Convex Optimization of the Full Centroidal Dynamics for Planning in Multi-Contact Scenarios

Gabriel Garcia, Robert Griffin, Jerry Pratt

***Abstract* — In this paper, we present an iterative QP optimization problem to generate the centroidal trajectories of a robot for highly dynamic behaviors using Multi-contact locomotion with minimization of the angular momentum around the center of mass. We are considering both, point and planar contacts with friction restriction and we model their forces using the friction cone and the Contact Wrench Cone respectively. The present QP configuration produces a feasible trajectory in any iteration, because we use an exact linearization of the Centroidal Dynamics, not approximation. We move the non-convexity of the angular momentum dynamics to the cost function, and we minimize a linearization of it. More iterations produce a better approximation of the minimal angular momentum. When contact placements and time between them are pre-specified, we extend the result to n-step capturability and planning. Thus, we are able to generate extreme motions that the world allows, such as parkour wall-runs for a biped robot, or near-vertical dam walking for a quadruped robot.**

I. Introduction

There is an extensive amount of literature involving methods for capturability and planning using reduced models with simplifications like zero angular momentum or no height variation. These approaches include the Linear Inverted Pendulum [1,2], the Divergent Component of Motion [3,4], or the Variable-Height Inverted Pendulum [5-8]. Angular momentum also has received attention in [2,9,10]. But almost all of these approaches consider only a single contact. There are some considerations about the double support phase as a single contact using the convex hull of the feet as the support region but this is not applicable when contacts are non-coplanar. Nevertheless, this approaches had been used in hardware applications.

These works consider transformations of the states of the robot to measure stability in walking. One of the most known approach is the Zero Moment Point (ZMP) criterion, but it lacks of generality in multi-contact scenarios. In [11] ZMP criterion was generalized to the *Contact Wrench Cone* (CWC) criterion, which corresponds to a projection of the possible wrenches that can be applied at every contact. If the centroidal dynamics belong to that CWC, then the motion is dynamically feasible and the robot is not breaking its contacts. In [12], authors compute a closed form formula for the wrenches applicable in a rectangle foot around its geometrical center under a linearized friction cone restriction. We will be taking this model, because it allows to represent the wrenches at the contacts as a linear restriction.

Approaches like [13-15] treat the problem of Multi-contact capturability in 0-step scenarios (without switching contacts). [13] solves the problem using the CWC, but considering a linear model. [14-15] also treat this problem, but authors assume that the best way to stop a robot is to minimize instantaneously its kinetic energy.

* This work was supported by IHMC.
The authors are with the Florida Institute for Human and Machine Cognition, 40 S Alcaniz St, Pensacola, FL 32502, United States
E-mail: {ggarcia,rgriffin,jpratt}@ihmc.us

In [16], a convex program is proposed for finding trajectories of the CoM. Authors force the CoM to follow a polynomial trajectory of a specific degree. They claim that it can solve the problem of 0-step and 1-step capturability, which is true when the robot CoM is able to follow the given polynomial trajectory. When the polynomial degree is increased, the non-linearities reappear.

Minimization of the angular momentum is one of the main features of human walking [17]. Because of this, the minimization angular momentum can produce human-like gaits, although not every motion must have a minimal angular momentum. In approaches like [18], authors define QCQPs (Quadratically Constrained Quadratic Program) for finding solutions to the centroidal dynamics based in its decomposition into a convex and a concave part, but they do not aim to minimize the angular momentum around the CoM and the complexity of the problem can be reduced even more as it is shown in [19].

In [19], authors define a QP optimization for minimizing an upper bound of the angular momentum around the CoM and for maximizing the CWC margin, which can be interpreted as the maximum disturbance at the CoM that the current contacts can resists. Their approach is dependent on some bounds chosen for the CoM, and the quality of the minimization is directly influenced by these bounds. Nevertheless, [19] represents one of the closest approaches that solves for the centroidal trajectories of a robot with minimum angular momentum, and it is able to produce extreme dynamic behaviors using a QP program.

Approaches like [20] consider trajectories kinematically generated and dynamically feasible. Although they consider the angular momentum generation, they don't minimize its value around the CoM. [21] also considers both, full kinematics and centroidal dynamics in a highly nonlinear problem. Although it can explicitly minimize the angular momentum, according to the authors the time that the solving takes goes from the range from minutes to hours.

The main contribution of this paper is presented as follows:

- Provide an iterative QP program to find a trajectory that locally minimizes the Angular Momentum around the Center of Mass in Multi-Contact Scenarios for planning with n-step when the time between switches in contacts is fixed and the contact placements are given.

In that sense, we are able to perform extreme dynamic behaviors. One of the consequences of this contribution is when the planning is done using only the current contacts and the robot must be in rest in the end of the trajectory, i.e. 0-step capturability with fixed final time. We provide a LP program the check feasibility on the 0-step capturability in Multi-Contact scenarios. We also provide an iterative QP program to find a trajectory that minimizes the Angular Momentum around the Center of Mass in Multi-Contact scenarios for 0-step capturability.

The structure of the paper is organized as follows: In Section II, we will present the background and the Dynamical system we are working on. We will show the contributions in Section III, defining the QP setup and presenting some simulated results. In Section IV we will discuss the results and we will talk about some open problems. In Section V we will conclude the present paper.

II. Full Centroidal Dynamics

A. Dynamical model / Background

Let's define first the Wrenches $\boldsymbol{\zeta_i}$ for each contact i as the stacked vector of force $\boldsymbol{f_i}$ and torque $\boldsymbol{\tau_i}$ around some point in each contact surface $\boldsymbol{CS}$:

$$\boldsymbol{\zeta_i} = \begin{bmatrix} \boldsymbol{f_i} \\ \boldsymbol{\tau_i} \end{bmatrix} \tag{1}$$

For a polygonal surface, there is always a closed form of the wrenches applicable around its origin. This can be represented as a *Contact Wrench Restriction* in the Wrench $\boldsymbol{\zeta_i}$, defined as:

$$\boldsymbol{W_{CWCi}\zeta_i} \leq \boldsymbol{b_{CWCi}} \tag{2}$$

Where $\boldsymbol{W_{CWCi}}, \boldsymbol{b_{CWCi}}$ corresponds to the Contact Wrench Cone matrix and vector respectively for each contact i between 1 and n. An example of these restriction can be found in equations (15-20) from [12], valid for a rectangular contact surface.

We have the following dynamical system with n contact surfaces, known as the *Centroidal Dynamics*:

$$\begin{bmatrix} \dot{\boldsymbol{L}} = \sum_i \left(\boldsymbol{\tau_i} + \left(\boldsymbol{r_{p0i}} - \boldsymbol{r}\right) \times \boldsymbol{f_i}\right) \\ \ddot{\boldsymbol{r}} = \frac{1}{m}\sum_i \boldsymbol{f_i} + \boldsymbol{g} \end{bmatrix}, \boldsymbol{W_{CWCi}\zeta_i} \leq \boldsymbol{b_{CWCi}} \tag{3}$$

Where:

$\boldsymbol{L}$: Angular momentum around the CoM

$\boldsymbol{r}, \dot{\boldsymbol{r}}$: Position and velocity of the CoM

$\boldsymbol{r_{p0i}}$: Center of each Contact Surface i, or more generally, point around each Contact Wrench Restriction is computed.

m: Mass of the robot

$\boldsymbol{g}$: Gravity vector.

$\boldsymbol{f_i}, \boldsymbol{\tau_i}$: Components of the Wrench $\boldsymbol{\zeta_i}$ applied at each contact.

We can stack the wrenches $\boldsymbol{\zeta_i}$ into a vector $\boldsymbol{\zeta_v}$, the vectors b_{CWCi} into a vector b_{bCWC} and stack in diagonal form the matrices W_{CWCi} into a W_{bCWC}. The vector $\boldsymbol{\zeta_v}$ and the condition of the Contact Wrench Restriction are written as:

$$\boldsymbol{\zeta_v} = \begin{bmatrix} \boldsymbol{\zeta_1} \\ \vdots \\ \boldsymbol{\zeta_n} \end{bmatrix} \tag{4}$$

$$\boldsymbol{W_{bCWC}\zeta_v} \leq \boldsymbol{b_{bCWC}} \tag{5}$$

We can define the following variables for a fixed $\boldsymbol{r_{p0}}$:

$$\boldsymbol{f} = \sum_i \boldsymbol{f_i} \tag{6}$$

$$\boldsymbol{\tau} = \sum_i \left(\boldsymbol{\tau_i} + \left(\boldsymbol{r_{p0i}} - \boldsymbol{r_{p0}}\right) \times \boldsymbol{f_i}\right) \tag{7}$$

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{\tau} \end{bmatrix} \tag{8}$$

The new dynamics will hold:

$$\begin{bmatrix} \dot{\boldsymbol{L}} = \boldsymbol{\tau} + \left(\boldsymbol{r_{p0}} - \boldsymbol{r}\right) \times \boldsymbol{f} \\ \ddot{\boldsymbol{r}} = \frac{1}{m}\boldsymbol{f} + \boldsymbol{g} \end{bmatrix}, \boldsymbol{W_{CWC}\zeta} \leq \boldsymbol{b_{CWC}} \tag{9}$$

Where $\boldsymbol{W_{CWC}}$ and $\boldsymbol{b_{CWC}}$ corresponds to the *Contact Wrench Cone* computed around $\boldsymbol{r_{p0}}$, where the centroidal dynamics must belong for stability, approach used in [19]. Note that $\boldsymbol{\zeta} = \boldsymbol{P\zeta_v}$ for some matrix $\boldsymbol{P}$ and $\boldsymbol{W_{CWC}}, \boldsymbol{b_{CWC}}$ corresponds to the polytope generated by projecting the polytope from (5) using the projection $\boldsymbol{P}$. This can be solved using the *double description method* [19,22].

The dynamics (9) for shows similarity of this approach with [19]. But for the optimization itself we are going to use (3), because the matrix $\boldsymbol{W_{CWC}}$ can be larger than $\boldsymbol{W_{bCWC}}$, especially when the contacts does not have a common normal direction. In any case, this approach is applicable to both systems.

Systems (3) and (9) are equivalent and they represent a robot model considering the following assumptions:

- Multicontact model using polygonal or point contacts.
- Coulomb friction for the definition of the Contact Wrench Restriction in each planar contacts.
- Variable CoP on the polygonal contact surface.
- Variable Height and Angular Momentum.

III. Control

In this subsection we present first the possible stabilization region for the 0-step capture problem, we will present the solution of this using a LP program only. Then we will be defining an iterative QP for minimizing the angular momentum around the CoM, and finally we will extend the result to n-step capturability, or equivalently, centroidal planning.

A. 0-Step capturability as a convex problem

Equation (3) defines the dynamics of the system and the restriction over the wrenches $\boldsymbol{\zeta_v}$ ensures that the contacts will not break. All terms are linear (note also that r_{p0i} is constant) except by the terms $\boldsymbol{r} \times \boldsymbol{f_i}$. We can remove this nonlinearity by using a transformation of the Angular momentum $\boldsymbol{L}$. Instead of computing $\boldsymbol{L}$ around the center of mass, we will compute it around a fixed point on the space. This approach had been used in [19], where the angular momentum is considered around the origin of the world frame of coordinates, and they minimize a bound on the angular momentum around the CoM.

We will call $\boldsymbol{L_2}$ to the angular momentum around the origin of coordinates. The definition of $\boldsymbol{L_2}$ using the state variables from (3) is:

$$\boldsymbol{L_2} = \boldsymbol{L} + m\boldsymbol{r} \times \dot{\boldsymbol{r}} \tag{10}$$

Let's note that the derivative of $\boldsymbol{L_2}$ holds:

$$\dot{\boldsymbol{L}}_{\boldsymbol{2}} = \dot{\boldsymbol{L}} + m\dot{\boldsymbol{r}} \times \dot{\boldsymbol{r}} + m\boldsymbol{r} \times \ddot{\boldsymbol{r}} \tag{11}$$

$$\dot{\boldsymbol{L}}_{\boldsymbol{2}} = \dot{\boldsymbol{L}} + m\boldsymbol{r} \times \ddot{\boldsymbol{r}} \tag{12}$$

Which means that the term $\boldsymbol{r} \times \ddot{\boldsymbol{r}}$ present in the centroidal dynamics actually have a close form integral:

$$\frac{d}{dt}\boldsymbol{r} \times \dot{\boldsymbol{r}} = \boldsymbol{r} \times \ddot{\boldsymbol{r}} \tag{13}$$

Taking component the second component of (9):

$$\ddot{\boldsymbol{r}} = \frac{1}{m}\boldsymbol{f} + \boldsymbol{g} \tag{14}$$

and inserting (14) in (12) we obtain new dynamics of the system:

$$\begin{bmatrix} \dot{\boldsymbol{L}}_2 = \sum_i(\boldsymbol{\tau}_i + \boldsymbol{r}_{p0i} \times \boldsymbol{f}_i) + m\boldsymbol{r} \times \boldsymbol{g} \\ \ddot{\boldsymbol{r}} = \frac{1}{m}\sum_i \boldsymbol{f}_i + \boldsymbol{g} \end{bmatrix} \tag{15a}$$

$$\boldsymbol{W}_{bcwc}\boldsymbol{\zeta}_v \leq \boldsymbol{b}_{bcwc} \tag{15b}$$

We can see that the new dynamics of the system are affine (linear but a constant term, $\boldsymbol{g}$). Also, the inputs $\boldsymbol{\zeta}_v$ have a linear restriction under linearization of the friction cones. So we can know if a given state is capturable or not in a given time by solving an LP program. We can introduce the new vector of states $\boldsymbol{x}$ as:

$$\boldsymbol{x} = [\boldsymbol{L}_2, \boldsymbol{r}, \dot{\boldsymbol{r}}]^T \tag{16}$$

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{\zeta}_v + \boldsymbol{c}, \boldsymbol{W}_{bCWC}\boldsymbol{\zeta}_v \leq \boldsymbol{b}_{bCWC} \tag{17}$$

The problem of 0-step capturability can be written as: Given an initial state $\boldsymbol{x}_{init}$, find feasible trajectories $\boldsymbol{x}(t)$ and $\boldsymbol{\zeta}_v(t)$ without breaking any contact such that $\boldsymbol{x}(t_f) = [0, \boldsymbol{r}_f, 0]^T$ for some final time t_f and for some final position inside the SSR $\boldsymbol{r}_f$.

We want the robot to be in rest, so $\dot{\boldsymbol{r}}_f = \boldsymbol{0}$. We want also to keep the robot permanently in rest, so we must be able to exert forces to keep it in rest. This is possible if and only if the CoM is inside the SSR. We also don't want the robot to be spinning, so $\boldsymbol{L}_f = \boldsymbol{0}$ around the CoM. This is translated to the Angular Momentum around the origin of coordinates as:

$$\boldsymbol{L}_{2f} = \boldsymbol{L}_f + m\boldsymbol{r}_f \times \dot{\boldsymbol{r}}_f = \boldsymbol{0} \tag{18}$$

In this approach, as we said before, we will fix the time t_f to a constant value, and also the final position $\boldsymbol{r}_f$ to anywhere in the SSR. Given this definition of 0-step capturability and the linearity of the dynamics and the restrictions in (3), the application of a QP method in the discretized system is straightforward when using any trajectory optimization method. Particularly, we will be using Direct Transcription.

First of all, we will discretize the time t in $p+1$ knot points $t_j, \forall j = \{0, \dots, p\}$, including the initial and final state. We will define a vector $\boldsymbol{X}$ containing the knot points of $\boldsymbol{x}(t)$ and $\boldsymbol{\zeta}_v(t)$ at every t_j as:

$$\boldsymbol{X} = [\boldsymbol{x}_0, \dots, \boldsymbol{x}_p, \boldsymbol{\zeta}_{v0}, \dots, \boldsymbol{\zeta}_{vp-1}]^T \tag{19}$$

Second, we will define the equality constraints of the QP program. consider the dynamics of the system as a restriction. We can use any approximation of the derivative of $\boldsymbol{x}$. For this application we will just consider the following discretization:

$$\dot{\boldsymbol{x}}_j = \frac{\boldsymbol{x}_{j+1} - \boldsymbol{x}_j}{T}, \forall j \in \{0, \dots, p-1\} \tag{20}$$

Where T corresponds to the time difference between t_{j+1} and t_j. Note that this difference not necessarily should be constant for each j. Inserting (20) on (17), we have:

$$\boldsymbol{x}_{j+1} - \boldsymbol{x}_j - T(\boldsymbol{A}\boldsymbol{x}_j + \boldsymbol{B}\boldsymbol{\zeta}_{vj}) = T\boldsymbol{c} \tag{21}$$

The initial and final restriction can be written as:

$$\boldsymbol{x}_0 = \boldsymbol{x}_{init} \tag{22}$$

$$\boldsymbol{x}_p = \boldsymbol{x}_f = [\boldsymbol{0}, \boldsymbol{r}_f, \boldsymbol{0}]^T \tag{23}$$

Restrictions (21), (22) and (23) are linear in the variables $\boldsymbol{x}_{j+1}$, $\boldsymbol{x}_j$, $\boldsymbol{\zeta}_{vj}$, $\boldsymbol{x}_0$ and $\boldsymbol{x}_p$, and all of them are components of the vector $\boldsymbol{X}$. In consequence (21), (22) and (23) can be represented using a single matrix $\boldsymbol{M}_{eq}$ and a single vector $\boldsymbol{N}_{eq}$ holding:

$$\boldsymbol{M}_{eq}\boldsymbol{X} = \boldsymbol{N}_{eq} \tag{24}$$

Third, we will define the constraints of the dynamics. Applying (5) to every wrench in time, the contact wrench restriction can be written as:

$$\boldsymbol{W}_{bCWC}\boldsymbol{\zeta}_{vj} \leq \boldsymbol{b}_{bCWC}, \forall j \in \{0, \dots, p-1\} \tag{25}$$

Kinematic limits can be partially addressed by considering restrictions on the position of the CoM $\boldsymbol{r}$, among as limitation in the world as floor, walls or obstacles. Also, we can consider bounds on its velocity $\dot{\boldsymbol{r}}$. We can use any linear approximation on these bounds as:

$$\boldsymbol{M}_{\boldsymbol{R}_j}\boldsymbol{x}_j \leq \boldsymbol{N}_{\boldsymbol{R}_j}, \forall j \in \{0, \dots, p\} \tag{26}$$

Again, restrictions (25) and (26) are linear in the variables $\boldsymbol{x}_j$ and $\boldsymbol{\zeta}_{vj}$, which are components of the vector $\boldsymbol{X}$, so (25) and (26) can be represented using a matrix $\boldsymbol{M}_{ineq}$ and a vector $\boldsymbol{N}_{ineq}$ holding:

$$\boldsymbol{M}_{ineq}\boldsymbol{X} \leq \boldsymbol{N}_{ineq} \tag{27}$$

Finally, we can define now the following optimization setup $\mathbb{P}_1$ with optimal value p_1^*:

$$\mathbb{P}_1: p_1^* = \min_{\boldsymbol{X}} 0, \text{ s.t} \tag{28}$$

$$\boldsymbol{M}_{eq}\boldsymbol{X} = \boldsymbol{N}_{eq}$$

$$\boldsymbol{M}_{ineq}\boldsymbol{X} \leq \boldsymbol{N}_{ineq}$$

$\mathbb{P}_1$ just checks feasibility of the conditions of the dynamics, the initial and final state, the wrenches belonging to their CWC, and kinematical restrictions. $\mathbb{P}_1$ can be solved as a LP program. If the program is feasible, the resulting solution $\boldsymbol{X}_{sol}$ contains the discretized trajectories of $\boldsymbol{x}_{sol}(t)$ and $\boldsymbol{\zeta}(t)$ that works as the solution of the 0-step capturability problem. By interpolating the components of $\boldsymbol{X}_{sol}$, we can obtain the time trajectories of the CoM, its velocity, its angular momentum and the wrenches applied to the world. This is one way to obtain a centroidal trajectory for tracking.

The final position $\boldsymbol{r}_f$ restriction (23) can be moved from the equality restrictions to the inequality side if we have modeled an inner approximation of the SSR in the contacts. But let's analize the final state of the system. If we desire more freedom on the final position and potentially more possibilities for the feasibility of $\mathbb{P}_1$, then we can change restriction (23) to:

$$-T(\boldsymbol{A}\boldsymbol{x}_p + \boldsymbol{B}\boldsymbol{\zeta}_{vp}) = T\boldsymbol{c} \tag{29}$$

With this, we are forcing the system to find a fixed point on the dynamics (17). A fixed point in this case corresponds to a point inside of the SSR. We are directly finding a final point inside instead of precomputing its possible values, approach done previously in works like [23]. Restriction (29) does not enforce the final angular momentum value, but from (18) we will enforce again:

$$\boldsymbol{L_{2f}} = \boldsymbol{0}$$

Note that we cannot add directly bounds to the angular momentum around the CoM $\boldsymbol{L}$, because it's definition involves a cross product which will produce a bilinear restriction. However, we can add bounds to any linear combination of the position $\boldsymbol{r}$, velocity $\dot{\boldsymbol{r}}$, and angular momentum around the world origin of coordinates $\boldsymbol{L_2}$.

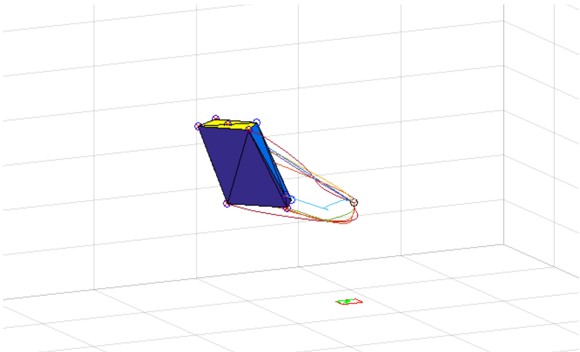

Figure 1: 0-step capturable CoMs for a fixed foot placement

C. *Minimization of the Angular Momentum in 0-step Capturability,*

In [19], authors minimize an upper bound on $\boldsymbol{L}$ by forcing the CoM to stay in a certain polytope or an elliptic region. The term $\boldsymbol{r} \times \dot{\boldsymbol{r}}$ is linearized by minimizing the maximum value of $\boldsymbol{L}$ in that region, which is placed in one of the edges of the polytope CoM region. The main issue with this approach is that it is highly dependent on the region chosen because the quality of the results is directly influenced by it. If we don't chose the region, then this approach will not work.

Other works like [16] approximate the CoM trajectory as a Bezier curve. They break the nonlinearity of $\boldsymbol{r} \times \dot{\boldsymbol{r}}$ by defining the degree of the Bezier polynomial to 6, matching some constant to some restrictions, and . The problem with this (and with any) polynomial approach is that it lacks of generality. When the degree of the polynomial is increased, the nonlinearity will persist and it cannot completely solve the 0-step and 1-step capturability.

We are going to provide an algorithm that minimizes a close approximation of the angular momentum around the CoM by performing an iterative QP program to find trajectories for $\boldsymbol{L}$, $\boldsymbol{r}$, and $\dot{\boldsymbol{r}}$, and the wrenches applied by the contacts $\boldsymbol{\zeta_v}$.

We want to minimize the value of $\int_0^{t_f} \|\boldsymbol{L}(t)\|^2_{\boldsymbol{Q_L}} dt$, which is the accumulate norm of the angular momentum around the CoM using a positive definite matrix $\boldsymbol{Q_L}$. In terms of the linear system (15), we want to minimize:

$$\min_{\zeta} \int_0^{t_f} \|\boldsymbol{L_2} - m\boldsymbol{r} \times \dot{\boldsymbol{r}}\|^2_{\boldsymbol{Q_L}} dt, \text{ s.t} \tag{30}$$

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{\zeta} + \boldsymbol{c}$$

$$\boldsymbol{W_{bcwc}}\boldsymbol{\zeta_v} \le \boldsymbol{b_{bcwc}}$$

$$\boldsymbol{x}(0) = \boldsymbol{x_0}$$

$$\boldsymbol{x}(t_f) = \boldsymbol{x_f}$$

This would be a QP program, but the term $\boldsymbol{r} \times \dot{\boldsymbol{r}}$ produces a quartic term on the decision variables because of the squared term. Even if we want to minimize the L_1 norm of the angular momentum, then we will have absolutes values in the objective function, and the problem cannot be handled using a QP configuration yet.

In order to solve the minimum accumulated angular momentum problem, one basic idea is to minimize the angular momentum around an *estimated position* of the CoM. For example, the middle point between its initial and its final position. That works as a good approximation and can be solved with a QP program, but let's go further. We know the initial and final positions and velocities. So we can *estimate* the position of the CoM as a polynomial of degree 3 $\boldsymbol{R_0}(t)$:

$$\boldsymbol{R_0}(t) = \boldsymbol{a_1} + \boldsymbol{a_2}t + \boldsymbol{a_3}t^2 + \boldsymbol{a_4}t^3 \tag{31}$$

Where the coefficients $\boldsymbol{a_i} \in \mathbb{R}^3$ produces a $\boldsymbol{R_0}(t)$ holding:

$$\boldsymbol{R_0}(0) = \boldsymbol{r_0}, \dot{\boldsymbol{R}}_{\boldsymbol{0}}(0) = \boldsymbol{r_0}$$

$$\boldsymbol{R_0}(t_f) = \boldsymbol{r_f}, \dot{\boldsymbol{R}}_{\boldsymbol{0}}(t_f) = \dot{\boldsymbol{r}}_{\boldsymbol{f}}$$

The solution of:

$$\min_{\zeta} \int_0^{t_f} \|\boldsymbol{L_2} - m\boldsymbol{R_0}(t) \times \dot{\boldsymbol{r}}\|^2_{\boldsymbol{Q_L}} dt, \text{ s.t} \tag{32}$$

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{\zeta} + \boldsymbol{c}$$

$$\boldsymbol{W_{bcwc}}\boldsymbol{\zeta} \le \boldsymbol{b_{bcwc}}$$

$$\boldsymbol{x}(0) = \boldsymbol{x_0}$$

$$\boldsymbol{x}(t_f) = \boldsymbol{x_f}$$

corresponds to the minimization of the angular momentum around $\boldsymbol{R_0}(t)$. We can define a matrix $\boldsymbol{V}(t)$ holding:

$$\boldsymbol{V_0}(t) = [\boldsymbol{I}_{3\times3}, \boldsymbol{0}_{3\times3}, -m[\boldsymbol{R_0}(t)]_\times]^T \tag{33}$$

The term $\int_0^{t_f} \|\boldsymbol{L_2} - m\boldsymbol{R_0}(t) \times \dot{\boldsymbol{r}}\|^2_{\boldsymbol{Q_L}} dt$ can be written as:

$$\int_0^{t_f} \|\boldsymbol{V_0}(t)^T\boldsymbol{x}\|^2_{\boldsymbol{Q_L}} dt = \int_0^{t_f} \boldsymbol{x}^T\boldsymbol{V_0}(t)\boldsymbol{Q_L}\boldsymbol{V_0}(t)^T\boldsymbol{x} dt \tag{34}$$

Note that the discrete case of this optimization is solvable by a QP program using again direct transcription. The constraints remain the same as (24) and (27). The objective function changes from 0 to:

$$\mathbb{P}_2 : p_2^* = \min_{\boldsymbol{X}} \sum_{j=0}^{p} \left\|\boldsymbol{L_{2j}} - m\boldsymbol{R_0}(t_j) \times \dot{\boldsymbol{r}}_j\right\|^2_{\boldsymbol{Q_L}}, \text{ s.t.}$$

$$\boldsymbol{M_{eq}}\boldsymbol{X} = \boldsymbol{N_{eq}}$$

$$\boldsymbol{M_{ineq}}\boldsymbol{X} \le \boldsymbol{N_{ineq}}$$

Which can be written as:

$$\mathbb{P}_2 : p_2^* = \min_{\boldsymbol{X}} \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{Q_{diag0}}\boldsymbol{X}, \text{ s.t.} \tag{35}$$

$$\boldsymbol{M_{eq}}\boldsymbol{X} = \boldsymbol{N_{eq}}$$

$$\boldsymbol{M_{ineq}}\boldsymbol{X} \le \boldsymbol{N_{ineq}}$$

Where $\boldsymbol{Q_{diag0}}$ is the stacked diagonal matrix filled with $\boldsymbol{V_0}(t_j)\boldsymbol{Q_L}\boldsymbol{V_0}(t_j)^T$ for each j in the knot points, and with zeros to complete the size of $\boldsymbol{X}$ (these zeros are the multipliers of the wrenches $\boldsymbol{\zeta_v}$ which are not involved in the objective function).

The matrices $\boldsymbol{M_{eq}}$ and $\boldsymbol{N_{eq}}$ are created by stacking the discretization of dynamics of the system, the initial and the final condition. The matrices $\boldsymbol{M_{ineq}}$ and $\boldsymbol{N_{ineq}}$ are created by the restrictions of the wrenches that the surfaces can apply and optionally some restrictions on the velocity and position of the CoM, compared to [19], where the definition of the region of the CoM is compulsory.

Using this program, the resulting CoM from $\boldsymbol{X_{sol}}$ will not necessarily match the cubic trajectory $\boldsymbol{R_0}(t)$. We are going to define $\boldsymbol{R_1}(t)$ as this resulting CoM trajectory from $\boldsymbol{X_{sol}}$, and we are going to solve $\mathbb{P}_2$ again, but using $\boldsymbol{R_1}$ instead of $\boldsymbol{R_0}$ in the definition of $\boldsymbol{V_0}$ and $\boldsymbol{Q_{diag0}}$, because the optimal solution tends to follow $\boldsymbol{R_1}$ instead of $\boldsymbol{R_0}$. We will repeat this process until $\boldsymbol{R_{k+1}}$ and $\boldsymbol{R_k}$ have reach a certain level of equality, measured by, for example, the maximum norm of their difference.

We can see that in some sense $\boldsymbol{R_k}(t) \times \dot{\boldsymbol{r}}$ is a linearization of $\boldsymbol{r} \times \dot{\boldsymbol{r}}$ around the trajectory $\boldsymbol{r} = \boldsymbol{R_k}(t)$.

We can achieve better results by improving the linearization of the angular momentum, We can use for more accuracy the linearization considering also the reference $\dot{\boldsymbol{r}} = \dot{\boldsymbol{R}}_{\boldsymbol{k}}(t)$. In this case we will linearize the cross term in the objective function as:

$$\boldsymbol{r} \times \dot{\boldsymbol{r}} \approx -\boldsymbol{R_k}(t) \times \dot{\boldsymbol{R}}_{\boldsymbol{k}}(t) + \boldsymbol{R_k}(t) \times \dot{\boldsymbol{r}} + \boldsymbol{r} \times \dot{\boldsymbol{R}}_{\boldsymbol{k}}(t) \quad (36)$$

Defining:

$$\boldsymbol{V_k}(t) = \left[\boldsymbol{I}_{3\times3}, m\left[\dot{\boldsymbol{R}}_{\boldsymbol{k}}(t)\right]_\times, -m[\boldsymbol{R_k}(t)]_\times\right]^T \quad (37)$$

$$\boldsymbol{v_k}(t) = m\boldsymbol{R_k}(t) \times \dot{\boldsymbol{R}}_{\boldsymbol{k}}(t) \quad (38)$$

we can see that the following approximation holds:

$$\int_0^{t_f} \|\boldsymbol{L_2} - m\boldsymbol{r} \times \dot{\boldsymbol{r}}\|_{\boldsymbol{Q_L}}^2 dt \approx \int_0^{t_f} \|\boldsymbol{V_k}(t)^T \boldsymbol{x} + \boldsymbol{v_k}(t)\|_{\boldsymbol{Q_L}}^2 dt \quad (39)$$

We can define the following matrix and vector:

$$\boldsymbol{Q_{bk}}(t) = \boldsymbol{V_k}(t)\boldsymbol{Q_L}\boldsymbol{V_k}(t)^T \quad (40)$$

$$\boldsymbol{q_{bk}}(t)^{\boldsymbol{T}} = 2\boldsymbol{v_k}(t)^T \boldsymbol{Q_L}\boldsymbol{V_k}(t)^T \quad (41)$$

and the term $\int_0^{t_f} \|\boldsymbol{V_k}(t)^T \boldsymbol{x} + \boldsymbol{v_k}(t)\|_{\boldsymbol{Q_L}}^2 dt$ can be written as:

$$\int_0^{t_f} \boldsymbol{x^T Q_{bk}}(t)\boldsymbol{x} + \boldsymbol{q_{bk}}(t)^{\boldsymbol{T}}\boldsymbol{x} + \boldsymbol{v_k}(t)\boldsymbol{Q_L}\boldsymbol{v_k}(t)^T dt \quad (42)$$

Note again that the discretization of this optimization is solvable also by a QP program using direct transcription. The constraints are still (24) and (27). Deleting the constant term respect to the variables $\boldsymbol{v_k}(t)\boldsymbol{Q_L}\boldsymbol{v_k}(t)^T$, the objective function now is:

$$\mathbb{P}_{3k}\colon p_3^* = \min_{\boldsymbol{X}} \sum_{j=0}^{p} \boldsymbol{x_j^T Q_{bk}}(t_j)\boldsymbol{x_j} + \boldsymbol{q_{bk}}(t_j)^{\boldsymbol{T}}\boldsymbol{x_j}, \text{ s.t.}$$

$$\boldsymbol{M}_{eq}\boldsymbol{X} = \boldsymbol{N}_{eq}$$

$$\boldsymbol{M}_{ineq}\boldsymbol{X} \le \boldsymbol{N}_{ineq}$$

Which can be written as:

$$\mathbb{P}_{3k}\colon p_3^* = \min_{\boldsymbol{X}} \boldsymbol{X^T Q_{diagk}X} + \boldsymbol{q_{colk}^T X}, \text{ s.t.} \quad (43)$$

$$\boldsymbol{M}_{eq}\boldsymbol{X} = \boldsymbol{N}_{eq}$$

$$\boldsymbol{M}_{ineq}\boldsymbol{X} \le \boldsymbol{N}_{ineq}$$

Where $\boldsymbol{Q_{diagk}}$ is the stacked diagonal matrix filled with $\boldsymbol{Q_{bk}}(t_j)$ and $\boldsymbol{q_{diagk}}$ is the stacked vector filled with $\boldsymbol{q_{bk}}(t_j)$ for each j in the knot points; and filled again with zeros to complete the size of $\boldsymbol{X}$.

Let´s call $\boldsymbol{X_{solk}}$ to the solution of $\mathbb{P}_{3k}$.

$$\boldsymbol{X_{solk}} = \underset{\boldsymbol{X}}{\operatorname{argmin}}\, \mathbb{P}_{3k} \quad (44)$$

Again, we are going to define $\boldsymbol{R_{k+1}}(t)$ and $\dot{\boldsymbol{R}}_{\boldsymbol{k+1}}(t)$ as resulting CoM position and velocity trajectory from $\boldsymbol{X_{solk}}$, and resolve $\mathbb{P}_3$ again, but this time with $\boldsymbol{Q_{diagk+1}}$ and $\boldsymbol{q_{colk+1}}$.

This form of iterative QP produces a solution that is adjusting to itself. In the theoretical case that $\boldsymbol{R_{k+1}}(t) = \boldsymbol{R_k}(t)$, $\boldsymbol{L_k}(t)$ and $\boldsymbol{R_k}(t)$ are the solution to the problem (30) of minimization of Angular momentum around the CoM, at least local optima, and the control law to follow is $\boldsymbol{\zeta_k}(t)$.

Any solution obtained by using $\mathbb{P}_{3k}$ is actually an approximation of the optimal solution but it is always *feasible*, because we have not linearized the dynamics with an approximation, but with an exact change of variables. This is why the $\mathbb{P}_{3k}$ works well, when you use a solution for any k, the trajectories can always be used; but the higher k you use, the less angular momentum around the CoM the robot will produce, at least as a local optima.

We have now an iterative QP program which minimizes the angular momentum around the CoM, but let's note that the solution cannot be unique. The optimal case when $\boldsymbol{L}(t) = 0$ can be solved in many ways, and actually the single contact case known as the 3D Variable-Height Inverted Pendulum has many ways to be controlled [5-8]. This is because the objective function is convex but not *strongly convex*, and some solvers will not be able to handle it. This is similar to a LQR problem with the input matrix $\boldsymbol{R} = \boldsymbol{0}$. So we can add some additional regularization terms to the objective function in order to make it strongly convex, like:

- Tracking the previous CoM position $\alpha_1\|\boldsymbol{r} - \boldsymbol{R_k}(t)\|_{Q_r}^2$
- Tracking the previous CoM velocity $\alpha_2\left\|\dot{\boldsymbol{r}} - \dot{\boldsymbol{R}}_{\boldsymbol{k}}(t)\right\|_{Q_{\dot{r}}}^2$
- Tracking the previous wrenches $\alpha_3\|\boldsymbol{\zeta} - \boldsymbol{\zeta_k}(t)\|_{R_\zeta}^2$
- Minimization of actuation or wrenches $\alpha_4\|\boldsymbol{\zeta}\|_R^2$

The first three points are written for stability purposes in the iteration. They don't change the final optimal value or the trajectories. We can initialize $\boldsymbol{\zeta_0}(t) = \boldsymbol{0}$, and this term will be corrected in the next iteration. The fourth point is optional, although it can be useful for avoiding a lot of use of actuation, it changes the resultant optimal trajectory. Also, strong restrictions on the wrenches based on maximum normal force can be directly added to the Contact Wrench Matrix. In general, for good performance in the angular momentum it is better to keep low values of α_i respect to the coefficient of angular momentum (in this case $\alpha_i \ll 1$)

Note that the addition of the four points does not produce an increase of the size of $\boldsymbol{Q_{diagk}}$ or $\boldsymbol{q_{colk}}$, this just fills the matrix with values related with the weights instead of zeros, and this will automatically match the size of $\boldsymbol{X}$. We will call the resultant matrices $\boldsymbol{Q_{diag2k}}$ or $\boldsymbol{q_{col2k}}$ respectively.

The final program has the same structure as $\mathbb{P}_{3k}$:

$$\mathbb{P}_{4k}: p_4^* = \min_{\boldsymbol{X}} \boldsymbol{X}^T \boldsymbol{Q}_{diag2k} \boldsymbol{X} + \boldsymbol{q}_{col2k}^T \boldsymbol{X}, \text{ s.t.} \quad (45)$$

$$\boldsymbol{M}_{eq} \boldsymbol{X} = \boldsymbol{N}_{eq}$$

$$\boldsymbol{M}_{ineq} \boldsymbol{X} \leq \boldsymbol{N}_{ineq}$$

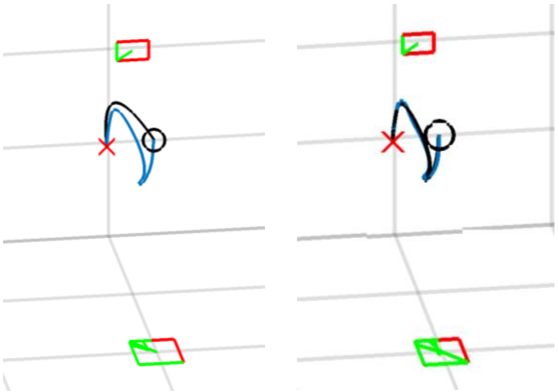

Figure 2. Resultant CoM trajectory in blue after minimizing Angular momentum around a previous trajectory in black (a) First iteration (b) Second iteration

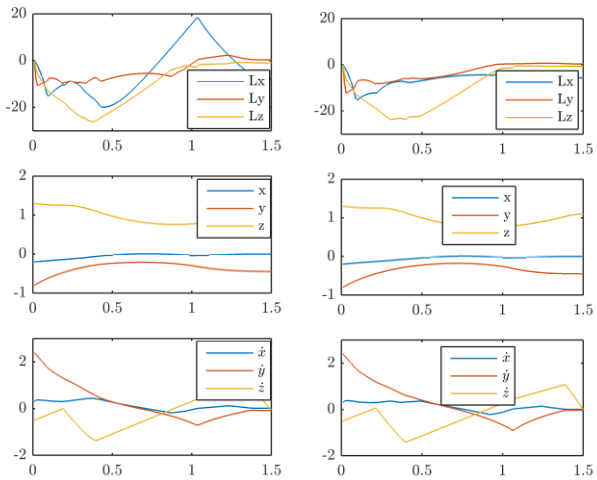

Figure 3. Angular momentum and CoM trajectories

In Fig. 2 we can see the results of the QP optimization (43) in the first two iterations of the 0-step capturability problem with minimal angular momentum. Although the CoM does not fit the estimated cubic trajectory perfectly, in the next iteration the CoM trajectory is really close to the previous trajectory.

D. *Extension to n-step Capturability and planning.*

The previous result can be directly extended to n-step capturability when the time between switches in contacts is fixed and the places where the contacts (footsteps in case of legs) are known. Perhaps the most interesting part of this subsection is that we are not going to change the objective function, so the computation of $\boldsymbol{Q}_{diag2k}$ and $\boldsymbol{q}_{col2k}$ will remain the same, but we are going to change the restrictions of the optimization. And not exactly change the current restrictions, but we are only going to add restrictions.

We are going to use the same dynamics but considering *all* the contacts and future placements at the same time:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{\zeta}_v + \boldsymbol{c} \quad (46)$$

$$\boldsymbol{W}_{bCWC} \boldsymbol{\zeta}_v \leq \boldsymbol{b}_{bCWC}$$

$$\boldsymbol{x}(0) = \boldsymbol{x}_0$$

So this time the Wrench Matrices $\boldsymbol{W}_{bCWC}$ and $\boldsymbol{b}_{bCWC}$ will contain all the contact information in the future placements. The matrices $\boldsymbol{M}_{eq}, \boldsymbol{N}_{eq}, \boldsymbol{M}_{ineq}$ and $\boldsymbol{N}_{ineq}$ are created in the same way as $\mathbb{P}_1$, but we are going to add the following restrictions:

Let be $t_{sw\delta}$ be the time when the switching between contacts occurs for each $\delta = 1, \ldots, m$, so in total there are m switches of contacts. We are going also to define $t_{sw0} = t_0$ and $t_{swm+1} = t_f$. Now, we are just going to force the wrenches that are not currently in contact to be zero.

Let Ω_δ be the set of the index of non-active contacts in the time interval $[t_{sw\delta}, t_{sw\delta+1}]$. This is translated to the following restrictions, which are linear:

$$\forall \delta: \ \boldsymbol{\zeta}_{vj}(t) = \boldsymbol{0}, \forall j \in \Omega_\delta \wedge t \in [t_{sw\delta}, t_{sw\delta+1}] \quad (47)$$

Restrictions can be compacted using a matrix $\boldsymbol{M}_{nac}$, which selects the Non-Active Contacts in the whole discretized vector $\boldsymbol{X} = [\boldsymbol{x}_0, \ldots, \boldsymbol{x}_p, \boldsymbol{\zeta}_{v0}, \ldots, \boldsymbol{\zeta}_{vp-1}]^T$

$$\boldsymbol{M}_{nac} \boldsymbol{X} = \boldsymbol{N}_{nac} = 0 \quad (48)$$

We will vertically stack the matrices $\boldsymbol{M}_{eq}$ and $\boldsymbol{M}_{nac}$ in a new matrix $\boldsymbol{M}_{eq2}$, and $\boldsymbol{N}_{eq}$ and $\boldsymbol{N}_{nac}$ in $\boldsymbol{N}_{eq2}$. Now we will define the iterative QP setup $\mathbb{P}_{5k}$ as:

$$\mathbb{P}_{5k}: p_5^* = \min_{\boldsymbol{X}} \boldsymbol{X}^T \boldsymbol{Q}_{diag2k} \boldsymbol{X} + \boldsymbol{q}_{col2k}^T \boldsymbol{X}, \text{ s.t.} \quad (49)$$

$$\boldsymbol{M}_{eq2} \boldsymbol{X} = \boldsymbol{N}_{eq2}$$

$$\boldsymbol{M}_{ineq} \boldsymbol{X} \leq \boldsymbol{N}_{ineq}$$

This approach has many variables and it is highly redundant, but directly shows that the problem can still be solved with this iterative QP method. We will go slightly further and provide a QP configuration using less variables for solving n-step capturability for the minimization of the Angular Momentum.

We will define a Hybrid System with modes:

$$M_\delta, \forall \delta \in 0, \ldots, m \quad (50)$$

M_δ are the sets containing the current active contacts of the robot in the world. n_δ is the number of current active contacts in the mode δ. We will be switching between modes when a “step” occurs, i.e. a contact is broken or a new one is added. For simplicity in this setup and because the assumption in this paper, the guards are just the time being greater than the switching time:

$$G_{\delta,\delta+1} = \{t: t - t_{sw\delta} > 0\} \quad (51)$$

The dynamics of the system for each mode $\delta \in 0, \ldots, m$ are:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}_\delta \boldsymbol{\zeta}_{v\delta} + \boldsymbol{c} \quad (52)$$

$$\boldsymbol{x}(0) = \boldsymbol{x}_0$$

$$\boldsymbol{W}_{bCWC\delta} \boldsymbol{\zeta}_{v\delta} \leq \boldsymbol{b}_{bCWC\delta}$$

Matrices $\boldsymbol{A}$ and $\boldsymbol{c}$ does not change. The only change respect to (17) is the matrix $\boldsymbol{B}_\delta$ which has the same structure as $\boldsymbol{B}$ from (17), but containing only the terms corresponding to the current active contacts $\boldsymbol{\zeta_{v\delta}}$, defined for each δ as:

$$\boldsymbol{\zeta_{v\delta}} = \{\boldsymbol{\zeta_{v\delta,k}}\}_{k=0}^{n_\delta} = \begin{bmatrix} \boldsymbol{\zeta_{v\delta,1}} \\ \vdots \\ \boldsymbol{\zeta_{v\delta,n_\delta}} \end{bmatrix}, k \in M_\delta \tag{53}$$

$\boldsymbol{W_{bCWC\delta}}$ is a stacked diagonal matrix containing the respective CWC restrictions (or friction restrictions only in case of point contacts) for each active contact for a given mode δ. $\boldsymbol{b_{bCWC\delta}}$ is the respective stacked vertical vector.

For the definition of the QP setup, we will define a vector $\boldsymbol{X}$ containing the knot points of the discretization of $\boldsymbol{x}(t)$ and $\boldsymbol{\zeta_{v\delta}}(t)$ as:

$$\boldsymbol{X} = \left[\boldsymbol{x_0}, \dots, \boldsymbol{x_p}, \boldsymbol{\zeta_{v0disc}}, \dots, \boldsymbol{\zeta_{vmdisc}}\right]^{\boldsymbol{T}} \tag{54}$$

Where $\boldsymbol{\zeta_{v\delta disc}}$ are the discretized vectors for each mode δ. Note that the length of each individual $\boldsymbol{\zeta_{v\delta disc}}$ depends on the time between switches already predefined by the programmer. Actually, $\boldsymbol{\zeta_{v\delta disc}}$ contains the discretization of the vector $\boldsymbol{\zeta_{v\delta}}(t)$ in the time interval $[t_{sw\delta}, t_{sw\delta+1}]$, and the term $[\boldsymbol{\zeta_{v0disc}}, \dots, \boldsymbol{\zeta_{vmdisc}}]^{\boldsymbol{T}}$ corresponds to the whole discretization of the wrenches in the time interval $\left[0, t_f\right]$ but without redundancy.

We use (52) to build matrices $\boldsymbol{M}_{heq}$, $\boldsymbol{N}_{heq}$, $\boldsymbol{M}_{hineq}$ and $\boldsymbol{N}_{hineq}$ in the same way we built matrices $\boldsymbol{M}_{eq}$, $\boldsymbol{N}_{eq}$, $\boldsymbol{M}_{ineq}$ and $\boldsymbol{N}_{ineq}$ from (17). We are using also the analogous restrictions to the final position inside the SSR (29) and bounding of the states (26). Matrices $\boldsymbol{Q_{hdiagk}}$ and $\boldsymbol{q_{hcolk}^T}$ are computed exactly in the same way as $\boldsymbol{Q_{diag2k}}$ and $\boldsymbol{q_{col2k}}$, but with their values matching the components of the new $\boldsymbol{X}$ defined in. We define now the following iterative program:

$$\mathbb{P}_{6k}: p_6^* = \min_{\boldsymbol{X}} \boldsymbol{X^T Q_{hdiagk} X} + \boldsymbol{q_{hcolk}^T X}, \text{ s.t.} \tag{55}$$

$$\boldsymbol{M}_{heq}\boldsymbol{X} = \boldsymbol{N}_{heq}$$

$$\boldsymbol{M}_{hineq}\boldsymbol{X} \leq \boldsymbol{N}_{hineq}$$

Essentially, we are defining exactly the same setup as $\mathbb{P}_{5k}$, but directly including restriction (48) in the variables in order to remove redundancy. By using only the active contacts, we are reducing the size of $\boldsymbol{M}_{eq}$ (because we are using $\boldsymbol{B_\delta}$ instead of $\boldsymbol{B}$) and we are reducing the dimensionality of the decision vector $\boldsymbol{X}$.

Let's also note that we are implicitly defining transitions between modes $T_{\delta,\delta+1}$ as a reset map equal to the previous state to the switch:

$$T_{\delta,\delta+1}(\boldsymbol{x}^-) = \boldsymbol{x}^- \tag{56}$$

However, linear models of impacts on the centroidal variables $T_{\delta,\delta+1}$ can be added in this QP setup as:

$$T_{\delta,\delta+1}(\boldsymbol{x}^-) = T_{\delta,\delta+1}\boldsymbol{x}^- \tag{57}$$

but we need to know how impacts change and redefine the angular momentum.

Fig. 4 shows a Wall-Run of 2.5 using Centroidal Dynamics with $\mathbb{P}_{6k}$ with minimization of the angular momentum. Maximum force of the legs is $2mg$, and the maximum force of the hand: $0.4mg$. This time we are not enforcing the final position to be a fixed point, but we are only enforcing it to remain in the SSR of the final contacts.

Fig. 5 shows a double Wall-Run in two walls (the XZ and YZ planes) after many iterations. We can see that the CoM keeps changing its final position in the first iterations. After some iterations it converges to a final position and the angular momentum is extremely minimized to human-gait values. Leg swing times must be done in 250 ms for the robot to be able to successfully perform the Wall-run and it must be able to apply forces of $4mg$.

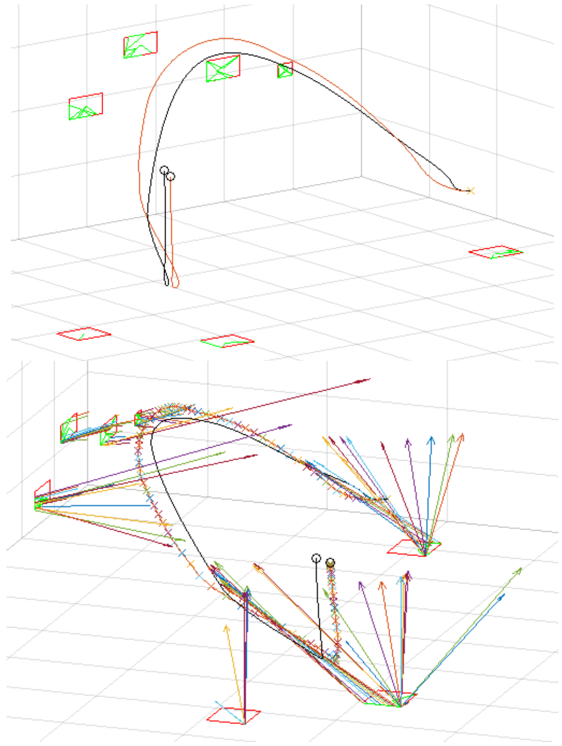

Figure 4: Wall-Run using Hand and feet (a) CoM trajectory in the second iteration (b) Forces applied on each contact

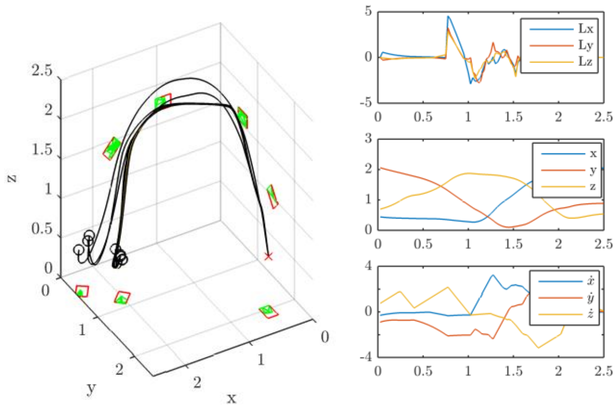

Figure 5: Double Wall-Run using feet (a) CoM trajectory after some iterations (b) State variables in the last iteration

We can see in Fig. 6 a robot walking in an almost-vertical plane based on the walking of goats in dams. In quadrupeds a lot of times it is possible to produce a resultant force going through the CoM. So in this case the angular momentum has the order of 10^{-3}. It can climb in plane of 82° considering a friction coefficient of $\mu = 1$, but we are taking this coefficient respect to *virtual plane* of 37°. This is because the terrain in average it has 82°, but it is irregular relative to the size of the foot, and it can have other virtual angles. Goats perform good foot placements to take advantage of those terrain irregularities, so the virtual plane is a reasonable consideration for the application of this work.

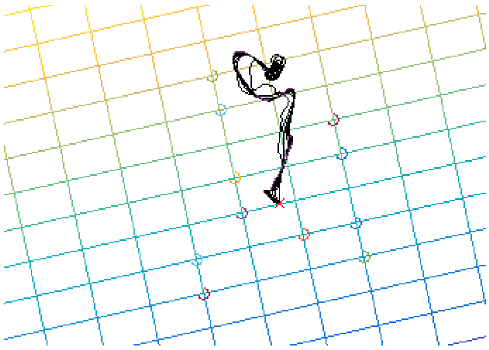

Figure 6: Quadruped robot climbing using point-contact feet only. CoM trajectory after some iterations.

IV. Discussion

In subsection A of Section III; for solving the 0-step capturability problem we have only consider the final time where rest must be reached. We have not considered a bound on the angular momentum, although this can help, it does not solve the problem definitively. The robot can spin according to a maximum angular momentum, which is not possible for a real robot, so another kind of bound must be necessary.

Defining an integral for the angular momentum is reasonable, where a flywheel is considered, and its angle is bounded. This overmodels the system and actually is not accurate, because the angular momentum is a *non-holonomic constraint*. Better approaches can be done when we could know a good way to project the physical limitations of the kinematic limits of the robot to the centroidal dynamics (specifically to the angular momentum), which not necessarily has a positive answer.

For now, minimizing angular momentum is reasonable according to [17] for walking human-like, but this not necessarily should be held in extreme motions like the double wall-run showed in Fig. 5. The robot actually must spin in the z component, but *how much does the robot spin* is hard question to solve, partially solvable with a fixed Inertia around the z axis (or bounding it between reasonable values), but this ignores completely the centroidal momentum matrix. Note also that the iterations can get stuck to a minimum, this can happen in the case of (30) suffering from minimum.

In section D, the problem of solving for complex dynamic behaviors becomes nonlinear when one of the following assumptions is removed:

- The times between switches in contacts are fixed.
- The contact placements are already planned.

The first one is the most critical. In case of highly dynamic behaviors, feasibility of $\mathbb{P}_{6k}$ is directly influenced by the switching times, so making them variables instead of fixed will highly help in the optimization. However, these values directly affect the Guards of the Hybrid system given in (51). When these guards are functions of the states (or time variable), the dynamics (52) cannot longer be compacted in the matrices $\boldsymbol{M}_{eq}$ and $\boldsymbol{N}_{eq}$.

In the second case we can estimate the contact placements by kinematic limits. Although approximations used in bipedal walking (like the instantaneous capture point or the ballistic trajectory) can be extended at least locally in Multicontact locomotion, they don't cover the general case. The questions of *when and where to step given constraints on the Centroidal Dynamics* is much harder to solve.

Another point of discussion is the optimization method used. Direct transcription is a simple but powerful method, although it needs a considerable number of knot points for reach higher levels of accuracy in the continuous dynamics. For solving this, we can use a method using partial polynomial trajectories such as Direct Collocation. This will help a lot in the reduction of knot points without losing accuracy in the discretization of the dynamics.

A last consideration corresponds to the number of active contacts and our control over them. A robot can use this approach if it has at least $6n_{pl} + 3n_{po}$, where n_{pl} is the number of planar contacts and n_{po} is the number of point contacts. Otherwise it will suffer a problem of underactuation and although we can use the Contact Wrench Sum (which requires only 6 degrees of actuation), there will exists risk of breaking some contact or falling.

V. Conclusion

In this paper we solve the 0-step capture problem for a fixed capturable time using a LP setup. We also provide an iterative QP configuration which solves for the trajectories of the centroidal dynamics and the wrenches applied at the contacts such that the resultant angular momentum is minimal. Additionally, the program can be stopped at any iteration, and the resultant trajectories are dynamically feasible, suboptimal but reasonable because of a good initialization provided as a cubic polynomial. We also extended our results to n-step capturability and planning. This allows us to compute trajectories using QP optimization for highly dynamic behaviors, that represent the motions that the world allows to perform in the case we have a robot able to perform those motions.

References

[1] S. Kajita and K. Tani. "Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode". In Proceeding of IEEE International Conference on Robotics and Automation (ICRA), vol. 2, pp. 1405-1411, 1991.

[2] J. Pratt, J. Carff, S. Drakunov and A. Goswami. "Capture Point: A Step toward Humanoid Push Recovery". In Proceedings of the IEEE-RAS/ RSJ International Conference on Humanoid Robots, pp. 200-207, 2006

[3] J. Englsberger, C. Ott, M.A. Roa, A. Albu-Schffer, G. Hirzinger. "Bipedal walking control based on capture point dynamics". In Intelligent Robots and Systems (IROS) 2011, IEEE/RSJ International Conference on, 2011

[4] J. Englsberger, G. Mesesan, C. Ott. "Smooth trajectory generation and push-recovery based on divergent component of motion". In Intelligent Robots and Systems (IROS) 2017, IEEE/RSJ International Conference on, 2017

[5] T. Koolen, M. Posa and R. Tedrake. "Balance control using center of mass height variation: limitations imposed by unilateral contact". In Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on, pp. 8-15, 2016.

[6] S. Caron and B. Mallein, "Balance control using both ZMP and COM height variations: A convex boundedness approach", in IEEE International Conference on Robotics and Automation, May 2018.

[7] S. Caron, A. Escande, L. Lanari and B. Mallein, "Capturability based Pattern Generation for Walking with Variable Height", in IEEE Transactions on Robotics, July 2019.

[8] G. Garcia, "A Control Approach for the Variable-Height Inverted Pendulum Based on Sliding Mode Control with Input Saturation", to Appear in Humanoid Robots (Humanoids), 2019 IEEE-RAS 19th International Conference on, 2019.

[9] S.-H. Lee and A. Goswami, “Reaction mass pendulum (rmp): An explicit model for centroidal angular momentum of humanoid robots,” in IEEE International Conference on Robotics and Automation (ICRA), 2007

[10] T. Seyde, A. Shrivastava, J. Englsberger, J. Pratt and R. Griffin, “Inclusion of angular momentum during planning for Capture Point based walking”. In: Robotics and Automation (ICRA) 2018, IEEE-RAS International Conference on, 2018

[11] H. Hirukawa, S. Hattori, K. Harada, S. Kajita, K. Kaneko, F. Kanehiro, K. Fujiwara, and M. Morisawa, “A universal stability criterion of the foot contact of legged robots-adios zmp,” in Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on. IEEE, 2006, pp. 1976–1983.

[12] S. Caron, Q.-C. Pham, and Y. Nakamura, “Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench cone for rectangular support areas,” in Robotics and Automation (ICRA), 2015 IEEE International Conference on. IEEE, 2015.

[13] A. Del Prete, S. Tonneau, and N. Mansard. Zero step capturability for legged robots in multicontact. IEEE Transactions on Robotics, 34(4):1021–1034, Aug 2018.

[14] D. Mansour, A. Micaelli, A. Escande, and P. Lemerle, “A computational approach for push recovery in case of multiple noncoplanar contacts,” IEEE-RAS International Conference on Humanoid Robots, 2011.

[15] D. Mansour, A. Micaelli, and P. Lemerle, “Humanoid push recovery control in case of multiple non-coplanar contacts,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2013

[16] P. Fernbach, S. Tonneau, and M. Tax, “CROC: Convex resolution of centroidal dynamics trajectories to provide a feasibility criterion for the multi contact planning problem,” in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Oct 2018, pp. 1–9.

[17] Hugh Herr and Marko Popovic. Angular momentum in human walking. Journal of Experimental Biology, 211(4):467–481, 2008.

[18] 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.

[19] H. Dai and R. Tedrake. “Planning robust walking motion on uneven terrain via convex optimization”, in Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on, 2016.

[20] 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), pp. 295– 302, 2014.

[21] A. Herzog, S. Schaal, and L. Righetti, “Structured contact force optimization for kino-dynamic motion generation,” in IEEE/RSJ Int. Conference on Intelligent Robots and Systems, 2016, pp. 2703–2710.

[22] K. Fukuda and A. Prodon. "Double description method revisited", in Combinatorics and computer science, pages 91–111. Springer, 1996.

[23] T. Bretl and S. Lall, “Testing static equilibrium for legged robots”, in Robotics, IEEE Transactions on, vol. 24, no. 4, pp. 794–807, 2008