Planning with the Center-of-Mass rather than Stances for Humanoids Walking on Uneven Terrains

Stéphane Caron*
Department of Mechano-Informatics
The University of Tokyo, Japan

Yoshihiko Nakamura
Department of Mechano-Informatics
The University of Tokyo, Japan

Abstract— *In the current literature for non-gaited humanoid motion planning, stances (*i.e.*, contact locations) are usually planned in a first step, after which joint-angle trajectories are interpolated or planned themselves. In this paper, we propose an alternative where planning is driven by center-of-mass motions rather than stances. Our approach uses a randomized motion planner as its first layer to explore the space of horizontal CoM coordinates (x_G, y_G). At a lower level, we propose a custom method to extend stances based on a desired CoM position. We evaluate the ability of the resulting planner in a rubble-field 3D environment with a model of the* HYDRA *humanoid robot.*

I. Introduction

For space robots or manipulators with fixed bases, the notions of *motion* and *actuated joint trajectory* seem to coincide: given the time evolution of joint-angle values, one can apply the equation of motion to univocally compute the complete motion of the robot. This is because, in such cases, the *spatial* features of interactions between the robot and the environment is stationary: interaction forces may vary in intensity, but not their application points, *e.g.*, the Center-of-Mass (CoM) for gravity or the screws of the fixed base for a manipulator.

However, motion and actuated-joint trajectory should not be confused when interactions vary both in space and intensity, as is the case for humanoid robots. Limbed robots locomote by breaking contact, moving their free limb to a new contact position, establishing contact, and so forth. For them, the geometric information necessary to describe a motion includes both joint-angle values and contact locations. A set of contact locations is called a stance. By locations, we mean indifferently point [7] or surface contacts (the latter can be reduced to the former under proper assumptions [5]).

In an early humanoid motion planner [13], the *stance* (*i.e.*, the set of contact locations) was fixed and a randomized planning algorithm (RRT [14]) was used to find feasible trajectories. This solution could only move the free-flying coordinates of the humanoid inside the reachable space delimited by the fixed contact condition. Later developments [2], [9] enabled a larger reachable space by using discrete stance changes: first, performing a graph search in a sample of the stance space (where nodes are stances and edges are steps), then planning a whole-body trajectory following the resulting step sequence.

A drawback in discretizing the stance space lies in the large impact of the sampling resolution on the algorithm's performance: too sparse a sampling and no solution may be found, too large a sampling and the execution time may be prohibitive. To palliate this, [1], [7] developed an alternative where the continuum of the stance space is explored, with guiding from heuristic cost and distance functions, rather than discretized.

In both approaches, planning takes place in the stance space and configurations are computed subsequently. Yet, the stance space is not an easy environment to plan in: its dimension grows linearly with the number of contact points or surfaces, and its free regions are conditioned by the geometric structure of the robot (*i.e.*, calls to an inverse-geometric solver are required to compute obstacles). In the present paper, we consider planning in the CoM space. That is, a CoM trajectory is determined at the top-priority layer of our method, and stances are computed subsequently to support the CoM motion.

The rest of the paper is organized as follows. We state the problem and summarize necessary definitions in section II. Then, we derive our stability criterion in section III and describe its integration in a joint CoM-stance planner in section IV. Finally, we provide some experimental validation in section V.

II. Problem statement

Consider a humanoid robot with n actuated degrees of freedom (DoF). We describe its configuration by an $(n+6)$-dimension vector of generalized coordinates $\boldsymbol{q}$, the last six components of which describe the position and orientation of the free-flying link in $\mathbb{R}^3 \times SO(3)$.

A contact point is defined by a tuple $(C_i, D_i(\boldsymbol{q}))$ where C_i is a fixed point in the world reference frame and $D_i(\boldsymbol{q})$ denotes the world coordinates of a point attached to the robot. The geometric contact constraint is $C_i = D_i(\boldsymbol{q})$, that is, the two points coincidate. Assuming it is satisfied, its time derivative yields the kinematic contact constraint $\mathbf{J}_i(\boldsymbol{q})\dot{\boldsymbol{q}} = 0$, where $\mathbf{J}_i$ is the contact jacobian.

We model the contact between the humanoid and its en-

*caron@ynl.t.u-tokyo.ac.jp

vironment as a set of contact points σ that is called a *stance* [9]. The equation of motion of the robot in the stance σ is

$$\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \mathbf{S}^\top \boldsymbol{\tau} + \sum_{i \in \sigma} \mathbf{J}_i^\top(\boldsymbol{q})\boldsymbol{f}_i, \qquad (1)$$

with $\mathbf{M}(\boldsymbol{q})$ is the joint-inertia matrix, $\boldsymbol{h}$ the resultant of gravity and Coriolis forces, $\mathbf{S}$ the projector on actuated coordinates, $\boldsymbol{\tau}$ the vector of actuated torques and $\boldsymbol{f}_i$ a force applied at the i^{th} contact. Due to the robot's power limits, torques are bounded quantities:

$$\boldsymbol{\tau}_{\min} \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}_{\max} \qquad (2)$$

Contact forces are also constrained to lie in friction cones. In this paper, we choose the pyramidal approximation to these cones:

$$|f_{ix}| \leq \mu f_{iz}, \quad |f_{iy}| \leq \mu f_{iz}, \qquad (3)$$

with μ the friction coefficient.

We say that the configuration $\boldsymbol{q}$ is *supported* by the stance σ, written $\boldsymbol{q} \in \mathcal{F}_\sigma$, when there exists a solution $\{\boldsymbol{\tau}, \boldsymbol{f}_1, \ldots, \boldsymbol{f}_k\}$ to Equations (1), (2) and (3). Furthermore, two stances σ and σ' are *adjacent* when there exists a configuration $\boldsymbol{q} \in \mathcal{F}_\sigma \cap \mathcal{F}_{\sigma'}$.

A given stance σ allows for a given volume of free-link motions, but it is necessary to change stances in order to move the free-link in its full reachable space. We therefore consider motions with (a finite number of) switches in the supporting stance. A joint-angle trajectory $t \mapsto \boldsymbol{q}(t)$ is *feasible* when there exists a piecewise-constant function $t \mapsto \sigma(t)$ such that

- any two subsequent stances in $\sigma(t)$ are adjacent, and
- for any time instant t, $\boldsymbol{q}(t)$ is feasible for $\sigma(t)$.

We will thereafter call *feasible motion* such a joint function $t \mapsto (\boldsymbol{q}(t), \sigma(t))$. The goal of this paper is to plan a feasible motion for a humanoid on rough terrain.

III. Stability criterion

Our stability condition proceeds from the theory of convex polyhedra, in particular its applications to polyhedral convex cones [10]. In the present work, we assume that friction is more a challenge than actuation power, in which case the torque constraints (2) are never saturated. The equation of motion (1) will then be satisfied if and only if the six lines corresponding to the free-flying coordinates are satisfied. As was shown in [16], these six lines can be concisely rewritten in terms of centroidal momentum:

$$\begin{aligned} m\ddot{\boldsymbol{p}}_{\mathrm{G}} &= \sum_{i \in \sigma} \mathbf{R}_i \boldsymbol{f}_i + m\boldsymbol{g}, \\ \dot{\mathcal{L}} &= \sum_{i \in \sigma} (\boldsymbol{p}_i - \boldsymbol{p}_{\mathrm{G}}) \times \mathbf{R}_i \boldsymbol{f}_i. \end{aligned}$$

Under static equilibrium, it is equivalent to

$$\begin{aligned} \boldsymbol{f}_{\mathrm{G}} &= -\sum_{i \in \sigma} \mathbf{R}_i \boldsymbol{f}_i, & (4) \\ \boldsymbol{\tau}_{\mathrm{G}} &= -\sum_{i \in \sigma} \boldsymbol{p}_i \times \mathbf{R}_i \boldsymbol{f}_i. & (5) \end{aligned}$$

where $\boldsymbol{w}_{\mathrm{G}} = (\boldsymbol{f}_{\mathrm{G}}, \boldsymbol{\tau}_{\mathrm{G}})$ is the gravitational wrench defined by

$$\begin{aligned} \boldsymbol{f}_{\mathrm{G}} &:= m\boldsymbol{g}, & (6) \\ \boldsymbol{\tau}_{\mathrm{G}} &:= m\boldsymbol{p}_{\mathrm{G}} \times \boldsymbol{g}. & (7) \end{aligned}$$

For configurations in static equilibrium, it is known since [3] that the CoM lies in a vertical cylinder with horizontal polygonal basis. The corresponding polygon was computed in [3] using a two-dimensional polytope projection algorithm. We shall now provide an alternative derivation of this polygon using the double-description method.

Proposition 1: A motion $\boldsymbol{q}(t)$ is feasible if and only if, at each time instant t, the horizontal coordinates of the center-of-mass $(x_{\mathrm{CoM}}, y_{\mathrm{CoM}})(\boldsymbol{q}(t))$ lie in a polygon $\mathcal{S}$ that is fully determined by the set of contact points $\mathcal{C}$.

All contact forces lie in a polyhedral cone given by Equation (3). Equivalently, this constraint can be written $\mathbf{A}\boldsymbol{f} \leq \mathbf{0}$. Polyhedral convex cone theory [10] provides a pivotal tool to manipulate such inequalities, which is formalized by the following propositions.

Proposition 2: For any matrix $\mathbf{A}$, there exists a dual matrix $\mathbf{A}^S$ such that,

$$\forall \boldsymbol{f},\ \mathbf{A}\boldsymbol{f} \leq \mathbf{0} \ \Leftrightarrow\ (\exists \boldsymbol{z} \geq \mathbf{0}, \boldsymbol{f} = \mathbf{A}^S \boldsymbol{z}).$$

Conversely, for any matrix $\mathbf{B}$, there exists a dual matrix $\mathbf{B}^F$ such that

$$\forall \boldsymbol{f},\ (\exists \boldsymbol{z} \geq \mathbf{0}, \boldsymbol{f} = \mathbf{B}\boldsymbol{z}) \ \Leftrightarrow\ \mathbf{B}^F \boldsymbol{f} \leq \mathbf{0}.$$

Furthermore, $(\mathbf{A}^S)^F = \mathbf{A}$ and $(\mathbf{B}^F)^S = \mathbf{B}$.

The representation $\mathbf{A}\boldsymbol{f} \leq 0$ is called the *face* or H-representation of a polyhedral cone, while its dual formulation $\boldsymbol{f} = \mathbf{B}\boldsymbol{z}, \boldsymbol{z} \geq 0$ is called the *span* or V-representation. The H-representation is efficient for checking membership of a point to the polyhedron, but inconvenient to apply linear transformations. The converse holds for the V-representation.

According to Equation (3), all contact forces satisfy $\mathbf{A}\boldsymbol{f}_i \leq \mathbf{0}$, or equivalently, $\boldsymbol{f}_i = \mathbf{A}^S \boldsymbol{z}_i$ for some $\boldsymbol{z}_i \geq \mathbf{0}$. Meanwhile, the gravitational wrench $\boldsymbol{w}_{\mathrm{G}}$ can be written using Equations (4) and (5) as

$$\boldsymbol{w}_{\mathrm{G}} = \mathbf{N}(\sigma)[\boldsymbol{f}_1^\top \ldots \boldsymbol{f}_k^\top]^\top,$$

where the matrix $\mathbf{N}(\sigma)$ only depends on the stance $\sigma = \{(\boldsymbol{p}_i, \mathbf{R}_i)\}$. One can thereafter write

$$\boldsymbol{w}_{\mathrm{G}} = \mathbf{N}(\sigma)\, \mathrm{diag}(\mathbf{A}^S, \ldots, \mathbf{A}^S)\, \boldsymbol{z}$$

with $\boldsymbol{z} = [\boldsymbol{z}_1^\top \dots \boldsymbol{z}_k^\top]^\top \geq \mathbf{0}$. Finally, one can derive the H-representation of the contact constraints as

$$\mathbf{C}_{\mathrm{G}}(\sigma)\boldsymbol{w}_{\mathrm{G}} \leq \mathbf{0},$$

with $\mathbf{C}_{\mathrm{G}}(\sigma) := \left[\mathbf{N}(\sigma)\,\mathrm{diag}\left(\mathbf{A}^S, \dots, \mathbf{A}^S\right)\right]^F$. Now, recalling the definition (6)-(7) of the gravitational wrench, one can write this inequality as

$$\mathbf{C}_{\mathrm{G},[1,3]}(\sigma)\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \mathbf{C}_{\mathrm{G},[4,6]}(\sigma)\begin{bmatrix} y_G \\ -x_G \\ 0 \end{bmatrix} \leq \mathbf{0},$$

that is, a two-dimensional polyhedron

$$\mathbf{A}_{\mathrm{G}}(\sigma)\,[x_G\ y_G]^\top \leq \boldsymbol{b}_{\mathrm{G}}.$$

We have therefore computed the polygonal support area in arbitrary contact stances. In practice, we used the double-description library `cddlib` [8] for its efficient conversion methods between the H- and V-representation.

IV. Motion planning on uneven terrains

As we consider a static equilibrium condition, the criterion we computed in the previous section only depends on the horizontal plane coordinates of the CoM. Consequently, we choose to explore the two-dimensional space (x_G, y_G) using a Rapidly-exploring Random Tree (RRT).

The tree looks for a path between an initial and target CoM positions. At each extension, arbitrary CoM coordinates (x_G, y_G) are sampled from an estimate of the globally reachable region and the planner tries to extend one of its reached states in order to get closer to (x_G, y_G). The overall process is reminded in Figure 1.

Algorithm 1 CoM-RRT

Input: $\boldsymbol{p}_{\mathrm{G,start}}, \boldsymbol{p}_{\mathrm{G,goal}}$
Output: A feasible motion $(\boldsymbol{q}(t), \sigma(t))$ such that $\boldsymbol{p}_{\mathrm{G}}(0) = \boldsymbol{p}_{\mathrm{G,start}}$ and $\boldsymbol{p}_{\mathrm{G}}(T) = \boldsymbol{p}_{\mathrm{G,goal}}$; or `Failure`

$\sigma_{\mathrm{start}} \leftarrow$ GENERATE_STANCE$(\boldsymbol{p}_{\mathrm{G,start}})$
$\mathcal{T} \leftarrow \{(\boldsymbol{p}_{\mathrm{G,start}}, \sigma_{\mathrm{start}})\}$
for $i = 1$ to N **do**
  $\boldsymbol{p}_{\mathrm{G}} \leftarrow$ SAMPLE$([X_{\mathrm{min}}, X_{\mathrm{max}}] \times [Y_{\mathrm{min}}, Y_{\mathrm{max}}])$
  EXTEND$(\mathcal{T}, \boldsymbol{p}_{\mathrm{G}})$
  if EXTEND$(\mathcal{T}, \boldsymbol{p}_{\mathrm{G,goal}})$ **then**
    return BACKTRACK_TRAJECTORY$(\mathcal{T}, \boldsymbol{p}_{\mathrm{G,goal}})$
  end if
end for
return `Failure`

Differences from the generic RRT of [14] will occur in the EXTEND and GENERATE_STANCE functions.

A. CoM-aware Stance Extensions

In the extension step, we use the k-nearest neighbors heuristic with $k = 10$ [15] to raise the likelihood of a successful extension. The underlying metric $\mathrm{d}(\boldsymbol{p}_{\mathrm{G}}, \sigma)$ is the

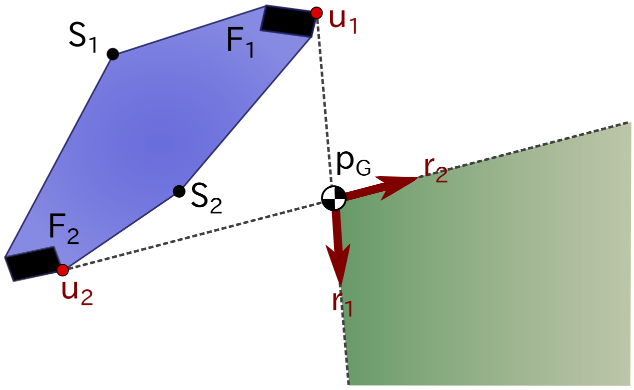

Fig. 1. Extension toward a candidate CoM of a four-link stance with four non-coplanar contacts: two rectangular feet and two point sticks. Contact points and surfaces are drawn in black. The blue polygon represents the stance's support area. The green cone depicts the "eclipse" of the polygon by the CoM. It corresponds to the positions at which the stick S_2 can be put so that the resulting stance stabilizes the candidate CoM.

distance from the target CoM position $\boldsymbol{p}_{\mathrm{G}}$ to the support polygon of the stance σ, computed as in section III.

Algorithm 2 EXTEND() function

Input: tree $\mathcal{T}$, CoM target $\boldsymbol{p}_{\mathrm{G}}$
Output: `Success` or `Failure`

$\mathcal{N} \leftarrow k$_NEAREST_STANCES$(\mathcal{T}, \boldsymbol{p}_{\mathrm{G}})$
$\mathcal{S} \leftarrow \{$EXTEND_STANCE$(\sigma, \boldsymbol{p}_{\mathrm{G}}), \sigma \in \mathcal{N}\}$
if $\mathcal{S} \neq \emptyset$ **then**
  $\mathcal{T} \leftarrow \mathcal{T} \cup \left\{\arg\min_{\sigma \in \mathcal{S}} \mathrm{d}(\boldsymbol{p}_{\mathrm{G}}, \sigma)\right\}$
  return `Success`
end if
return `Failure`

The core routine here is EXTEND_STANCE, which we will now explain. Let us denote by $\boldsymbol{v}_1, \dots, \boldsymbol{v}_m$ the vertices of the support area, in trigonometric order. We define the *extension set* of the stance as

$$\mathsf{ext}(\sigma) = \{v \mid \boldsymbol{p}_{\mathrm{G}} \in \text{CONVEX\_HULL}(\{\boldsymbol{v}_1, \dots, \boldsymbol{v}_m, v\}\}$$

This extension set turns out to be a cone that one can compute as follows. First, compute the sequence of signed distances

$$s_i := \frac{\boldsymbol{v}_{i+1} - \boldsymbol{v}_i}{\|\boldsymbol{v}_{i+1} - \boldsymbol{v}_i\|} \times (\boldsymbol{p}_{\mathrm{G}} - \boldsymbol{v}_i),$$

with the convention that the $m+1$ index loops to 1. When $\boldsymbol{p}_{\mathrm{G}}$ is outside of the support polygon, this sequence has positive and negative entries. Furthermore, all positive (resp. negative) elements are consecutive. Then, define the two points were the signed distance changes sign as

$$\begin{aligned} \boldsymbol{u}_1 &:= \boldsymbol{v}_{j_1} \text{ s.t. } s_{j_1} > 0 \wedge s_{j_1 - 1} < 0 \\ \boldsymbol{u}_2 &:= \boldsymbol{v}_{j_2} \text{ s.t. } s_{j_2} < 0 \wedge s_{j_2 - 1} > 0 \end{aligned}$$

Finally, $\mathsf{ext}(\sigma)$ is given by the cone defined by the apex $\boldsymbol{p}_{\mathrm{G}}$ and the two rays $\boldsymbol{r}_1 = (\boldsymbol{p}_{\mathrm{G}} - \boldsymbol{u}_1)$ and $\boldsymbol{r}_2 = (\boldsymbol{p}_{\mathrm{G}} - \boldsymbol{u}_2)$. Figure 3 illustrates this procedure.

We extend the stance by either setting a free link $\ell \in \mathsf{free}(\sigma)$ or moving a contacting link to intersect the extension cone. In the latter case, we consider only the links that can be moved without affecting the side of the support polygon opposite to the target CoM. This set is computed by mapping each vertex $\boldsymbol{v}_i$ to its closest link $\ell(\boldsymbol{v}_i)$ (for the point-to-volume 3D euclidean metric) and using the signed distances to determine the side of each vertex. *In fine*,

$$\mathsf{candidates}(\sigma) := \mathsf{free}(\sigma) \cup \{\ell(\boldsymbol{v}_j) \mid s_j < 0\} - \{\ell(\boldsymbol{u}_1), \ell(\boldsymbol{u}_2)\}$$

Our overall stance-extension algorithm is summarized in Figure 3.

Algorithm 3 EXTEND_STANCE() function

Input : initial stance σ, CoM target $\boldsymbol{p}_{\mathrm{G}}$
Output : stance σ' stabilizing $\boldsymbol{p}_{\mathrm{G}}$, or `Failure`

```
(j1, j2), (r1, r2) ← ext(σ)
for each link ℓ ∈ candidates(σ) do
    (x', y') ← SAMPLE_CONE(pG, r1, r2)
    t ← GROUND_POSE_AT(x', y')
    σ' ← GENERATE_POSTURE(σ ∪ {(ℓ, t)})
    if σ' was found then
        return σ'
    end if
end for
return Failure
```

B. *Posture Generation*

As illustrated by the call to GENERATE_POSTURE in the above pseudo-code, all the pipeline we described rests on an inverse-geometry solver. Posture Generation is the geometric problem of finding a vector of generalized coordinates $\boldsymbol{q}$ satisfying a set of constraints such as DOF limits or, in a given stance σ, that all contacts $i \in \sigma$ are made. We solve the posture generation problem by inverse kinematics using the prioritized kinematic control framework from [11]. The reader is referred to [4] for a more general approach to posture generation in multi-contact scenarii.

V. Experiment

We conduct our experiments in OpenRave [6] with a model of the HYDRA humanoid robot developed in our laboratory (see Figure 2). In its current version, the robot has 37 actuated DOFs.

The setting of the experiment is a rubble field, as depicted in Figure 3. Following the idea of [12], we replace the robot's hands by walking sticks. The sticks give the robot a wider geometric range, in turn allowing for larger support areas. We ran the RRT algorithm from section IV, with $\boldsymbol{p}_{\mathrm{G,start}}$ set to the middle of the rubble field. For now, our

Fig. 2. False-color view of the HYDRA humanoid robot.

implementation of the GENERATE_STANCE function samples foot and stick poses around the CoM until a solution is found.

Figure 3 shows a stance sequence computed by our solution, with the underlying RRT depicted in Figure 4. The humanoid starts in a configuration where its legs stand on two non-coplanar rubble blocks and are almost crossed. It first moves its right, then left stick before performing a a left step. Two stick moves later, it performs a second left step, this time reaching a second block left of the first one, immediately followed by a right step putting both feet on the same block. Finally, after repositionning the two sticks, it performs an additional right step on the next block forward.

VI. Conclusion

In this paper, we proposed an alternative to stance-space planning for humanoid locomotion. Our approach combines a Rapidly-exploring Random Tree planning in the horizontal-CoM space with a custom CoM-based extension routine to compute underlying stances at each node. We evaluated the ability of our planner in a rubble-field environment with a model of the HYDRA humanoid robot.

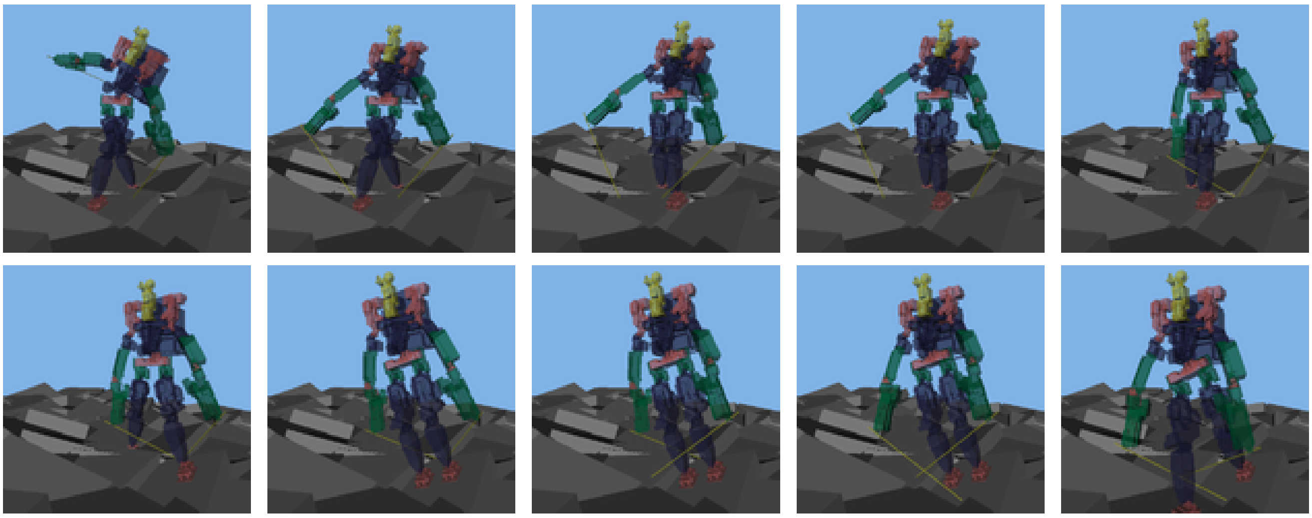

Fig. 3. Stance sequence generated by our planner. A stick-carrying variant of the HYDRA humanoid evolves on a randomly-generated rubble-field. Our planner's state space is the two-dimensional plane of horizontal CoM position. In the sequence above, each stance results from an extension toward a CoM subgoal (see IV-A).

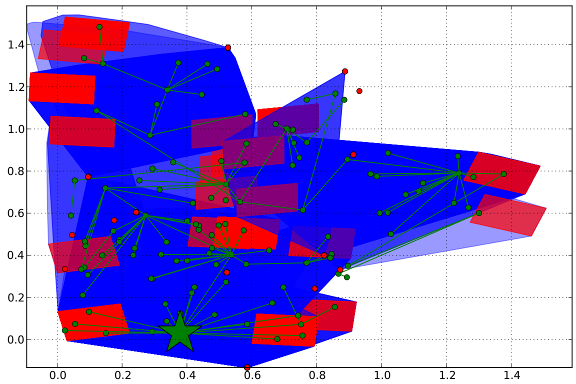

Fig. 4. Exploration of the RRT in the horizontal plane for the stance sequence depicted in Figure 3. CoM positions and trajectories are represented by green dots and lines, respectively. The green star corresponds to the starting CoM position. Contact locations, either rectangular surfaces or stick points, are drawn in red while the blue areas correspond to superposed support polygons (plotted with transparency for readability).

References

[1] Karim Bouyarmane and Abderrahmane Kheddar. Multi-contact stances planning for multiple agents. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5246–5253. IEEE, 2011.

[2] Timothy Bretl. Motion planning of multi-limbed robots subject to equilibrium constraints: The free-climbing robot problem. *The International Journal of Robotics Research*, 25(4):317–342, 2006.

[3] Timothy Bretl and Sanjay Lall. A fast and adaptive test of static equilibrium for legged robots. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 1109–1116. IEEE, 2006.

[4] Stanislas Brossette, Adrien Escande, Joris Vaillant, François Keith, Thomas Moulard, and Abderrahmane Kheddar. Integration of non-inclusive contacts in posture generation. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 933–938. IEEE, 2014.

[5] Stéphane Caron, Quang-Cuong Pham, and Yoshihiko Nakamura. Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench for rectangular support areas. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on, (To Appear in)*. IEEE, 2015.

[6] Rosen Diankov. *Automated Construction of Robotic Manipulation Programs*. PhD thesis, Carnegie Mellon University, Robotics Institute, August 2010.

[7] Adrien Escande, Abderrahmane Kheddar, and Sylvain Miossec. Planning contact points for humanoid robots. *Robotics and Autonomous Systems*, 61(5):428–442, 2013.

[8] Komei Fukuda. Cddlib reference manual. *Report version 093a, McGill University, Montréal, Quebec, Canada*, 2003.

[9] Kris Hauser, Timothy Bretl, Jean-Claude Latombe, Kensuke Harada, and Brian Wilcox. Motion planning for legged robots on varied terrain. *The International Journal of Robotics Research*, 27(11-12):1325–1349, 2008.

[10] Shinichi Hirai. *Analysis and planning of manipulation using the theory of polyhedral convex cones*. PhD thesis, Kyoto University, 1991.

[11] Oussama Kanoun. Real-time prioritized kinematic control under inequality constraints for redundant manipulators. In *Robotics: Science and Systems*, volume 7, page 145, 2012.

[12] Oussama Khatib and Shu-Yun Chung. Suprapeds: Humanoid contact-supported locomotion for 3d unstructured environments. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2319–2325. IEEE, 2014.

[13] J.J. Kuffner, S. Kagami, K. Nishiwaki, M. Inaba, and H. Inoue. Dynamically-stable motion planning for humanoid robots. *Autonomous Robots*, 12(1):105–118, 2002.

[14] S.M. LaValle and J.J. Kuffner. Randomized kinodynamic planning. *The International Journal of Robotics Research*, 20(5):378–400, 2001.

[15] Quang-Cuong Pham, Stéphane Caron, and Yoshihiko Nakamura. Kinodynamic planning in the configuration space via velocity interval propagation. In *Robotics: Science and System*, 2013.

[16] P-B Wieber. Holonomy and nonholonomy in the dynamics of articulated motion. In *Fast motions in biomechanics and robotics*, pages 411–425. Springer, 2006.