Leveraging Cone Double Description for Multi-contact Stability of Humanoids with Applications to Statics and Dynamics

Stéphane Caron† Quang-Cuong Pham‡ Yoshihiko Nakamura*
* Department of Mechano-Informatics, University of Tokyo, Japan
† School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore

Abstract—We build on previous works advocating the use of the Gravito-Inertial Wrench Cone (GIWC) as a general contact stability criterion (a “ZMP for non-coplanar contacts”). We show how to compute this wrench cone from the friction cones of contact forces by using an intermediate representation, the surface contact wrench cone, which is the minimal representation of contact stability for each surface contact. The observation that the GIWC needs to be computed only once per stance leads to particularly efficient algorithms, as we illustrate in two important problems for humanoids: “testing robust static equilibrium” and “time-optimal path parameterization”. We show, through theoretical analysis and in physics simulations, that our method is more general and/or outperforms existing ones.

I. INTRODUCTION

Planning motions for legged robots requires taking into account three types of constraints: (i) avoid self-collisions and undesired collisions with the environment; (ii) respect actuator limits; (iii) keep balance (or avoid falling). While type (i) and type (ii) constraints are also experienced by fixed-base robots, type (iii) constraints are peculiar to legged robots and will constitute the main focus of this paper.

One way for legged robots to keep balance is to ensure that the links through which they make contact do not move with respect to the environment. A popular approach to guarantee this condition is to ensure that the Zero Moment Point (ZMP) be contained within the convex hull of the support area. This approach however presents several serious limitations:

- it only applies when all contacts are coplanar, and thus cannot be used when multiple non-coplanar contacts are involved;
- it only constraints 2 out of the 6 degrees of freedom (DoF) of a surface contact. More precisely, if the normal direction to the contact surface is denoted Z, then the rotation around X and Y are constrained by the ZMP condition, but nothing can be said about translations along X, Y, Z, as well as rotations around Z. This limitation is particularly serious at low friction.

There have been many attempts to generalize the ZMP condition to overcome the above limitations. One approach is to consider individual contact forces distributed at the contact surfaces (see e.g., [6, 7, 8]), or surface contact wrenches [3], and require that these forces (or wrenches) satisfy the Coulomb condition. However, this approach lacks the concision and computational efficiency associated with the ZMP condition because of the large numbers of variables to consider (three times the number of individual contact forces or six times the number of contact wrenches).

Analyzing separately the unactuated part of the equation of motion (corresponding to the position and orientation of the base-link) reveals that, under the assumption that torque limits are sufficiently large, the stability of the contacts depends only on the gravo-inertial wrench [9]. This wrench, an object of dimension six, should lie in a convex cone (the Gravito-Inertial Wrench Cone or GIWC) generated by the contact constraints. This condition was proposed as a “universal stability criterion” by the authors of [6]. However, the representation of the gravo-inertial wrench that they used at that time was highly redundant (all contact forces as input variables) and their proof of stability was incorrect [4]. A derivation of the GIWC from contact forces was recently given in [10, 11], where the authors also proposed an algorithm to compute the GIWC using an efficient implementation of the double description method [12]. In both cases, the authors neglected the angular momentum: [10] assumed a point-mass model at the CoM, i.e., only the translational DoF of a surface contact. More precisely, if the normal direction to the contact surface is denoted Z, then the rotation around X and Y are constrained by the ZMP condition, but nothing can be said about translations along X, Y, Z, as well as rotations around Z [3]. This limitation is particularly serious at low friction [4, 5].

1Equations (8) and (9) of [6] assume that the end-effector displacement $\Delta x_k$ at contact $k$ is the same for all contacts, i.e., $\forall k, \Delta x_k = \Delta x_G$. This is true for single rigid bodies but false for articulated systems. For example, imagine a robot in contact with the floor, the ceiling, and two walls in front and behind it. The normal vectors are then $n_1 = (0, 0, 1), n_2 = (0, 0, -1), n_3 = (1, 0, 0)$ and $n_4 = (-1, 0, 0)$. Equation (8) would yield $\Delta x_G = 0$, i.e., the Center-of-Mass cannot move, but a redundantly-actuated humanoid can obviously move its CoM while making these four contacts.
resulting in a simplified three-dimensional cone, while [11] dealt only with static equilibrium.

In the present paper, we propose to use an intermediate representation, the surface contact wrench [3], which eliminates the redundancy at the level of each surface contact and which constitutes a useful abstraction. Next, studying more closely the GIWC, we note that this cone only depends on the surface contact wrench cones and on the absolute positions and orientations of the contacts; in particular, it depends neither on the robot joint angles, nor on the position of its Center-of-Mass, nor on the direction of the gravity vector. Thus, the GIWC needs only be computed once per stance. This observation enables us to propose more general and/or more efficient algorithms for the problems of “testing robust static equilibrium” and “time-optimal path parameterization”.

Relation to other works involving cone representations

Using conversions between the face and span forms of a cone (hence the term “double description”) to study contact stability was first suggested in [13]. In [14], other contact modes than the stable mode (for instance rolling, sliding, etc.) were studied by this method. However, the authors of these references were concerned with the problem of fixturing workpieces and did not consider the specificity of legged locomotion (e.g., surface contacts, gravito-inertial wrench, etc.) These works also relied on the cone algorithms implemented in [13], which is too slow to handle more than six contact points, and which have been superseded by recent advances in computational geometry [12]. In [3], the cone of surface contact wrenches was studied, but through algebraic manipulations rather than the double description method.

Another approach to deal with multi-dimensional cones consists in projecting them onto a plane using recursive polygon expansions [15,7]. Note that this algorithm is peculiar to the dimension two and is not easily generalizable to higher dimensions ([15] does mention the possibility of generalization but we are not aware of any work in this direction). Nevertheless, it proved to be useful in the problems of “testing static equilibrium” [15] and “time-optimal path parameterization” [7,8].

The above references highlight how crucial it is to find a suitable cone representation for a given problem. The 2D-polygonal cross-section in [15] may be the minimum polyhedral representation when it comes to simple static equilibria, but it cannot account for robust static equilibria [16], which require computing 3D polyhedra.

In the problem of “time-optimal path parameterization” (TOPP), the works of [7,8] project the set of possible “motions” onto the (\(\hat{s}^2\), \(\hat{s}\)) plane, where \(s(t)\) denotes the position along a fixed path. The main limitation is that, contrary to [15], the projection needs to be performed at each discretized position \(s\) along the path, and not simply once per stance. In the present communication, we show how, by using the GIWC and neglecting torque limits, one can decrease computation times from seconds to tens of milliseconds.

Organization of the paper

In Section [1] we recall the relationship between stability of surface contacts and convex cones, as well as some basic definitions and results of polyhedral convex cone theory. In Section [11] we show how to compute the Gravito-Inertial Wrench Cone from the contact friction cones by using an intermediate representation, namely the surface contact wrench cone. We then use the combined GIWC to revisit two important problems arising in legged robotics: “testing robust static equilibrium” in Section [IV] and “time-optimal path parameterization” in Section [V]. We validate the latter by a dynamic motion with non-coplanar contacts that we successfully run in a dynamics simulator. Finally, we discuss in Section [VI] some directions for further development.

II. BACKGROUND

A. Stability of surface contacts

As humanoid robots usually contact the environment through surfaces (e.g., the soles of their feet) rather than through points, we first recall some definitions and results regarding the stability of surface contacts (see [3] for more details).

Definition 1: In this paper, we consider the Coulomb friction model. A contact force \(f\) is said valid when

\[
\begin{align*}
\|f^n\| &\leq \mu f^n \\
\|f^t\| &\leq \mu f^n,
\end{align*}
\]

where \(\mu\) is the friction coefficient, \(f^n\) and \(f^t\) denote respectively the normal and tangential components of the force with respect to a reference frame attached to the contact surface. In the following development, we will use the polyhedral approximation of the latter inequality, i.e.,

\[|f_x| \leq \mu f^n, \quad |f_y| \leq \mu f^n,\]

Physically speaking, a surface contact is a continuum of infinitesimal contact forces encoded by two fields: a scalar pressure field \(p(x,y)\) and a two-dimensional stress field \(\sigma(x,y)\). Here, \((x,y)\) are 2D coordinates on the surface \(S\). The fields \(p\) and \(\sigma\) are the continuous analogues of \(f^n\) and \(f^t\).

Definition 2: The pair of fields \((p(x,y), \sigma(x,y))\) is said valid when, for any \((x,y) \in S\),

\[
\begin{align*}
|p(x,y)| &\geq 0 \\
\|\sigma(x,y)\| &\leq \mu p(x,y)
\end{align*}
\]

When the environment and contacting link are rigid, the interaction between them is fully described by a contact wrench \(w = (f, \tau)\) where

\[
\begin{align*}
f &\text{ def } \int_S \nu(x,y) dx dy, \\
\tau &\text{ def } \int_S \mathbf{OC}_{xy} \wedge \nu(x,y) dx dy,
\end{align*}
\]

\(A\) stance is the set of contact status for all possible contact links, along with the position and orientation of the contacts. A change of stance occurs when e.g., a new link enters into contact with the environment, a pre-existing contact is broken, a sliding contact becomes a fixed contact, etc.
with the three-dimensional vectors \( \mathbf{v} = (\mathbf{\sigma}, p) \) and \( \overrightarrow{OC_{xy}} \) \((x, y, 0)\). We say that the field \( \mathbf{v} \) sums up to \( w \), which get us to the following definition

**Definition 3:** A contact wrench \( w \) is said valid if there exist valid fields that sum up to \( w \). △

Assume that the convex hull of the contact area is a convex polygon. Then the following proposition shows that, as far as the validity of the wrench is concerned, it is sufficient to consider forces at the vertices of the polygon instead of the continuous fields.

**Proposition 1:** A contact wrench \( w \) is valid if and only if there exist valid contact forces at the vertices of the polygon that sum up to \( w \). △

Consider now a robot with \( n \) actuated joints making \( m \) surface contacts with the environment. The equation of motion of the robot is given by

\[
M(q)\ddot{q} + H(q, \dot{q}) = S^T \tau_a + \sum_{i \text{ contact}} J_i^T w_i, \tag{3}
\]

where \( q \) is a vector of dimension \( n + 6 \) describing the configuration of the robot (the first \( n \) coordinates for the actuated joints, the last \( 6 \) coordinates for the position and orientation of the base link), \( S \) is a \( n \times (n+6) \) selection matrix, \( J_i \) is the \( 6 \times (n+6) \) contact Jacobian for contact \( i \) (i.e., the Jacobian for the transformation of the surface reference frame), \( w_i \) is the contact wrench for contact \( i \) (expressed in the surface frame).

The relationship between valid contact wrenches and (weak [17][14]) contact stability is illustrated by the following

**Definition 4 (Feasibility under stable contacts):** Assume that the robot is in a given state \((q, \dot{q})\) where \( J_i \dot{q} = 0 \) for all contacts \( i \). Then an acceleration \( \ddot{q} \) is feasible under stable contacts if (1) \( J_i \ddot{q} = -H_i \dot{q} \) and (2) there exist torques \( \tau_a \) within the torque limits and valid contact wrenches \( w_i \) that satisfy the equation of motion \( (3) \). △

In other words, wrench and torque validity are the dynamic inequalities coupled with the kinematic binding of the positions of contacting links. (See [17][14][3] for more discussion on contact stability.) As mentioned in the Introduction, we shall always make the assumption that torque limits are sufficiently large, so that our focus is on the existence of valid contact wrenches. In Section III we shall characterize this existence by a single stability criterion: the Gravito-Inertial Wrench Cone.

**B. Cone double description**

We recall here some definitions and results in the theory of Polyhedral Convex Cones. For more details, the reader is referred to [13][14]. From now on, all the “cones” we mention will implicitly be polyhedral.

**Definition 5:** A polyhedral convex cone \( C \) is defined by a set of inequalities

\[
C = \text{face}(u_1, \ldots, u_m) = \{ x : u_1^T x \leq 0, \ldots, u_m^T x \leq 0 \}. \tag{4}
\]

3The “it” part is true when one authorizes Dirac fields. If one does not authorize Dirac fields, whether this implication is true is still an open question.

for some vectors \((u_1, \ldots, u_m)\). △

Cones can be equivalently defined as positive combination of a family of base vectors.

**Theorem 1 (Weyl-Minkowski):** For every polyhedral convex cone \( C \), there exists a set of vectors \((v_1, \ldots, v_n)\) such that

\[
C = \text{span}(v_1, \ldots, v_n) = \left\{ \sum_{i=1}^{n} z_i v_i : z_1 \geq 0 \ldots z_n \geq 0 \right\}. \tag{5}
\]

If one stacks the \( u_i \) horizontally into a matrix \( U \) and the \( v_i \) vertically into a matrix \( V \), the above definitions become in matrix form (vector inequalities are element-wise)

\[
C = \text{face}(U) = \{ x : U x \leq 0 \} = \text{span}(V) = \{ V z : z \geq 0 \}. \tag{6}
\]

The span and face forms have each their own advantages. It is trivial to test whether a vector \( x \) belongs to a cone face \( \text{face}(U) \) (suffices to check \( U x \leq 0 \)) but the same operation requires solving a linear program in span form. Meanwhile, the span form is compatible with linear combinations: if \( y = Ax \), then \( x \) belongs to the cone \( \text{span}(V) \) if and only if \( y \) belongs to the cone \( \text{span}(AV) \). Doing the same in face form is a more involved operation.

An important result in polyhedral cone theory is the possibility to convert between the face and span forms of a cone (hence the term “double description”), which we shall use extensively in Section III.

**Proposition 2:** Given a matrix \( U \), one can compute explicitly a matrix \( U^S \) such that \( \text{span}(U^S) = \text{face}(U) \).

Given a matrix \( V \), one can compute explicitly a matrix \( V^F \) such that \( \text{face}(V^F) = \text{span}(V) \). △

An algorithm to compute \( U^S \) and \( V^F \) was developed in [13]. Its complexity is exponential – the problem being NP-complete – and in practice, computation times are prohibitive when the number of span or face vectors is greater than 20. In our implementation, we used the cdd library [12], which performs fast conversions even for more than 100 span or face vectors (see e.g., Tables I and II).

**III. COMPUTING THE GRAVITO-INERTIAL WRENCH CONE USING DOUBLE DESCRIPTION**

We now show how to compute the Gravito-Inertial Wrench Cone using surface contact wrenches. For each link \( k \), denote by \( p_k \) the position of its Center-of-Mass (CoM) in the laboratory frame, \( R_k \) the orientation matrix of the link frame with respect to the laboratory frame, \( \omega_k \) the angular velocity of the link in the link frame, \( I_k \) the inertia matrix of the link in the link frame. Let \( p_{\text{CoM}} \) be the position of the CoM of the robot, and \( L \), its angular momentum calculated with respect to \( p_{\text{CoM}} \), i.e.,

\[
p_{\text{CoM}} \overset{\text{def}}{=} \frac{1}{m} \sum_{\text{link } k} m_k p_k, \tag{4}
\]

\[
L \overset{\text{def}}{=} \sum_{\text{link } k} (p_k - p_{\text{CoM}}) \times m_k \dot{p}_k + R_k \omega_k. \tag{5}
\]
Definition 6 (and Proposition): The gravito-inertial wrench \( w_{\text{GI}} \) (\( f_{\text{GI}}, \tau_{\text{GI}} \)), computed with respect to the origin of the laboratory frame, is defined by

\[
\left\{
\begin{array}{l}
 f_{\text{GI}} \overset{\text{def}}{=} m (g - \hat{p}_{\text{CoM}}), \\
 \tau_{\text{GI}} \overset{\text{def}}{=} p_{\text{CoM}} \times m (g - \hat{p}_{\text{CoM}}) - \dot{\hat{p}}.
\end{array}
\right.
\]

The set of gravito-inertial wrenches that correspond to feasible joint accelerations under given stable contacts (cf. Definition 4) is a cone, called the Gravito-Inertial Wrench Cone (GIWC) \( \triangle \).

Our approach can be summarized as follows (see also Fig. 2). First, we convert the face representation of the friction cones for individual contact forces into their span form. In span form, combining the individual cones into the surface wrench cones is immediate. We thus get one contact wrench cone for every contacting link.

Next, based on the equation of motion for the free-floating coordinates, we express the linear mapping between all contact wrenches and the gravito-inertial wrench. Applying this mapping to the contact wrench cones gives us the GIWC in span form. Finally, we convert the span form of the GIWC to its face form, which can be readily used in e.g., QP or TOPP solvers. Note that only the last two steps need to be executed at each change of stance.

A. Span form of the surface contact wrench cone

We first show the following proposition.

Proposition 3: Assuming polyhedral Coulomb friction, the set of valid contact wrenches is a polyhedral convex cone. Its expression depends only on the surface geometry and friction model. \( \triangle \)

Proof: Consider a valid wrench \( w \). By Proposition 1 there exists a set of valid contact forces \( f_j \) at the vertices of the polygon that sum up to \( w \). The summation procedure is linear and can thus be encoded by a matrix \( A_{\text{surf}} \), i.e., \( w = A_{\text{surf}} f_{\text{all}} \), where \( f_{\text{all}} \) is the stacked vector of all the contact forces \( f_j \). On the other hand, the validity of the contact forces of Definition 1 can be put in the following matrix form (assuming polyhedral approximation) \( u_{\text{all}} \leq 0 \). It follows that \( w \in \text{span}(A_{\text{surf}} u_{\text{all}}) \).

Conversely, consider a \( w \in \text{span}(A_{\text{surf}} u_{\text{point}}) \), which means that there exists \( z \geq 0 \) such that \( w = A_{\text{surf}} u_{\text{point}} z \). Let \( f = U_z \). Since \( z \geq 0 \), \( f \in \text{span}(U_{\text{point}}) = \text{face}(U_{\text{point}}) \). Thus \( u_{\text{point}} f \leq 0 \). Combined with the fact that \( w = A_{\text{surf}} U_z \), this shows that \( w \) is valid. \( \square \)

One can also note \( U_{\text{point}} \overset{\text{def}}{=} U_z \) and \( U_{\text{surf}} \overset{\text{def}}{=} A_{\text{surf}} U_{\text{point}} = A_{\text{surf}} U_z \). The surface wrench cone is the minimal representation of the contact constraint, in the sense that its dimension (six) is equal to the contact DoF [3]. Note that \( U_{\text{surf}} \) can be reduced at this stage by eliminating the columns that can be written as non-negative combinations of the others. Note also that, as the matrix \( U_{\text{surf}} \) depends only on the surface geometry and friction coefficient, its computation and reduction need to be done only once for all.

B. From surface contact wrenches to gravito-inertial wrenches

The last six equations of the robot dynamics (3) can be formulated as

\[
m \hat{p}_{\text{CoM}} = \sum_{\text{contact } i} R_i f_i + m g, \tag{8}
\]

\[
\dot{\hat{p}} = \sum_{\text{contact } i} (p_i - \hat{p}_{\text{CoM}}) \times R_i f_i + R_i \tau_i. \tag{9}
\]

One can next rewrite equations (8) and (9) in terms of the gravito-inertial wrench

\[
\left\{
\begin{array}{l}
 f_{\text{GI}} = - \sum_{\text{contact } i} R_i f_i, \\
 \tau_{\text{GI}} = - \sum_{\text{contact } i} p_i \times R_i f_i + R_i \tau_i.
\end{array}
\right.
\]

In matrix form, this is equivalent to

\[
w_{\text{GI}} = \sum_{\text{contact } i} \begin{bmatrix} -R_i & 0 \\ \hat{p}_{\text{CoM}} & -R_i \end{bmatrix} w_i = A_{\text{stance}} w_{\text{all}},
\]

where \( w_{\text{all}} \) is the stacked vector of the contact wrenches \( w_i \).

The key observation here is that \( A_{\text{stance}} \) only depends on the \( p_i \) and \( R_i \), which are fixed for each stance.

Proposition 4: The gravito-inertial wrench is linearly determined by the contact wrenches. Furthermore, this linear mapping only depends on the absolute positions and orientations of the contacting links. \( \triangle \)

C. Face form of the GIWC

We can now prove the following proposition

Proposition 5: The set of gravito-inertial wrenches that correspond to feasible joint accelerations is a polyhedral convex cone, which depends only on (1) the absolute positions and orientations of the contacting links, and (2) the local geometry and friction properties at the contact surfaces. The face form of this cone can be explicitly computed. \( \triangle \)

Proof: From Proposition 3, the contact wrench \( w_i \) is valid if and only if \( w_i \in \text{span}(V^S_i) \), which implies that
Thus, \( w_{GL} = A_{\text{stance}}w_{all} \in \text{span}(A_{\text{stance}}V_{all}). \) Define now \( U_{\text{stance}} \equiv (A_{\text{stance}}V_{all})^K. \) Then, there exist valid wrenches at the contacts if and only if \( U_{\text{stance}}w_{GL} \leq 0. \) □

Since \( A_{\text{stance}} \) depends only on the \( p_i \) and \( R_i, \) \( U_{\text{stance}} \) depends only on (1) the absolute positions \( p_i, \) and orientations \( R_i \) of the contacting links, and (2) the local geometry and friction properties at the contact surfaces. Thus, it needs to be computed only once per stance.

### IV. TEST OF ROBUST STATIC EQUILIBRIUM

A robot is in static equilibrium if, at zero velocity and acceleration, the gravity wrench can be “generated” by valid contact wrenches. If the terrain is flat, static equilibrium is achieved if the CoM of the robot lies above the convex hull of the robot’s feet (support area), or in other words, if the CoM belongs to an infinite right cylinder whose axis is parallel to gravity and whose cross-section is the support area. If the terrain is not flat, then the CoM positions that induce static equilibrium are still an infinite right cylinder with axis parallel to gravity, but whose cross-section is no longer the support area. The computation of the cross-section is given in [15]. Once this cross-section is computed, one can test static equilibrium quickly by checking whether the \( (x, y) \) coordinates of the CoM is within the cross-section.

Static equilibrium is said to be robust if not only the gravity wrench, but any wrench in some neighborhood around the gravity wrench can be generated by valid contact wrenches [16]. In some limited situations, while simple static equilibrium may be satisfied (e.g., when the CoM is at the boundaries of the support polygon in flat terrain), robust static equilibrium adds an informed safety margin. Uncertainties such as model inaccuracies or unknown disturbances can be modeled within this framework.

In [16], the authors studied robust equilibrium for a robot in a 2D environment (one horizontal and one vertical direction) and for a polytopic neighborhood. They showed that the set of CoM positions that induce robust static equilibrium is non-convex and infinite in general (e.g., when the CoM is at a boundary of the support polygon in flat terrain), robust static equilibrium adds an informed safety margin. Uncertainties such as model inaccuracies or unknown disturbances can be modeled within this framework.

In the case of static equilibrium, we have \( \dot{p}_{\text{CoM}} = 0 \) and \( \dot{L} = 0. \) Equations (8) and (9) become

\[
0 = \sum_{\text{contact } i} R_i f_i + mg,
\]

\[
0 = \sum_{\text{contact } i} (p_i - p_{\text{CoM}}) \times R_i f_i + R_i \tau_i,
\]

i.e., \( (mg, mp_{\text{CoM}} \times g) = A_{\text{stance}}w_{all}. \)

We are now interested not only in the generation of the gravity wrench \( (mg, mp_{\text{CoM}} \times g) \) but also in that of any wrench in a polytopic neighborhood \( N(p_{\text{CoM}}) \) around it. Consider for simplicity the neighborhood defined as the convex hull of \( \{(mg_1, mp_{\text{CoM}} \times g_1), \ldots (mg_k, mp_{\text{CoM}} \times g_k)\} \) where \( g_1, \ldots, g_k \) are vectors around \( g. \)

Consider the sets \( M_k \) defined as

\[
M_k \equiv \left\{ p : U_{\text{stance}} \left( \frac{g_k}{p \times g_k} \right) \geq 0 \right\}
\]

and their intersection \( M \equiv \bigcap_{k \in [1,K]} M_k. \) We show that \( M \) is the set of CoM positions that ensure robust static equilibrium.

**Proof:** assume that \( p \in M. \) Consider a wrench \( w^* \in N(p). \) There exists \( \lambda_1 \geq 0, \ldots, \lambda_k \geq 0 \) such that

\[
w^* = m(\lambda g_1 + \ldots \lambda g_K, p \times (\lambda g_1 + \ldots \lambda g_K))
\]

Since \( p \in M, \) we have that, for all \( k, \)

\[
U_{\text{stance}} \left( \frac{g_k}{p \times g_k} \right) \geq 0,
\]

thus there exist valid contact wrenches \( w_{all_1}, \ldots, w_{all_k} \) such that \( -(mg_k, mp \times g_k) = A_{\text{stance}}w_{all}. \) By convexity, the contact wrench \( w_{all} = \sum \lambda_k w_{all_k} \) is valid. On the other hand, by linearity, we have \( w^* = A_{\text{stance}}w_{all}. \) Together, these last two equations show that the wrench \( w^* \) can be generated. Since \( w^* \) is arbitrary in \( N(p), \) this implies that \( p \) induces robust equilibrium.

Conversely, assume that \( p \) induces robust equilibrium, i.e., all wrenches \( w \in N(p) \) can be generated. In particular there exists a valid contact wrench \( w_{all_k} \), that generates \( (mg_k, -mp \times g_k). \) By construction of \( U_{\text{stance}}, \) this implies that \( p \in M_k. \) By repeating this reasoning, one can show that \( p \in M_2, \ldots, p \in M_K, \) which implies that \( p \in M. \) □

It can be noted that the \( M_k \) are infinite right prisms with axis parallel to \( g_k. \) Thus, if the \( g_k \) are not collinear, then \( M \) will not be a right prism, but a polyhedron. Nevertheless, from the development presented above, testing robust static equilibrium requires simply to pre-compute \( U_{\text{stance}}, \) and subsequently, for each candidate CoM position, to evaluate \( K \) matrix multiplications and comparisons as given in (12).

Note that the algorithm in [15] would require to perform \( K \) polytope projections in the pre-computation phase.

Fig. 3 illustrates the proposed robust equilibrium test. For two surface contacts (right arm and right foot), the matrix \( U_{\text{stance}} \) had dimension \( 105 \times 6 \) and could be computed in \( 3.5 \) ms on our 8-core 3.00 GHz microprocessor. Subsequently, each test took 0.1 ms.

### V. TIME-OPTIMAL PATH PARAMETERIZATION (TOPP)

Consider a path \( P \) – represented as the underlying path of a trajectory \( p(s) \in [0, s_{\text{end}}] \) – in the configuration space. Assume that \( p(s) \in [0, s_{\text{end}}] \) is \( C^1 \) and piecewise \( C^2 \)-continuous. We are interested in time-parameterizations of \( P, \) which are increasing scalar functions \( s : [0, T] \rightarrow [0, s_{\text{end}}], \) under kinodynamic constraints.
constraints. If the constraints can be expressed in the form
\[ \ddot{s} \mathbf{a}(s) + s^2 \mathbf{b}(s) + c(s) \leq 0, \]  \tag{13} 
then there exists efficient methods and implementations to find the time-optimal parameterization \( s(t) \) (see [13] and references therein).

A. TOPP reduction

Reducing constraints to the form \( \ddot{s} \mathbf{a}(s) + s^2 \mathbf{b}(s) + c(s) \leq 0 \) is relatively straightforward for fully-actuated systems, but harder for overactuated systems, which include closed-chain manipulators or legged robots with more than one surface contact (see e.g., [19]). In [21], the author adapts the 2D projection algorithm of [15] to achieve this reduction. However, the projection needs to be performed at each discretized position \( s \) along the path, which is time-consuming. Using the development of [13] we now present a method that needs cone computations only once per stance: for each discretized position \( s \), only some matrix multiplications are required to compute the vectors \( \mathbf{a}(s), \mathbf{b}(s), c(s) \).

Consider a path \( \mathbf{p}(s) \) of the CoM (note that we have dropped the subscript CoM for simplicity). Differentiating twice, we have
\[ \ddot{\mathbf{p}} = \mathbf{p}_s \ddot{s} + \mathbf{p}_{ss} s^2. \]

The angular momentum \( \mathcal{L} \) can be expressed in Jacobian form \( \mathcal{L} = \mathbf{J}_c(q) \dot{q} = \mathbf{J}_c(q) \mathbf{q}_s \ddot{s} \). Therefore, one can always write
\[ \ddot{\mathcal{L}} = \mathbf{l}_1 \ddot{s} + \mathbf{l}_2 s^2, \]
for some functions \( \mathbf{l}_1 \) and \( \mathbf{l}_2 \). In general, there is no function \( \mathbf{l} \) such that \( \mathbf{l}_1 = \mathbf{l} \) and \( \mathbf{l}_2 = \mathbf{l}_{ss} \) (in other words, \( \mathcal{L} \) is not integrable). Exceptions include the cases when \( \mathcal{L} = 0 \) or when \( \mathcal{L} \) is the angular momentum of a single rigid body.

Substituting the expressions of \( \ddot{\mathbf{p}} \) and \( \ddot{\mathcal{L}} \) into (6) and (7), we have
\[ \mathbf{w}_{\text{CoM}} = \begin{pmatrix} \mathbf{g} - \mathbf{p}_s \ddot{s} - \mathbf{p}_{ss} s^2 \mathbf{\ddot{s}} \\ \mathbf{m} \times \mathbf{g} - \mathbf{p}_s \ddot{s} - \mathbf{p}_{ss} s^2 \mathbf{\ddot{s}} - \mathbf{l}_1 \ddot{s} - \mathbf{l}_2 s^2 \end{pmatrix}. \]

Thus, the condition \( \mathbf{U}_{\text{stance}} \mathbf{w}_{\text{GI}} \leq 0 \) can be rewritten as
\[ -\ddot{s} \mathbf{U}_{\text{stance}} \begin{pmatrix} \mathbf{m} \mathbf{p}_s \\ \mathbf{m} \times \mathbf{p}_s + \mathbf{l}_1 \end{pmatrix} - s^2 \mathbf{U}_{\text{stance}} \begin{pmatrix} \mathbf{m} \mathbf{p}_{ss} \\ \mathbf{m} \times \mathbf{p}_{ss} + \mathbf{l}_2 \end{pmatrix} + \mathbf{U}_{\text{stance}} \begin{pmatrix} \mathbf{m} \mathbf{g} \\ \mathbf{m} \times \mathbf{q} \end{pmatrix} \leq 0, \] \tag{14} 
which is in the canonical form of [15].

B. Trajectory generation

As equation (14) illustrates, the centroidal trajectory, i.e., the joint trajectory of the linear and angular momenta \( \mathbf{m} \mathbf{p}(t), \mathcal{L}(t) \), is the only piece of information required to formulate the contact stability constraint. Our first attempts were therefore to interpolate a centroidal trajectory, re-time it with TOPP to satisfy contact constraints, and then interpolate a whole-body trajectory with the same linear and angular momenta by Inverse Kinematics (IK). This approach is however hampered by the difficulty in interpolating the angular momentum \( \mathcal{L}(t) \). Because of its non-holonomy, it is impossible to integrate it into a position variable, as is the case with the linear momentum. We experimented with the suggestion from [20] to regulate \( \mathcal{L} = 0 \), but it resulted in large free limb movements with a tendency to get the limbs in inconvenient positions (e.g., hands behind the back), making the overall control task harder.

We subsequently opted for a different pipeline: interpolate the CoM and end-effector trajectories first, compute a corresponding whole-body trajectory \( \mathbf{q}(s) \) by inverse kinematics, and finally enforce contact stability along this trajectory by TOPP. The angular momentum \( \mathcal{L}(\mathbf{q}, \mathbf{q}_s) \) will then result from the configurations computed by the IK solver.

Numerical TOPP solvers require relatively smooth velocity and acceleration profiles. Discontinuities in velocity or acceleration are allowed and properly dealt with [13], but we found that the acceleration profiles returned by a velocity-based IK solver are too erratic for proper use with TOPP. To avoid this, we used an acceleration-based IK solver. The whole-body trajectory \( \mathbf{q}(s) \) is computed as the double-integral of an acceleration trajectory \( \mathbf{\dot{q}}(s) \), where accelerations are computed as solutions to the following QP problem. Provided a reference trajectory \( \mathbf{p}(s) \) and \( \mathbf{f}_{\text{link}}(s) \) for the CoM and steered non-contacting link, minimize
\[ w_{\text{CoM}} \| \mathbf{J}_{\text{CoM}} \mathbf{q}_{ss} + \gamma (\mathbf{\dot{p}}^*(t) - \mathbf{J}_{\text{CoM}} \mathbf{q}_s) + \mathbf{q}_s^T \mathbf{H}_{\text{CoM}} \mathbf{q}_s \|_2^2 \]
\[ + w_{\text{link}} \| \mathbf{J}_{\text{link}} \mathbf{q}_{ss} + \gamma (\mathbf{f}_{\text{link}}^*(t) - \mathbf{J}_{\text{link}} \mathbf{q}_s) + \mathbf{q}_s^T \mathbf{H}_{\text{link}} \mathbf{q}_s \|_2^2, \]
such that
\[ \forall c \in \text{contacts}, \quad \mathbf{J}_c \mathbf{q}_s = -\gamma \mathbf{J}_c \mathbf{q}_s + \mathbf{q}_s^T \mathbf{H}_c \mathbf{q}_s, \quad K_{ss}(\mathbf{q}_{\min} - \mathbf{q}_s) \leq \mathbf{q}_s \leq K_{ss}(\mathbf{q}_{\max} - \mathbf{q}_s), \]
\[ \mathbf{q}_{\max} \triangleq -K_s (\mathbf{q} - \mathbf{q}_{\min}), \quad \mathbf{q}_{\min} \triangleq -K_s (\mathbf{q} - \mathbf{q}_{\min}). \]

Here, \( \mathbf{J}_c \) (resp. \( \mathbf{H}_c \)) denote the Jacobians (resp. Hessians) of constraints, where constraint labels are \( c \) for contact, CoM for the center-of-mass and link for the free end-effector. This problem can be readily addressed by many off-the-shelf QP
solvors. We used CVXOPT\cite{cvxopt} which is free software and could deal efficiently with both equality and inequality constraints. The reader is referred to \cite{cvxopt} for a more general solution to QP-based inverse kinematics.

In all experiments, we used $K_s = K_{ss} = \gamma = 10 s^{-1}$, $w_{\text{CoM}} = 1$ and $w_{\text{link}} = 0.1$. CoM and end-effector trajectories were interpolated as simple line segments.

C. Experiment 1: stair climbing

We first illustrate our method on a common stair climbing motion. The staircase (red boxes, reconstructed from point cloud data) has a step height of 24 cm. The motion is quasi-statically stable (it can be executed at arbitrary slow velocities) and alternates single and double support segments where the projection of the CoM is moved linearly from one support foot to the other. The retimed motion is shown in Figure 4 Computation times are reported in Table I where we detail the three consecutive computations: that of the gravito-inertial wrench matrix $U_{\text{stance}}$, of the constraint vectors $a(s), b(s), c(s)$ ($T_{\text{abc}}$) and of the re-timing by our numerical TOPP solver ($T_{\text{TOPP}}$).

TABLE I

<table>
<thead>
<tr>
<th>Segment (s)</th>
<th>Size of $U_{\text{stance}}$</th>
<th>$T_{U_{\text{stance}}}$</th>
<th>$T_{\text{abc}}$</th>
<th>$T_{\text{TOPP}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0 – 0.4</td>
<td>$114 \times 6$</td>
<td>4.2</td>
<td>2.0</td>
<td>130</td>
</tr>
<tr>
<td>0.4 – 1.0</td>
<td>$16 \times 6$</td>
<td>1.3</td>
<td>2.0</td>
<td>350</td>
</tr>
<tr>
<td>1.0 – 1.2</td>
<td>$162 \times 6$</td>
<td>4.5</td>
<td>2.2</td>
<td>560</td>
</tr>
<tr>
<td>1.2 – 2.0</td>
<td>$16 \times 6$</td>
<td>1.3</td>
<td>2.0</td>
<td>360</td>
</tr>
<tr>
<td>2.0 – 2.8</td>
<td>$162 \times 6$</td>
<td>4.3</td>
<td>2.2</td>
<td>150</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>15.6</td>
<td>10.4</td>
<td>1550</td>
</tr>
</tbody>
</table>

D. Experiment 2: box climbing using tilted support

We now apply our method to the setting depicted in Figure 4. The robot climbs a 10-cm box using two contacts: its right arm, set on a 90-cm high horizontal ledge, and its left foot, set on a 25° inclined stepping surface. The motion generated in this experiment may seem unnatural, as the robot could step on the box directly. Yet, the motivation for this setting is two-fold. First, stability throughout this motion cannot be checked by ZMP, as frictional contacts may be lost (the “sufficient friction” assumption does not apply) and the contact surfaces non-coplanar. Second, the solution to this constrained problem is not feasible at low velocities, which means CoM-based methods such as \cite{giwc} cannot be applied.

Figure 5 shows the execution of the motion in our physics simulator. In this scenario, the friction coefficient of environment bodies (respectively the floor, box, arm support and inclined plane) was set to $\mu = 0.9$. Due to the high speed of the retimed segment (the combined duration of the right-foot and left-foot steps is only two seconds), the position controller was not responsive enough to track the reference trajectory exactly. To palliate this, we made time-optimal trajectories slower by using a smaller friction coefficient $\mu = 0.7$ in the computations of the GIWC for TOPP (namely, $\mu = 0.7$ and contact surfaces scale by 0.7) in order to get a “safer” motion.

A further difficulty, compared to other experiments were humanoids walk on slightly inclined surfaces, was that we could not use the stabilizer module of the humanoid, which is designed for horizontal floors. Thus, the motion represented in Figure 5 was run with pure open-loop position control.

The transition between the first and second foot steps is the most challenging part of the motion. Its configurations are not statically stable, and it therefore needs a non-zero minimum velocity to be performed without falling. As CoM trajectories are straight lines in our design, the problem of finding a feasible whole-body motion for this segment boiled down to finding a suitable CoM velocity $v$. Manual trials being unsuccessful, we chose this velocity heuristically as $v^* = \arg \max_v \beta_v(0, 1)$, where $\beta$ is an internal vector field from TOPP representing the maximum acceleration achievable along the path (see \cite{heuristic} for details). This heuristic provided feasible solutions in practice. Furthermore, from sampling neighboring vectors, we estimated the solid angle of valid CoM velocity vectors around $v^*$ to 0.1 steradian, i.e., less than 0.8% of the orientation space.

E. Comparison to previous work

Tables I and II show the performances of our method on the box and stair climbing motions. We compare these performances with those reported in previous work by Hauser \cite{hauser} where the recursive polygon expansion algorithm is used to retime humanoid trajectories under distributed contact forces and actuator limits. The two main differences between this
work and our method are that (1) we calculate the GIWC rather than a recursive polygon expansion, and (2) we use a numerical TOPP solver rather than a Sequential Linear Programming (SLP) solver. In order to make run times more comparable, we used the same path discretization resolution $N = 100$.

In accordance with previous work, we break down computation times as follows

- **Pre-computation of feasible sets ($T_{U_{\text{stance}}} + T_{\text{abc}}$):** the time reported in [7] is 2.40 s, while our solution takes around 30 ms in both settings to perform this operation. When actuator torque is not limiting, it is therefore one to two orders of magnitude faster than previous work.

- **Computation of the velocity profile ($T_{\text{TOPP}}$):** the time reported in [7] is 2.46 s, while our solution takes between 1 s and 1.5 s in the previous climbing motions. Two factors are concurring here to put execution times on the same magnitude: we use a numerical TOPP solver, an approach that is usually orders of magnitude faster than SLP; however, we do not prune redundant inequalities, which is a side benefit of recursive polygon expansion.

Overall, our approach cuts down to tens of milliseconds the pre-computations where previous work would spent half of the computing budget, while having comparable performances on the remaining time-optimal re-timing. This comparison should however be considered with much precaution since our robot, task, and computing environment are different from [7]. A more controlled comparison is currently under preparation.

### VI. Conclusion

In this paper, we presented a derivation of the Gravito-Inertial Wrench Cone from contact constraints. Maintaining the gravito-inertial wrench within this cone is known as a general stability criterion, a “generalized ZMP”. Armed with this cone, we revisited two important problems arising in motion planning for humanoid robots. In the problem of “testing static equilibrium”, we were able to extend the results of [15, 16] to the case of robust three-dimensional static equilibrium. In the problem of “time-optimal path parameterization” (TOPP), we derived a new reduction of the contact constraints into the classical form, which requires cone computations only once per stance, thereby dramatically improving the performance of TOPP in multi-contact [7, 8]. We compared our computation times to state-of-the-art related approaches, and validated our results by realistic simulated motions on a full-size humanoid model.

There are many directions for further development. The results on time-optimal control obtained here can be straightforwardly integrated into the framework of Admissible Velocity Propagation [22], which will open the possibility for non-quasi-static motion planning through contact changes. One can also generalize the “stable surface contact wrench cone” of Section III-A into a useful abstraction: any contact mode between two rigid bodies (stable, sliding, rolling, etc.) can indeed be encoded by a wrench cone [13]. One can thus decompose a humanoid contact problem into two independent ones: (i) studying the relationship between the GIWC and arbitrary contact wrench cones, and (ii) studying how a particular contact mode (stable, sliding, rolling, etc.) gives rise to a particular contact wrench cone.

Besides the above extensions, we believe that the wrench cone condition developed through the works of [13, 14, 6] and in the present paper has the potential to be to multi-contact stability what the ZMP is to planar locomotion.
REFERENCES


