ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

A COMPUTATIONAL STRATEGY FOR TRAJECTORY OPTIMIZATION OF UNDERACTUATED MULTIBODY SYSTEMS WITH CONTACTS

Silvia Manara[1], **Alessio Artoni**[1], **and Marco Gabiccini**[1,2,3]

[1]Dipartimento di Ingegneria Civile e Industriale
Largo Lucio Lazzarino 1, 56122 Pisa, Italy
e-mail: silvia.manara@for.unipi.it
{a.artoni,m.gabiccini}@ing.unipi.it

[2] Research Center "E. Piaggio", University of Pisa
Largo Lucio Lazzarino 1, 56122 Pisa, Italy

[3] Department of Advanced Robotics, Istituto Italiano di Tecnologia
Via Morego 30, 16163 Genova, Italy

Keywords: Optimization, Trajectory Planning, Optimal Control, Computational Methods.

Abstract. *In this paper we propose a strategy for improving the computational efficiency of direct methods for trajectory optimization of multibody systems. We particularly focus on those applications where the system necessarily has to interact with the surrounding environment through intermittent contacts. The problem is hereby formulated such that just the initial and final states of the system over a given time interval are prescribed, so as to let the solver automatically synthesize the best contact sequence to accomplish the considered task. The proposed computational strategy consists in: (i) solving a preliminary optimization problem that roughly approximates the original one, but differs from it by one or more conveniently chosen parameters and is faster to solve; (ii) using the obtained solution as an initial guess for the actual (full-fledged) optimal control problem. The performance of the method is evaluated in a simulated planar system, whose peculiarity is to be trivially underactuated. An extensive investigation is presented which shows how a proper choice of the parameters in the preliminary optimization can lead to a significant reduction in the computational effort required to solve the problem. The results we present assess both the validity and the robustness of the proposed method.*

1 INTRODUCTION

The analysis of multibody systems interacting with the surrounding environment through intermittent contacts gives rise to non-trivial issues, mainly caused by the discontinuities due to friction and impulsive dynamics. However, in contexts such as robotic manipulation and locomotion, interactions between the system and parts of the environment through unilateral contact forces are essential in order to accomplish a given task.

The complexity of the trajectory planning problem in such contexts is due to the fact that, over time, the system may be in an enormous number of possible configurations, separated by collision events occurring when a new contact is established or a prior one is broken. However, reducing the search space dimension by *a priori* scheduling the contact sequence to be followed by the system may sometimes be difficult and, more notably, so restrictive to lead to the planning of unnatural behaviors. On the other hand, the combinatorial explosion of possible configurations caused by unspecified contacts can hardly be faced by sampling-based planners [1, 2]. Even the recently proposed multi-modal planners [3] can cope with just a limited number of contact modes.

Although promising methods for reducing the search space have been recently proposed [4], the robotics community is showing an increasing interest in optimization-based methods [5]. In fact, such methods, also known as *direct transcription* methods, are not significantly affected by the presence of *a priori* unknown interaction forces, since the formulation of the problem as an optimal control one allows the contact forces to be simply embedded in the structure of the problem itself, without altering it. Different approaches have been proposed to represent unilateral contact forces. In [6], where the contact invariant optimization is presented, a penalty term in the cost function accounts for the feasibility of the contact forces. In other works [7, 8], contact forces were added to the problem variables and their constitutive equations were listed among the constraints of the optimization problem. Since gradient-based methods can, in general, very hardly handle discontinuities, special care must be devoted to modeling contact forces. In [7], a complementarity formulation of contact (resulting in a Linear Complementarity Problem, LCP) is employed, whereas in [8] two different contact models, respectively based on complementarity and on a continuous penalty-based formulation, are evaluated. In [9], where an optimization framework for trajectory planning is proposed and applied to a simplified biped system performing a one-step walking motion, smoothing functions are introduced to relax the complementarity conditions involving normal and tangential contact forces.

Other investigations about the potential of gradient-based methods have mainly focused on trajectory planning of underactuated systems. The particular significance of this kind of systems is that they are paradigmatic of locomotion problems, where at least the degrees of freedom (DoF) of the floating base are not actuated. In [10], where a very complex humanoid model is considered, the problem is approached by considering just the centroidal dynamics of the system, as the resulting equations describe the effect of external wrenches (due to contact and gravitational forces) on the underactuated degrees of freedom. Instead in [11], the full system dynamics is considered. Here, a very detailed model of human lower limbs is used and contact invariant optimization is employed to make it discover complex motions, including walking. Different ways of ensuring dynamic feasibility have been proposed as well. In [11], consistently with the contact invariant approach, the violation of the equations of motion is heavily penalized in the cost function, whereas in other works the dynamics of the systems is added as a set of nonlinear constraints to the optimal control problem. Among these last mentioned works, an interesting contribution is given in [12], where an extremely simplified model of a

floating base legged robot is employed and, without specifying the contact sequence, different gaits are identified through direct trajectory optimization. Other assumptions are made, such as periodicity, which make this work focus more on the optimization of the stride itself rather than on finding more general walking behaviors.

Recent research efforts in robotics [13] have been devoted to provide further insight into the performance of direct optimization methods. In this paper, we present a study that aims to assess the performance through extensive numerical tests of different direct trajectory optimization schemes in terms of computational time and sensitivity to parameters. Additionally, an original strategy is devised that substantially reduces the computational time to convergence. Both the validity and the robustness of the proposed method are evaluated through simulation of a 3-DoF underactuated 2D system.

2 MODEL DESCRIPTION

2.1 System structure and task definition

The computational performance of different direct trajectory optimization schemes, which is the main object of the present paper, was evaluated for planning the trajectory of a benchmark system, represented in Fig. 1.

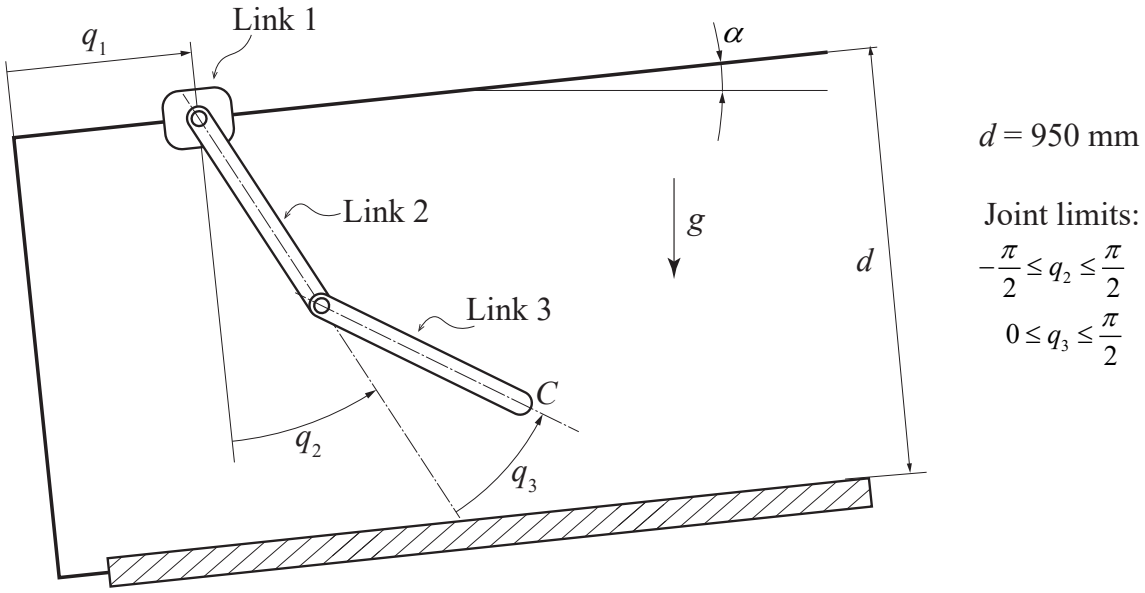

Figure 1: PRR model.

Link 1	Link 2	Link 3
$m_1 = 5$ kg	$m_2 = 3$ kg	$m_3 = 3$ kg
	$l_2 = 500$ mm	$l_3 = 500$ mm

Table 1: Geometric and inertial properties of the system.

This is a 3-DoF planar PRR system, consisting of three links connected together by two actuated revolute (R) joints and attached to a supporting frictionless track through a non-actuated

prismatic (P) joint.

From a given initial state, the system has to reach a specified final position, which is 3 m away along the direction of the track, within a fixed time horizon, and stop there. It is worth pointing out that this problem is not as simple as it could appear, as the system under investigation is trivially underactuated, and therefore it necessarily has to exploit the interaction with the underlying floor in order to produce the thrust force required to climb up the slope until the target final position is reached.

2.2 Formulation of the system dynamics

For the description of the system dynamics, we employ the Euler-Lagrange equations for holonomic systems:

$$B(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = Q(q,\dot{q},u) \tag{1}$$

where $q \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ collect, respectively, the configuration variables and the input torques at the revolute joints, $B(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is the matrix of Coriolis and centrifugal terms and $G(q) \in \mathbb{R}^n$ is the vector of gravitational forces. $Q(q,\dot{q},u) \in \mathbb{R}^n$ is the vector of generalized forces, whose j-th component can be expressed as:

$$Q_j(q,\dot{q},u) = \begin{cases} J_{v_C}(q)[:,j]^\top f_C(q,\dot{q}) & \text{if } j \in \bar{\mathscr{A}}, \\ J_{v_C}(q)[:,j]^\top f_C(q,\dot{q}) + u_j & \text{if } j \in \mathscr{A} \end{cases} \tag{2}$$

where J_{v_C} is the linear velocity Jacobian of the contact point C, f_C is a vector that represents the force acting at C because of the contact with the ground, and u_j is the input torque applied to the j-th joint. $\mathscr{A}$ and $\bar{\mathscr{A}}$ are, respectively, the set of actuated and non-actuated degrees of freedom, having cardinalities $|\mathscr{A}| = m$ and $\left|\bar{\mathscr{A}}\right| = n - m$.

2.3 Contact force model

The optimization of trajectories for systems with intermittent contacts requires a careful modeling of contact forces. In this paper we employ a penalty-based model, as it provides smoothness which conveniently fits with the Newton-type algorithms used to solve the optimization problem.

We approximate the unilateral linear contact model by the following differentiable constitutive relation (Fig. 2) between the normal component of the contact force f_n and the normal gap g_n separating the candidate contact point C from the ground (it depends on the internal configuration of the system, i.e. $g_n = g_n(q)$)

$$f_n(y) = f_0 \log_2\left(1 + 2^{-\frac{\kappa}{f_0} g_n}\right) \tag{3}$$

where f_0 is the force value at $g_n = 0$, and $-\kappa$ is the contact stiffness as $g_n \to -\infty$.
The parameters f_0 and κ provide a straightforward way to properly tune the model. For the problem under investigation, we chose to set $f_0 = 20$ N and $\kappa = 10^4$ N/m, as this tuning results in a realistic modeling of contact actions, without exacerbating the numerical stiffness of the problem. With these values, in fact, a very small penetration ($g_n = -5$ mm) yields a value of the normal force ($f_n \approx 50$ N) of the same order of magnitude as the forces actually required in our problem. Also, with this choice of the parameters, the effect of contact forces acting at a distance — which is one of the main drawbacks of the penalty-based formulation of contact —

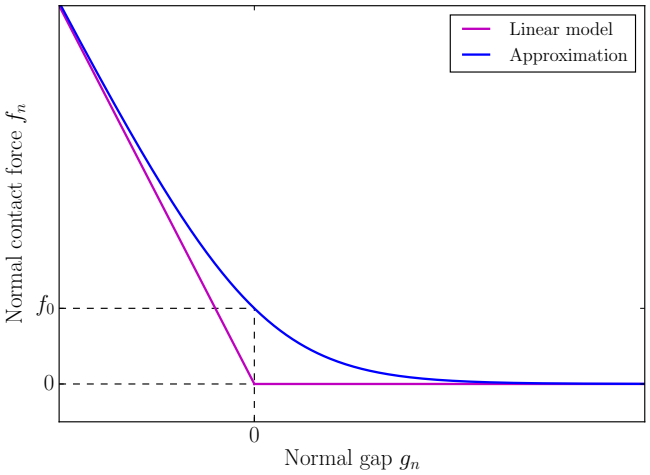

Figure 2: Normal contact force model.

is so small to be negligible: its value at $g_n = 10$ mm is less than 1 N ($f_n \approx 0.9$ N) and rapidly vanishes as the normal gap increases.

Also the tangential component of the contact model is approximated, according to a regularization strategy of the stick-slip behavior, as in [14]. For a 2D problem, the relation between the tangential component of the contact force f_t and the tangential velocity V_t of the contact point can be modeled as follows:

$$f_t(g_N, V_t) = \mu f_n(g_N)\gamma(V_t), \quad \text{with} \quad \gamma(V_t) = -\tanh\left(\frac{V_t}{\widehat{V}_t}\right) \tag{4}$$

Here, μ is the coefficient of friction, γ is a smooth function that approximates the Coulomb model (Fig. 3), and $\widehat{V}_t$ is a reference sliding velocity at which the tangential force f_t is 76% of its asymptotic value. This tangential force model requires tuning as well. We set $\widehat{V}_t = 5 \cdot 10^{-3}$ m/s,

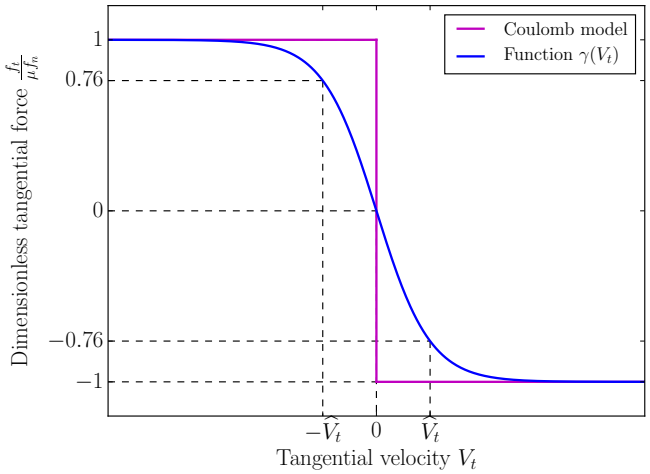

Figure 3: Coulomb friction model and its smooth approximation.

as the corresponding model appears to be a fair balance between physical plausibility and numerical tractability. The friction coefficient was fixed to $\mu = 1$.

3 FORMULATION OF THE OPTIMAL CONTROL PROBLEM

3.1 Discretization

The solution of the optimal control problem through a direct transcription method requires the parameterization of the continuous optimization problem into a finite dimensional one, through discretization of the time horizon $[0, T]$ into N time intervals $[t_k, t_{k+1}]$, so that $t_0 = 0$, $t_N = T$ and $k \in \{0, 1, \ldots, N-1\}$. It is worth emphasizing that problems involving intermittent contacts require very small time steps, because of the typical stiffness of the associated differential equations. For this reason, in our problem we chose to discretize the time horizon $T = 6$ s into $N = 200$ uniform steps, each having duration $h = T/N = 30$ ms.

Hence, discrete time versions of both the dynamics of the system and the contact force model are needed. We parameterize these equations by a collocation scheme based on a single collocation point, chosen in the middle of each time interval. Over each discretization interval, we assume the states $(q, \dot{q})$ to vary linearly, and the controls (which include the torques at the revolute joints u, together with the components of the contact force f_C) to be constant. According to this scheme, *kinematic reconstruction* and *dynamic equations* of the system can be written as follows:

$$q_{k+1} - q_k - h\bar{\dot{q}}_k = 0 \tag{5}$$

$$\bar{B}_k(\dot{q}_{k+1} - \dot{q}_k) + h(\bar{C}_k\bar{\dot{q}}_k + \bar{G}_k - \bar{Q}_k) = 0 \tag{6}$$

where, for simplicity of notation, we denoted $q_k = q(t_k)$, $\dot{q}_k = \dot{q}(t_k)$, $\bar{q}_k = (q_{k+1} + q_k)/2$ and $\bar{\dot{q}}_k = (\dot{q}_{k+1} + \dot{q}_k)/2$. Moreover, $\bar{B}_k = B(\bar{q}_k)$, $\bar{C}_k = C(\bar{q}_k, \bar{\dot{q}}_k)$, $\bar{G}_k = G(\bar{q}_k)$, and $\bar{Q}_k = Q(\bar{q}_k, \bar{\dot{q}}_k, u_k)$.

3.2 Optimization

Let us define a vector of decision variables $w \in \mathbb{R}^\nu$, which collects the sequence of unknowns $(q_0, \dot{q}_0, u_0, f_{C_0}, \ldots, q_k, \dot{q}_k, u_k, f_{C_k}, \ldots, q_N, \dot{q}_N)$. Since we have $2n(N+1)$ discretized state variables, mN discretized control actions, and N variables for each component of the contact force, the total number of optimization variables is:

$$\nu = 2n(N+1) + (m+2)N \tag{7}$$

The nonlinear program (NLP) we need to solve in order to obtain the optimal trajectory — and the corresponding sequence of control actions to be taken — can be formalized as follows:

$$\begin{aligned} \min_{w} \quad & f(w) \\ \text{subject to} \quad & g_{\min} \leq g(w) \leq g_{\max} \\ & w_{\min} \leq w \leq w_{\max} \end{aligned} \tag{8}$$

$g(w)$ is a set of nonlinear constraints, including Eqs. (5)–(6), together with the discretized versions of the normal and tangential contact force models (Eqs. (3)–(4)). In addition, by properly enforcing $w_{\min}$ and $w_{\max}$, we can assign the initial and final configurations and impose joint and torque limits (torques are constrained in the range $[-50, 50]$ Nm).
$f : \mathbb{R}^\nu \to \mathbb{R}$ is the objective function to be minimized. In our case, it is made up of five different contributions, aimed at minimizing, respectively: input torques, accelerations, dissipation due to sliding, input torque variations and contact force variations. The objective function was

calibrated by properly scaling and weighting the different contributions, so as to obtain natural and smooth behaviors.

Problem (8) results in a large-scale but very sparse NLP, which we solve through the interior-point solver IPOPT [15]. For the calculation of derivatives, we benefit from the Algorithmic Differentiation (AD) implemented in the CasADi framework [16].

4 COMPUTATIONAL STRATEGY: TWO-STAGE OPTIMIZATION

The computational strategy we propose in this paper aims to reduce the computational effort required for the solution of the trajectory optimization problem. The NLP we consider as the reference problem we want to solve is characterized by the following choice of the parameters:

$$\begin{aligned} \text{Discretization} \quad & N = 200 \\ \text{Normal contact} \quad & f_0 = 20 \text{ N} \\ & \kappa = 10^4 \text{ N/m} \\ \text{Tangential contact} \quad & \widehat{V}_t = 5 \cdot 10^{-3} \text{ m/s} \end{aligned} \tag{9}$$

The computational strategy we devised is to solve this problem through a *two-stage* optimization. The first stage consists in solving a preliminary optimization problem which roughly approximates the original one, but is simplified by slight changes in some parameters, so as to speed up the solution. In order to suitably choose the values to assign to these parameters, we investigated the beneficial effects of:

- reducing the number of time steps N. This simplifies the problem by decreasing its dimension, as clearly appears from (7). We will refer to the number of time steps used in the first optimization as N';
- relaxing the contact parameters, as this reduces the numerical stiffness of the problem. More in details, we smooth the tangential contact model by increasing the value of the reference tangential velocity $\widehat{V}_t$ (see Eq. (4)) in the first stage. We will indicate this value using the symbol $\widehat{V}_t'$. Regarding the normal contact model, we chose to relax it by reducing the value of the asymptotic stiffness κ to the value κ' in the preliminary optimization, while leaving the parameter f_0 unchanged.

Once this preliminary optimization is solved, we proceed to the second stage, using the just computed solution to warm-start the final optimization problem, whose parameters are summarized in (9). The results we present show how this two-stage strategy, with a proper choice of the parameters in the preliminary optimization, can lead to a significant reduction in the computational time required to obtain the solution.

5 RESULTS

In this section, we present the results of our analysis. All of them were obtained on a notebook computer with a 2.40 GHz Intel(R) Core(TM) i7-5500U processor and 8 GB of RAM. The performance is evaluated here in terms of CPU time to convergence.

5.1 Reference solution

The reference solution was obtained in a single-stage optimization. Namely, this solution was computed by solving the optimal control problem, characterized by the parameters detailed

in (9), without exploiting the advantage of solving a preliminary optimization with a coarser time grid, nor a relaxed contact model. The slope of both the track and the ground was set to $\alpha = \pi/30$.

The initial guess was selected so that all of the variables were zero except for the configuration variables q: the internal configuration angles were set to their initial state values (equal to those in the final state), whereas the prismatic coordinate q_1, representing the position of the PRR system along the track, was initialized using a cubic interpolation between the initial and final conditions. This choice was made with the precise intention of providing the solver with the minimum domain knowledge on how to accomplish the task, since we would like it to discover the optimal gait by itself.

The solution was obtained after 300 s CPU time (671 iterations), and it is depicted in Fig. 4. The corresponding values of some of the optimization variables over the time horizon are plotted in Fig. 5.

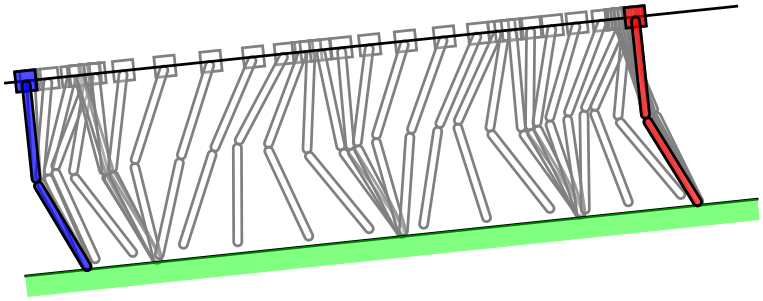

Figure 4: In gray, frames of the reference optimal trajectory. In blue and red, respectively, the initial and final configurations prescribed to the system.

5.2 Two-stage optimization strategy

Let us now compare the performance of the proposed two-stage optimization strategy against the reference single-stage solution. The strategy described in Sec. 4 consists in solving a preliminary approximated optimization, and using its solution to warm-start the full-fledged optimal control problem. The preliminary optimization differs from the reference problem by one or more parameters: in particular, we investigated the effect of enlarging the time steps and relaxing the parameters of the contact model. The first stage of optimization is initialized in the same naive way as illustrated in Sec. 5.1. The obtained solution is then interpolated on a finer grid and used to initialize the solver of the second optimization, whose parameters are the reference ones listed in (9).

The results are presented in tables, which are organized as follows:

- each table refers to a specific value of κ', that is the value adopted for the normal contact stiffness in the preliminary optimization: Table 2 reports the results obtained when the normal contact model is left unchanged through the two stages, whereas Tables 3 and 4 refer to results obtained by dividing the normal contact stiffness respectively by 2 and by 4 in the first stage;

- in each table, the results obtained for different levels of relaxation of the tangential contact force model in the first optimization are organized in columns (labelled by the corre-

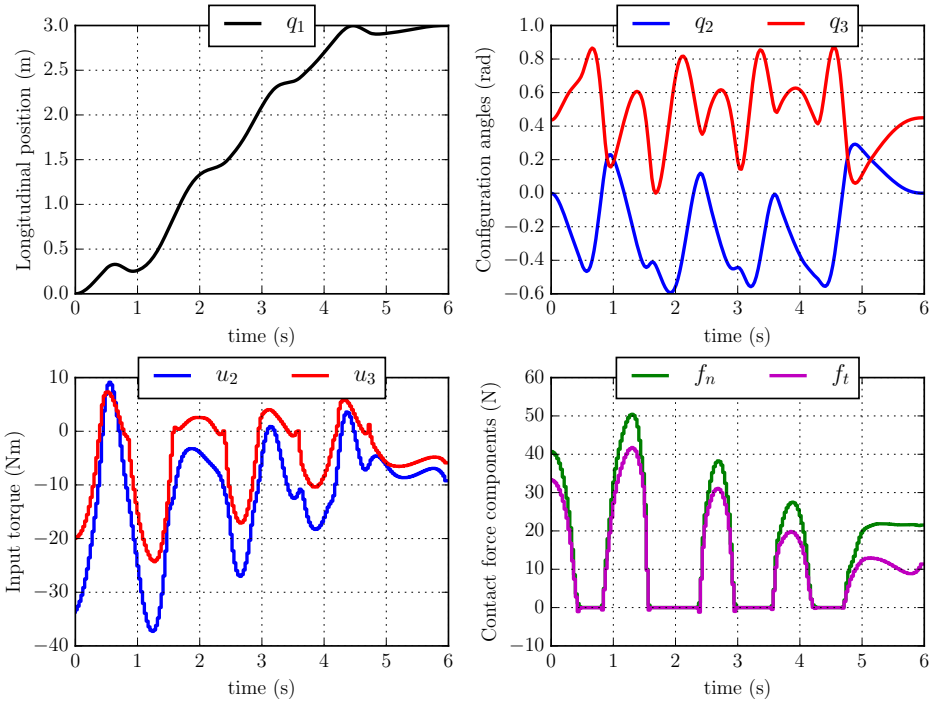

Figure 5: Reference solution: configuration variables, input torques and contact force as functions of time.

sponding value of $\widehat{V}_t'$);

- in each table, the results obtained for different numbers of time steps N' in the first optimization are organized in rows;
- in each cell of the table, the performance of the corresponding two-step optimization in terms of CPU time to convergence is reported (in seconds).
  Specifically, the first (second) term stands for the computational time required to solve the first (second) optimal control problem. The word FAILED indicates that the solver was not able to find any solution to the preliminary optimization within the maximum number of iterations imposed (5000).

All of these results refer to the case $\alpha = \pi/30$.

5.3 Remarks

Let us now focus on what we can deduce from our analysis. First of all, we notice that in most cases, compared to the reference solution (obtained after 300 s CPU time), the two-stage strategy leads to a significant reduction in the computational time required to obtain a sensibly similar solution: despite direct optimization methods suffer the risk of being trapped in local minima, all of the optimal solutions found are comparable in terms of value of the objective function. The mean computational time saving over all those cases in which the solver found a solution to the first stage is of 163.5 s (54.5% of the reference CPU time). In the most favourable case, which corresponds to adopting $N' = 100$, $\kappa' = \kappa/2$ and $\widehat{V}_t' = 2\widehat{V}_t$ in the preliminary phase, this strategy led to an 85.3% reduction in the CPU time required to compute a solution, as can be seen from Table 3.

	$\widehat{V}_t' = \widehat{V}_t$	$\widehat{V}_t' = 2\widehat{V}_t$	$\widehat{V}_t' = 4\widehat{V}_t$
$N' = 75$	$32 + 110 = 142$	$34 + 33 = 67$	$20 + 125 = 145$
$N' = 100$	$97 + 82 = 179$	$87 + 21 = 108$	$19 + 51 = 70$
$N' = 125$	$63 + 27 = 90$	$74 + 50 = 124$	$76 + 74 = 120$

Table 2: Results obtained for $\kappa' = \kappa$ (CPU time in seconds).

	$\widehat{V}_t' = \widehat{V}_t$	$\widehat{V}_t' = 2\widehat{V}_t$	$\widehat{V}_t' = 4\widehat{V}_t$
$N' = 75$	$37 + 372 = 409$	$19 + 57 = 76$	$16 + 51 = 67$
$N' = 100$	$230 + 60 = 290$	$28 + 16 = 44$	$15 + 68 = 83$
$N' = 125$	FAILED	$82 + 22 = 104$	$20 + 65 = 85$

Table 3: Results obtained for $\kappa' = \kappa/2$ (CPU time in seconds).

	$\widehat{V}_t' = \widehat{V}_t$	$\widehat{V}_t' = 2\widehat{V}_t$	$\widehat{V}_t' = 4\widehat{V}_t$
$N' = 75$	$55 + 41 = 96$	$73 + 28 = 101$	$11 + 71 = 82$
$N' = 100$	$194 + 35 = 229$	$35 + 120 = 155$	$46 + 60 = 106$
$N' = 125$	FAILED	$72 + 28 = 100$	$22 + 317 = 339$

Table 4: Results obtained for $\kappa' = \kappa/4$ (CPU time in seconds).

We could observe that in those cases in which the strategy was not effective — the solver was not able to find a solution to the preliminary optimization or the total CPU time exceeded 300 s — this was plausibly due to an undesirable mix of a smoothed normal contact ($\kappa' = \kappa/2$ or $\kappa' = \kappa/4$) with a stiff tangential contact ($\widehat{V}_t' = \widehat{V}_t$), see first column of Table 3 and Table 4. In fact, in these conditions, because of both the layout of the system and the fact that sliding behaviors are penalized in the objective function, the solver finds some difficulties in computing solutions which satisfy the constraints while minimizing the objective function. In fact, reducing the normal contact stiffness has the side effect of increasing the intensity of the normal force acting at a distance. Without relaxing the tangential contact model, even small values of the tangential velocity V_t at a distance from the surface of the ground result in a sensible contribution in the term of the objective function which penalizes the dissipation due to sliding. Even in the event that the solver succeeds, the solution could represent an unnatural trajectory (the foot has to move very far from the ground), constituting an inadequate initial guess for the second optimization, which may require a large amount of time to convergence (see for example the case $N' = 75, \kappa' = \kappa/2, \widehat{V}_t' = \widehat{V}_t$ in Table 3).

Concerning the number of steps N' to be used in the first stage, we notice that in those cases where the contact force model is accurate (first column in Table 2), the best choice is to use a high number of steps in the first optimization: since the first problem to be solved, although complex, is very similar to the actual one, its solution is a very good initial guess for the second stage, and the solver requires a short time to find a solution to the second optimization. On the other hand, the more the contact model is smoothed in the first stage, the more it is convenient to use fewer integration steps: since the first problem is a very simplified version of the original one, performing a consistent coarse discretization of it seems to be the best choice (last column in Table 4).

5.4 Robustness of the method

In order to test the validity and robustness of the proposed method, we picked the best performing mix of parameters from our analysis and employed it in the two-stage optimization strategy for a slightly different problem, where the system and the task to be accomplished (move 3 m forward along the track in $T = 6$ s, then stop) are the same, but the slope angle α is different. Without performing any analogous analysis for the modified problem, we exploit the results from the study conducted for the reference problem to tune up a two-stage optimization strategy and apply it in the new scenarios. In the first stage, we solved a preliminary optimization where the parameters were modified as follows:

$$\begin{aligned} \text{Discretization} \quad & N' = N/2 \\ \text{Normal contact} \quad & \kappa' = \kappa/2 \\ \text{Tangential contact} \quad & \widehat{V}_t' = 2\widehat{V}_t \end{aligned} \tag{10}$$

as this combination of parameters led to the best performance for the reference problem, and it seems to be a reasonable choice according to the outlined remarks. Of course, this may not be the best choice for the modified problem, yet our aim is here to prove that, from a computational perspective, solving the direct optimization problem in a two-stage fashion, with a reasonable slackening of the parameters in the first stage, is convenient compared to a single-stage optimization.

Also in these modified cases, the proposed strategy proved to be very efficient in improving the computational performance of the trajectory optimization, therefore assessing the robustness of the method. The results we present are arranged in tables, comparing the CPU time (in seconds) required to obtain the optimal solution for the single and two-stage optimization. Table 5 refers to a problem where the slope angle is half the reference value ($\alpha = \pi/60$), whereas Table 6 reports the results obtained in a problem where the slope angle is doubled ($\alpha = \pi/15$).

Reference solution (single-stage optimization)	Two-stage optimization $N' = N/2$, $\kappa' = \kappa/2$, $\widehat{V}_t' = 2\widehat{V}_t$
403	$31 + 60 = 91$

Table 5: Results obtained for $\alpha = \pi/60$ (CPU time in seconds).

Reference solution (single-stage optimization)	Two-stage optimization $N' = N/2$, $\kappa' = \kappa/2$, $\widehat{V}_t' = 2\widehat{V}_t$
494	$32 + 43 = 75$

Table 6: Results obtained for $\alpha = \pi/15$ (CPU time in seconds).

6 CONCLUSIONS

In this paper, we have devised and numerically assessed a computational strategy to improve the performance of direct trajectory optimization of systems interacting with the surrounding environment through intermittent contacts. This computational strategy consists in subsequently solving two optimal control problems of increasing complexity. A penalty-based

formulation for modeling the contact forces is employed throughout the paper. A straightforward method for simplifying the problem in the preliminary optimization was proposed and several tests were performed on a 2D underactuated multibody system. The results we presented prove the effectiveness and the robustness of the proposed strategy.

Our current research effort is focused on the application of similar strategies for Model Predictive Control of underactuated multibody systems with contacts, where computational efficiency is essential in order to cope with the strict requirements of replanning the trajectory online.

REFERENCES

[1] S. M. LaValle. *Planning algorithms*. Cambridge University Press, 2006.

[2] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research (IJRR)*, 30(7):846–894, 2011.

[3] K. Hauser and V. Ng-Thow-Hing. Randomized multi-modal motion planning for a humanoid robot manipulation task. *The International Journal of Robotics Research (IJRR)*, 30(6):678–698, 2011.

[4] S. Tonneau, N. Mansard, C. Park, D. Manocha, F. Multon, and J. Pettré. A reachability-based planner for sequences of acyclic contacts in cluttered environments. In *International Symposium on Robotics Research (ISRR 2015)*, 2015.

[5] J. T. Betts. *Practical methods for optimal control and estimation using nonlinear programming*. SIAM, 2nd edition, 2010.

[6] I. Mordatch, E. Todorov, and Z. Popović. Discovery of complex behaviors through contact-invariant optimization. *ACM Transactions on Graphics (TOG)*, 31(4):43–50, 2012.

[7] M. Posa, C. Cantu, and R. Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research (IJRR)*, 33(1):69–81, 2014.

[8] M. Gabiccini, A. Artoni, G. Pannocchia, and J. Gillis. A computational framework for environment-aware robotic manipulation planning. In *International Symposium on Robotics Research (ISRR 2015)*, 2015.

[9] C. Mummolo, L. Mangialardi, and J.H. Kim. Concurrent contact planning and trajectory optimization in one step walking motion. In *Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE 2015)*, 2015.

[10] H. Dai, A. Valenzuela, and R. Tedrake. Whole-body motion planning with centroidal dynamics and full kinematics. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2014.

[11] I. Mordatch, J. M. Wang, E. Todorov, and V. Koltun. Animating human lower limbs using contact-invariant optimization. *ACM Transactions on Graphics (TOG)*, 32(6):203–210, 2013.

[12] W. Xi and C. D. Remy. Optimal gaits and motions for legged robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, 2014.

[13] D. Pardo, L. Möller, M. Neunert, A. Winkler, and J. Buchli. Evaluating direct transcription and nonlinear optimization methods for robot motion planning. *arXiv:1504.05803v1 [math.OC]*, 2015.

[14] P. Wriggers. *Computational Contact Mechanics*. Springer, New York, 1999.

[15] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.

[16] J. Andersson. *A General-Purpose Software Framework for Dynamic Optimization*. PhD thesis, Arenberg Doctoral School, KU Leuven, Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium, October 2013.