

# SL1M: Sparse L1-norm Minimization for contact planning on uneven terrain

Steve Tonneau<sup>1</sup>, Daeun Song<sup>2</sup>, Pierre Fernbach<sup>3</sup>, Nicolas Mansard<sup>3</sup>, Michel Taïx<sup>3</sup> and Andrea Del Prete<sup>4</sup>

**Abstract**—One of the main challenges of planning legged locomotion in complex environments is the combinatorial contact selection problem. Recent contributions propose to use integer variables to represent which contact surface is selected, and then to rely on modern mixed-integer (MI) optimization solvers to handle this combinatorial issue. To reduce the computational cost of MI, we exploit the sparsity properties of L1 norm minimization techniques to relax the contact planning problem into a feasibility linear program. Our approach accounts for kinematic reachability of the center of mass (COM) and of the contact effectors. We ensure the existence of a quasi-static COM trajectory by restricting our plan to quasi-flat contacts. For planning 10 steps with less than 10 potential contact surfaces for each phase, our approach is 50 to 100 times faster than its MI counterpart, which suggests potential applications for online contact re-planning. The method is demonstrated in simulation with the humanoid robots HRP-2 and Talos over various scenarios.

## I. INTRODUCTION

Motion planning of legged robots in arbitrary contexts is still an open problem. Specifically, we are concerned with planning contact locations that allow the robot to move towards its target without falling.

Since the problem is nonlinear, local approaches often lead to dead-ends. Nonlinear optimization formulations are able to provide impressive results at convergence [1], [2], but the solvers are prone to fall in local minima. Sampling-based approaches [3], [4], including our own work [5], [6], as well as A\*-based approaches [7], [8], [9], have demonstrated significant successes on real robots for specific sets of scenarios. An appealing approach to tackle the contact planning problem in the most general way is to use mixed-integer (MI) programming. Deits et al. have demonstrated the potential for such approach with a purely kinematic formulation [10]. Ponton et al. then extended MI to account for the centroidal dynamics of the robots [11], [12].

MI can theoretically address entirely the *multi-contact planning* problem, extending locomotion to non-gaited behaviors, involving the use of hands and other effectors. A practical limitation of MI programming is the computation time, resulting from the branch-and-bound techniques to handle the combinatorial aspect of the problem (here the issue of selecting contact surfaces), especially when the inner problem is hard (such as when nonlinear dynamics have to be handled). Several hundreds of milliseconds are required

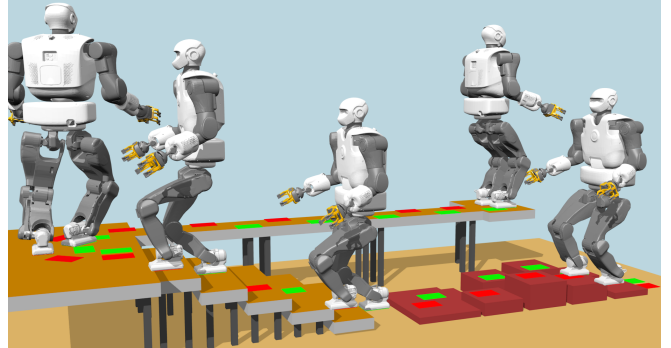


Fig. 1: A contact sequence computed by our approach, with keyframes of the Talos robot executing the plan.

to plan one or two steps, while ideally we would like to plan the next contact more reactively.

The objective of this paper is to mitigate the issue of computation time for MI, while preserving its potential to tackle contact planning in a general manner. To achieve this, we propose a convex relaxation of the problem with an L1-norm minimization formulation, which has the desirable property of often converging to sparse solutions [13], [14], [15]. The sparse solution can then be exploited to decide which surfaces to select.

This reformulation, which we call SL1M (for Sparse L1-norm Minimization) is faster than the MI approach for planning 10 steps with less than 10 potential contact surfaces for each phase, up to 100 times in the most favorable cases. SL1M comes with two drawbacks. First, optimality is only approximated, as the computed solution is a compromise between the imperative of finding a sparse solution and the considered objective function. Second, in some cases involving large numbers of contacts and potential contact surfaces (more than 15 steps and more than 10 potential contact surfaces), the solver can fall in unsatisfying minima (i.e. not converging to entirely sparse solutions), though we believe those minima can be escaped by combining the contact planner with high-level approaches [16].

### A. Contributions

Our main contribution is the reformulation of the MI contact planning program into a convex feasibility linear program (Section IV), for which we provide an analysis of the benefits and drawbacks with respect to MI (Section IX).

To guarantee that the contact plans are feasible, SL1M continuously handles the dynamics of the robot using a quasi-static constraint that applies on quasi-flat surfaces (for which

<sup>1</sup>IPAB, The University of Edinburgh, Scotland

<sup>2</sup>Department of Computer Science and Engineering, Ewha Womans University, Korea

<sup>3</sup>LAAS-CNRS / Université de Toulouse, France

<sup>4</sup>Industrial Engineering Department, University of Trento, Italy

the friction cone contains the gravity). This is achieved by extending the 2-PAC approach [17] to handle variable foot positions (Section V-A.2), while we leave for future work the extension of the formulation to fully dynamic cases such as [11], which is not the focus of this paper.

Lastly, we combine the contact planner with a guide-path planner to automatically initialize the contact planning problem with relevant contact surfaces (Section IX-C).

## II. RATIONALE

Given the initial and final sets of admissible contact postures,  $\mathcal{I}$  and  $\mathcal{G}$ , SLIM computes a sequence of contact locations connecting  $\mathcal{I}$  to  $\mathcal{G}$  such that a feasible motion exists between each consecutive contact posture. The following assumptions are made:

- Dynamics constraints are verified by exhibiting a “quasi-static” trajectory for the center of mass (COM).
- The planner is limited to “quasi-flat” contact surfaces (for which the friction cone contains the gravity). This limitation is due to our dynamics approximation and not to the l1-norm approximation.
- The kinematics constraints on the COM are approximated as linear inequalities attached to the frame of each effector [18], [17]. Similarly the relative positions of the effectors is linearly constrained.
- The yaw orientations of the contacts at each step is a given, as opposed to [10]. Not handling orientation is a design choice, motivated by the objective to combine the contact planner with a higher-level method (detailed in Section IX-C).
- As for [10] the planner requires as input a set of convex potential contact surfaces (e.g., they may be given by a sampling-based guide-path planner).
- The user must provide an initial guess of the order followed by the effectors to create contacts (in our experiments we assume a cyclic gait pattern, which is not limiting for biped robots).

## III. DEFINITIONS AND NOTATIONS

### A. Important note on the mathematical formulation

To simplify the equations, we present the problem for the case of gaited bipedal walking. Section VII discusses the generalization to acyclic locomotion.

### B. Problem variables

*Contact phases:* The motion of the robot is decomposed into a discrete sequence of  $n_{ph}$  **contact phases**. The contact phase  $k \in [1, n_{ph}]$  is associated with one or several effectors in contact. Exactly one contact is broken and relocated from phase  $k$  to  $k + 1$ . For bipedal walking each contact phase is a double support phase (both feet in contact).

*End-effectors:* Any effector is identified by an index  $j \in [1, n_{eff}]$ , where  $n_{eff}$  is the number of effectors. The position of the effector moving (swing foot) between phases  $k - 1$  and  $k$  is noted  $\mathbf{p}^k = [p_x^k, p_y^k, p_z^k] \in \mathbb{R}^3$ . The position of the non-moving effector (support foot) is thus  $\mathbf{p}^{k-1}$ .  $\mathbf{p}^0$  is the position of the support effector for the first phase.

*Convex hulls:* The convex hull of all the effectors in contact at phase  $k$  is noted  $\mathbf{conv}^k$ .

*Center Of Mass (COM):* The COM of the robot  $\mathbf{c} \in \mathbb{R}^3$  moves continuously with a negligible acceleration, under a quasi-static assumption. The quasi-static assumption is only used to provide a certificate of feasibility, such that the resulting motions do not follow this constraint.

*Contact surfaces:* The possible contact locations are convex surfaces  $\mathcal{S}_i$ ,  $1 \leq i \leq n_{surf}$ . A surface  $\mathcal{S}_i$  is defined by  $n_i + 1$  planes: the plane aligned with the surface defines an equality constraint, while the  $n_i$  planes that bound the surface define as many inequality constraints:

$$\mathcal{S}_i : \{\mathbf{p} \in \mathbb{R}^3, \mathbf{S}_i \mathbf{p} \leq \mathbf{s}_i, \mathbf{d}_i^T \mathbf{p} = e_i\}$$

*Slack variables:* Our formulation uses two kinds of slack variables. Positive variables are denoted  $\boldsymbol{\alpha}^k = [\alpha_1^k, \dots, \alpha_n^k]^T \in \mathbb{R}^{n+}$  and the others  $\boldsymbol{\beta}^k = [\beta_1^k, \dots, \beta_n^k]^T \in \mathbb{R}^n$ , with  $k$  the related contact phase.

*Cardinality:* The cardinality operator  $\mathbf{card}(\boldsymbol{\alpha})$  gives the number of entries of  $\boldsymbol{\alpha}$  that are strictly greater than 0.

## IV. CONTACT SURFACE SELECTION CONSTRAINTS

When in contact, the position  $\mathbf{p}$  of a foot must belong to a candidate contact surface  $\mathcal{S}_i$ . In the following we detail the convex relaxation of this non-convex constraint.

### A. Logical OR as a “Minimal number of violations” problem

We look for a point  $\mathbf{p}$  belonging to exactly one surface of a set  $\mathbf{S} = \bigcup_{i=1}^{n_{surf}} \mathcal{S}_i$ . We assume that the surfaces  $\mathcal{S}_i$  do not intersect. The problem is written:

$$\begin{aligned} &\text{find } \mathbf{p}, \\ &\text{s. t. } \mathbf{p} \in \mathcal{S}_1 \quad \vee \quad \mathbf{p} \in \mathcal{S}_2 \dots \quad \vee \quad \mathbf{p} \in \mathcal{S}_{n_{surf}} \end{aligned} \quad (1)$$

With  $\vee$  the logical *or* operator. Solving problem (1) is equivalent to finding  $\mathbf{p}$  such that as many constraints  $\mathbf{p} \in \mathcal{S}_i$  as possible are satisfied, which corresponds to 1 constraint. This optimum is only reached if  $\mathbf{p}$  effectively belongs to one of the surfaces. This is in turn equivalent to finding  $\mathbf{p}$  such that the *minimum number of violations* occurs, which is  $n_{surf} - 1$ . This can be written as a cardinality minimization problem:

$$\begin{aligned} &\text{find } \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta} \\ &\text{min } \mathbf{card}(\boldsymbol{\alpha}) \\ &\text{s. t. } \mathbf{S}_i \mathbf{p} \leq \mathbf{s}_i + \mathbf{1} \boldsymbol{\alpha}_i \quad \forall i \\ &\quad \mathbf{d}_i^T \mathbf{p} = e_i + \beta_i \quad \forall i \\ &\quad -\alpha_i \leq \beta_i \leq \alpha_i \quad \forall i \end{aligned} \quad (2)$$

where  $\mathbf{1}$  is a vector of appropriate size filled with 1. Let us explain why (2) and (1) are equivalent. Assume a given  $i$  such that  $\mathbf{p} \notin \mathcal{S}_i$ . Then, either  $\alpha_i \neq 0$  or  $\beta_i \neq 0$ ; moreover, the last constraint of (2) (equivalent to  $|\beta_i| \leq \alpha_i$ ) implies that if  $\beta_i \neq 0$ , then  $\alpha_i \neq 0$ . We conclude that if  $\mathbf{p} \notin \mathcal{S}_i$  then  $\alpha_i \neq 0$ . If instead  $\mathbf{p} \in \mathcal{S}_i$ , then  $\beta_i = 0$  and  $\alpha_i$  is going to be set to zero by the solver because this decreases the objective function. The function  $\text{card}(\alpha)$  thus counts the number of surfaces on which  $\mathbf{p}$  does not lie. Therefore, the optimal value of  $\text{card}(\alpha)$ , which is  $n_{\text{surf}} - 1$ , is obtained by choosing  $\mathbf{p}$  to lie on one of the surfaces, which is exactly what (1) does.

Rather than addressing this problem with a branch and bound approach (as in MI programming), we can use a convex relaxation of the problem, approximating  $\text{card}(\alpha)$  with the L1-norm  $\|\alpha\|_1$ . Moreover, since the elements of  $\alpha$  are nonnegative, we have  $\|\alpha\|_1 = \mathbf{1}^\top \alpha$ . This approximation of (2) is thus a Linear Program:

$$\begin{aligned} \min \quad & \mathbf{1}^\top \alpha \\ \text{s. t.} \quad & \mathbf{S}_i \mathbf{p} \leq \mathbf{s}_i + \mathbf{1} \alpha_i \quad \forall i \\ & \mathbf{d}_i^\top \mathbf{p} = e_i + \beta_i \quad \forall i \\ & -\alpha_i \leq \beta_i \leq \alpha_i \quad \forall i \end{aligned} \quad (3)$$

Because L1-minimization leads to sparse solutions, at the optimum we can hope that one  $\alpha_i$  will be (sufficiently close to) 0. This indicates that the surface  $\mathcal{S}_i$  has been selected.

## V. REACHABILITY CONSTRAINTS

Two kinds of additional constraints are considered in the model. First, the position of the COM is constrained with respect to the contact points, following the 2-PAC approach [17]. This allows to continuously guarantee feasibility of the COM trajectory while only considering two discrete COM positions at each contact phase. We recall the method for completeness and extend it to handle variable foot translations under the quasi-flat constraint. Then, the position of each effector is constrained with respect to the other effector in contact.

### A. Center Of Mass constraints

To guarantee that the equilibrium and balance constraints are continuously satisfied for a contact phase  $k$ , we will use the 2-PAC formulation. In the following we recall that we only need to choose 2 COM positions for each phase, namely  $\mathbf{c}^{k,0}$  and  $\mathbf{c}^{k,1}$ , to guarantee continuous feasibility.

1) *Equilibrium constraints:* For quasi-flat contact surfaces, a sufficient condition for the COM to allow for static equilibrium is:  $\mathbf{c}^k \in \text{conv}^k$  [19]. For bipedal walking, this boils down to having the COM on top of the support effector. In this case  $\mathbf{c}^{k,0}$  is constrained to lie above the support polygon of  $\mathbf{p}^{k-1}$  (i.e. the support foot used in the transition from phase  $k-1$  to  $k$ , which was the swing foot for phase  $k-1$ ) at the beginning of phase  $k$ . We then constrain  $\mathbf{c}^{k,1}$  to be above  $\mathbf{p}^k$  at the end of phase  $k$  (Fig. 2):

$$\begin{aligned} \mathbf{F}_i^{k-1}(\mathbf{c}^{k,0} - \mathbf{p}^{k-1}) &\leq \mathbf{f}_i^{k-1} + \mathbf{1} \alpha_i^{k-1} \\ \mathbf{F}_i^k(\mathbf{c}^{k,1} - \mathbf{p}^k) &\leq \mathbf{f}_i^k + \mathbf{1} \alpha_i^k \end{aligned} \quad (4)$$

where  $\mathbf{F}_i^k$  and  $\mathbf{f}_i^k$  are the matrix and vector defining the polygonal shape of the foot associated to phase  $k$  on surface  $\mathcal{S}_i$ . Note that these constraints depend only on the xy coordinates of the COM and the foot positions.

By convexity of the static equilibrium region, the straight line  $[\mathbf{c}^{k,0}, \mathbf{c}^{k,1}]$  continuously satisfies the static equilibrium constraint. Similarly, the straight lines  $[\mathbf{c}^{k-1,1}, \mathbf{c}^{k,0}]$  and  $[\mathbf{c}^{k,1}, \mathbf{c}^{k+1,0}]$  are also feasible because the COM stays above the support effector for all the duration of the single support phase (Fig. 2).

2) *Reachability constraints:* We additionally constrain  $\mathbf{c}^{k,0}$  and  $\mathbf{c}^{k,1}$  to guarantee kinematic reachability. We stress that the kinematic constraints are only approximated here, thus the “guarantees” that we mention for feasibility are only valid for this simplified representation of the robot. The COM positions are linearly constrained as follows. First, for each effector we compute offline a polytope that approximates the reachable COM workspace: a large number of configurations of the robot are randomly sampled, and those who are collision-free and correspond to a “quasi-flat” pair of contacts are kept. For each of those configurations, the COM is expressed in the frame of a given effector. The convex hull of all the computed COM positions approximates the COM workspace in the effector frame. For each effector  $j$ , we thus obtain a 3D polytope  ${}^j\mathcal{R} : \{\mathbf{c} \in \mathbb{R}^3, {}^j\mathbf{R}\mathbf{c} \leq {}^j\mathbf{r}\}$ .

At contact phase  $k$ , for each contact surface  $\mathcal{S}_i$  the orientation of the foot frame is constant<sup>1</sup>. We note  $\mathcal{R}_i^k$  the rotated polytope associated with contact  $\mathbf{p}^k$  at phase  $k$ , assuming it lies on surface  $\mathcal{S}_i$ . The translation is variable, thus the constraints depend linearly on the effector positions. Both COM positions  $\mathbf{c}^{k,m}, m \in \{0,1\}$  at phase  $k$  are thus constrained by the two active contacts  $\mathbf{p}^k$  and  $\mathbf{p}^{k-1}$ :

$$\mathbf{R}_i^l(\mathbf{c}^{k,m} - \mathbf{p}^l) \leq \mathbf{r}_i^l + \mathbf{1} \alpha_i^l \quad \forall i, \forall l \in \{k-1, k\} \quad (5)$$

<sup>1</sup>The yaw is a given of the problem, while the roll and pitch are given by the surface orientation

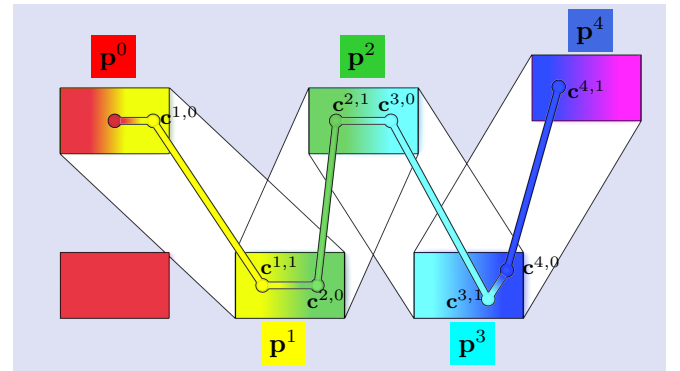


Fig. 2: From an initial double support phase (red), 4 double support contact phases are planned. Each contact is used for 2 consecutive phases. The feasible COM quasi-static trajectory is given by a polyline. Color gradients are used when the COM moves inside the support effector during the simple support phase, and plain colors are used to depict the COM trajectory during a double support phase.

Here again, the slack variable  $\alpha$  is used such that only the constraints related to the selected contact surfaces are applied. By convexity of the kinematic constraints, if they are satisfied for  $\mathbf{c}^{k,0}$  and  $\mathbf{c}^{k,1}$  for all  $k$  then they are continuously satisfied.

### B. Relative foot position constraints

Similarly to the case of the COM reachability, we use a sampling-based approach to approximate the reachable workspace of each foot with respect to the others. For effector  $j$ , we obtain a polytope  ${}^j\mathcal{Q} : \{\mathbf{p} \in \mathbb{R}^3, {}^j\mathbf{Q}\mathbf{p} \leq {}^j\mathbf{q}\}$  that constrains the other effector. If  $j$  is the moving effector at phase  $k$  on surface  $\mathcal{S}_i$ , we abusively write  $\mathcal{Q}_i^k = {}^j\mathcal{Q}_i$  for clarity. We then apply the same reasoning as for the COM to obtain the following constraints at each phase:

$$\mathbf{Q}_i^{k-1}(\mathbf{p}^k - \mathbf{p}^{k-1}) \leq \mathbf{q}_i^{k-1} + \mathbf{1}\alpha_i^{k-1} \quad \forall i \quad (6)$$

### C. An important simplification

If the contact surfaces for the effectors are punctual (as for most quadruped robots), or if the candidate surfaces share the same orientation, then the polytope constraints  $\mathcal{R}_i^k$  (respectively  $\mathcal{Q}_i^k$  and  $\mathcal{F}_i$ ) are the same for all  $i$ . In this case we do not need to use the selection variable  $\alpha$  in equations 4, 5 and 6.

## VI. THE COMPLETE FEASIBILITY PROBLEM

Combining all the constraints, we can now write the complete feasibility problem that defines SLIM. For the sake of simplicity, and without loss of generality, we assume that all contact surfaces are potential candidates for all phases.

$$\begin{aligned} \text{find } & \mathbf{u} = [\mathbf{p}^1, \dots, \mathbf{p}^{n_{ph}}, \alpha^1, \dots, \alpha^{n_{ph}}, \beta^1, \dots, \beta^{n_{ph}}, \\ & \mathbf{c}^{1,0}, \dots, \mathbf{c}^{n_{ph},0}, \mathbf{c}^{1,1}, \dots, \mathbf{c}^{n_{ph},1}] \\ \text{min } & \sum_{k=1}^{n_{ph}} \mathbf{1}^T \alpha^k + \gamma l(\mathbf{u}) \\ \text{s. t. } & \{\mathbf{p}^1, \mathbf{c}^{1,0}\} \in \mathcal{I} \\ & \{\mathbf{p}^{n_{ph}-1}, \mathbf{p}^{n_{ph}}, \mathbf{c}^{n_{ph},1}\} \in \mathcal{G} \\ & \forall k, i : \\ & \quad \mathbf{S}_i \mathbf{p}^k \leq \mathbf{s}_i + \mathbf{1}\alpha_i^k \\ & \quad \mathbf{d}_i^T \mathbf{p}^k = e_i + \beta_i^k \\ & \quad -\alpha_i^k \leq \beta_i^k \leq \alpha_i^k \\ & \quad \mathbf{F}_i^{k-1}(\mathbf{c}^{k,0} - \mathbf{p}^{k-1}) \leq \mathbf{f}_{i,k-1}^i + \mathbf{1}\alpha_i^{k-1} \\ & \quad \mathbf{F}_i^k(\mathbf{c}^{k,1} - \mathbf{p}^k) \leq \mathbf{f}_i^k + \mathbf{1}\alpha_i^k \\ & \quad \mathbf{Q}_i^k(\mathbf{p}^k - \mathbf{p}^{k-1}) \leq \mathbf{q}_i^k + \mathbf{1}\alpha_i^k \\ & \quad \forall l \in \{k-1, k\}, \forall m \in \{0, 1\} : \\ & \quad \mathbf{R}_i^l(\mathbf{c}^{k,m} - \mathbf{p}^l) \leq \mathbf{r}_i^l + \mathbf{1}\alpha_i^l \end{aligned} \quad (7)$$

where  $l$  is an optional quadratic or linear objective function and  $\gamma$  is a small weighing value.

$\mathcal{I}$  and  $\mathcal{G}$  define initial and goal state conditions. The initial/goal values for contact locations and COM can be given exactly, or loosely specified. For instance the last contact can be constrained to lie on a given contact surface, which is convenient for specifying the goal state in general.

## VII. EXTENSION TO ACYCLIC MULTI-CONTACT LOCOMOTION

So far we have presented an approach to plan contacts for bipedal locomotion on quasi-flat terrains. However, we believe that this approach can be extended to multi-contact locomotion, as long as contact surfaces remain quasi-flat.

To do this, the main change needed regards the equilibrium constraints (Section V-A.1):  $\mathbf{c}^k \in \text{conv}^k$ . If we assume point-like contacts, these constraints can be written as:

$$\mathbf{c}^k = \sum_j w_j^k \mathbf{p}_j^k, \quad \sum_j w_j^k = 1, \quad w_j^k \geq 0 \quad \forall j, \quad (8)$$

where the  $\mathbf{p}_j^k$ 's are the positions of the effectors in contact at phase  $k$ , and  $\mathbf{w}^k$  is a unit weighting vector. Since the contact positions are variable, (8) is bilinear, hence non-convex. To maintain the formulation convex, we could choose a conservative approach. For each phase we can fix  $\mathbf{w}^k$ , such that the COM xy coordinates are directly determined from the contact positions. A similar approach could be used for finite-size contact surfaces: we can express the COM as a convex combination of virtual contact points  $\bar{\mathbf{p}}_j^k$ , which must lie inside the associated effector surfaces with position  $\mathbf{p}_j^k$ :

$$\begin{aligned} \mathbf{c}^k &= \sum_j w_j^k \bar{\mathbf{p}}_j^k \\ \mathbf{F}_i^k(\bar{\mathbf{p}}_j^k - \mathbf{p}^k) &\leq \mathbf{f}_i^k + \mathbf{1}\alpha_i^k \end{aligned} \quad (9)$$

Apart from this new formulation of the equilibrium constraints, problem (7) could be used almost as it is for multi-contact problems—even though we plan to test this extension in future work.

## VIII. FIXING THE SPARSITY

### A. Near sparse convergence

Once problem (7) is solved, in the ideal case, for each phase  $k$  one and only one  $\alpha_i^k$  value is equal or sufficiently close to 0. We can then fix the sparsity by assigning to each phase its selected surface and solve problem (7) without the slacks  $\alpha$  and  $\beta$  again to obtain an exact solution.

It might be the case that for some phases no  $\alpha_i^k$  is close enough to 0. This indicates either that the problem is unfeasible or that the convex relaxation leads to a non-sparse optimum. If the number of concerned phases is low, the non-sparse minimum might be close enough to a sparse optimum. We then solve a small combinatorial problem. We test all the possible combinations of surface selection for the unresolved phases by assigning one and only one single contact surface to each unresolved phase, and we solve again problem (7). We can either test all combinations or stop at the first feasible solution, in which case the resolution remains computationally efficient.

In our tests we noticed that, if the problem was feasible, choosing the surfaces associated to the lowest  $\alpha_i^k$  led to the desired sparse solution in the majority of cases.

### B. Handling non-sparse optimum

If too many phases are unresolved (i.e. no  $\alpha_i^k \approx 0$ ) the non-sparse optimum might be too far from a sparse optimum.

This happens with a brute force formulation of our main test scenario (Fig. 1), where the start and goal configurations are spatially close, but a long path is required to connect them. This is a classical case where nonlinear solvers would fail by trying to directly connect the states.

This issue can be mitigated by initializing the solver with a relevant initial guess. Fortunately an intuitive solution can be built either manually or automatically, by limiting the set of contact surfaces available at each phase. If we separate the scene into 2 distinct sets of potential contact surfaces (“before” and “after” the top of the staircase), the solver is able to converge to a valid solution.

Proper and automatic initialization is thus essential for large problems, and discussed in Section IX-C.

## IX. RESULTS

SL1M has been successfully tested in simulation for two humanoid robots with different kinematic constraints, namely HRP-2 and Talos [20]. In this section we also present a quantitative comparison of the performances of our approach with respect to a mixed integer implementation.

### A. Implementation details

Problem (7) has been implemented in python, using the sparse, open source linear solver GLPK [21]<sup>2</sup>. For comparison the problem has also been implemented in its MI form using gurobi [22], interfaced with cvxpy [23]. Tests were run on a laptop with an i7 processor on ubuntu 16.04. Our code will be open sourced upon acceptance of the paper.

### B. Quantitative analysis and comparison with MI

To determine empirically the capabilities of SL1M and compare it with the MI approach, we designed two experiments and computed the ratio between the obtained computation times for each formulation. The computation times for SL1M include the extra steps required to fix the sparsity described in Section VIII-A. The presented times are averaged over 10 runs.

*Toy problem:* We consider an environment composed of a single flat surface, on which the robot is required to walk between 2 and 38 steps. Variations of this scenario involve iteratively splitting the surface to artificially create more contact candidates.

The results are presented by the color map in Fig. 3 - top. The computation times range from 50 ms to about 17 s for MI, and from 0.47 ms to 585 ms for SL1M. In this scenario, SL1M is always faster, with a speed up factor between 1.5 and 100. The most favorable setups are the smaller problems.

We conclude that, for this toy problem, SL1M outperforms MI in finding a feasible contact sequence, without any significant drawback.

<sup>2</sup>Once the sparsity is fixed, we use Quadprog with a quadratic cost that favors contact positions at the center of the contact surfaces

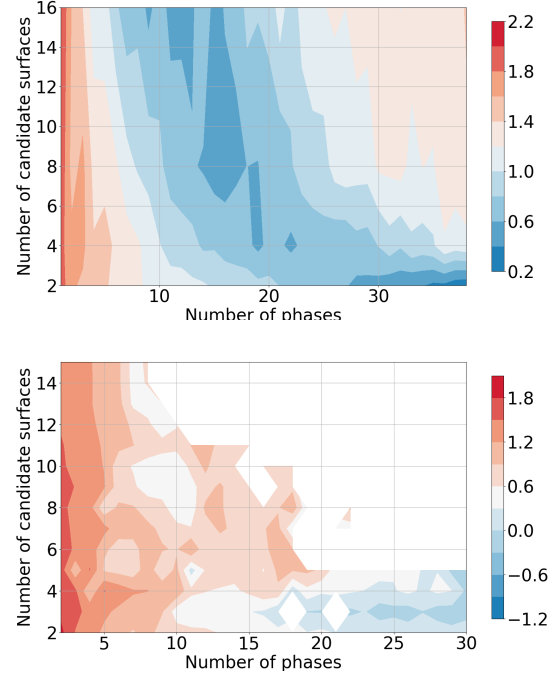


Fig. 3: Ratio of computation times obtained for SL1M over the times for the MI formulation for the toy planar scenario (up) and the LAAS experiment room (down) in logarithmic 10 scale.

*Challenging scenario:* We now consider the LAAS experiment room shown in Fig. 1. Again, we change the size of the problem by changing the number of steps and possible contact surfaces. To achieve this, we start from a known feasible solution of the problem connecting the first and last poses of Fig. 1. We obtain 30 contact phases with exactly one contact candidate for each phase. To increase the number of surface candidates for a phase  $k$ , we add those of the surrounding phases  $k - 1$  and  $k + 1$ , and iterate as required. The problem is thus always feasible. To reduce the number of steps, we simply cut the last phases.

The results are presented by the color map in Fig. 3 - bottom. For this scenario, when the L1-norm did not converge to a sparse solution, we decided to mark the scenario as unsolved if the combinatorial involved more than 4000 possible cases, otherwise we let the combinatorial run (with the possibilities sorted according to the  $\alpha_i^k$  values) until a feasible solution was found. The upper right part of the plot is thus left blank, indicating a “failure” of the L1-norm. On the other hand MI was always successful. In this case the computation times range from 43 ms to 15 s for MI, and from less than 1 ms to 3.5 s for SL1M. Fig. 3 shows some cases where the MI is faster (up to 10 times) as the number of phases increase. The drawback of SL1M is also highlighted by the failures registered for large problems. However we still observe that SL1M outperforms significantly MI for small and medium problems (up to 40 times faster).

These results suggest that SL1M is especially suited for



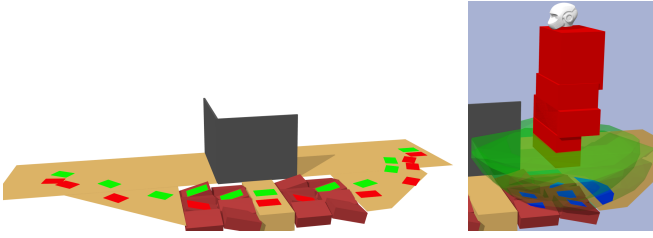


Fig. 4: Left: Contact sequences on uneven surfaces for Talos. Right: For the root location, the approximated reachable workspace of the feet (green) gives a set of potential contact surfaces (blue). The right shape approximates the collision avoidance constraint, similarly to [10].

scenarios involving few potential contact surfaces at each phase, while additional work is required to analyze in depth the failures that occur for larger problems, and propose a mean to address them, for instance by considering stricter norms to enforce the sparsity [15].

### C. Qualitative validation of the contact sequence

To validate SLIM qualitatively, the generated contact sequences are given as input to an open-source whole-body motion generator<sup>3</sup> presented in [24], [25]. For lack of space we cannot describe this generator in detail here.

In our demonstrations, rather than manually providing the potential contact surfaces and orientations, we initialize (7) with a sampling-based guide path planner [16]. Given a 3D mesh as well as start and goal configurations for the root of the robot, the planner computes a 6D collision-free trajectory for this root. The planner is able to return, for each position of the root along the path, a set of potential contact surfaces for each effector (Fig. 4 - right). Because the contact surfaces correspond to collision-free positions of the root, the output of the contact planner is more likely to avoid collisions (Fig. 4). The guide path is thus discretized into  $n_{ph}$  contact phases, from which we extract the potential contacts for each phase. The root yaw orientation at each phase  $k$  becomes the orientation constraint for the moving foot  $\mathbf{p}^k$ .

The companion video demonstrates that contact plans such as the ones displayed in Fig. 1, 4 and 5 can be extended to dynamically consistent and collision-free motions. The motions with Talos have been validated on the simulator Gazebo, using a torque controller provided by the robot's manufacturer PAL robotics.

## X. CONCLUSIONS AND FUTURE WORK

We present SLIM, a convex relaxation of the mixed integer (MI) programming approach for planning contact sequences for legged robots. For the issue of finding a feasible contact sequence, L1-norm minimization seems particularly well suited for small and medium size problems, where SLIM outperforms MI by at least one order of magnitude, at the cost of relegating optimality to a secondary objective.

<sup>3</sup><https://github.com/loco-3d/multicontact-locomotion-planning>

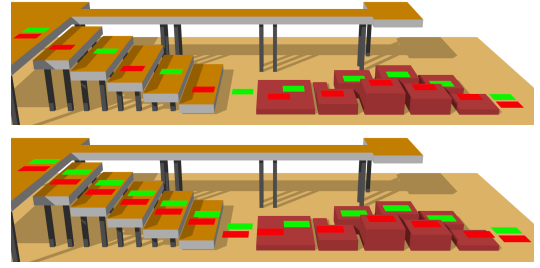


Fig. 5: Example of contact sequences in the LAAS experimental room for Talos (top) and HRP-2 (bottom). Interestingly the kinematic constraints of each robot lead to different strategies.

We plan to extend the presented approach and its assessment in different ways. First, our comparison with the MI approach has shown that any approach can outperform the other one depending on the situation. We want to carry out further tests to understand under which conditions the L1 optimization fails to provide a sparse solution, and explore whether it could be enforced using stricter norm formulations [15].

Another promising direction is the integration of contact planning and guide-path planning, which we have already started to explore in this paper. Using a guide-path planner we can drastically reduce the number of candidate contact surfaces for each contact. This greatly simplifies the contact planning problem, and can make the difference between failure and success. However, it also reduces the exploration space of the contact planner, which may be detrimental. Quantifying the benefits of guide-path planning is subject of ongoing investigation.

While this work has focused on bipedal walking, the presented approach can be applied to any acyclic multi-contact locomotion (as long as contacts are quasi-flat, see Section VII). Thus, we plan to test our method with other gaits and with quadruped robots. In the future we would also like to extend the formulation to integrate fully dynamic constraints in the planning by using a convex formulation of the problem [26].

Finally, other robotics problems such as collision-free inverse kinematics [27] or trajectory planning [28], [29] can be addressed using Mixed Integer programming. Investigating the use of L1-norm minimization for those problems provides an exciting direction for further research.

## XI. ACKNOWLEDGEMENTS

The authors would like to thank Joseph Mirabel and Jean-Michel Tonneau for their help, as well as PAL-Robotics for providing a simulator for Talos.

This work is supported by the National Research Foundation (NRF) in South Korea (2017R1A2B3012701) and the H2020 project Memmo (ICT-7801684).

## REFERENCES

- [1] I. Mordatch, E. Todorov, and Z. Popović, “Discovery of complex behaviors through contact-invariant optimization,” *ACM Trans. on Graph.*, vol. 31, no. 4, pp. 43:1–43:8, 2012.
- [2] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli, “Gait and Trajectory Optimization for Legged Systems through Phase-based End-Effector Parameterization,” *IEEE Robotics and Automation Letters*, pp. 1–1, 2018. [Online]. Available: <http://ieeexplore.ieee.org/document/8283570/>
- [3] K. Hauser, T. Bretl, K. Harada, and J.-C. Latombe, “Using motion primitives in probabilistic sample-based planning for humanoid robots,” in *WAFR*, ser. Springer Tracts in Advanced Robot., S. Akella, N. M. Amato, W. H. Huang, and B. Mishra, Eds., vol. 47. Springer, 2006.
- [4] A. Escande, A. Kheddar, and S. Miossec, “Planning contact points for humanoid robots,” *Robotics and Autonomous Systems*, vol. 61, no. 5, pp. 428 – 442, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0921889013000213>
- [5] S. Tonneau, A. Del Prete, J. Pettré, C. Park, D. Manocha, and N. Mansard, “An efficient acyclic contact planner for multiped robots,” *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 586–601, June 2018.
- [6] J. Carpentier and N. Mansard, “Multicontact locomotion of legged robots,” *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1441–1460, 2018.
- [7] J. Chestnutt, K. Nishiwaki, J. Kuffner, and S. Kagami, “An adaptive action model for legged navigation planning,” in *2007 7th IEEE-RAS International Conference on Humanoid Robots*, Nov 2007, pp. 196–202.
- [8] Y. Lin and D. Berenson, “Humanoid navigation planning in large unstructured environments using traversability - based segmentation,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 7375–7382.
- [9] R. J. Griffin, G. Wiedebach, S. McCrory, S. Bertrand, I. Lee, and J. E. Pratt, “Footstep planning for autonomous walking over rough terrain,” *CoRR*, vol. abs/1907.08673, 2019. [Online]. Available: <http://arxiv.org/abs/1907.08673>
- [10] R. Deits and R. Tedrake, “Footstep planning on uneven terrain with mixed-integer convex optimization,” in *Humanoid Robots (Humanoids), 14th IEEE-RAS Int. Conf. on*, Madrid, Spain, 2014.
- [11] B. Ponton, A. Herzog, S. Schaal, and L. Righetti, “A convex model of humanoid momentum dynamics for multi-contact motion generation,” in *Proceedings of the 2016 IEEE-RAS International Conference on Humanoid Robots*, 2016.
- [12] B. Aceituno-Cabezas, C. Mastalli, H. Dai, M. Focchi, A. Radulescu, D. Caldwell, J. Cappelletto, J. C. Grieco, G. Fernández-López, and C. Semini, “Simultaneous Contact, Gait and Motion Planning for Robust Multi-Legged Locomotion via Mixed-Integer Convex Optimization,” *IEEE Robotics and Automation Letters*, pp. 1 – 1, Dec. 2017. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01674935>
- [13] J. M. F. Bach, R. Jenatton and G. Obozinski, “Convex optimization with sparsity-inducing norms,” *Optimization for Machine Learning*, 2011.
- [14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004. [Online]. Available: <https://web.stanford.edu/~boyd/cvxbook/>
- [15] M. Skouras, B. Thomaszewski, S. Coros, B. Bickel, and M. Gross, “Computational design of actuated deformable characters,” *ACM Trans. Graph.*, vol. 32, no. 4, pp. 82:1–82:10, Jul. 2013. [Online]. Available: <http://doi.acm.org/10.1145/2461912.2461979>
- [16] P. Fernbach, S. Tonneau, A. Del Prete, and M. Taïx, “A kinodynamic steering-method for legged multi-contact locomotion,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2017, pp. 3701–3707.
- [17] S. Tonneau, P. Fernbach, A. Del Prete, J. Pettré, and N. Mansard, “2pac: Two-point attractors for center of mass trajectories in multi-contact scenarios,” *ACM Trans. Graph.*, vol. 37, no. 5, pp. 176:1–176:14, Oct. 2018. [Online]. Available: <http://doi.acm.org/10.1145/3213773>
- [18] C. Brasseur, A. Sherikov, C. Collette, D. Dimitrov, and P. Wieber, “A robust linear mpc approach to online generation of 3d biped walking motion,” in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, Nov 2015, pp. 595–601.
- [19] A. Del Prete, S. Tonneau, and N. Mansard, “Fast Algorithms to Test Robust Static Equilibrium for Legged Robots,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden, 2016.
- [20] O. Stasse, T. Flayols, R. Budhiraja, K. Giraud-Esclasse, J. Carpentier, J. Mirabel, A. Del Prete, P. Souères, N. Mansard, F. Lamiriaux, J.-P. Laumond, L. Marchionni, H. Tome, and F. Ferro, “TALOS: A new humanoid research platform targeted for industrial applications,” in *International Conference on Humanoid Robotics, ICHR, Birmingham 2017*, ser. IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids), Birmingham, United Kingdom: IEEE, Nov. 2017. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01485519>
- [21] A. Makhorin, “Glpk (gnu linear programming kit),” <http://www.gnu.org/software/glpk/glpk.html>, 2008.
- [22] L. Gurobi Optimization, “Gurobi optimizer reference manual,” 2019. [Online]. Available: <http://www.gurobi.com>
- [23] S. Diamond and S. Boyd, “CVXPY: A Python-embedded modeling language for convex optimization,” *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.
- [24] J. Carpentier, A. Del Prete, S. Tonneau, T. Flayols, F. Forget, A. Mifsud, K. Giraud, D. Atchuthan, P. Fernbach, R. Budhiraja *et al.*, “Multi-contact locomotion of legged robots in complex environments—the loco3d project,” 2017.
- [25] P. Fernbach, S. Tonneau, O. Stasse, J. Carpentier, and M. Taïx, “C-CROC: Continuous and Convex Resolution of Centroidal dynamic trajectories for legged robots in multi-contact scenarios,” Aug. 2019, working paper or preprint. [Online]. Available: <https://hal.laas.fr/hal-01894869>
- [26] P. Fernbach, S. Tonneau, and M. Taïx, “Croc: Convex resolution of centroidal dynamics trajectories to provide a feasibility criterion for the multi contact planning problem,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.
- [27] H. Dai, G. Izatt, and R. Tedrake, “Global inverse kinematics via mixed-integer convex optimization,” *The International Journal of Robotics Research*, vol. 0, no. 0, p. 0278364919846512, 0. [Online]. Available: <https://doi.org/10.1177/0278364919846512>
- [28] R. Deits and R. Tedrake, “Efficient mixed-integer planning for uavs in cluttered environments,” *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 42–49, 2015.
- [29] W. Sun, G. Tang, and K. Hauser, “Fast UAV trajectory optimization using bilevel optimization with analytical gradients,” *CoRR*, vol. abs/1811.10753, 2018. [Online]. Available: <http://arxiv.org/abs/1811.10753>