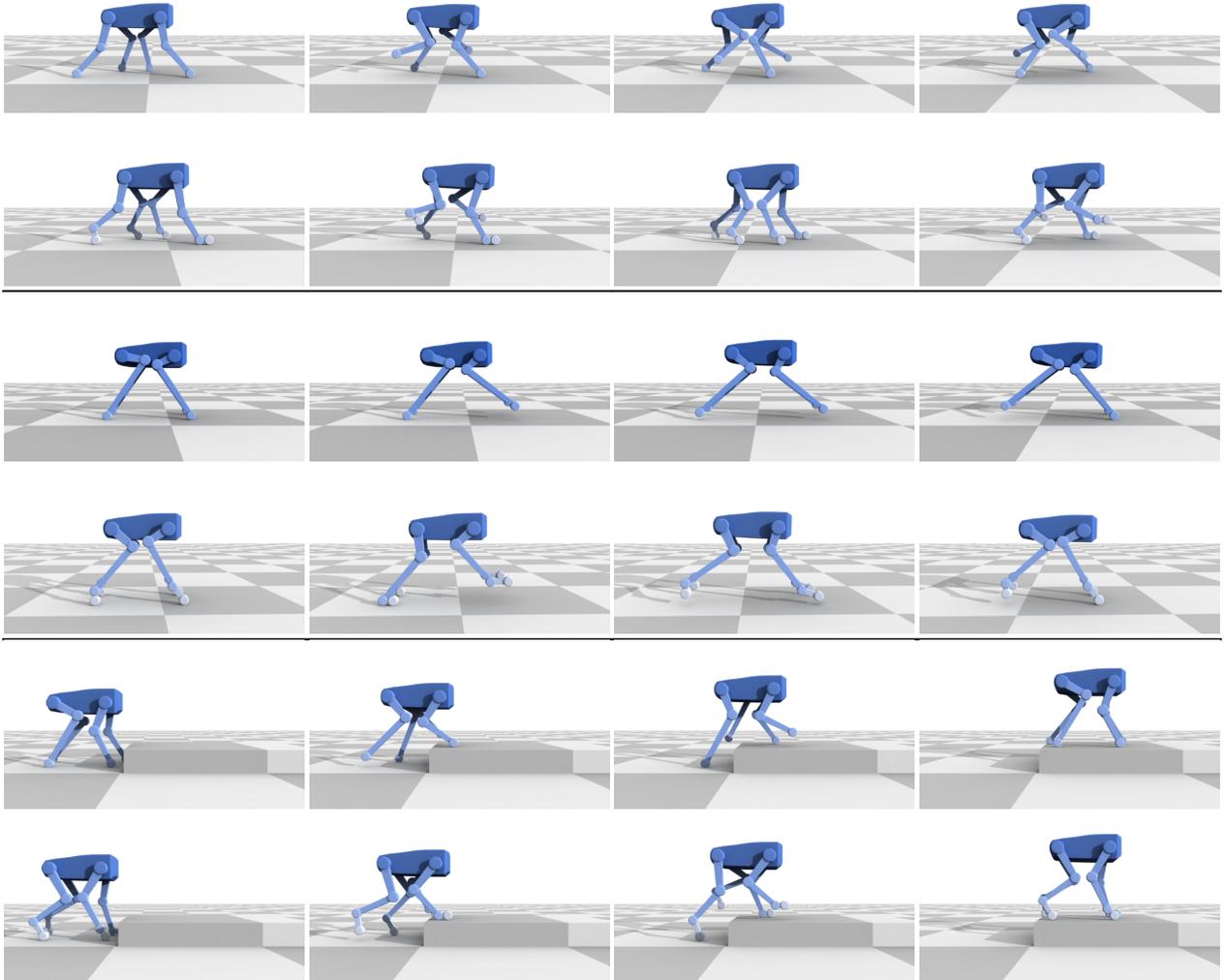|	Task	Robot	Leg			Objective
Name	Target Height (m)	# Links / Leg	Thigh (m)	Shin (m)	Foot (m)	Cost (N²m²)
Jump	1.0	2	0.304	0.511	-	57.5
Jump	1.0	3	0.339	0.294	0.254	30.9

Fig. 7.   The optimized designs and motions for quadrupeds: (First Row) The two-link legged robot for walking (Second Row) The three-link legged robot for walking (Third Row) The two-link legged robot for bounding (Fourth Row) The three-link legged robot for bounding (Fifth Row) The two-link legged robot for stair climbing (Sixth Row) The three-link legged robot for stair climbing

Task			Robot	Rear Leg			Front Leg			Objective
Name	Speed (m/s)	Gait	# Links / Legs	Thigh (m)	Shin (m)	Foot (m)	Thigh (m)	Shin (m)	Foot (m)	Cost (N^2m^2)
Walk	1.4	Trot	2	0.292	0.342	-	0.273	0.358	-	1.61×10^4
Walk	1.4	Trot	3	0.293	0.308	0.118	0.349	0.307	0.090	2.38×10^4
Bound	1.0	Bound	2	0.222	0.649	-	0.183	0.587	-	4.15×10^4
Bound	1.0	Bound	3	0.243	0.521	0.090	0.210	0.458	0.162	5.30×10^4
Stair	0.9	Trot	2	0.264	0.536	-	0.2904	0.353	-	4.15×10^4
Stair	0.9	Trot	3	0.336	0.385	0.169	0.329	0.345	0.090	3.75×10^4

B. Quadruped Robot

We also applied our framework to the optimization of quadruped robot designs. The input structure of a quadruped robot can have either two-link or three-link legs (Fig. 1), and the left-right symmetry is assumed. The total mass of the robot is set to 30.0kg and each motor mass is 0.7kg. The motors are connected by aluminum bars with a $4cm^2$ cross section, and a length that can range from 10cm to 80cm. In addition, we made a few assumptions to remove ambiguous solutions. First, knees are always bent inwards, and the ankles are always bent so that the feet point forward. We also fixed the locations of the hips with horizontal offsets of ±25cm from the robot's root. We also fixed the distance between the front and rear hip joints to 50cm and between the left and right hip joints to 25 cm to constrain the size of the base link.

The input tasks are walking (walk), a bounding gait (bound), and stair climbing (stair). The target speeds are 1.4m/s, 1.0m/s, and 0.9m/s, respectively. All tasks assume a trotting gait, except the bounding gait task (Fig. 2). The desired foot clearance height h_{max} is 20cm for walking, and 10cm for bouding. The step in the stair climbing task is 20cm high. Each task describes one cycle of locomotion with 20 frames, except for the the stair climbing task which comprises two cycles with 40 frames total.

We optimized the designs of two-link and three-link legged quadrupeds for all tasks. The optimized designs and motions are illustrated in Fig. 7, and the data is presented in Table II. For the walking task, the optimal designs have thighs and shins of relatively equal lengths, which produces the advantage of a larger workspace, therefore providing the ability to take longer steps. For the bounding gait, the optimal designs have very short thighs for applying large forces at feet. For the stair climbing task, the framework produces a design that has longer rear legs, especially for the three-link robot.

In general, the optimal designs indicate that three-link legged quadrupeds require more torque than two-link legged quadrupeds because three-link legs are heavier than two-link legs due to an additional servo and link. However, three-link legs show slightly better performances over two-link legs (near 10%) for the stair climbing task. For this task, the robot needs to exert large forces when the legs are bent. In this scenario, three-link legs are favorable, which is also the case for the jumping monopod.

For two-link legged quadruped walking, the optimal ratios of the rear and front leg links are 1.17 and 1.31, respectively.

These ratios are defined as d_{shin}/d_{thigh} for a given leg. We tested the optimality of this design by applying quadruped designs with higher and lower link ratios to the same task. For this comparison, the front and back legs are kept symmetric. For each test ratio, we reran the optimization to get the best cost function for the given quadruped. We plot these cost values with respect to link ratio in Fig. 8. The plot shows an optimal ratio between 1.0 and 1.5, which is similar to our optimal design. In the worse test case, having a short shin (ratio: 0.33) produces a cost value about three times higher than that of the optimal solution.

	Two-Link Legged Quadruped Design			
Task	Base	Walk	Bound	Stair
Walk	1.65×10^4	1.61×10^4	3.16×10^4	1.79×10^4
Bound	5.31×10^4	4.95×10^4	4.15×10^4	4.71×10^4
Stair	4.62×10^4	4.42×10^4	9.71×10^4	4.15×10^4

	Three-Link Legged Quadruped Design			
Task	Base	Walk	Bound	Stair
Walk	2.51×10^4	2.38×10^4	5.20×10^4	3.13×10^4
Bound	6.89×10^4	7.63×10^4	5.30×10^4	7.38×10^4
Stair	4.35×10^4	4.61×10^4	7.02×10^4	3.75×10^4

We applied the optimal design for each task to the other remaining tasks to verify each design's optimality. However, directly applying the optimized link lengths, shown in Table II, is not a fair comparison because the various tasks require different ranges of motion. Instead, we take only the ratio of the link lengths from the optimal designs, and reoptimize to find the other parameters $(\mathbf{d}, \mathbf{s}, \mathbf{h})$ for each
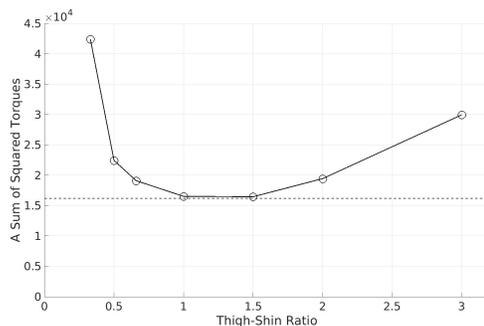

Fig. 8. For two-link legged quadruped walking we plot cost values with respect to thigh-shin ratio. The dashed horizontal line represents the cost of the optimal solution (ratio: 1.17 and 1.31).

task. The baseline design has a $thigh : shin : foot$ ratio of $3 : 3 : 1$, which approximates the average leg link ratio of a German shepherd [16]. The performance comparison of the optimal designs for all tasks is presented in Table III. In general, the optimal designs perform 10% to 20% better than other designs on the task for which they were optimized. However, a few extreme designs require more than twice the torque of the optimal design for a given task, e.g. when the two-link legged bounding-gait design is used for walking and stair climbing.

C. Limitations and Future Work

To perform our analysis, we introduced several modeling simplifications. First, the motion trajectory used to specify each task is generated by considering only the centroidal dynamics and the locations of the end effectors. This simplification allows us to more easily compare different robot designs against each other. However, we note that better results are likely to be obtained by concurrently or iteratively optimizing the full motion trajectory and a robot's morphological features. These approaches will allow us to plan optimal motions without collisions, which is not explicitly modeled in the current implementation for simplification. This is a direction we plan to investigate thoroughly in future work.

Our work focused on simplified dynamics of planar robots. For future consideration, we plan on extending our framework to 3D. This change will increase the parameter space by having new free variables, such as rotational axes or servo offsets. It will also introduce additional concerns, such as lateral balancing.

Another simplification is the prescribed structure of the robot. A quadruped's structure, such as its knee-bending directions or the number of links in a given leg, affects the capabilities and performance of the robot significantly. For this reason, current quadruped robots exhibit diverse configurations, as seen throughout the versions of Boston Dynamics' quadruped robots [17].

Each of the designs are currently optimized for a single task. As indicated by our preliminary study, robots designed for one task can perform other tasks as well, but their performance is inferior, sometimes quite significantly. Nevertheless, robots are often required to be versatile. We will therefore extend our method to optimize morphological features for multiple tasks.

The optimization criterion we have used in this work quantifies the performance with which motor tasks are executed. Strategies observed in nature, such as walking with straight legs, emerge automatically. Nevertheless, robustness is equally important, and singular configurations can be more vulnerable to sensor or actuator noise. As an avenue for future work, we therefore plan to explicitly incorporate terms that quantify robustness in the optimization process.

VI. CONCLUSION

We developed a mathematical framework to automate the design of legged robots for specific locomotion-based tasks. With the objective of minimizing joint torques while performing these tasks, our framework is able to co-design a robot's configurations and associated full body motions. To make this challenging design problem tractable, we introduced a number of simplifying assumptions. Under these assumptions, we conducted a set of experiments on monopod and quadruped robots performing various tasks such as jumping, walking, bounding, and climbing stairs.

Our findings show that the cost function varies substantially with the design parameters, indicating the need for careful finetuning of a robot's morhpological features. For example, we have found that using an optimized design with two links per leg leads to better performance for simple walking tasks, while optimized three link leg morphologies are better for jumping and walking over variable terrain.

REFERENCES

[1] A. Crespi, K. Karakasiliotis, A. Guignard, and A. J. Ijspeert, "Salamandra Robotica II: An amphibious robot to study salamander-like swimming and walking gaits," *IEEE Transactions on Robotics*, 2013.
[2] S. Seok, A. Wang, M. Y. Chuah, D. J. Hyun, J. Lee, D. M. Otten, J. H. Lang, and S. Kim, "Design Principles for Energy-Efficient Legged Locomotion and Implementation on the MIT Cheetah Robot," *IEEE/ASME Transactions on Mechatronics*, 2014.
[3] K. Graichen, S. Hentzelt, A. Hildebrandt, N. Kärcher, N. GaiBert, and E. Knubben, "Control design for a bionic kangaroo," *Control Engineering Practice*, 2015.
[4] D. Kuehn, F. Bernhard, A. Burchardt, M. Schilling, T. Stark, M. Zenzes, and F. Kirchner, "Distributed computation in a quadrupedal robotic system," *International Journal of Advanced Robotic Systems*, 2014.
[5] C. J. Paredis and P. K. Khosla, "An Approach for Mapping Kinematic Task Specifications into a Manipulator Design," *International Conference on Advanced Robotics*, 1991.
[6] M. Ceccarelli and C. Lanni, "A multi-objective optimum design of general 3R manipulators for prescribed workspace limits," *Mechanism and Machine Theory*, vol. 39, no. 2, pp. 119–132, 2004.
[7] E. Van Henten, D. Vant Slot, C. Hol, and L. Van Willigenburg, "Optimal manipulator design for a cucumber harvesting robot," *Computers and Electronics in Agriculture*, vol. 65, pp. 247–257, 2009.
[8] G. Yang and I. M. Chen, "Task-based optimization of modular robot configurations: minimized degree-of-freedom approach," *Mechanism and Machine Theory*, vol. 35, no. 4, pp. 517–540, 2000.
[9] S. G. Kim and J. Ryu, "New dimensionally homogeneous jacobian matrix formulation by three end-effector points for optimal design of parallel manipulators," *IEEE Transactions on Robotics and Automation*, 2003.
[10] J.-F. Collard, P. Fisette, and P. Duysinx, "Contribution to the Optimization of Closed-Loop Multibody Systems: Application to Parallel Manipulators," *Multibody System Dynamics*, vol. 13, pp. 69–84, 2005.
[11] Y. Yun and Y. Li, "Optimal design of a 3-PUPU parallel robot with compliant hinges for micromanipulation in a cubic workspace," *Robotics and Computer-Integrated Manufacturing*, 2011.
[12] K. C. Olds, "Global Indices for Kinematic and Force Transmission Performance in Parallel Robots," *IEEE Transactions on Robotics*, vol. 31, pp. 494–500, 2015.
[13] C. D. Jung, W. J. Chung, J. S. Ahn, M. S. Kim, G. S. Shin, and S. J. Kwon, "Optimal mechanism design of in-pipe cleaning robot," *IEEE International Conference on Mechatronics and Automation*, 2011.
[14] D. Kim, H. Hong, H. S. Kim, and J. Kim, "Optimal design and kinetic analysis of a stair-climbing mobile robot with rocker-bogie mechanism," *Mechanism and Machine Theory*, 2012.
[15] N. Hansen, S. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es)," *Evolutionary Computation*, 2003.
[16] *FCI-Standard, http://fci.be/Nomenclature/Standards/166g01-en.pdf*.
[17] *BostonDynamics, https://www.youtube.com/user/BostonDynamics/*.