# Quadratic Programming-Based Inverse Dynamics Control for Legged Robots with Sticking and Slipping Frictional Contacts

Samuel Zapolsky[1] and Evan Drumwright[2]

***Abstract*— Inverse dynamics control is an extremely effective nonlinear control strategy if the inverse dynamics computation can be performed with sufficient speed. We describe our method for inverse dynamics control of legged robots that can deal with both sticking and slipping frictional contacts and mitigates the problems introduced by indeterminate rigid body contact. We improve this work, which previously used quadratically constrained quadratic programs (QCQPs), to use faster-to-solve quadratic programs via linear algebraic simplifications and a nullspace. We also show that Lemke's Algorithm is significantly faster than alternatives and permits multi-stage optimizations within control loops running as fast as 125Hz on commodity hardware.**

## I. INTRODUCTION

The control scheme with the potential to track a trajectory most accurately is inverse dynamics control. The only caveats are that the model must be sufficiently faithful to reality and that the inverse dynamics control is fast enough to run in high frequency control loops. This paper focuses on (1) inverse dynamics control for legged robots interacting with rigid environments using predicted contact forces (contact forces cannot currently be sensed with sufficient accuracy and without considerable lag for inverse dynamics), (2) the susceptibility of the underlying rigid body contact models to *indeterminate configurations* and the effects of this problem, and (3) the development of fast methods to compute inverse dynamics under such constraints. We verify our method using a simulated quadruped that demonstrates a robotic morphology which readily exhibits indeterminate contact with its environment.

## II. NOTATION

Our notation below will assume that all quantities (unless otherwise specified) are given in *independent coordinates* (velocities, accelerations, inertias, and forces), *i.e.*, the minimum coordinates necessary to fully describe the state of a multi-rigid-body system.

| | |
|---|---|
| $\boldsymbol{c}_N, \boldsymbol{c}_S, \boldsymbol{c}_T$ | the (to be determined) magnitudes of the impulsive forces applied at points of contact in the normal, first tangent, and second tangent direction of each contact; $\boldsymbol{c}_S$ and $\boldsymbol{c}_T$ each have two components—corresponding to positive and negative magntidues along the first and second tangent directions |
| $\mathbf{M}$ | the independent inertia matrix (combined robot and contacting rigid body independent inertias) |
| $\boldsymbol{v}$ | the current independent velocity (concatentated velocity of the robot and bodies in contact) |
| $\boldsymbol{v}^*$ | the independent velocity (concatentated velocity of the robot and the bodies in contact) of the system after the contact model and external forces/torques have been applied |
| $\dot{\boldsymbol{q}}$ | the current robot joint velocity |
| $\ddot{\boldsymbol{q}}$ | the desired robot joint acceleration |
| $\boldsymbol{\tau}$ | the (to be determined) vector of torques to be applied to the robot's joints |
| $\boldsymbol{f}_{\text{ext}}$ | the vector of independent external forces/torques (combined robot and contacting rigid body independent forces/torques) |
| $\mathbf{N}, \mathbf{S}, \mathbf{T}$ | the Jacobian relating contact forces to independent forces along the contact normals and first and second tangent directions, respectively |
| $\mathbf{Q}, \mathbf{K}_S, \mathbf{K}_T$ | the Jacobian relating contact forces to independent forces along the direction of relative velocity (sliding contacts only) and first and second tangent directions (sticking contacts only), respectively |
| $\boldsymbol{\mu}$ | $n$ Coulomb friction coefficients |

## III. RELATED WORK

### A. Inverse dynamics control for bipeds

Idealized bipeds (*i.e.*, with point contacts as the feet) do not experience indeterminate contact configurations at the feet, as more than two contact points are required for indeterminacy. Contact forces that yield a desired trajectory using inverse dynamics control is computable (as shown in [12]) given a trajectory consistent with no-slip constraints at the feet (or with inconsistent accelerations using least-squares minimization). Bipedal walking systems can utilize our methodology but would not need to implement our approach for mitigating inconsistent configurations.

### B. Inverse dynamics control for quadrupeds

The work of Righetti *et al.* [11] also focused on inverse dynamics control while in contact with rigid environments. That work sought a linear system solution to the coupled problems of determining contact forces and motor torques. Additionally, [11] developed a framework that permits quickly optimizing any mixed linear/quadratic function of motor torques and contact forces using fast numerical linear algebra algorithms. Their work does not incorporate unilateral contact constraints; nor do they use any knowledge

[1] Samuel Zapolsky is a Ph. D. student in Computer Science at George Washington University, Washington, DC samzapo@gwmail.gwu.edu

[2] Evan Drumwright is faculty in Computer Science at George Washington University, Washington, DC drum@gwu.edu

of surface friction to better model the contact forces (which permits tracking desired trajectories more closely).

In contrast to that research, our previous work [17] has used a "true" (*i.e.*, unilaterally constrained) contact model for developing an inverse dynamics-based controller for quadruped robots. Our contact model is able to predict bodies separating, does not apply tensile contact forces, is guaranteed to be feasible (a set of contact forces always exists), does not suffer from Painlevé-type paradoxes [10] in the presence of slipping friction, guarantees frictional forces are dissipative (forces cannot be predicted that would add energy to the system), and does not require considering an exponential (in the number of contact points) tree of possible solutions. Our previous work used a quadratically constrained quadratic programming (QCQP)-based second phase. Our current work converts that model into a convex, linearly constrained quadratic program (QP) that can be solved with multiple, fast, and free software libraries.

## IV. BACKGROUND: RIGID CONTACT MODELS

Theory of rigid contact assumes that bodies cannot be interpenetrating, and thus are either in contact at a point, an edge, or one or more surfaces. Figure 1 shows two bodies A and B in contact at a single point $\boldsymbol{p}$. We define a contact frame $F$ at $\boldsymbol{p}$ with one axis ($\hat{\boldsymbol{n}}$) pointing along the contact normal toward A and the other two axes, $\hat{\boldsymbol{s}}$ and $\hat{\boldsymbol{t}}$ aligned arbitrarily. If we define the relative velocity between the two bodies projected along the contact normal as $\dot{\phi}_{\hat{n}}$, the bodies are separating at $\boldsymbol{p}$ if $\dot{\phi}_{\hat{n}} > 0$ and impacting if $\dot{\phi}_{\hat{n}} < 0$. If the bodies are separating, no contact forces need be applied; if the bodies are impacting, an impact law must be used. If the bodies are neither impacting nor separating but $\ddot{\phi}_{\hat{n}} < 0$, contact forces must be applied to prevent interpenetration. The following section describes such a model.

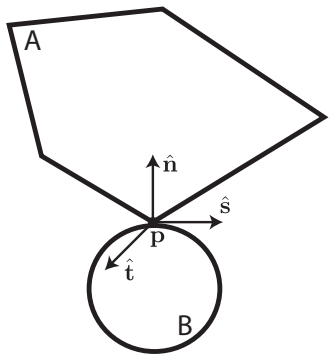


Fig. 1. Two rigid bodies A and B in contact at a single point $\boldsymbol{p}$ with contact normal $\hat{\boldsymbol{n}}$.

### A. Models for resting contact without impulsive forces

Rigid body contact under Coulomb friction can be modeled using the following *mixed linear complementarity problem* (MLCP):

$$\begin{bmatrix} \mathbf{M} & \mathbf{W} & -\mathbf{K}_S{}^{\mathsf{T}} & -\mathbf{K}_T{}^{\mathsf{T}} & \mathbf{0} \\ \mathbf{N} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{K}_S & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{X}_S \\ \mathbf{K}_T & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{X}_T \\ \mathbf{0} & \boldsymbol{\Omega}_K & -\mathbf{X}_S{}^{\mathsf{T}} & -\mathbf{X}_T^{\mathsf{T}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{c}_N \\ \boldsymbol{c}_S \\ \boldsymbol{c}_T \\ \boldsymbol{\lambda} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{f}_{\text{ext}} \\ \dot{\mathbf{N}}\boldsymbol{v} \\ \dot{\mathbf{K}}_S\boldsymbol{v} \\ \dot{\mathbf{K}}_T\boldsymbol{v} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\alpha} \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma} \\ \boldsymbol{\delta} \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{c}_N \\ \boldsymbol{c}_S \\ \boldsymbol{c}_T \\ \boldsymbol{\lambda} \end{bmatrix} \geq \mathbf{0}, \quad \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma} \\ \boldsymbol{\delta} \end{bmatrix} \geq \mathbf{0}, \quad \begin{bmatrix} \boldsymbol{c}_N \\ \boldsymbol{c}_S \\ \boldsymbol{c}_T \\ \boldsymbol{\lambda} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma} \\ \boldsymbol{\delta} \end{bmatrix} = 0$$

where $\mathbf{W} \triangleq \boldsymbol{\Omega}_L \boldsymbol{Q}^{\mathsf{T}} - \boldsymbol{N}^{\mathsf{T}}$, ($\boldsymbol{\Omega}_L$ is a $n \times r_L$ dimensional matrix, where $r_L$ is the number of sliding contacts, for which entry $(i, j) = \mu_i$ if the $i^{\text{th}}$ contact is the $j^{\text{th}}$ sliding contact and zero otherwise), $\boldsymbol{a}$ is the (to be determined) independent acceleration of the system, $\mathbf{X}_S$ and $\mathbf{X}_T$ are block diagonal:

$$\mathbf{X}_S = \begin{bmatrix} \mathbf{X}_{S_1} & & \\ & \ddots & \\ & & \mathbf{X}_{S_n} \end{bmatrix}$$

$$\mathbf{X}_T = \begin{bmatrix} \mathbf{X}_{T_1} & & \\ & \ddots & \\ & & \mathbf{X}_{T_n} \end{bmatrix}$$

(the blocks are defined immediately below) and $\boldsymbol{\Omega}_K$ is a $n \times r_K$-dimensional ($r_K$ is the number of sticking contacts) matrix; entry $(i, j) = \mu_i$ if the $i^{\text{th}}$ contact is sticking and is set to zero otherwise.

We linearize the friction cone using a circumscribing $k$-edge polygon (see Figure 4), which we define using $2 \times \frac{k+4}{4}$ dimensional matrices $\mathbf{X}_{S_i}$ and $\mathbf{X}_{T_i}$ to correspond to the $i^{\text{th}}$ contact as follows:

$$\mathbf{X}_{S_i} = \begin{bmatrix} \cos 0 & \cos\theta & \cos 2\theta & \dots & \cos\frac{\pi}{2} \\ \cos 0 & \cos\theta & \cos 2\theta & \dots & \cos\frac{\pi}{2} \end{bmatrix}$$

$$\mathbf{X}_{T_i} = \begin{bmatrix} \sin 0 & \sin\theta & \sin 2\theta & \dots & \sin\frac{\pi}{2} \\ \sin 0 & \sin\theta & \sin 2\theta & \dots & \sin\frac{\pi}{2} \end{bmatrix}$$

where $\theta = \frac{\pi}{\frac{k}{2}+1}$. In the special case that $k = 4$, we can further reduce the matrices to the $2 \times 1$ dimensional matrices:

$$\mathbf{X}_{S_i} = \mathbf{X}_{T_i} = \begin{bmatrix} \cos 0 \\ \cos 0 \end{bmatrix} = \begin{bmatrix} \sin\frac{\pi}{2} \\ \sin\frac{\pi}{2} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

When there is no sliding friction, the MLCP model is solvable and copositive-plus (using a nearly identical proof to that described in [15]), which means it can be solved in expected polynomial time using Lemke's Algorithm [5], [2]. When sliding friction is present the MLCP may not possess a solution due to the existence of inconsistent configurations [14]. This issue led to the movement to the impulsive force/velocity domain [7] and *time stepping methods* [15], which can provably avoid inconsistent configurations. As a side benefit, sustained ($\dot{\phi}_{\hat{n}} = 0,\ \ddot{\phi}_{\hat{n}} < 0$) and impacting ($\dot{\phi}_{\hat{n}} < 0$) contacts can be treated under a single regime, meaning—as a practical matter—one need not classify small floating point values into sustained *vs.* impacting contacts.

Another issue with rigid body contact *that remains even when time stepping methods are used* is that of indeterminacy: as Figure 2 illustrates, multiple, even infinite, sets of contact forces might satisfy the model. This scenario is identical to that of a quadruped standing on all four legs, though indeterminacy can occur with the robot supported by three legs as well. Thus, this particular issue can arise under both standing and creeping gaits.

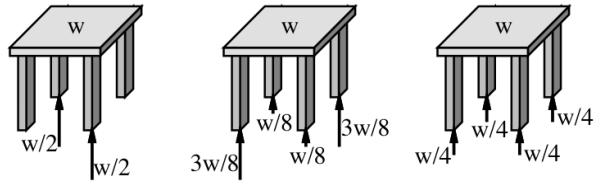


Fig. 2. An illustration (taken from [6]) of indeterminacy in rigid body contact models. Infinite variations of force magnitudes yield solutions to this problem.

For copositive-plus MLCP-based models, even if one can devise a deterministic rule to select a solution (*e.g.*, pick the solution with the lowest $\ell_2$-norm), it is not clear that computing such a solution can be performed in polynomial time (even in the expected case). The *status quo* is that the linear complementarity problem (LCP) solver converges to an arbitrary solution (which is a function of the pivoting rule and the initial basis for Lemke's Algorithm).

When we incorporate a rigid body contact model into an inverse dynamics controller, the implication of indeterminacy is potentially rapidly switching, and thus mechanically deleterious, torque values. This is not just a theoretical issue: we have observed contact force predictions switching rapidly between two opposing sets of legs (front left/hind right and front right/hind left), leading to highly discontinuous motor torques; this problem is illustrated by the extreme values of the time derivative of motor torques when our simulated quadruped is standing on all four feet (see Figure 3).

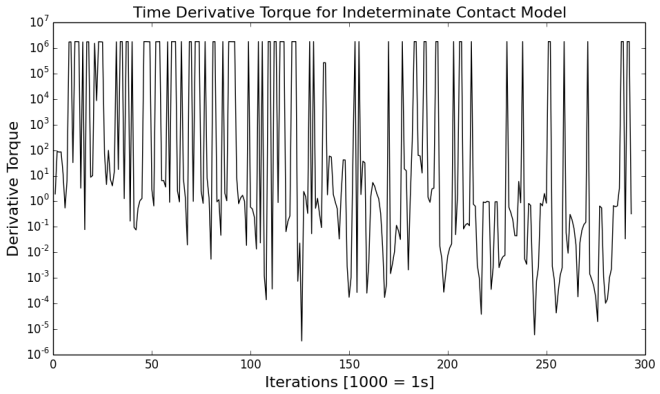


Fig. 3. The norm of the time derivative of the motor torques (over all joints) computed by inverse dynamics using an indeterminate contact model

### B. Convex contact model

Our previous work [3] introduced a convex contact model (also used by Todorov [16] for model predictive control with contact) that exhibits worst-case polynomial time complexity. Our model, which yields a convex QP, is reproduced below:

$$\underset{\boldsymbol{c}_N,\boldsymbol{c}_S,\boldsymbol{c}_T}{\text{minimize}} \ \frac{1}{2}\boldsymbol{v}^{*\mathsf{T}}\mathbf{M}\boldsymbol{v}^* \tag{1}$$

$$\text{such that } \boldsymbol{v}^* = \boldsymbol{r} + \mathbf{M}^{-1}(\mathbf{N}^\mathsf{T}\boldsymbol{c}_N + \mathbf{S}^\mathsf{T}\boldsymbol{c}_S + \mathbf{T}^\mathsf{T}\boldsymbol{c}_T) \tag{2}$$

$$\mathbf{N}\boldsymbol{v}^* \geq \mathbf{0} \tag{3}$$

$$\mu_i c_{N_i} \geq \mathbf{X}_{S_i}\boldsymbol{c}_{S_i} + \mathbf{X}_{T_i}\boldsymbol{c}_{T_i} \ (\text{for } i = 1 \ldots n) \tag{4}$$

$$\boldsymbol{c}_N \geq \mathbf{0} \tag{5}$$

$$\mathbf{1}^\mathsf{T}\boldsymbol{c}_N \leq \kappa \tag{6}$$

where $\boldsymbol{r} \triangleq \boldsymbol{v} + h\mathbf{M}^{-1}\boldsymbol{f}_{\text{ext}}$, for $h > 0$ effectively yields a time stepping method that avoids the issue of inconsistent configurations. This model uses the principle of maximum dissipation [13] to determine contact forces that minimize the post-contact kinetic energy of the system (Equation 1). Equation 2 provides the relationship between the contact forces and the change in velocity. Equation 3 specifies that the post contact velocities should not lead to interpenetration. Equation 4 describes the Coulomb friction constraint. Equation 5 specifies that the forces applied along the contact normal should be compressive. Finally, Equation 6—where $\kappa$ is the sum of force magnitudes applied along the contact normals for the frictionless instance of the contact problem—ensures contact forces are conservative for frictionless contact.

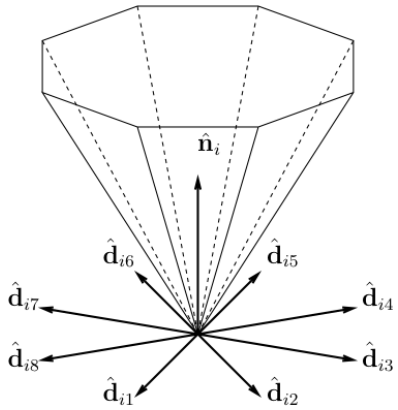


Fig. 4. A linearized friction cone, which enables the quadratic inequality Coulomb friction constraints to be approximated using a variable number of linear inequality constraints. Figure taken from [15].

Finally, we note that this problem was transformed from a QCQP in [3] into a standard QP using the linearized friction cone. *The advantage of QPs over QCQPs is that the latter currently require either interior-point or SQP [9] solvers; both currently exhibit variable running times poorly suited to real-time control loops, in our experience.* QPs, in contrast, can be solved by algorithms that run fairly deterministically, including active set methods [9] and splitting methods [8].

## V. METHOD: INVERSE DYNAMICS MODEL

The inverse dynamics model follows from adding the inverse dynamics constraint (Equation 7) to the existing

contact model (reproduced below):

$$\begin{aligned}
\underset{\boldsymbol{x},\boldsymbol{c}_N,\boldsymbol{c}_S,\boldsymbol{c}_T}{\text{minimize}}\ & \frac{1}{2}{\boldsymbol{v}^*}^{\mathsf{T}}\mathbf{M}\boldsymbol{v}^* \\
\text{such that}\ & \mathbf{N}^{\mathsf{T}}\boldsymbol{v}^* \geq \mathbf{0} \\
& \boldsymbol{c}_N \geq \mathbf{0} \\
& \mu_i c_{N_i} \geq \mathbf{X}_{S_i}\boldsymbol{c}_{S_i} + \mathbf{X}_{T_i}\boldsymbol{c}_{T_i} \quad (\text{for } i = 1 \ldots n) \\
& {\boldsymbol{c}_N}^{\mathsf{T}}\mathbf{1} \leq \kappa \\
& \boldsymbol{v}^*_{\text{robot.q}} = \dot{\boldsymbol{q}} + h\ \ddot{\boldsymbol{q}} \quad \text{(inverse dynamics)} \qquad (7)
\end{aligned}$$

where

$$\mathbf{M}(\boldsymbol{v}^* - \boldsymbol{v}) = \mathbf{N}\boldsymbol{c}_N + \mathbf{S}\boldsymbol{c}_S + \mathbf{T}\boldsymbol{c}_T + h(\boldsymbol{f}_{\text{ext}} + \begin{bmatrix}0\\ \boldsymbol{x}\end{bmatrix}) \quad (8)$$

$$\boldsymbol{v}^* \triangleq \begin{bmatrix}{\boldsymbol{v}^*}^{\mathsf{T}}_{\text{robot.fb}} & {\boldsymbol{v}^*}^{\mathsf{T}}_{\text{robot.q}} & {\boldsymbol{v}^*}^{\mathsf{T}}_{\text{cb}}\end{bmatrix}^{\mathsf{T}} \quad (9)$$

$$\boldsymbol{f}_{\text{ext}} \triangleq \begin{bmatrix}{\boldsymbol{f}_{\text{fb}}}^{\mathsf{T}} & \boldsymbol{\tau}^{\mathsf{T}}_{\text{ext}} & {\boldsymbol{f}_{\text{cb}}}^{\mathsf{T}}\end{bmatrix}^{\mathsf{T}} \quad (10)$$

For the recurrent forces vector ($\boldsymbol{f}_{\text{ext}}$), acting on the robot and the objects in its environment, the abbreviation "fb" refers to a six-dimensional vector of the robot's floating base; "ext" refers to forces not determined by the robot's actuation; "cb" refers to a $m \times 6$-dimensional vector of the $m$ rigid bodies contacting the robot.

We have shown in [3] that the underlying convex contact model always possess a solution and that the model does only negative work. The addition of the inverse dynamics constraint (Equation 7) does not change this result. As in the original contact model, the first order approximation of next velocity avoids inconsistent configurations that can occur in rigid body dynamics contact with Coulomb friction.

The worst-case time complexity of solving this particular model is $O(n^{3.5})$—assuming that the binary length of the input grows linearly with $n$—where $n$ in the number of contact points using interior-point methods [1]. Practically faster active-set methods (*e.g.*, Lemke's Algorithm [5]) run in expected $O(n^3)$ time (Cottle claims that the expected iterations for Lemke's Algorithm is $O(n)$ [2]), but exhibit worst-case $2^{O(n)}$ time complexity. High frequency control loops limit $n$ to approximately four for present optimization techniques and computational hardware.

*Simplifying the computation (previous work)*

Our previous work [17] showed how to simplify and reduce the size of the optimization problem, as well as remove linear equality constraints (which eliminates significant variables if transforming the QP to a LCP via optimization duality theory), using the following definitions:

$$\begin{aligned}
\mathbf{R} &\triangleq \begin{bmatrix}\mathbf{N} & \mathbf{S} & \mathbf{T}\end{bmatrix} \\
\boldsymbol{z} &\triangleq \begin{bmatrix}\boldsymbol{c}_N^{\mathsf{T}} & \boldsymbol{c}_S^{\mathsf{T}} & \boldsymbol{c}_T^{\mathsf{T}}\end{bmatrix}^{\mathsf{T}} \\
\mathbf{M} &\triangleq \begin{bmatrix}\mathbf{A} & \mathbf{B} \\ \mathbf{B}^{\mathsf{T}} & \mathbf{C}\end{bmatrix}
\end{aligned}$$

$$\begin{aligned}
\mathbf{M}^{-1} &\triangleq \begin{bmatrix}\mathbf{D} & \mathbf{E} \\ \mathbf{E}^{\mathsf{T}} & \mathbf{F}\end{bmatrix} \\
\boldsymbol{j} &\triangleq \boldsymbol{v}_b + \begin{bmatrix}\mathbf{D} & \mathbf{E}\end{bmatrix}(h\boldsymbol{f}_{ext} + \begin{bmatrix}\mathbf{0} \\ h\boldsymbol{f}_{ID}\end{bmatrix}) \\
\boldsymbol{k} &\triangleq \boldsymbol{v}_q + \begin{bmatrix}\mathbf{E}^{\mathsf{T}} & \mathbf{F}\end{bmatrix}(h\boldsymbol{f}_{ext} + \begin{bmatrix}\mathbf{0} \\ h\boldsymbol{f}_{ID}\end{bmatrix})
\end{aligned}$$

The components of $\boldsymbol{v}^*$ are then defined as follows:

$$\boldsymbol{v}^*_b = \boldsymbol{j} + \begin{bmatrix}\mathbf{D} & \mathbf{E}\end{bmatrix}(\mathbf{R}\boldsymbol{z} + \begin{bmatrix}\mathbf{0} \\ h\boldsymbol{x}\end{bmatrix}) \quad (11)$$

$$\boldsymbol{v}^*_q = \boldsymbol{k} + \begin{bmatrix}\mathbf{E}^{\mathsf{T}} & \mathbf{F}\end{bmatrix}(\mathbf{R}\boldsymbol{z} + \begin{bmatrix}\mathbf{0} \\ h\boldsymbol{x}\end{bmatrix}) = \boldsymbol{v}_q + h\boldsymbol{a}_q \quad (12)$$

Our previous work also showed that—using the latter equation—we can solve a linear equation for the unknown actuator forces ($\boldsymbol{x}$) by first solving for the contact forces:

$$\boldsymbol{x} = \frac{\mathbf{F}^{-1}(\boldsymbol{v}^*_q - \boldsymbol{k} - \begin{bmatrix}\mathbf{E}^{\mathsf{T}} & \mathbf{F}\end{bmatrix}\mathbf{R}\boldsymbol{z})}{h} \quad (13)$$

Substituting this solution for $\boldsymbol{x}$ from Equation 12 into Equation 11, we arrive at:

$$\boldsymbol{v}^*_b = \boldsymbol{j} + \begin{bmatrix}\mathbf{D} & \mathbf{E}\end{bmatrix}\mathbf{R}\boldsymbol{z} + \mathbf{E}\mathbf{F}^{-1}(\boldsymbol{v}^*_q - \boldsymbol{k} - \begin{bmatrix}\mathbf{E}^{\mathsf{T}} & \mathbf{F}\end{bmatrix}\mathbf{R}\boldsymbol{z})$$

To move further toward simplifying our second phase optimization, we will define a new matrix and a new vector:

$$\begin{aligned}
\mathbf{Z} &\triangleq \left(\begin{bmatrix}\mathbf{D} & \mathbf{E}\end{bmatrix} - \mathbf{E}\mathbf{F}^{-1}\begin{bmatrix}\mathbf{E}^{\mathsf{T}} & \mathbf{F}\end{bmatrix}\right)\mathbf{R} \\
\boldsymbol{p} &\triangleq \boldsymbol{j} + \mathbf{E}\mathbf{F}^{-1}(\boldsymbol{v}^*_q - \boldsymbol{k})
\end{aligned}$$

Now, $\boldsymbol{v}^*_b$ can be defined simply, and solely in terms of $\boldsymbol{z}$, as:

$$\boldsymbol{v}^*_b = \mathbf{Z}\boldsymbol{z} + \boldsymbol{p}$$

We now represent the objective function (Equation 1) in block form as:

$$f(.) \triangleq \frac{1}{2}\begin{bmatrix}\boldsymbol{v}^*_b \\ \boldsymbol{v}^*_q\end{bmatrix}^{\mathsf{T}}\begin{bmatrix}\mathbf{A} & \mathbf{B} \\ \mathbf{B}^{\mathsf{T}} & \mathbf{C}\end{bmatrix}\begin{bmatrix}\boldsymbol{v}^*_b \\ \boldsymbol{v}^*_q\end{bmatrix}$$

which, when expanded, yields:

$$f(.) \triangleq \frac{1}{2}{\boldsymbol{v}^*_b}^{\mathsf{T}}\mathbf{A}\boldsymbol{v}^*_b + \boldsymbol{v}^*_b\mathbf{B}^{\mathsf{T}}\boldsymbol{v}^*_q + \frac{1}{2}{\boldsymbol{v}^*_q}^{\mathsf{T}}\mathbf{C}\boldsymbol{v}^*_q$$

We can ignore the last term of the above equation, as $\boldsymbol{v}^*_q$ is independent of $\boldsymbol{z}$. Expanding the remaining terms using Equation 11, we arrive at the new objective function:

$$\begin{aligned}
f(.) &\triangleq \frac{1}{2}\boldsymbol{z}^{\mathsf{T}}\mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{Z}\boldsymbol{z} + \boldsymbol{z}^{\mathsf{T}}\mathbf{Z}^{\mathsf{T}}\mathbf{A}\boldsymbol{p} + \boldsymbol{z}^{\mathsf{T}}\mathbf{Z}^{\mathsf{T}}\mathbf{B}\boldsymbol{v}^*_q \\
&\triangleq \frac{1}{2}\boldsymbol{z}^{\mathsf{T}}\mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{Z}\boldsymbol{z} + \boldsymbol{z}^{\mathsf{T}}(\mathbf{Z}^{\mathsf{T}}\mathbf{A}\boldsymbol{p} + \mathbf{Z}^{\mathsf{T}}\mathbf{B}\boldsymbol{v}^*_q)
\end{aligned}$$

subject to the following constraints:

$$\begin{aligned}
\mathbf{N}^{\mathsf{T}}\begin{bmatrix}\mathbf{Z}\boldsymbol{z} + \boldsymbol{p} \\ \boldsymbol{v}^*_q\end{bmatrix} &\geq 0 \\
\boldsymbol{z}_i &\geq 0 \quad (\text{for } i = 1 \ldots n) \\
\mu_i\boldsymbol{z}_i \geq \mathbf{X}_{S_i}\boldsymbol{z}_{S_i} + \mathbf{X}_{T_i}\boldsymbol{z}_{T_i} &\quad (\text{for } i = 1 \ldots n)
\end{aligned}$$

Symmetry and positive semi-definiteness of the quadratic programming problem follows from symmetry and positive definiteness of $\mathbf{A}$. Once the solution to this QP is determined, $\boldsymbol{x}$ can be recovered via Equation 13. The actuator forces determined via inverse dynamics are then $\boldsymbol{f}_{ID} + \boldsymbol{x}$.

*Mitigating rigid body indeterminacy (through a second phase):*

In the case that the matrix $\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}$ is singular, the contact model is only convex rather than *strictly convex* [1]; conceptually, contact forces that predict that two legs, three legs, or four legs support the robot are all equally valid. This section describes a method to optimize within the contact model's solution space while favoring solutions that predict contact forces at all contacting feet (and thus preventing the rapid torque cycling observed in Section IV-A). This phase must be run sequentially, incorporating the solution computed in Section V (hereby denoted as *Phase I*) into a second phase.

Our previous work [17] used exactly this approach to eliminate the inconsistency in the rigid contact model (albeit arbitrarily) by using a second optimization phase to make the model strictly convex. However, that work used a quadratic inequality constraint, yielding a QCQP that may be insufficiently fast for high frequency control loops. We now demonstrate how to use the nullspace of $\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}$ to perform this second optimization *without explicitly considering the quadratic inequality constraint*; thus, QP-based modeling is retained. Assume that the matrix $\mathbf{P}$ gives the nullspace of $\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}$. The vector of contact forces will now be given as $\boldsymbol{z} + \mathbf{P}\boldsymbol{w}$, where $\boldsymbol{w}$ will be the optimization vector.

The kinetic energy from applying the contact impulses is:

$$\begin{aligned}\epsilon_2 &= \frac{1}{2}(\boldsymbol{z}+\mathbf{P}\boldsymbol{w})^\mathsf{T}\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}(\boldsymbol{z}+\mathbf{P}\boldsymbol{w}) \\ &\quad + (\boldsymbol{z}+\mathbf{P}\boldsymbol{w})^\mathsf{T}(\mathbf{Z}^\mathsf{T}\mathbf{A}\boldsymbol{p} + \mathbf{Z}^\mathsf{T}\mathbf{B}\boldsymbol{v}_q^*) \\ &= \frac{1}{2}\boldsymbol{z}^\mathsf{T}\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}\boldsymbol{z} + \boldsymbol{z}^\mathsf{T}(\mathbf{Z}^\mathsf{T}\mathbf{A}\boldsymbol{p} + \mathbf{Z}^\mathsf{T}\mathbf{B}\boldsymbol{v}_q^*) \\ &\quad + \boldsymbol{w}^\mathsf{T}\mathbf{P}^\mathsf{T}(\mathbf{Z}^\mathsf{T}\mathbf{A}\boldsymbol{p} + \mathbf{Z}^\mathsf{T}\mathbf{B}\boldsymbol{v}_q^*)\end{aligned}$$

Note that the terms $\frac{1}{2}\boldsymbol{w}^\mathsf{T}\mathbf{P}^\mathsf{T}\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}\mathbf{P}\boldsymbol{w}$ and $\boldsymbol{z}\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}\mathbf{P}\boldsymbol{w}$ are not included above. Both are zero because $\mathbf{P}$ is in the nullspace of $\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}$. We wish for the energy dissipated in the second phase, $\epsilon_2$, to be equal to the energy dissipated in the first phase, $\epsilon_1$. Thus, we want $\epsilon_2 - \epsilon_1 = 0$. Algebra yields:

$$\boldsymbol{w}^\mathsf{T}\mathbf{P}^\mathsf{T}(\mathbf{Z}^\mathsf{T}\mathbf{A}\boldsymbol{p} + \mathbf{Z}^\mathsf{T}\mathbf{B}\boldsymbol{v}_q^*) = 0 \tag{14}$$

First defining $\boldsymbol{y}$ as:

$$\boldsymbol{y} \triangleq \frac{\mathbf{F}^{-1}\Big(\boldsymbol{v}_q^* - \boldsymbol{k} - \begin{bmatrix}\mathbf{E}^\mathsf{T} & \mathbf{F}\end{bmatrix}\mathbf{R}(\boldsymbol{z}+\mathbf{P}\boldsymbol{w})\Big)}{h} \tag{15}$$

the following optimization problem arises:

$$\underset{\boldsymbol{w}}{\text{minimize}}\ \frac{1}{2}\boldsymbol{y}^\mathsf{T}\boldsymbol{y} \tag{16}$$

$$\text{subject to } (\boldsymbol{p}^\mathsf{T}\mathbf{A}^\mathsf{T} + \boldsymbol{v}_q^{*\mathsf{T}}\mathbf{B}^\mathsf{T})\mathbf{Z}\mathbf{P}\boldsymbol{w} = \mathbf{0} \tag{17}$$

$$\mathbf{N}^\mathsf{T}\begin{bmatrix}\mathbf{Z}(\boldsymbol{z}+\mathbf{P}\boldsymbol{w})+\boldsymbol{p} \\ \boldsymbol{v}_q^*\end{bmatrix} \geq \mathbf{0} \tag{18}$$

$$(\boldsymbol{z}+\mathbf{P}\boldsymbol{w})_i \geq \mathbf{0} \text{ (for } i = 1\ldots 5n) \tag{19}$$

$$\mu_i(\boldsymbol{z}+\mathbf{P}\boldsymbol{w})_i \geq \mathbf{X}_{S_i}(\boldsymbol{z}+\mathbf{P}\boldsymbol{w})_{S_i} + \ldots \mathbf{X}_{T_i}(\boldsymbol{z}+\mathbf{P}\boldsymbol{w})_{T_i} \text{ (for } i = 1\ldots n) \tag{20}$$

However, we use a proof that $\mathbf{Z}\cdot\ker(\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}) = \mathbf{0}$ (see Appendix) to render $n+1$ of $7n+1$ linear constraints (Equations 17 and 18) moot.

*Feasibility and time complexity*

It should be clear that a feasible point ($\boldsymbol{w} = \mathbf{0}$) always exists for the optimization problem. The dimensionality ($n \times n$ in the number of contact points) of $\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}$ yields a nullspace computation of $O(n^3)$ and represents one third of the Phase II costs. For quadrupeds with single point contacts, the dimensionality of $\boldsymbol{w}$) is typically at most two, and thus the total number of optimization variables is at most $6n+2$ (each linear constraint introduces six KKT dual variables for the simplest friction cone approximation). The optimization—which, as noted in Section V, runs in expected $O(n^3)$ time—is responsible for two thirds of the second-phase costs.

## VI. EXPERIMENTS

We simulated a quadrupedal robot supported by four feet, modeled as point contacts, on a rigid planar surface using our simulation software, `Moby`. Torque output was collected given the convex and strictly convex systems for Phase I and Phase II, respectively. The change of torque output over time for each model is shown in Figure 5, which illustrates that Phase II generates smooth motor torques in indeterminate rigid body contacting configurations. We used `MATLAB`'s `fmincon` optimization tool to compare the performance of different classes of optimizers on our controller, and we also compared these optimizers against a `MATLAB` implementation of Lemke's Algorithm (*i.e.*, the `LEMKE` library [4]). Timing results can be found in Table I. Accuracy results from this system are published in [17]; here we present results on achieving rapid running time. When compared to work on non-redundant systems such as a bipedal walker (*e.g.* [12]), our system will only utilize Phase I without indeterminacy resolution; in such cases, our system can solve the inverse dynamics QP with sub-millisecond runtimes, allowing for control rates of greater than 1000Hz.

The strictly convex system used in the smoothing optimization (Phase II) that we present in this paper requires us to either constrain for—or optimize within the nullspace, as we described in Section V—the quadratic objective of the convex, indeterminate contact model (Phase I). Timings for Phase I and II optimizations, on indeterminate and determinate contact models, respectively, are given in Table I. The last row of the table uses Lemke's Algorithm with the method presented in this paper and is performed within the nullspace of $\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}$. We find a large increase in optimization speed due to the removal of the quadratic inequality constraint (and facilitation of Lemke's Algorithm). The nullspace does necessitate an additional singular value decomposition (the nullspace is less susceptible to numerical error when computed in this way compared to using a QR factorization), requiring $0.2277$ms of time on average and a

| Phase | Optimizer | min [ms] | mean [ms] | max [ms] |
|---|---|---|---|---|
| Phase I: Indeterminate | interior-point | 6.7443 | 89.1671 | 169.4384 |
| contact model | active-set | 4.8382 | 7.3098 | 253.9801 |
| | trust region reflexive | 7.9662 | 93.4645 | 478.2784 |
| | LEMKE | 0.5441 | 1.0073 | 10.6818 |
| Phase II: Determinate | interior-point | 12.9814 | 172.2423 | 706.4246 |
| contact model | sequential quadratic programming | 147.1 | 855.4 | 8973.6 |
| | **LEMKE** | **0.5390** | **1.9070** | **10.2718** |

TABLE I

TIMINGS IN MILLISECONDS FOR DIFFERENT SOLVERS FOR THE QUADRUPED STANDING EXPERIMENT. SIMULATION RUNNING WITH FIXED STEP SIZE OF 0.001 FOR 0.3 SECONDS (300 ITERATIONS). THE METHOD WE ADVOCATE IN THIS PAPER IS TYPESET IN BOLDFACE.

maximum time of 0.3694ms over 300 trials; thus, the SVD operation accounts for roughly a third the cost of Phase II.

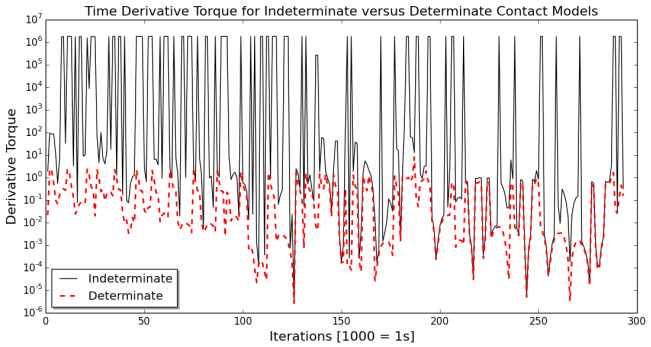


Fig. 5. We observe a 5 order of magnitude reduction in the maximum of the time derivative of torque when using the optimization with our determinate contact model, as opposed to a "standard" indeterminate contact model.

## VII. CONCLUSIONS

We have presented a method of transforming a convex but indeterminate contact model for legged robot inverse dynamics to a strictly convex optimization, *and described how to pose both as readily solvable QPs* using a nullspace, which is rapidly computable. Using QPs in place of QCQP's improves performance by over a magnitude compared to MATLAB's well regarded optimization tools. Performance in C++ is even faster: we are able to solve Phase I and Phase II optimizations in fewer than 8ms on commodity hardware, permitting control loops of 125Hz or faster for legged robots with redundant supports. Thus our results also show that we can perform multi-stage, quadratic optimization at real-time control speeds.

## APPENDIX

*Theorem:* $\mathbf{Z} \cdot \ker(\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}) = \mathbf{0}$ for arbitrary matrix $\mathbf{A}$.

*Proof:* Assume the singular value decomposition of $\mathbf{Z}$ is $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\mathsf{T}$. $\mathbf{\Sigma}$ has the form $\begin{bmatrix}\mathbf{\Sigma}^* & \mathbf{0}\end{bmatrix}$, where $\mathbf{\Sigma}^*$ is a diagonal matrix of non-zero singular values, because $\mathbf{Z}$ has fewer rows than columns. We can then write the product as follows:

$$\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z} = (\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\mathsf{T})^\mathsf{T}\mathbf{A}\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\mathsf{T} \tag{21}$$

$$= \mathbf{V}\mathbf{\Sigma}^\mathsf{T}\mathbf{U}^\mathsf{T}\mathbf{A}\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\mathsf{T} \tag{22}$$

The product in the middle has the form:

$$\mathbf{\Sigma}^\mathsf{T}\mathbf{U}^\mathsf{T}\mathbf{A}\mathbf{U}\mathbf{\Sigma} = \begin{bmatrix}\mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \tag{23}$$

where $\mathbf{B}$ is an irrelevant matrix block. $\mathbf{V}$ is orthonormal (a basic result of the SVD). If we partition $\mathbf{V}$ into left columns and right columns ($\mathbf{V} = \begin{bmatrix}\mathbf{V}_L & \mathbf{V}_R\end{bmatrix}$, the non-singular sub-block of the product $\mathbf{Z}^\mathsf{T}\mathbf{A}\mathbf{Z}$ is $\mathbf{V}_L\mathbf{B}\mathbf{V}_L^\mathsf{T}$. The nullspace must then be $\mathbf{V}_R$, which is also a basis of the nullspace of $\mathbf{Z}$. ■

## REFERENCES


[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[2] R. W. Cottle, J.-S. Pang, and R. Stone. *The Linear Complementarity Problem*. Academic Press, Boston, 1992.

[3] E. Drumwright and D. A. Shell. Modeling contact friction and joint friction in dynamic robotic simulation using the principle of maximum dissipation. In *Proc. of Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2010.

[4] P. L. Fackler and M. J. Miranda. LEMKE. http://people.sc.fsu.edu/ burkardt/m_src/lemke/lemke.m.

[5] C. E. Lemke. Bimatrix equilibrium points and mathematical programming. *Management Science*, 11:681–689, 1965.

[6] B. Mirtich. *Impulse-based Dynamic Simulation of Rigid Body Systems*. PhD thesis, University of California, Berkeley, 1996.

[7] J. J. Moreau. *Standard inelastic shocks and the dynamics of unilateral constraints*, pages 173–221. Springer-Verlag, New York, 1983.

[8] K. G. Murty. *Linear Complementarity, Linear and Nonlinear Programming*. Heldermann Verlag, Berlin, 1988.

[9] J. Nocedal and S. J. Wright. *Numerical Optimization, 2nd ed.* Springer-Verlag, 2006.

[10] P. Painlevé. Sur le lois du frottement de glissemment. *C. R. Académie des Sciences Paris*, 121:112–115, 1895.

[11] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal. Optimal distribution of contact forces with inverse-dynamics control. *Intl. J. Robot. Res.*, 32(3):280–298, 2013.

[12] B. Stephens and C. Atkeson. Dynamic balance force control for compliant humanoid robots. In *Proc. IEEE/RSJ Intl. Conf. Intelligent Robots and Systems (IROS)*, 2010.

[13] D. E. Stewart. Rigid-body dynamics with friction and impact. *SIAM Review*, 42(1):3–39, Mar 2000.

[14] D. E. Stewart. Time-stepping methods and the mathematics of rigid body dynamics. In A. Guran, B. Feeny, A. Klarbring, and Y. Ishida, editors, *Impact and Friction of Solids, Structures, and Intelligent Machines*. World Scientific, 2000.

[15] D. E. Stewart and J. C. Trinkle. An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *Intl. Journal for Numerical Methods in Engineering*, 39(15):2673–2691, 1996.

[16] E. Todorov. A convex, smooth and invertible contact model for trajectory optimization. In *Proc. IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Shanghai, 2011.

[17] S. Zapolsky, E. Drumwright, I. Havoutis, J. Buchli, and C. Semini. Inverse dynamics for a quadruped robot locomoting on slippery surfaces. In *Proc. Intl. Conf. Climbing Walking Robots (CLAWAR)*, Sydney, Australia, 2013.