

Kinematics and Methods for Combined Quasi-Static Stance/Reach Planning in Multi-Limbed Robots

Krishna Shankar, Joel W. Burdick
California Institute of Technology, Pasadena, CA 91125
{krishna, jwb}@robotics.caltech.edu

Abstract—This paper provides kinematic analysis and local motion planning methods for multi-limbed robots. In particular, we consider combined stance and reach tasks for robotic mechanisms whose limbs can be used either as legs or manipulator arms. An example of such a system is the *RoboSimian* robot participating in the DARPA Robotics Challenge (Figure 1). We develop relationships which model the key quasi-statics and kinematics of these mechanisms: the *stance map*, the *stance Jacobian*, and the *reach Jacobian*, as well as the *stance constrained center-of-mass Jacobian*. We also introduce characterizations of multi-limbed mechanism configurations in terms of the properties of these maps: *local dexterity* and *limbness*. This paper also introduces local planning methods which seek to balance the motion of legs, body, and arms of such mechanisms so as to realize manipulation goals while also maintaining awareness of stance stability issues. Examples with a simple planar model illustrate the methods.

I. INTRODUCTION



Fig. 1. RoboSimian. Designed by NASA\JPL-Caltech

Figure 1 depicts the *RoboSimian* robot, a competitor in the DARPA Robotics Challenge (DRC). Each of this robot's four limbs can be used either as a leg, or as a manipulator (a 3-fingered hand is attached to the distal end of each limb). To locomote over rough terrain, this vehicle can use standard quasi-static gait planning and coordination techniques. However, many tasks which this robot must accomplish involve using some limbs as legs to provide a stable stance, while one or more other limbs simultaneously carry out a manipulation task, such as turning a water valve, or picking up a fire hose.

In practice, this robot and its peers will initially be used in hazardous environments for activities, like disaster relief,

where there is a human operator in the loop, minimizing the need for complete high level autonomy. However, the joint motions required to move an end-effector as desired can be extremely counterintuitive due to the complex workspace of such mechanisms, their high number of mechanical degrees of freedom, and the need to account for stance stability throughout the task. Hence, *local planning methods* are needed to assist the human operators, and these methods would serve as a precursor to completely autonomous future operation of such vehicles.

The local planning problem that motivates this paper is the following:

Problem Statement. Consider a multi-limbed robot with $n \geq 4$ limbs in the presence of gravity. Let the end effectors at the end of the limbs involved in stance be called *feet*, and let the surfaces contacted by the feet be termed *footholds* or *holds*. Assume that 3 or more feet are placed on holds so as to form a quasi-statically stable stance. One limb is used to carry out a manipulation task. How should all of the limbs and joints be adjusted during the manipulation task so as to satisfy the manipulation task constraints while also accommodating postural stability? In particular, the robot should move so as to

- Maintain the same contacts at the footholds,
- Maintain a suitable stance stability margin,
- Satisfy end-effector task motion and force constraints.

This paper develops the basic kinematic relationships associated with these problems, and some preliminary planning solutions based on these relations. Our goal is to facilitate the provision of guarantees for arbitrary legged robots.

Relation to Prior Work. Quasi-static multi-legged locomotion has been well-studied, the foundation for which was laid in [1], [2] and built upon in many works, for example [3], [4]. The analysis of static equilibrium and quasi-static stability for legged robots with attention to contacts (friction, curvature etc.) has been approached from both theoretical [5]–[8] and computational [7], [9] standpoints. The local planning methods introduced in this paper build upon the prior characterizations of stance equilibria, as we seek to determine those motions which satisfy manipulation criteria while also attempting to maintain the center of mass within the support region.

The relationship between the center of mass motion and

joint motion is central to the problem of free floating space manipulators (e.g. [10]). In contrast to the space manipulator problem, the center of mass location for the systems that we study is not conserved, and its motion must also take into account the reaction forces generated by the stance limbs.

Manipulation from a legged robot platform has not been explored deeply, but some work has been done for humanoids ([11]–[13]). Accounting for the position of the center of mass is very important for those working with humanoids, and Ref. [14] details the relationship between joint velocities and center of mass motion in relation to notions of dynamic disturbance rejection; a number of other works build on this result. [15] provides methods for dynamic locomotion, and describes the construction of a whole body Jacobian that is applied as part of a hierarchical redundancy-resolution based controller.

We take advantage of the fact that multi-limbed standing manipulation has many similarities to dextrous manipulation by multi-fingered robotic hands. Many authors [4], [9], [16]–[18] have previously noted these similarities. One of the contributions of this paper is to make these analogies explicit.

The class of mechanisms studied in this paper invariably have many degrees of freedom, and often have internal motions that are analogous to the self motions of redundant manipulators. Our local planning methods borrow several concepts from the classical literature on manipulator redundancy resolution [19], [20]. But the governing kinematic equations have a slightly different linear algebraic structure, requiring a new set of formulae to be derived.

While sampling based planners have proven effective in many high dimensional problems [21], [22], we have found them to be slow in this particular problem: Robosiman has 28 joints - sampling without very special bias could be prohibitively inefficient¹. Moreover, a straightforward sampling planner would not take stance stability issues into account. Thus, there is a need to develop the appropriate physical relationships and local planning techniques for this domain.

Structure of the Paper. Section II develops the basic kinematic relationships that govern combined stance and reach. Section III induces some basic stance/reach properties which can be assessed using these relations. Section IV derives the stance-constrained center of mass Jacobian matrix. Section V introduces several local kinematic motion planning problems which build upon the results of Sections II and IV, while Section VI illustrates the methods on a simple planar multi-limbed mechanism.

II. MULTI-LIMBED STANCE AND REACH KINEMATICS

Figure 2 depicts an n -limbed robot with one limb used as a manipulator, while the other limbs provide a (potentially) quasistatically stable stance. The goal of this section is to derive the kinematic relationships which relate end-effector

¹e.g. RRT-connect often takes a very long time or fails altogether to move a particular end-effector towards a goal pose kinematically, even when mass and contact are not accounted for.

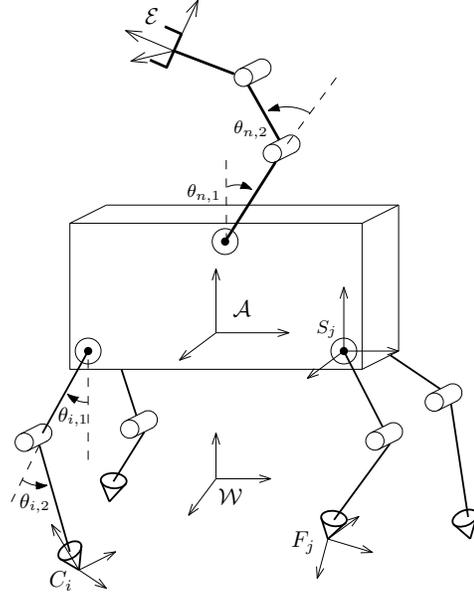


Fig. 2. Key Reference Frames for Stance and Reach.

motions to mechanism joint motions. Section IV will derive analogous relationships for center of mass motion. While we focus on the case of a single manipulation limb, the methods can be extended to multiple manipulating limbs.

Let a *world* (fixed) reference frame be denoted by \mathcal{W} . We assume that a terrain model is available from which one can derive the location, height, and contact normals associated to each foothold. A user-selected frame attached to the robot’s main body, or *abdomen*, is denoted by \mathcal{A} (see Fig. 2). For each limb, the *shoulder* reference frame, S_j , is located at the point where the j^{th} limb attaches to the abdomen body, and is *fixed* with respect to \mathcal{A} . Without loss of generality, the stance legs are indexed $j = 1, \dots, n-1$, while the reaching limb has index n . The end effector frame at the distal end of the n^{th} limb is denoted by \mathcal{E} . For limbs that are positioned on holds, the *foot frame*, denoted F_j , is rigidly attached to the foot body, with origin at the point of ground contact². The i^{th} *contact frame*, C_i , is fixed with origin at the contact between the i^{th} limb and the ground, and is oriented with the surface of the i^{th} hold. The joint angles corresponding to the i^{th} limb are $\vec{\theta}_i = (\theta_{i,1}, \dots, \theta_{i,m_i})^T$ where m_i is the number of joints in the i^{th} limb.

To find the relationship between end-effector motions and robot joint motions, let the homogeneous transformation between end-effector frame \mathcal{E} and fixed frame \mathcal{W} be denoted $g_{\mathcal{W}\mathcal{E}} \in se(3)$. Using Figure 2, $g_{\mathcal{W}\mathcal{E}}$ can be expanded as:

$$g_{\mathcal{W}\mathcal{E}} = g_{\mathcal{W}C_i} g_{C_i F_j} g_{F_j S_i} g_{S_i \mathcal{A}} g_{\mathcal{A} \mathcal{E}} \cdot \quad (1)$$

The twist coordinates of the end-effector velocity, with respect to \mathcal{W} , as seen in the end effector frame (a.k.a. ‘body’

²We assume point contact between the feet and ground in this paper, but the methods be readily extended to other types of contact

velocity), is ³

$$\widehat{V}_{w\varepsilon} = g_{w\varepsilon}^{-1} \dot{g}_{w\varepsilon} . \quad (2)$$

Hereafter, every velocity is assumed to be a body velocity unless otherwise noted by a superscript. Expanding Eq. (2) using Eq. (1), applying the chain rule, and converting to twist vector form yields:

$$\begin{aligned} V_{w\varepsilon}^{\varepsilon} &= \text{Ad}_{g_{\varepsilon c_i}} V_{w c_i} + \text{Ad}_{g_{\varepsilon f_i}} V_{c_i f_i} \\ &+ \text{Ad}_{g_{\varepsilon s_i}} V_{f_i s_i} + \text{Ad}_{g_{\varepsilon A}} V_{s_i A} + V_{A\varepsilon} . \end{aligned} \quad (3)$$

Velocities $V_{w c_i}$ and $V_{s_i A}$ are zero since they describe motion between frames that are relatively fixed. Letting $J_n(\vec{\theta}_n)$ denote the Spatial Jacobian matrix⁴ of the n^{th} limb (being used as a manipulator) $V_{A\varepsilon}$ can be expressed as

$$V_{A\varepsilon} = \text{Ad}_{g_{A\varepsilon}}^{-1} J_n(\vec{\theta}_n) \dot{\vec{\theta}}_n$$

Similarly, letting $J_i(\vec{\theta}_i)$ denote the Jacobian matrix of the i^{th} leg,

$$V_{f_i s_i} = -V_{s_i f_i}^s = -J_i(\vec{\theta}_i) \dot{\vec{\theta}}_i ,$$

The velocity of the foot frame with respect to the contact frame can be expanded as

$$\begin{aligned} V_{c_i f_i} &= V_{c_i A}^{f_i} + V_{A s_i}^{f_i} + V_{s_i f_i} \\ &= \text{Ad}_{g_{f_i A}} V_{c_i A} + \text{Ad}_{g_{f_i s_i}} J_i(\vec{\theta}_i) \dot{\vec{\theta}}_i \end{aligned} \quad (4)$$

where $V_{A s_i}^{f_i} = 0$ for all i since the shoulder frame is rigidly attached to the abdomen. Substituting these results into Eq. (3) yields

$$\begin{aligned} V_{w\varepsilon} &= \text{Ad}_{g_{f_i \varepsilon}}^{-1} (\text{Ad}_{g_{f_i s_i}} J_i(\vec{\theta}_i) \dot{\vec{\theta}}_i + \text{Ad}_{g_{f_i A}} V_{c_i A}) \\ &- \text{Ad}_{g_{\varepsilon s_i}} J_i(\vec{\theta}_i) \dot{\vec{\theta}}_i + \text{Ad}_{g_{A\varepsilon}}^{-1} J_n(\vec{\theta}_n) \dot{\vec{\theta}}_n \end{aligned} \quad (5)$$

Combining coordinate transformations and noting that the first and third terms in (5) cancel, one obtains:

$$V_{w\varepsilon} = \text{Ad}_{g_{\varepsilon A}} V_{c_i A} + \text{Ad}_{g_{A\varepsilon}}^{-1} J(\vec{\theta}_n) \dot{\vec{\theta}}_n . \quad (6)$$

At each foothold, we assume a *contact constraint*, which implies that the foot cannot move in directions along which forces can be applied and supported by the contact:

$$B_{c_i}^T V_{f_i c_i} = 0 \quad (7)$$

where B_{c_i} is the *wrench basis* at the i^{th} foot⁵.

Expanding the transform $g_{c_i f_i}$ in terms of a different chain of reference frames, one finds that

$$\begin{aligned} g_{c_i f_i} &= g_{f_i s_i} g_{s_i A} g_{A w} g_{w c_i} . \\ \widehat{V}_{f_i c_i} &= g_{c_i f_i}^{-1} \dot{g}_{c_i f_i} . \end{aligned}$$

³This paper uses the notation and methodology of Chapter 2 in [23]. In general, V_{ab}^c is a *twist* that specifies the velocity of frame a with respect to frame b observed in frame c .

⁴The *Spatial Jacobian* is defined and discussed in [23]

⁵The columns of the wrench basis matrix form a basis for all wrenches that can be applied at a contact, as expressed in the contact frame [23]. A wrench basis can model frictionless, point contact with Coulomb friction, soft contact, and many other types of ground contact models

Expanding out the right-hand side of the velocity, converting to twist vector form, and dropping velocities between relatively fixed frames results in:

$$\begin{aligned} V_{f_i c_i} &= \text{Ad}_{g_{s_i c_i}}^{-1} V_{f_i s_i} + \text{Ad}_{g_{A c_i}}^{-1} V_{s_i A} + \text{Ad}_{g_{w c_i}}^{-1} V_{A w} + V_{w c_i} \\ &= -\text{Ad}_{g_{s_i c_i}}^{-1} J(\vec{\theta}_i) \dot{\vec{\theta}}_i + \text{Ad}_{g_{w c_i}}^{-1} V_{A w} . \end{aligned} \quad (8)$$

Substituting Eq.s (4) and (8) into equation (7) and equating results yields:

$$\begin{aligned} B_{c_i}^T \text{Ad}_{g_{s_i c_i}}^{-1} J_i(\vec{\theta}_i) \dot{\vec{\theta}}_i &= B_{c_i}^T \text{Ad}_{g_{w c_i}}^{-1} V_{A w} \\ &= -B_{c_i}^T \text{Ad}_{g_{w c_i}}^{-1} \text{Ad}_{g_{w A}} V_{w A} \\ &= -B_{c_i}^T \text{Ad}_{g_{A c_i}}^{-1} V_{w A} . \end{aligned}$$

Next, define the *Stance Jacobian* matrix (which is analogous to the *hand Jacobian* in multi-fingered grasping) as

$$J_S(x_0, \vec{\theta}) = \begin{bmatrix} B_{c_1}^T \text{Ad}_{g_{s_1 c_1}}^{-1} J_1(\vec{\theta}_1) & & 0 \\ & \ddots & \\ 0 & & B_{c_{n-1}}^T \text{Ad}_{g_{s_{n-1} c_{n-1}}}^{-1} J_{n-1}(\vec{\theta}_{n-1}) \end{bmatrix} .$$

where $\vec{\theta} = (\vec{\theta}_1, \dots, \vec{\theta}_{n-1})$, and x_0 describes the location of the abdomen frame relative to the world frame (which implicitly defines the contact points and normals via the terrain model). Now equation (7) becomes

$$J_S(x_0, \vec{\theta}) \dot{\vec{\theta}} = - \left[B_{c_1}^T \text{Ad}_{g_{A c_1}}^{-1} \cdots B_{c_{n-1}}^T \text{Ad}_{g_{A c_{n-1}}}^{-1} \right] V_{w A} .$$

We define the *stance map* as

$$S = - \left[\text{Ad}_{g_{A c_1}}^T B_{c_1} \cdots \text{Ad}_{g_{A c_{n-1}}}^T B_{c_{n-1}} \right] .$$

The stance map transforms contact forces to wrenches on the abdomen –it is analogous to the grasp map in multi-fingered robotic manipulation. Using these definitions yields a more descriptive version of equation (7):

$$J_S(x_0, \vec{\theta}) \dot{\vec{\theta}} = S^T V_{w A} . \quad (9)$$

Eq. (9) describes the relationship between the motions of leg joints and the motion of the abdomen frame, assuming that the footholds are maintained.

We return to the consideration of the end-effector's velocity. Applying the the fact that contacts and the world frame are fixed relative to each other to Eq. (6) and rearranging yields the following important relationship:

$$\text{Ad}_{g_{A\varepsilon}} V_{w\varepsilon} = V_{c_i A} + J_n(\vec{\theta}_n) \dot{\vec{\theta}}_n . \quad (10)$$

Now, define the *stance-constrained Reach Jacobian*, or simply *Reach Jacobian* as follows:

$$J_{\mathcal{R}}(\Theta, x_0) = \left[J_S(x_0, \vec{\theta}) \quad S^T J_n(\vec{\theta}_n) \right] \quad (11)$$

where $\Theta = (\vec{\theta}_1, \dots, \vec{\theta}_n)$ is the vector of all mechanism joint velocities. Using (9) and (10), results in the following new relationship:

$$J_{\mathcal{R}}(\Theta, x_0) \dot{\Theta} = S^T V_{w\varepsilon}^A \quad (12)$$

We call (12) the *stance constrained Reach Constraint*, or *Reach Constraint*, as it describes all of the ways that the robot can reach for a certain goal by coordinating the motions of its legs and its manipulator limb (assuming that the footholds are maintained).

III. PROPERTIES OF A STANCE

This section provides some *local*, or *instantaneous* notions of the properties of the combined stance/reach system analyzed in the last section. Most of these properties are analogous to those of multi-fingered robotic grasps, with modest extensions required to handle the presence of the reaching limb.

Definition 1: A limbed robot stance is **limber** at a configuration $(\vec{\theta}, x_0)$ when for any motion of the abdomen $V_{\mathcal{W}\mathcal{A}}^{\mathcal{A}}$, there exists $\vec{\theta}$ satisfying equation (9).

Definition 2: A limbed robot stance is **locally dexterous** with respect to a free limb at a configuration (Θ, x_0) when for any motion of the end effector, $V_{\mathcal{W}\mathcal{E}}^{\mathcal{A}}$, there exists $\dot{\Theta}$ satisfying (12).

A robot is limber if it can locally move its abdomen in all directions via leg joint motions while maintaining its footholds, and it is locally dextrous if the reaching end-effector can instantaneously move in any direction by allowed mechanism joint motions. The following propositions result directly from Definitions 1 and 2, as well as Eq.s (9) and (12).

Proposition 1: Suppose that S is onto. A stance is limber at a configuration (Θ, x_0) if and only if

$$\text{Range}(S^T) \subseteq \text{Range}(J_S(\vec{\theta}, x_0)).$$

Proposition 2: Suppose that S is onto. A stance is locally dexterous with respect to a limb at a configuration (Θ, x_0) if and only if

$$\text{Range}(S^T) \subseteq \text{Range}(J_{\mathcal{R}}(\Theta, x_0)). \quad (13)$$

Furthermore, for a given configuration, a robot is locally dexterous if any of the following hold

- i) it is limber.
- ii) $J_n(\theta_n)$ is onto the pre-image of $\text{Range}(S^T)$ (or equivalently, onto the complement of $\text{Null}(S^T)$).

This fact is a consequence of Eq. (12). The most interesting situations occur when

- The robot is locally dexterous, but not limber, and the free limb's Jacobian matrix is not onto the complement of $\text{Null}(S^T)$. At these configurations, both the body and the reaching arm must work in concert to realize arbitrary end-effector motions.
- The robot is limber and the free limb Jacobian matrix is onto the pre-image of the $\text{Range}(S^T)$. In this case, there are many ways of producing the desired end effector velocity. The solution might depend on other goals, and this possibility is explored in Section V

Other properties which are standard in multi-fingered grasping also extend to this situation. Let FC_{c_i} denote the friction

cone associated with each contact c_i , and let FC denote the totality of friction cone constraints: $FC = FC_{c_1} \times \dots \times FC_{c_{n-1}}$. Assuming that the robot is standing on an immovable object or solid ground, The net wrench on the abdomen body applied at the contacts is:

$$F_{\mathcal{W}} = S\vec{f} \quad \vec{f} \in FC$$

where $\vec{f} = (f_1, \dots, f_{n-1})^T$ are the contact forces at the footholds.

Definition 3: A stance is **wrench-resistant** if it can counteract any external wrench F_e , i.e. there exists $f \in FC$ such that

$$Sf = -F_e$$

Proposition 3: A stance is wrench-resistant if and only if S restricted to FC is full rank, i.e.

$$S(FC) = \mathbb{R}^6$$

IV. THE CENTER OF MASS STANCE JACOBIAN

It is well known that quasi-static stability of a given stance depends upon the location of the mechanism's center of mass relative to the *support region* generated by the stance's footholds [1], [5]–[7], [9]. Hence, it is crucial to find a relationship between mechanism joint motions and center-of-mass motion. This relationship can then be used during the task planning process to maintain or enhance a suitable stability margin while realizing manipulation goals.

The center of mass of a system of N rigid bodies, $C \in \mathbb{R}^3$, is given by $C = (1/M) \sum_{i=1}^N \rho_i m_i$ where m_i and ρ_i are the mass and position of the i^{th} object respectively, and M is the total system mass: $M = \sum_{i=1}^N m_i$. Letting $g_{\mathcal{W}i}$ denote the location of each bodies's center of mass frame, the system's center of mass position in homogenous coordinates is

$$\bar{C} = g_{\mathcal{W}C} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

where $g_{\mathcal{W}C}$ is the displacement of a reference frame located at the system's center of mass. Hereafter, the center of mass will be described in terms of $g_{\mathcal{W}C}$, with the understanding that only the translational component of $g_{\mathcal{W}C}$ is relevant to the analysis.

Let $m_{\mathcal{A}}$ be the mass of the abdomen, located at the origin of the frame \mathcal{A} . The mass of the j^{th} link on the i^{th} limb is denoted $m_{i,j}$, and $g_{\mathcal{W}(i,j)}$ maps positions in homogenous coordinates from the $(i,j)^{\text{th}}$ link frame (whose origin is located at the link's center of mass) to \mathcal{W} .

$$g_{\mathcal{W}C} = \frac{m_{\mathcal{A}}}{M} g_{\mathcal{W}\mathcal{A}} + \frac{1}{M} \sum_{i=1}^N \sum_{j=1}^{n_i} m_{i,j} g_{\mathcal{W}(i,j)}$$

where M is the total robot mass. Using this notation, the velocity of the center of mass frame can be expressed as:

$$V_{\mathcal{W}C}^C = \frac{m_{\mathcal{A}}}{M} \text{Ad}_{g_{\mathcal{C}\mathcal{A}}} V_{\mathcal{W}\mathcal{A}}^{\mathcal{A}} + \frac{1}{M} \sum_{i=1}^N \sum_{j=1}^{n_i} m_{i,j} \text{Ad}_{g_{\mathcal{C}(i,j)}} V_{\mathcal{W}(i,j)}^{(i,j)} \quad (14)$$

where C denotes ‘center-of-mass’. Define the *link Jacobian* as the matrix which maps the velocities of the first j joints in the i^{th} limb to the velocity of the j^{th} link in the \mathcal{A} frame.

$$J_{i,j}(\theta_{i,1}, \dots, \theta_{i,j})$$

With this notation,

$$\begin{aligned} V_{\mathcal{W}(i,j)}^{\mathcal{W}} &= V_{\mathcal{W}\mathcal{A}}^{\mathcal{W}} + \text{Ad}_{g_{\mathcal{W}\mathcal{A}}} V_{\mathcal{A}(i,j)}^{\mathcal{A}} \\ &= V_{\mathcal{W}\mathcal{A}}^{\mathcal{W}} + \text{Ad}_{g_{\mathcal{W}s_i}} J_{i,j}(\theta_{i,1}, \dots, \theta_{i,j}) \dot{\theta}_{i \rightarrow j}. \end{aligned}$$

where

$$\theta_{i \rightarrow j} = (\theta_{i,1}, \dots, \theta_{i,j})$$

is shorthand notation to denote the fragment of the i^{th} limb’s joint velocity vector containing the first j joint velocities. Introducing the link Jacobian into (14), transforming (14) to the \mathcal{A} frame, and then multiplying both sides by the transpose of the stance map (so that the stance Jacobian can be used), one obtains:

$$S^T V_{\mathcal{W}c}^{\mathcal{A}} = J_S(\theta) \dot{\theta} + \frac{1}{M} \sum_{i=1}^N \sum_{j=1}^{n_i} m_{i,j} S^T \text{Ad}_{g_{\mathcal{A}s_i}} J_{i,j}(\theta_{i \rightarrow j}) \dot{\theta}_{i \rightarrow j} \quad (15)$$

Recall [23] that for a serial chain manipulator, the spatial Jacobian matrix takes the form:

$$J_{st}^s = [\xi_1 \ \dots \ \xi_n']$$

where

$$\xi_i = \text{Ad}_{(e^{\hat{\xi}_1 \theta_1} \dots e^{\hat{\xi}_{i-1} \theta_{i-1}})} \xi_i.$$

Define a *mass weighted Jacobian* matrix as follows

$$\bar{J}_k = \left[(\sum_{j=1}^{n_k} \frac{m_{k,j}}{M}) \xi_{1,k} \ \dots \ (\sum_{j=i}^{n_k} \frac{m_{k,j}}{M}) \xi'_{i,k} \ \dots \ (\frac{m_{k,n_k}}{M}) \xi'_{k,n_k} \right]$$

Equation (15) becomes

$$S^T V_{\mathcal{W}c}^{\mathcal{A}} = J_S(\theta) \dot{\theta} + S^T \sum_{i=1}^n \text{Ad}_{g_{\mathcal{A}s_i}} \bar{J}_i(\theta_i) \dot{\theta}_i$$

which can be written in the following compact form

$$S^T V_{\mathcal{W}c}^{\mathcal{A}} = J_C(\theta) \dot{\theta}$$

by introducing the *Stance-Constrained Center of Mass Jacobian*, Equation (16), where

$$\tilde{J}_k = \left[(\sum_{j=1}^{n_k} \frac{m_{k,j} + M}{M}) \xi_{k,1} \ \dots \ (\frac{m_{k,n_k} + M}{M}) \xi'_{k,n_k} \right].$$

V. LOCAL MOTION PLANNING

This section uses the relationships developed in the previous section to develop *local* planning methods for stance-constrained manipulation. Intuitively speaking, the goal of these local planning methods is to move the end-effector so as to satisfy task goals and/or constraints, while also simultaneously moving the center of mass so as to enhance, maintain, or gracefully degrade stance stability. There are many different conceivable types of problems and analytical approaches that one may take to realize these intuitively obvious goals. The remainder of this section introduces, roughly in order of increasing complexity, a few basic local

task planning problems and their relation solutions. These solutions would be useful components of a local planning toolkit for stance-constrained manipulation.

A. Problem #1: The Minimum Norm Solution

This first solution, which ignores issues of stance equilibrium or stability, primarily establishes results which will be useful below. Given a stance for the first $(n-1)$ joints, and a desired end-effector velocity, $V_{\mathcal{W}\mathcal{E}}$, for the n^{th} manipulating limb, the goal is to find the minimum norm joint velocities that exactly move the end-effector at this desired velocity. This problem is exactly analogous to the classical minimum norm redundancy resolution solution, except that the different kinematic constraint between the end-effector and the robot’s joints yield a solution with slightly different algebraic form. Formally, we seek the solution to the problem

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2} \dot{\Theta}^T \dot{\Theta} \\ \text{subject to} \quad & S^T \tilde{V}_{\mathcal{W}\mathcal{E}}^{\mathcal{A}} = J_{\mathcal{R}}(x_0, \Theta) \dot{\Theta} \end{aligned} \quad (17)$$

The solution is

$$\dot{\Theta}^* = J_{\mathcal{R}}^+ S^T \tilde{V}_{\mathcal{W}\mathcal{E}}^{\mathcal{A}}$$

where $J_{\mathcal{R}}^+$ is the Moore-Penrose pseudo-inverse, and the arguments of $J_{\mathcal{R}}$ are dropped for clarity. This solution can be readily found via lagrange multipliers. Let

$$\mathcal{L} = (1/2) \dot{\Theta}^T \dot{\Theta} + \nu^T (S^T \tilde{V}_{\mathcal{W}\mathcal{E}}^{\mathcal{A}} - J_{\mathcal{R}}(x_0, \Theta) \dot{\Theta}).$$

A necessary condition for a minimizer $\dot{\Theta}^*$ is that $\partial \mathcal{L} / \partial \dot{\Theta}^* = 0$, which implies that

$$\dot{\Theta}^* - J_{\mathcal{R}}(x_0, \Theta)^T \nu = 0.$$

Using this equation, $\dot{\Theta}^*$ is expressed in terms of ν and substituted into \mathcal{L} to obtain the lagrange dual function $g(\nu)$. The optimal dual variable ν^* can be found by solving $\frac{\partial g}{\partial \nu}(\nu^*) = 0$. Thereafter, $\dot{\Theta}^*$ is known. Note that one can solve the analogous problem where the center of mass must move in a desired direction using the minimum norm solution, without regard to the end-effector motion, by changing indices. The general solution to the kinematic relationship $S^T \tilde{V}_{\mathcal{W}\mathcal{E}}^{\mathcal{A}} = J_{\mathcal{R}}(x_0, \Theta) \dot{\Theta}$ is comprised of the minimum norm solution and any vector in the null space of $J_{\mathcal{R}}(x_0, \Theta)$:

$$\dot{\Theta}^* = J_{\mathcal{R}}^+ S^T \tilde{V}_{\mathcal{W}\mathcal{E}}^{\mathcal{A}} + (I - J_{\mathcal{R}}^+ J_{\mathcal{R}}) z$$

with z an arbitrary vector having the same dimension as $\dot{\Theta}$.

B. Problem #2: The Task Priority Solution

This problem is motivated by the classical Task Priority solution for redundant robot manipulators [19]. The goal is to require the manipulator limb’s end-effector to track a desired velocity, and as a second priority, have the center of mass move as closely as possible in a desired direction—one that presumably maintains or enhances stance stability.

find $\dot{\Theta}$

$$\begin{aligned} \text{subject to} \quad & \text{Priority 1 : } S^T \tilde{V}_{\mathcal{W}\mathcal{E}}^{\mathcal{A}} = J_{\mathcal{R}}(x_0, \Theta) \dot{\Theta} \quad (18) \\ & \text{Priority 2 : } S^T \tilde{V}_c^{\mathcal{A}} = J_C(x_0, \Theta) \dot{\Theta} \end{aligned}$$

$$J_C(\theta) = \begin{bmatrix} B_{c_1}^T \text{Ad}_{g_{c_1 s_1}} \tilde{J}_1(\theta_1) & B_{c_1}^T \text{Ad}_{g_{c_1 s_2}} \tilde{J}_2(\theta_2) & \dots & B_{c_1}^T \text{Ad}_{g_{c_1 s_{n-1}}} \tilde{J}_{n-1}(\theta_{n-1}) & \dots & B_{c_1}^T \text{Ad}_{g_{c_1 s_n}} \tilde{J}_n(\theta_n) \\ B_{c_2}^T \text{Ad}_{g_{c_2 s_1}} \tilde{J}_1(\theta_1) & B_{c_2}^T \text{Ad}_{g_{c_2 s_2}} \tilde{J}_2(\theta_2) & & & & \\ \vdots & & \ddots & & & \vdots \\ \dots & & & B_{c_M}^T \text{Ad}_{g_{c_{n-1} s_{n-1}}} \tilde{J}_{n-1}(\theta_{n-1}) & \dots & B_{c_{n-1}}^T \text{Ad}_{g_{c_{n-1} s_n}} \tilde{J}_n(\theta_n) \end{bmatrix} \quad (16)$$

Adapting the classical redundancy resolution solution to the slightly more complicated constraint formulae, one obtains:

$$\dot{\Theta}^* = J_{\mathcal{R}}^+ S^T \tilde{V}_{w\varepsilon}^A + [J_C(I - J_{\mathcal{R}}^+ J_{\mathcal{R}})]^+ (S^T \tilde{V}_c^A - J_C J_{\mathcal{R}}^+ S^T \tilde{V}_{w\varepsilon}^A) \quad (19)$$

It follows that for the problem to be feasible, i.e., that the center of mass velocity can be exactly tracked, the stance map must be full rank (the robot is wrench-resistant), and there must exist some $x \in \mathbb{R}^{m \times n}$ such that

$$\tilde{V}_{w\varepsilon}^A = (S^T)^+ J_C \left(J_{\mathcal{R}}^+ S^T \tilde{V}_{w\varepsilon}^A + (I - J_{\mathcal{R}}^+ J_{\mathcal{R}}) x \right)$$

Of course, the priorities can be reversed, and the same solution used by changing indices.

This solution could be useful in the following way. Let the desired center of mass velocity, \tilde{V}_c^A , be specified as a unit vector pointing away from the nearest face of the support polygon (into the interior of the polygon). Then, solution (19) will track the desired end-effector velocity, if at all possible, while attempting to maintain, or even enhance, the stance's margin.

C. Problem #3: The Balanced Priority Solution

In many situations, it may be difficult to track the desired manipulation trajectory, while also maintaining stability, and minimizing joint motions. Hence, in practice it makes sense to define a weighted optimization goal that blends the priorities of tracking, stance, and joint motions:

$$\begin{aligned} \text{minimize} \quad & \alpha \|V_{w\varepsilon}^A - \tilde{V}_{w\varepsilon}^A\|_2^2 + \beta \|V_{cm} - \tilde{V}_{cm}\|_2^2 + \gamma \|\dot{\Theta}\|_2^2 \\ \text{subject to} \quad & J_{\mathcal{R}}(\Theta, x_0) \dot{\Theta} = S^T V_{w\varepsilon}^A \\ & J_C(\Theta, x_0) \dot{\Theta} = S^T V_{w\varepsilon}^A \end{aligned} \quad (20)$$

This solution to this problem can be tuned to vary the importance of different goals during task execution, or to approximate a particular problem from the previous section. Define

$$x = \begin{pmatrix} V_{w\varepsilon}^A \\ V_c^A \\ \dot{\Theta} \end{pmatrix} \quad b = \begin{bmatrix} \tilde{V}_{w\varepsilon}^A \\ \tilde{V}_{cm} \\ 0 \end{bmatrix} \quad P = \begin{bmatrix} \alpha I_n & 0 & 0 \\ 0 & \beta I_n & 0 \\ 0 & 0 & \gamma I_m \end{bmatrix}$$

$$F = \begin{bmatrix} S^T & 0 & -J_{\mathcal{R}}(x_0, \Theta) \\ 0 & S^T & -J_C(x_0, \Theta) \end{bmatrix}$$

In these variables, Equation (20) is equivalent to

$$\begin{aligned} \text{minimize} \quad & x^T P x - x^T b \\ \text{subject to} \quad & F x = 0 \end{aligned} \quad (21)$$

This is an equality constrained quadratic program (ECQP), and it is easy to show that it has a unique solution whenever

$\alpha, \beta, \gamma \geq 0$ and F has full row rank. Geometrically, the solution is a projection of the vector b onto the null-space of F . The nature of the projection is defined by the matrix P above; when the weights are nonzero but equal, then the projection is orthogonal. When the weights are non-zero but not equal, the projection is *oblique* — it is stretched along components of x and b with large weights, and shrunk along the components with small weights. The problem sometimes has solutions when weights are set to zero, in these cases the solution though correct is useless as the corresponding variable is shrunk to zero.

The solution to this problem can be computed explicitly by enforcing the KKT optimality conditions. Strong duality holds in this case whenever the problem is feasible, and the optimal x^* can be obtained by solving the dual problem. Let

$$\begin{bmatrix} G_{11} & G_{12} \\ G_{12}^T & G_{22} \end{bmatrix} = \begin{bmatrix} \frac{1}{\alpha} S^T S + \frac{1}{\beta} J_{\mathcal{R}}(x_0, \Theta) J_{\mathcal{R}}(x_0, \Theta)^T & \frac{1}{\gamma} J_{\mathcal{R}}(x_0, \Theta) J_C(x_0, \Theta)^T \\ \frac{1}{\gamma} J_C(x_0, \Theta) J_{\mathcal{R}}(x_0, \Theta)^T & \frac{1}{\beta} S^T S + \frac{1}{\gamma} J_C(x_0, \Theta) J_C(x_0, \Theta)^T \end{bmatrix}$$

Define H to be the Schur complement of G_{11} in the 'G' matrix above,

$$H = G_{11} - G_{12} G_{22}^{-1} G_{12}^T$$

Define

$$\begin{aligned} \nu_2^* &= H^{-1} \left(\frac{1}{\beta} S^T \tilde{V}_c^A - \frac{1}{\alpha} G_{12}^T G_{11}^{-1} S^T \tilde{V}_{w\varepsilon}^A \right) \\ \nu_1^* &= G_{11}^{-1} \left(\frac{1}{\alpha} S^T \tilde{V}_{w\varepsilon}^A - \frac{1}{\alpha} G_{12} S^T \tilde{V}_c^A \right) \end{aligned}$$

(these correspond to optimal Lagrange multipliers). Then, since $x = P^{-1}(b + F^T \nu^*)$ it follows that

$$\begin{pmatrix} V_{w\varepsilon}^A \\ V_{cm} \\ \dot{\Theta} \end{pmatrix} = \begin{pmatrix} \frac{1}{\alpha} \tilde{V}_{w\varepsilon}^A + S \nu_1^* \\ \frac{1}{\beta} \tilde{V}_c^A + S \nu_2^* \\ -J_{\mathcal{R}}(x_0, \Theta) \nu_1^* - J_C(x_0, \Theta) \nu_2^* \end{pmatrix}$$

Proposition 4: (Sufficient condition for unique optimizer) If $\text{Null}(J_{\mathcal{R}}(\Theta, x_0)) \cap \text{Null}(J_C(\Theta, x_0)) = \{\mathbf{0}\}$, then F has full row rank.

This follows from the fact that $\text{Null}(J_{\mathcal{R}}(\Theta, x_0)) = \text{Range}(J_{\mathcal{R}}(\Theta, x_0)^T)^\perp$, and likewise for $J_C(\Theta, x_0)$. This fact provides the following guarantee: *provided there is no local motion that keeps the end effector and the center of mass stationary, (21) has a unique solution.* It is very unlikely that such motion exists.

D. Possible Extensions

Using the relationships derived in this paper, and starting from the formulations in the first part of this section, one can extend these approaches to incorporate many other issues into the local motion planning problem. For example, it is known that quasi-static equilibrium postures of a multi-legged robot lie inside a vertical prism, whose boundary can be approximated to arbitrary precision as a polytope. Thus, one could formulate this constraint in the form:

$$AV_c^A \preceq c. \quad (22)$$

This constraint, coupled with Equation (21), yields a convex quadratic problem which is practically solvable. Note that the presence of obstacles can also be formulated in the style of Equation (22), with the end-effector velocity substituted for the center of mass velocity, allowing for the end-effector to simultaneously avoid obstacle while carrying out its manipulator task.

VI. EXAMPLES

We consider a planar 3-limbed robot, somewhat analogous to Robosimian, shown in figure 3. This simple robot can walk quasi-statically, maintaining a support region with non-zero width. We assume the robot stands on point contacts with friction. The limb links all have mass equal to one tenth that of the abdomen. We consider two different motions, one largely translating the end-effector left/right, and one translating the end-effector vertically. The end-effector and mass center motions produced by the minimum norm solution and the balanced priority solution (the equality constrained quadratic programming approach), with priority weights $\alpha = 20$, $\beta = 50$, $\gamma = 200$ are compared. These examples are *kinematically* simulated.⁶

The simulations use a simple controller: the robot repeatedly adjusts its joint angles by $'C\Delta\theta'$, where $\Delta\theta$ is suggested by the particular optimization based on the direction to the goal, and C is a scaling coefficient that reduces monotonically as the end effector reaches the goal. In reality, we imagine that a trajectory curve from the start to finish would be specified for the end effector (and perhaps the center of mass if the ECQP method is being used) the end-effector (resp. center of mass) would servo about this trajectory by matching the tangent at every step.

The results are shown in figure 4. In each case, the robot's initial configuration is shown in black, intermediate positions are green, and the final position is blue. The end-effector trajectory is shown in red, and the center of mass motion is given by a solid streak of black circles. It is evident that the QP method results in smooth motion of both end-effector and center-of-mass. The minimum norm solution moves towards the goal more aggressively, jerking the center of mass and end-effector. The QP method results in a much smaller net deviation of the center of mass than the minimum norm approach.

⁶Whereas in reality the robot would maintain contact due to physics, here we ensure that applied joint-angle changes lie in the null-space of the Jacobian of the *structure equation* (see [23]).

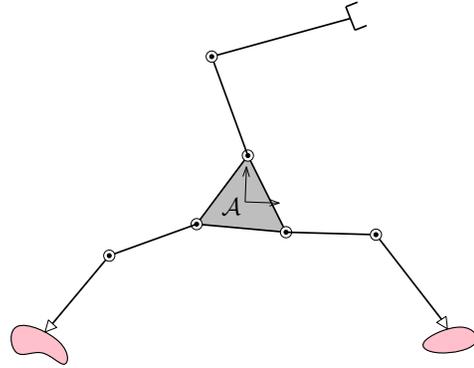


Fig. 3. A Planar 3-legged Example.

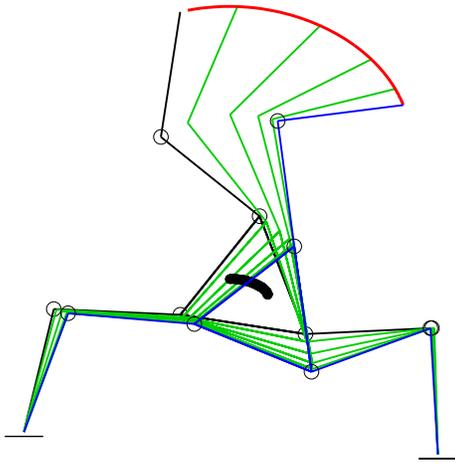
VII. CONCLUSION AND FUTURE WORK

This paper provided a framework for planning with multi-limbed robots during tasks that combine quasistatic balance and reach. The kinematics that govern the relationship between joint velocities and the velocity of a given point on the robot are provided. The motion of the center of mass is also explained, through the derivation of the stance-constrained center of mass Jacobian. The derived results are very general, and can be applied to any robot on almost any terrain. We also developed local motion planning methods that are analogous to classical redundancy resolution methods, but adapted for the more complex kinematic relationships which describe the motions of these mechanisms. Ongoing work aims to implement this framework on the RoboSimian robot.

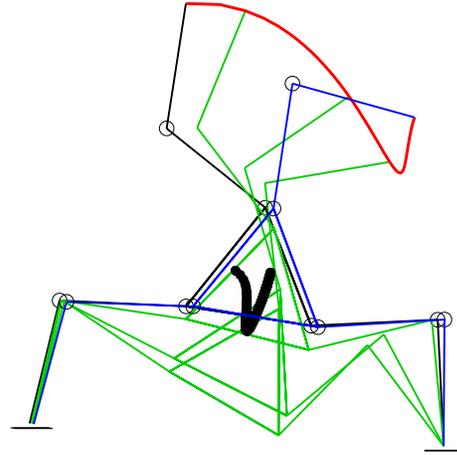
Issues related to forces, including the formal statement of static equilibrium follow very naturally from this work, as wrenches and twists are dual. These issues will be explored deeply in forthcoming work.

REFERENCES

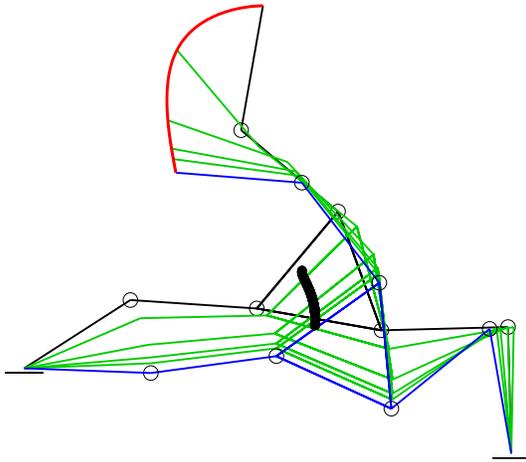
- [1] R. B. McGhee and A. A. Frank, "On the stability properties of quadruped creeping gaits," *Mathematical Biosciences*, vol. 3, pp. 331–351, 1968.
- [2] A. A. Frank, "Automatic control systems for legged locomotion machines," 1968.
- [3] R. B. McGhee and G. I. Iswandhi, "Adaptive locomotion of a multi-legged robot over rough terrain," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 9, no. 4, pp. 176–182, 1979.
- [4] K. Waldron, "Force and motion management in legged locomotion," *Robotics and Automation, IEEE Journal of*, vol. 2, no. 4, pp. 214–220, 1986.
- [5] Y. Or and E. Rimon, "Robust multiple-contact postures in a two-dimensional gravitational field," in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 5. IEEE, 2004, pp. 4783–4788.
- [6] E. Rimon, R. Mason, J. W. Burdick, and Y. Or, "A general stance stability test based on stratified morse theory with application to quasi-static locomotion planning," *Robotics, IEEE Transactions on*, vol. 24, no. 3, pp. 626–641, 2008.
- [7] Y. Or and E. Rimon, "Computation of multiple-contact frictional equilibrium postures in three-dimensional gravitational environments," *Department of Mechanical Engineering*, 2006.
- [8] R. Mason, E. Rimon, and J. Burdick, "Stable poses of 3-dimensional objects," in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 1. IEEE, 1997, pp. 391–398.
- [9] T. Bretl and S. Lall, "Testing static equilibrium for legged robots," *Robotics, IEEE Transactions on*, vol. 24, no. 4, pp. 794–807, 2008.
- [10] E. Papadopoulos and S. Dubowsky, "Dynamic singularities in free-floating space manipulators," in *Space Robotics: Dynamics and Control*. Springer, 1993, pp. 77–100.



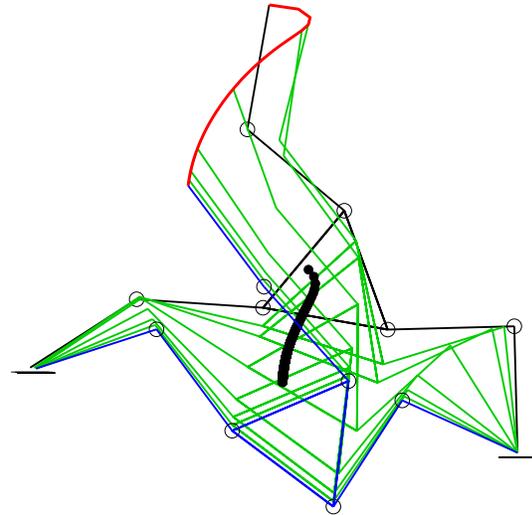
(a) ECQP, Goal at $(1, 6)$. COM deviates from initial position by 0.71 units.



(b) Minimum Norm, Goal at $(1, 6)$. COM deviates from initial position by 1.1 units.



(c) ECQP, Goal at $(-5, 4)$. COM deviates from initial position by 0.88 units.



(d) Minimum Norm, Goal at $(-5, 4)$. COM deviates from initial position by 1.82 units.

Fig. 4. Planning for a Planar Robot. A comparison of Minimum Norm solution and the Quadratic Program. The right foot of the robot is located at $(0, 0)$, and the end-effector's initial position is at $(7, -4)$.

- [11] K. Harada, S. Kajita, F. Kanehiro, K. Fujiwara, K. Kaneko, K. Yokoi, and H. Hirukawa, "Real-time planning of humanoid robot's gait for force-controlled manipulation," *Mechatronics, IEEE/ASME Transactions on*, vol. 12, no. 1, pp. 53–62, 2007.
- [12] K. Harada, S. Kajita, K. Kaneko, and H. Hirukawa, "Dynamics and balance of a humanoid robot during manipulation tasks," *Robotics, IEEE Transactions on*, vol. 22, no. 3, pp. 568–575, 2006.
- [13] J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba, and H. Inoue, "Motion planning for humanoid robots," in *Robotics Research*. Springer, 2005, pp. 365–374.
- [14] T. Sugihara and Y. Nakamura, "Whole-body cooperative balancing of humanoid robot using cog jacobian," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3. IEEE, 2002, pp. 2575–2580.
- [15] A. Shkolnik and R. Tedrake, "Inverse kinematics for a point-foot quadruped robot with dynamic redundancy resolution," in *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, 2007, pp. 4331–4336.
- [16] V. Kumar and K. Waldron, "Sub-optimal algorithms for force distribution in multifingered grippers," in *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, vol. 4. IEEE, 1987, pp. 252–257.
- [17] K. Hauser, T. Bretl, J.-C. Latombe, and B. Wilcox, "Motion planning for a six-legged lunar robot," in *Algorithmic Foundation of Robotics VII*. Springer, 2008, pp. 301–316.
- [18] A. M. Johnson, G. C. Haynes, and D. E. Koditschek, "Standing self-manipulation for a legged robot," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 272–279.
- [19] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *Int. J. Robotics Research*, vol. 6, no. 2, pp. 2–15, 1987.
- [20] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," *J. Intelligent and Robotic Systems*, vol. 3, pp. 201–212, 1990.
- [21] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.
- [22] J. J. Kuffner Jr and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2. IEEE, 2000, pp. 995–1001.
- [23] R. M. Murray and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- [24] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.