Exploring Nonintuitive Optima for Dynamic Locomotion

Guillaume Bellegarda, Nihar Talele and Katie Byl

Abstract— **This paper explores controllability and efficient locomotion of a two-link planar robot balancing on a single, unactuated wheel. By actuating an internal degree of freedom, the model can indirectly produce ground reaction forces yielding net accelerations and decelerations, to achieve locomotion. As with classic humanoid locomotion, this toy system is particularly challenging to control due to the need to balance continuously while controlling forward locomotion speed. However, unlike typical legged or rolling locomotion solutions, it is not immediately obvious how best to exploit actuation, internal reconfigurations, and motions to produce and control forward velocity along the ground, providing a useful benchmarking system for exploring novel optimization techniques. We use a direct collocation optimization framework to study this toy system, both to achieve a range of feasible locomotion solutions for nonintuitive dynamic robot models, and to investigate optimization of physical robot parameterizations, in the sense of improving locomotion efficiency. The framework and example presented throughout are designed with an aim toward bridging the gap between non-intuitive, data-driven optimization and model-based methods for design and control of underactuated and dynamically-stable locomotion.**

I. INTRODUCTION

Balance and dynamic locomotion are fundamental challenges that make humanoid locomotion uniquely challenging. Our goal in this paper is to develop tools that can optimize both the mechanical design of a highly dynamic robot and optimal motion trajectories, simultaneously. We hypothesize that simple models, when chosen with care, are extremely useful for identifying fundamental principles, and also that many interesting phenomena for producing balance and locomotion (in particular, through creative use of internal degrees of freedom) may yet remain undiscovered. Here, we focus on a simple yet unusual model, which includes the challenge of underactuated balance and potential use of internal degrees of freedom without the additional complication of (dissipative/stabilizing) collision events, toward investigating novel dynamic phenomena for locomotion.

Much work has been done in recent years to optimize locomotion control for underactuated systems. In this work, we are particularly interested in methods with potential to explore and reveal non-intuitive solutions, to use computing power to improve performance. A few examples are given below.

This work is funded in part by NSF NRI award 1526424 and in part by an NSF CAREER award (CMMI 1255018).

[1]Guillaume Bellegarda, Nihar Talele and Katie Byl are with the Robotics Laboratory, Dept. of Electrical and Computer Engineering, University of California at Santa Barbara (UCSB) gbellegarda@umail.ucsb.edu, nihar@umail.ucsb.edu, katiebyl@ucsb.edu

Within our group, we have explored computationally efficient mesh-based methods to produce agile bounding gait sequences [1] and to optimize for robustness of biped gaits on rough terrain with uncertainty [2], [3]. These works employ switching between a limit set of low-level controllers, with the advantage of enabling brute force searches for optimal switching policies to negotiate complex terrain. However, they require intuitive parameterization of the low-level controllers, inspiring a need for more automated gait generation.

As an example of automation of gait generation, Xi et al. demonstrate an approach to produce optimally energy efficient gaits without a preordained ground contact sequence, using a locomotion model with a floating base and complementarity conditions to assure ground contact constraints are obeyed, and using a direct collocation framework for optimization [4]. In a similar spirit, Mordatch et al. solve for non-gait-specific biped locomotion trajectories, using compact criteria (e.g., a desired speed, to be achieved at minimal energetic cost) and temporary fictional external forces that are systematically eliminated during optimization convergence to a physically-realizable solution [5].

Various other research efforts have focused on learning and optimizing trajectories and gaits for robot motion planning, including techniques such as (stochastic) policy gradient reinforcement learning [6], [7], Bayesian optimization for learning gaits under uncertainty [8], and continuous control with deep reinforcement learning [9].

Deep learning has also provided routes toward optimally planning for physically realizable motions of robotic systems, as highlighted by the range of locomotion demonstrated within a popular Google DeepMind AI locomotion video associated with [10]. These animations include examples on seemingly non-optimal – or at least non-intuitive – motions. For example, upper-body limbs sometimes flail dramatically, although they never contact the environment. While these results are definitely intriguing, there is arguably a practical desire to know whether such nonintuitive motions can be verified to have a specific (and significant) beneficial physical function or not. As a preview of our results, our work produces similarly "suspicious-looking" motions, which can be verified to assist in balance during efficient locomotion for our simple, planar model.

In this paper, we take direct inspiration from the work done in [11], [12], and [13], where full body control for underactuated systems is achieved via trajectory optimization and stabilization under constraints. [11] in particular introduces an algorithm (DIRCON) that extends the direct collocation method, incorporating manifold constraints to produce nominal trajectories with third-order integration accuracy.

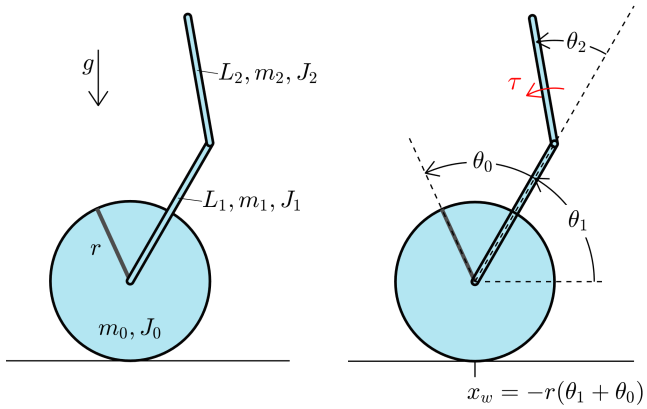

Fig. 1. The Acrollbot: a two-link acrobot on a passive wheel.

The system we study in this paper, shown in Figure 1, resembles the two-link acrobot model, except that it is mounted on a passive wheel that can roll on the ground instead of being mounted to a (passive) fixed pivot as in the classic version of the acrobot model [14]. Conceptually, the model is similar to a person on a unicycle, except that the unicycle includes no pedals or other direct actuation at its axis, and locomotion control must be achieved indirectly, by moving various joints to exploit passive damping at the axle and/or forces generated indirectly at the ground contact due to the no-slip assumption for the wheel.

The rest of this paper is organized as follows. Section II presents the Acrollbot system with equations of motion and comments on controllability. Section III presents the optimization framework used to develop motions for the system, as well as to find optimal parameters. Resulting motions, system parameters, comparisons to "normal" systems, and parameter sensitivity for various "optimal" systems are discussed in Section IV, and Section V concludes the paper with comments on the accompanying submitted video clip, and a discussion of ongoing and future work.

II. Acrollbot System

A. Acrollbot Model

Our mathematical model, which we call an "acrollbot", can best be described as an acrobot [14] mounted on a passive wheel. It is shown in Figure 1. The base of an acrobot is attached to the wheel at a "free axle", meaning there is no direct actuation between the wheel and the L_1 link. The only torque input, τ, is applied directly between the L_1 and L_2 links, as shown at right in Fig. 1. The wheel is modeled as a solid disk with mass m_0, radius r, and inertia $J_0 = \frac{1}{2}m_0 r^2$. Link 1 is modeled as a slender rod with mass m_1, length L_1 (center of mass [COM] at $L_{c1} = \frac{1}{2}L_1$), and inertia about COM of $J_1 = \frac{1}{12}m_1 L_1^2$. Link 2 is modeled as a slender rod with mass m_2, length L_2 (COM at $L_{c2} = \frac{1}{2}L_2$), and inertia $J_2 = \frac{1}{12}m_2 L_2^2$. θ_1 is the counter-clockwise (CCW) angle measured from the positive x-axis to link 1, analogous to link 1 of an acrobot. θ_0 is the wheel angle, measured CCW relative to θ_1, and θ_2 is the angle of the upper (second/last) link, similar to link 2 of an acrobot, measured CCW relative to θ_1.

The equations of motion are derived via a Lagrangian approach and are summarized in the subsection that follows.

B. Equations of Motion

The equations of motion for the system can be written in matrix form as:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \Xi \tag{1}$$

where $q = [\theta_0, \theta_1, \theta_2]^\mathsf{T}$. The nonconservative torques on the righthand side include viscous (linear) damping at the wheel axle, proportional to the relative velocity of the wheel ($\dot{\theta}_0$), and a torque appied between links 1 and 2 (u); i.e., Ξ is (3x1) with:

$$\Xi_1 = -f_0\dot{\theta}_0, \qquad \Xi_2 = \quad 0, \qquad \Xi_3 = u \tag{2}$$

The torques due centrifugal and Coriolis effects are:

$$C(q,\dot{q})\dot{q} = [C_1, C_2, C_3]^\mathsf{T} \tag{3}$$

where:

$$\begin{aligned}
C_1 =& r m_2 L_{c2} \cos(\theta_1+\theta_2)\left(\dot{\theta}_1+\dot{\theta}_2\right)^2 + \ldots \\
& r\left(m_1 L_{c1} + m_2 L_1\right)\cos(\theta_1)\,\dot{\theta}_1^2 \\
C_2 =& C_1 - m_2 L_1 L_{c2}\sin(\theta_2)\left(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2\right) \\
C_3 =& m_2 L_1 L_{c2}\sin(\theta_2)\,\dot{\theta}_1^2
\end{aligned}$$

with COM locations at L_{c1} and L_{c2}, as defined in Section II-A. Similarly, the gravity dependent terms are $G(q) = [G_1, G_2, G_3]^\mathsf{T}$, where

$$\begin{aligned}
G_1 =& 0 \\
G_2 =& G_3 + \left(m_1 L_{c1} + m_2 L_1\right) g\cos(\theta_1) \\
G_3 =& m_2 L_{c2} g\cos(\theta_1+\theta_2).
\end{aligned}$$

Finally, the configuration-dependent terms in inertia matrix $D(q)$ are given below, where D_{mn} corresponds to element $D(m,n)$ of the matrix:

$$\begin{aligned}
D_{11} =& J_0 + (m_0+m_1+m_2)\,r^2 \\
D_{12} =& D_{11} + D_{13} + L_1 m_2 r\sin(\theta_1) + L_{c1} m_1 r\sin(\theta_1) \\
D_{13} =& L_{c2} m_2 r\sin(\theta_1+\theta_2) \\
D_{21} =& D_{12} \\
D_{22} =& D_{11} + 2D_{13} + J_1 + J_2 + L_1^2 m_2 + L_{c1}^2 m_1 + L_{c2}^2 m_2 + \ldots \\
& 2L_1 L_{c2} m_2\cos(\theta_2) + 2\left(L_1 m_2 + L_{c1} m_1\right) r\sin(\theta_1) \\
D_{23} =& J_2 + L_{c2}^2 m_2 + L_1 L_{c2} m_2\cos(\theta_2) + L_{c2} m_2 r\sin(\theta_1+\theta_2) \\
D_{31} =& D_{13} \\
D_{32} =& D_{23} \\
D_{33} =& m_2 L_{c2}^2 + J_2.
\end{aligned}$$

We can linearize this system about some equilibrium configuration, in which all angular velocities are zero and the center of mass of the system is aligned at the same x coordinate as the center of the wheel. We define the states of the system as:

$$X = \left[\theta_0, \theta_1, \theta_2, \dot{\theta}_0, \dot{\theta}_1, \dot{\theta}_2\right]^\mathsf{T} \tag{4}$$

and X_{eq} is some valid equilibrium configuration in which the system can balance (with zero velocity for each DOF). One example is a fully upright configuration:

$$X_{eq} = \left[q_{eq}^{\mathsf{T}}, \dot{q}_{eq}^{\mathsf{T}}\right]^{\mathsf{T}} = \left[0, \frac{\pi}{2}, 0, 0, 0, 0\right]^{\mathsf{T}} \tag{5}$$

Note that any choice is valid for the relative wheel angle, θ_0. Toward deriving state space equations, we can first write the linearized equations of motion in the following form:

$$M\ddot{q} + \beta\dot{q} + Kq = T = [0, 0, u]^{\mathsf{T}} \tag{6}$$

where $M = D(q_{eq})$, i.e., the inertia matrix $D(q)$ evaluated at the equilibrium configuration of interest. Terms due to $C(q,\dot{q})\dot{q}$ drop out when linearizing about zero velocity, so that matrix β contains only the viscous damping term at the wheel, while K is the Jacobian of $G(q)$ evaluated at $q = q_{eq}$. For X_{eq} as defined in Equation 5:

$$\beta = \begin{bmatrix} f_0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{7}$$

$$K = \begin{bmatrix} 0 & 0 & 0 \\ 0 & (-m_2 g L_1 - m_1 L_{c1} g - m_2 g L_{c2}) & (-m_2 g L_{c2}) \\ 0 & (-m_2 g L_{c2}) & (-m_2 g L_{c2}) \end{bmatrix} \tag{8}$$

Equation 6 can be written in terms of the state vector as:

$$\tilde{M}\dot{X} = \tilde{A}X + \tilde{B}u \tag{9}$$

$$\begin{bmatrix} I_{3x3} & 0 \\ 0 & M \end{bmatrix} \dot{X} = \begin{bmatrix} 0 & I_{3x3} \\ -K & -\beta \end{bmatrix} X + [0,0,0,0,0,1]^{\mathsf{T}} u, \tag{10}$$

and the linearized state space equations (i.e., for $\dot{X} = AX + Bu$) are then $A = \tilde{M}^{-1}\tilde{A}$, and $B = \tilde{M}^{-1}\tilde{B}$.

For example, linearizing about upright and using $g = 9.8$ (m/s^2), $m_0 = m_1 = m_2 = 1$ (kg), $L_1 = L_2 = 1$ (m), $r = 0.2$ (m), $J_0 = \frac{1}{2}m_0 r^2$, $J_1 = J_2 = \frac{1}{12}m_1 L_1^2$ and $f_0 = 0.3$ (Nm/(rad/s)):

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -71.736 & 19.992 & -7.152 & 0 & 0 \\ 0 & 24.696 & -14.112 & 1.872 & 0 & 0 \\ 0 & -32.928 & 48.216 & -3.096 & 0 & 0 \end{bmatrix} \tag{11}$$

$$B = \left[0, 0, 0, 10.32, -5.52, 15.36\right]^{\mathsf{T}} \tag{12}$$

The closed-loop poles of the linearized system can be set arbitrarily to regulate the system about some desired equilibrium point (e.g., via pole placement, or LQR control) if and only if the controllability matrix has full rank, i.e., rank 6 for our system (with 3 angles and 3 angular velocities as states). With non-zero wheel damping (f_0) in β, the controllability matrix for A and B has rank 5. When $f_0 = 0$, the the fourth column of A becomes $[1,0,0,0,0,0]^{\mathsf{T}}$ (with B and the remainder of A remaining unchanged), and the controllability matrix for the linearized system has rank 4.

C. Controllability

As summarized in Section II-B above, the full linearized system without friction is not controllable. However, with some damping friction at the passive wheel, the controllability matrix of the linearized system becomes rank 5. While this does not allow the system to be stabilized about some arbitrary x_w position for the wheel, an LQR controller using the 5 other states of the system allows it to stabilize about the vertical equilibrium position.

Additionally, using partial feedback linearization (PFL), it is also possible to control the system using only the acrobot-specific states, namely $[\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2]$. These states can be used to stabilize the system about the vertical equilibrium point, when damping is present at the wheel. Specifically, rank 5 of the controllability matrix here corresponds to a lack of ability to control the overall x_0 wheel location of the system, although velocity of the wheel can be driven to zero here. With no damping or friction at the wheel, the linearized system can still be stabilized in an upright configuration, although now neither x_0 nor dx_0/dt can be regulated at (i.e., driven to) zero.

III. Optimization Framework

A. Problem Setup

We discretize the dynamics and use direct collocaton along with backward Euler integration as suggested in [15] to generate trajectories for the walking motion. The problem is formulated as

$$\text{find} \quad q, \dot{q}, u \tag{13}$$

$$\text{such that} \quad D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = Hu \tag{14}$$

$$\phi(q) = 0 \tag{15}$$

where $q \in \mathbb{R}^n$ is the vector of generalized coordinates, $D(q) \in \mathbb{R}^{n\times n}$ is the mass inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ represents the centrifugal and Coriolis forces, $G \in \mathbb{R}^n$ contains gravitational forces, and $H \in \mathbb{R}^n$ is the input mapping. $\phi(q)$ is a vector of constraints: we assume no slipping of the wheel, and constraints are imposed to make sure that the normal reaction force at the point of contact is always positive.

The initial and end states ($[\theta_1, \theta_2, \dot{\theta}_0, \dot{\theta}_1, \dot{\theta}_2]^{\mathsf{T}}$) of any motion are also constrained to both be some (identical) vertical equilibrium point, to enable either a single motion or a repeated limit-cycle behavior, and we solve for the base of the wheel to move a certain distance d_{stride} in a fixed time interval through our framework. However, we note that the same framework can also find trajectories for arbitrary starting and ending system positions and velocities, if other limit cycles or continuing motions (when the system is not starting or ending at rest) are desired.

B. Cost Function

An important consideration in finding an optimal trajectory is: what exactly is most important to optimize? There are several important characteristics and metrics of the system,

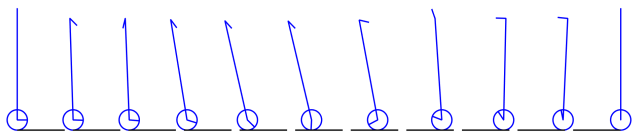

Fig. 2. Snapshots of the two-link acrollbot during locomotion.

and optimizing for different metrics will typically result in different solutions. We set up our cost function, J, as follows:

$$L = \left([\dot{\theta}_0, \dot{\theta}_1, \dot{\theta}_2]U\right)^2 = (\theta_2 u)^2 \tag{16}$$

since there is a single input, u; i.e., $U = [0, 0, u]^\mathsf{T}$. Then,

$$J = \frac{\int_0^t \sum_{i=1}^N \sqrt{L+\varepsilon}\ dt}{(m_0+m_1+m_2)gd_{stride}}. \tag{17}$$

We use $\varepsilon = 10^{-6}$ as a regularization term to help smooth the cost function as suggested in [16]. N is the number of sample points for the trajectory, and d_{stride} is the horizontal distance traversed by the center of the wheel.

C. Open System Parameters

Given a robotic system, researchers can control and optimize trajectories to make the system perform various objectives with high degrees of certainty. But how do we know that this given system is the best possible one? Does the robot's forearm have the optimal length and mass distributon for efficient control? What are the effects on the general potential of the system for robust and dynamic movements? These all lead to a more general question: what effects and consequences does changing system parameters have on metrics such as efficiency, robustness, and agility?

Motivated by this general question, we return to our acrollbot system. We leave not only the states/trajectories of each joint as open parameters in the system, but also the system parameters themselves: $r, L_1, L_2, m_0, m_1, m_2$, with $r, L_1, L_2 \in [0.01, 1]$ meters, and $m_0, m_1, m_2 \in [0.01, 1]$ kilograms.

Leaving all of these parameters open and finding an optimal solution with respect to energy, for example, returns not only an optimal trajectory, but also the "best" robot design to use for that motion.

IV. Results

Figure 2 shows a typical solution found for optimal robot design and corresponding locomotion. Here, nearly all mass is concentrated as a small L_2 link. Unlike most inverted pendulum systems, such as a Segway or cart-pendulum system, the center of mass (COM) does not lean forward as the system accelerates to the left. Instead, the mass lags behind the wheel contact, essentially driving the system forward much like an off-balance ladder would slide on the ground. Stability is maintained through an unanticipated vertical oscillation of the top-most mass of link 2. We have simulated various solutions using MATLAB's ode45, and the oscillation stabilizes the system using the same phenomenon used in Kapitza's pendulum [17]; i.e., vertical

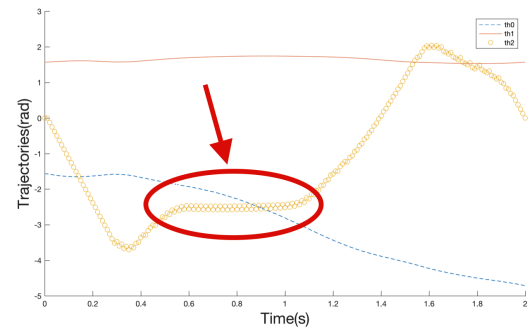

Fig. 3. Example optimal trajectory. The red circle and arrow highlight oscillations in the most distal link.

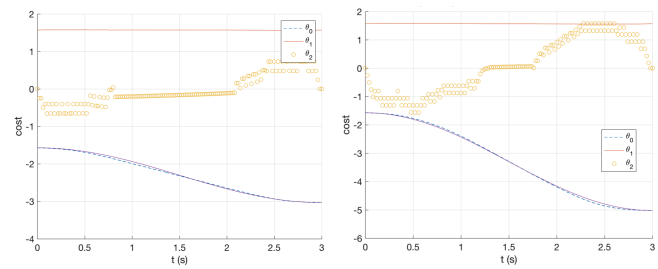

Fig. 4. Optimal joint trajectories to move the acrollBot system 1.0 meters (at left, CoT=0.029) vs 3.0 meters (right, CoT=0.054) in 3 seconds, to illustrate general behaviors. θ_1 remains nearly constant (about a vertical angle of $\frac{\pi}{2}$). A solid, purple line, which represents a half-cosine waveform, is overlaid on the dashed trajectory for θ_0, to illustrate the relative smoothness of overall wheel motion along the ground.

oscillation of a mass stabilizes an unstable pendulum. In our case, this oscillation can also be exploited to control acceleration or deceleration, depending on the offset angle of oscillation, and we used our optimal solutions to develop generalized control algorithms for locomotion, which are illustrated briefly within a video attachment for this paper.

For most locally-optimal solutions, L_2 is set near its minimal allowed value, with nearly all mass concentrated at this link. Figure 3 shows a typical trajectory result from optimization, when L_2 is constrained to be at least 0.1 [m]. Of note are oscillations, which suspiciously seem to correspond to the knot points of our direct collocation solution. After much investigation, oscillations are a repeatable phenomenon and can take on a range of frequencies. We have tested high-resolution simulations (ode45 set with 1E-8 tolerances) of feedback laws based on these motions, which stabilize and locomote the system, as expected.

A. Optimization Parameter "Typical" Results

In most cases, our locally-optimal solutions put most of the mass at the last link, and made this link (L_2) almost as short as possible. Figure 4 introduces the typical behaviors seen in these optimal trajectories. L_2 oscillates during both the initial acceleration period and final deceleration, and the overall motion of the wheel is rather smooth. (We overlay cosine waves on θ_0, wheel angle, in this figure, for comparison.)

Shown in Fig. 5 is a sample "optimal" trajectory for each of the joints of the system, along with the input

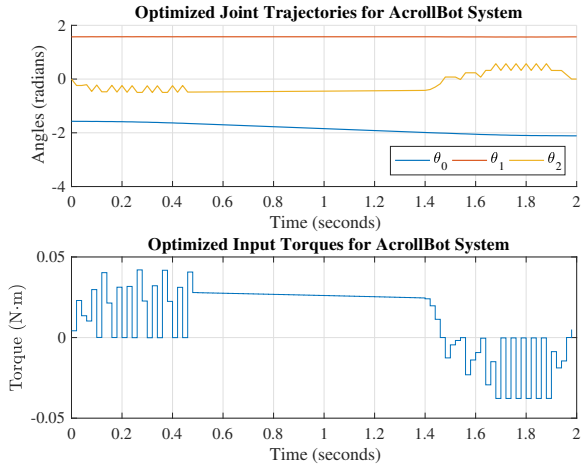

Fig. 5. Optimal joint trajectories to move AcrollBot system 0.3 meters in 2 seconds, where the corresponding optimal parameters are $m_0 = 0.01$ (kg), $m_1 = 0.01$ (kg), $m_2 = 1.0$ (kg), $r = 0.5537$(m), $L_1 = 0.8522$ (m), $L_2 = 0.0123$ (m), as well as the corresponding input torques at link 2 that results in the trajectories. Cost of Transport is 0.0432.

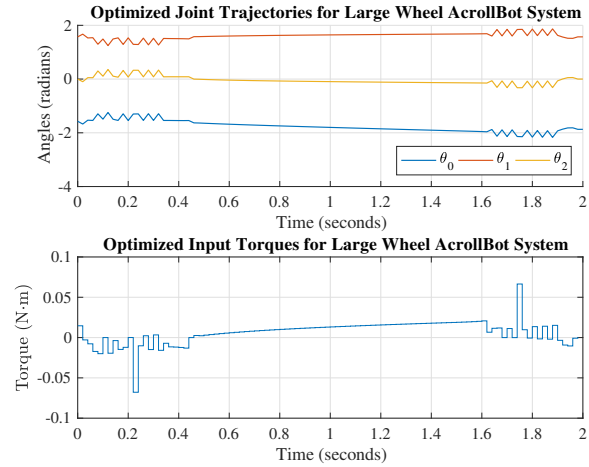

Fig. 6. Optimal joint trajectories to move large wheel acrollBot system 0.3 meters in 2 seconds, where the corresponding optimal parameters are $m_0 = 0.01$ (kg), $m_1 = 0.01$ (kg), $m_2 = 1.0$ (kg), $r = 1.0$(m), $L_1 = 0.0180$ (m), $L_2 = 0.1192$ (m), as well as the corresponding input torques at link 2 that results in the trajectories. Cost of Transport is 0.0197.

torques at link 2 that result in the system's motion. With masses and lengths constrained as described in Sec. III-C $[r, L_1, L_2, m_0, m_1, m_2 \in [0.01, 1]$, SI units], the system found the following "optimal" parameters: $m_0 = 0.01$ (kg), $m_1 = 0.01$ (kg), $m_2 = 1.0$ (kg), $r = 0.5393$ (m), $L_1 = 0.8099$ (m), $L_2 = 0.0137$ (m). The framework typically generates trajectories for parameters in approximately these ranges, with minimal mass on the wheel and link 1, maximal mass on link 2, an intermediate value in the range for r, large L_1, and minimal L_2. We observe results in these ranges for any values of distance or time, so the parameters are not tied to specific motion requirements. The trajectory input torques and resulting system reactions always follow the same oscillation patterns at the beginning and end of the trajectory, in the optimal cases. To start the motion, rapid oscillations in torque put energy into the system and provide enough momentum to get the system to roll in one direction, and the "opposite" reactions cause the system to slow down and stop at the desired distance. Between these beginning and ending oscillations, the torque stays approximately constant, simply transitioning and positioning link 2 from the midpoint of one set of oscillations to the other. The length of this transition period decreases as the desired lateral distance to move increases, placing an upper bound on the furthest distance the system can travel for a given total time interval.

B. Rare Optima: "Large Wheel, Short Links"

Although the local optima that the framework finds in repeated (and randomized) searches typically results in the ranges as mentioned in the preceding section, very occasionally returned is a lower COT solution consisting of a very large radius wheel $r = 1.0$ (m), with a minimal length such as $L_1 = 0.0180$ (m) and a lower value $L_2 = 0.1192$ (m), $(L_2 >> L_1)$. The associated cost of transport of this large wheel system is typically less than half that of the best, perhaps "more realistic" system previously discussed (0.0197 vs.0.0432 for the realistic system). The mass distribution stays the same for both of these systems. Physically, with such a large wheel and small internal links, there is no longer really a stabilization issue, but the constraints are still there to keep the system balanced. From comparing Fig. 5 and Fig. 6, the large wheel system requires fewer oscillations to start/stop the system, with one main peak toque at the beginning and end of the motion. The magnitudes have also switched signs, though both move the system in the same direction.

C. Flywheel

Since the optimization framework typically produces trajectories for system parameters with all of the mass located at m_2 and a minimal length L_2, we consider that perhaps a flywheel at the end of link 1 might be more optimal than a second link. From a controllability perspective, this would also help with system stabilization, since the flywheel is not constrained to a certain angle in order for the system to balance about an equilibrium point, as in the acrollbot system.

At first, we take the optimal parameters from the acrollbot as found by the optimization framework for m_0, m_1, m_2, r, L_1 and leave the inertia of the flywheel, J_2 as an open parameter. However, the trajectories resulting from these sets of parameters results in a much larger Cost of Transport for the flywheel system, no matter what value J_2 takes. Compared to minimum COT value of 0.0432 for the acrollbot system we found for $T = 2$ seconds, $d_{stride} = 0.3$ meters, the flywheel system returns trajectories costing 1.2750, which is very significantly worse.

Leaving all parameters open, however, the framework finds trajectories of lower cost than those of the optimal two-link acrollbot, as shown in Figures 7 and 8. The same constraints are set as before on the ranges for each parameter, except now we define a characteristic length $\widetilde{L}_2 \in [0.01, 1]$,

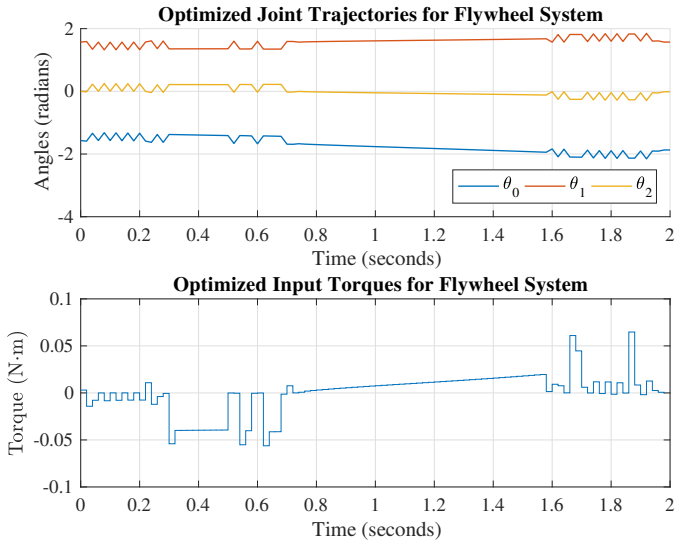

Fig. 7. Optimal joint trajectories to move flywheel system 0.3 meters in 2 seconds, where the corresponding optimal parameters are $m_0 = 0.01$ (kg), $m_1 = 0.01$ (kg), $m_2 = 1.0$ (kg), $r = 1.0$ (m), $L_1 = 0.0189$ (m), $\widetilde{L}_2 = 0.9999$ (m), as well as the corresponding input torques for the flywheel that results in the trajectories. Associated Cost of Transport is 0.0132.

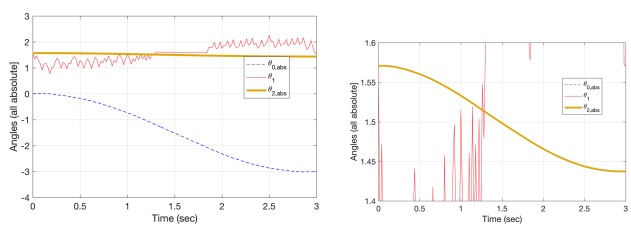

Fig. 8. Optimal trajectories for flywheel version of acrollbot, showing absolute angles. Here, L_1 oscillates while holding the flywheel out to the side, to offset the system COM. Both the wheel and flywheel rotation fairly smoothly. At right, the directions of rotation for the wheel and flywheel are in the same direction, which defies physical intuition! Additional simulations in MATLAB confirm this phenomenon, however. (See also supplemental video.)

to describe a flywheel centered at the end of link 1, with $J_2 = \frac{1}{2}m_2\widetilde{L}_2^2$. The optimal flywheel system parameters are $m_0 = 0.01$ (kg), $m_1 = 0.01$ (kg), $m_2 = 1.0$ (kg), $r = 1.0$ (m), $L_1 = 0.0189$ (m), $\widetilde{L}_2 = 0.9999$ (m), with a resulting COT of 0.0132. From a physical standpoint, the optimal system consists of a large-radius (but nearly massless) wheel, with a slightly offset from center, high-inertia flywheel. Leaving the range of $\widetilde{L}_2$ extremely large (on the order of 1E3 or more) does not seem to have a significant effect on the COT, though J_2 does tend toward a large value, given this range. This serves as inspiration for the next section of this paper, where we check the sensitivity of Cost of Transport to the system's optimal parameters found by the optimization framework. Specifically, how does COT vary as each of the available parameters is de-tuned from its optimal solution?

D. Parameter Sensitivity

As a sanity check, we repeatedly run the optimization many times with the model parameters fixed to particular values. Specifically, we keep 5 out of the 6 optimal parameters fixed and test across a range of values in a general neighborhood of the remaining optimal one, to record its effect on the Cost of Transport. Figure 9 shows such a plot for the acrollbot model. The parameter that has the greatest effect on the COT as it varies from its optimal value is L_2, whereas m_2, r, L_1 can vary without having any apparent effect on the COT. A longer L_2 means that the center of mass of the link is farther from the joint, so it is reasonable that it takes larger torques to control its movement. m_0 and m_1, which are always optimized to be the minimum value possible, also effect the COT as they increase in mass.

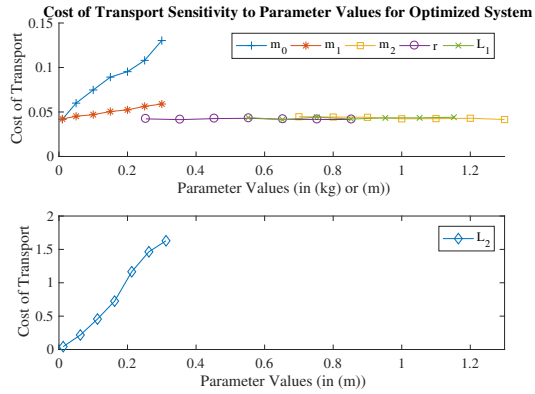

Fig. 9. Cost of Transport sensitivity to system parameters for the optimized acrollbot system with original parameters $m_0 = 0.01$ (kg), $m_1 = 0.01$ (kg), $m_2 = 1.0$ (kg), $r = 0.5537$ (m), $L_1 = 0.8522$ (m), $L_2 = 0.0123$ (m) for $T = 2, d_{stride} = 0.3(m)$. Increasing the length of L_2 even minimally has a significant effect on the COT. Increasing $m0$ or $m1$ also has some effect on the COT, though we see that the specific values of m_2, r, L_1 have minimal effect on the COT.

Figure 10 shows the Cost of Transport sensitivity for the flywheel model with a large radius wheel and a small flywheel. m_2, r, J_2 have almost no effect on the COT when offset from their optimal values, as found by the framework. Increasing L_1 from its minimal value, which puts the flywheel further from the center of the large wheel, increases the COT most. m_0 and m_1, which were again optimized to their minimums possible, also increase the COT as they increase, but at a slower rate than for L_1.

E. Comparison to a "Normal" (Hand-Designed) System

To put the trajectories and system parameters discovered via the optimization framework in perspective, we compare our optimal system with one that might be built in the real world as a first pass. Intuitively, one might pick L_1 and L_2 to be about the same, similar to a typical acrobot design. Specifically, we choose the following system parameters as a "typical" design choice: $m_0 = 0.2$ (kg), $m_1 = 0.4$ (kg), $m_2 = 0.4$ (kg), $r = 0.2$ (m), $L_1 = 0.4$ (m), $L_2 = 0.4$ (m). The total mass and height of this system is approximately the same as that of our optimal system. For these parameter choices, our optimization framework discovers trajectories with an associated Cost of Transport of 4.3090, which is on the order of 1E2 times worse than any trajectories returned with optimal parameters, emphasizing the importance of optimizing design parameters in tandem with motion trajec-

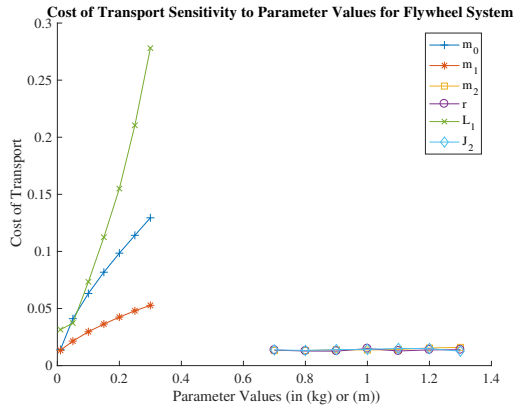

Fig. 10. Cost of Transport sensitivity to system parameters for the optimized flywheel-on-a-wheel system with original parameters $m_0 = 0.01$ (kg), $m_1 = 0.01$ (kg), $m_2 = 1.0$ (kg), $r = 1.0$ (m), $L_1 = 0.0189$ (m), $L_2 = 0.9999$ (m) for $T = 2, d_{stride} = 0.3(m)$. For this system, length L_1 is the most sensitive parameter, with respect to COT. Increasing $m0$ or $m1$ also has some effect on the COT, though we see that the specific values of m_2, r, J_2 have minimal effect on the COT.

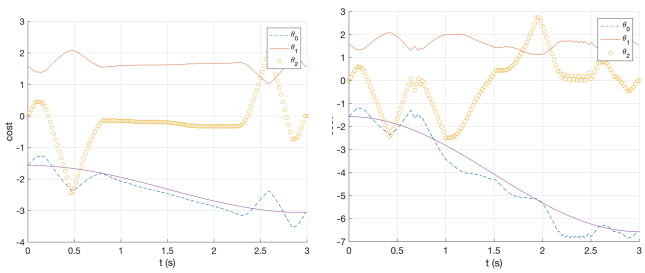

Fig. 11. Trajectories for hand-picked design parameters. Optimal motions are more extreme and exaggerated, and high-frequency oscillations in the last link are no longer observed. At left, a slower motion ($d = 0.3$(m) in $T = 3$(s)); at right, faster locomotion ($d = 1.0$(m) in $T = 3$(s)).

tories.

Figure 11 shows example trajectories for such a hand-picked system. Of note is the lack of high-frequency oscillations here. Instead, more exaggerated (i.e., higher amplitude), slower-frequency motions of the last link are used for simultaneous balance and locomotion.

F. Effects of Passive Damping on System

Clearly, an appropriately chosen passive mechanical spring would help generate the required oscillations for the system more efficiently. However, it is less obvious how damping might help or hurt the system. In general, damping seems likely to reduce efficiency, since energy is dissipated. And yet, damping at the passive wheel (f_0) increases the controllability of the linearized system, as discussed in Section II-C. Sweeping independently across values of f_0 and f_2 (damping in parallel with the actuator) for travel of $d = 0.3$ [m] in $T = 2$ [s], using a fixed set of parameters (r,L_1,L_2,m_0,m_1,m_2) that are optimal in the zero-damping case, we found a linear relationship between f_2 and the COT, while f_0 did not effect the COT very much across the range tested, as shown in Figure 12.

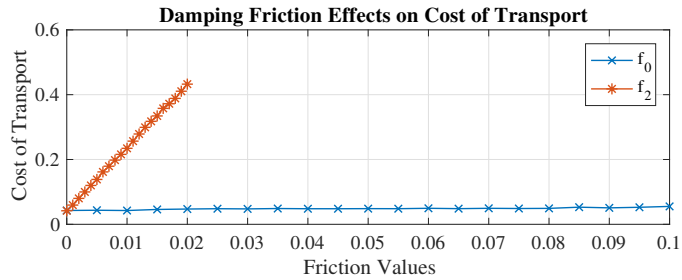

Fig. 12. Cost of Transport vs. friction for the optimized acrollbot system. Damping at the wheel has minimal effect on the COT, whereas increasing damping friction at link 2 where the input torques are present has a large effect on the COT.

V. DISCUSSION

A short mp4-format video is provided within the supplementary materials for this paper, which illustrates various motions described within the paper. Below, we highlight some details, following the order of presentation in the video.

First, note the basic elements of locomotion, which include two key aspects. (1) COM is offset in a direction opposite the net direction of acceleration. To remain balanced, most inverted pendulums instead bias COM toward the direction of acceleration ("leaning forward"), so the net ground reaction force (GRF) acts along the line of the pendulum. (2) Vertical oscillations stabilize the system. Next, when non-optimal design parameters are used, one can see the more exaggerated, bobbing motions required (illustrated in Fig. 11). Also, note that because most of optimal design solutions for the two-link acrollbot include a tiny L_2 value, close to 0.01(m), which is barely visible in video, we highlight the angle of this link in such cases by including a dashed, red line. For the acrollbot-with-flywheel, take care to note that L_1 is very tiny (see Sec. IV-C).

Finally, note that the simulations provided at the end (from 52 seconds onward) all involve control laws and dynamics simulations implemented using ode45 in MATLAB, as independent confirmation of some of the more surprising dynamic locomotion techniques we observe. Specifically, we can control forward and backward locomotion for both the "Typical" (Sec. IV-A) and "Flywheel" (Sec. IV-C) systems, and we even found an example where oscillations increase the basin of attraction for a simple biped walker, although investigating direct applications to biped locomotion in more depth is a topic for future work, outside the scope of this paper.

Several ongoing goals in humanoid locomotion directly motivate this work. In particular, while parameterizations (lengths and mechanical impedance properties) of a humanoid will clearly affect performance, better tools for optimizing such design parameters are needed. Also, optimization tools frequently suffer from local minima, in turn requiring appropriate initialization for useful results. By contrast, our framework allows us to optimize both design parameters (e.g., lengths and masses) and motion trajectories simultaneously, and it works with completely randomized

initial conditions (e.g., trajectories which initially violated dynamic constraints), which simplifies our search and in turn has allowed us to discover multiple local optima. Finally, methods based on deep learning or other data-driven approaches typically do not elucidate basic physics-based phenomena. In response, we have studied an intentionally non-intuitive yet simple model, which captures key aspects of balance and underactuation. By studying a simplified toy system, we have been successful in carefully analyzing and validating non-intuitive solutions, and we have discovered novel motion planning strategies that we, as human designers, would truly never have anticipated.

Current work within our group is successfully applying this framework to a 5-link biped model. Key future goals include improving optimization metrics, toward quantifying robustness and agility, in addition to the more easily-calculable energetic costs typically used, and better understanding of the trade-offs and principals at play in optimal humanoid locomotion, particularly as terrain properties, sensor noise and external disturbances (e.g., process noise) each vary.

ACKNOWLEDGMENT

The authors thank Andy Ruina and Matt Kelly for fruitful discussions on the dynamic phenomena observed.

REFERENCES

[1] K. Byl, T. Strizic, and J. Pusey, "Mesh-based switching control for robust and agile gaits," in *Proc. American Control Conference (ACC) [in press]*, 2017.

[2] C. O. Saglam and K. Byl, "Quantifying and optimizing robustness of bipedal walking gaits on rough terrain," in *Proc. Int. Symp. Robotics Research (ISRR)*, 2015.

[3] ——, "Robust policies via meshing for metastable rough terrain walking," in *Proc. Robotics: Science and Systems X (RSS X)*. RSS, 2014, p. Paper 49.

[4] W. Xi, Y. Yesilevskiy, and C. D. Remy, "Selecting gaits for economical locomotion of legged robots," *Int Journal of Robotics Research [online first]*, Nov 2015, doi 10.1177/0278364915612572.

[5] I. Mordatch, J. M. Wang, E. Todorov, and V. Koltun, "Animating human lower limbs using contact-invariant optimization," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 6, p. 203, 2013.

[6] R. Tedrake, T. W. Zhang, and H. S. Seung, "Stochastic policy gradient reinforcement learning on a simple 3d biped," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, Sept 2004, pp. 2849–2854 vol.3.

[7] N. Kohl and P. Stone, "Policy gradient reinforcement learning for fast quadrupedal locomotion," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 3, April 2004, pp. 2619–2624 Vol.3.

[8] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth, "Bayesian optimization for learning gaits under uncertainty," *Annals of Mathematics and Artificial Intelligence*, vol. 76, no. 1, pp. 5–23, Feb 2016. [Online]. Available: https://doi.org/10.1007/s10472-015-9463-9

[9] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *ArXiv e-prints*, Sep. 2015.

[10] N. Heess, D. TB, S. Sriram, J. Lemmon, J. Merel, G. Wayne, Y. Tassa, T. Erez, Z. Wang, S. M. A. Eslami, M. Riedmiller, and D. Silver, "Emergence of Locomotion Behaviours in Rich Environments," *ArXiv e-prints*, Jul. 2017.

[11] M. Posa, S. Kuindersma, and R. Tedrake, "Optimization and stabilization of trajectories for constrained dynamical systems," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1366–1373.

[12] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *International Journal of Robotics Research*, vol. 33, no. 1, pp. 69–81, 2014.

[13] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, "Optimization-based full body control for the DARPA robotics challenge," *Journal of Field Robotics*, vol. 32, no. 2, pp. 293–312, 2015.

[14] M. W. Spong, "Swing up control of the acrobot using partial feedback linearization," in *Proc. American Control Conference (ACC)*, 1994, pp. 2158–2162.

[15] M. Posa and R. Tedrake, "Direct trajectory optimization of rigid body dynamical systems through contact," in *Algorithmic foundations of robotics X*. Springer, 2013, pp. 527–542.

[16] M. Srinivasan, *Why walk and run: energetic costs and energetic optimality in simple mechanics-based models of a bipedal animal*. Cornell University Ithaca, NY, 2006.

[17] P. L. Kapitza, "A pendulum with oscillating suspension," *Uspekhi Fizicheskikh Nauk*, vol. 44, pp. 7–20, 1951.