2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)
Beijing, China, November 6-9, 2018

Nonlinear Optimization using Discrete Variational Mechanics for Dynamic Maneuvers of a 3D One-Leg Hopper

Iordanis Chatzinikolaidis, Theodoros Stouraitis, Sethu Vijayakumar, and Zhibin Li

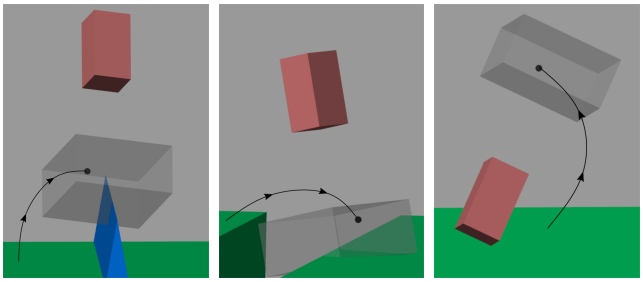
***Abstract*— We present an optimization-based motion planning framework for producing dynamically rich and feasible motions for a 3D one-leg hopper in challenging terrains. We formulate dynamic motion planning as a nonlinear optimization problem that computes position and orientation of the centroidal model, position of the limb, contact forces, contact locations, and timings of the gait in one unified framework. The dynamics are represented as a single rigid body, while the equations of motion are derived using discrete mechanics with a variational quaternion-based integrator for the orientation. We validate the capabilities by planning complex motions in three challenging tasks: jumping over an obstacle, leaping over a gap, and performing a somersault. All contact forces generated by the proposed optimization are verified with accurate numerical simulation to prove the feasibility of the generated agile motions with respect to the kinematic, dynamic, and environmental constraints.**

I. Introduction

Ground reference points [1], *e.g.* divergent component of motion (DCM) [2], instantaneous capture point (ICP) [3], and zero-moment point (ZMP) [4] are useful physical quantities for planning and control of discrete contact motions in legged locomotion. Such representations are intuitive and allow straightforward planning of references, while being effective for generating stable motions [5].

However, as the motions become more complex, *e.g.* involving contact points on non-coplanar surfaces or no contact points for a short period of time (in situations like running and hopping), the ground reference points—usually developed on a 2D projection basis—become less effective. As a result, they might require further extensions (the on-line adaptation presented in [6] is required to match the LIP model dynamics with that of the robot) or their assumptions might be invalidated. Such extensions can incur larger computational costs, while speed is one of the most attractive characteristics of these methods [7]. Moreover, the projected quantities have reduced dimensionality, which can not fully represent the 6-dimensional information of either spatial motion or contact wrench.

Therefore, a more sensible approach is to directly compute ground reaction forces and the resulting physical motions. This is applicable not only in walking scenarios but also in running and hopping, non-periodic motions, *etc.* Equally important is that angular momentum is usually neglected or enforced to be zero; this heavily restricts the regime of possible motions. Examples of motions not easily planned with ground reference point methods are shown in Fig. 1, computed by our optimization-based framework.



(a) (b) (c)

Fig. 1: A variety of dynamic maneuvers using one unified optimization framework: (a) Jumping over an obstacle; (b) Leaping over a gap; (c) Performing a somersault. We denote the body by a red box, the contact point by a black dot, and the workspace of the leg by the light-gray box.

Still, reasoning first for the kinematic part and afterwards for the dynamics in a hierarchical fashion has been used to obtain a variety of motions. In [8], first a root path was found using a sampling-based algorithm and then a discrete sequence of whole-body configurations was generated. Due to the kinematic formulation, only static stability was enforced at the discrete configurations. More importantly, as with every hierarchical approach, it is not clear how solutions of the former stages restrict the solution of the latter stages of the pipeline.

As the models become more complicated, non-convexity is introduced which make the problem very hard to solve efficiently. In [9], the authors tried to absorb the discrete and non-convex aspects of the problem using a mixed integer convex optimization formulation. The mixed-integer formulation is used to absorb the multiple non-convexities present in the problem: related to gait (the footstep sequence, kinematic constraints, *etc.*) and the non-convexities of the terrain. Rather than computing individual motions for each leg, they use a predefined set of gait sequences. Furthermore, the convex segmentation of the terrain assumes that it can be approximated by a small number of convex polygons; otherwise the number of variables increases dramatically. Yet, mixed-integer problems are still NP-hard to solve and require relaxations, heuristics, *etc.*

A related line of research is focused on tackling the problem using continuous methods. The initial approach

*This research is supported by the EPSRC as part of the CDT in Robotics and Autonomous Systems (EP/L016834/1) and the EPSRC UK RAI Hub in Offshore Robotics for Certification of Assets (ORCA) (EP/R026173/1).

The authors are with the Edinburgh Centre for Robotics and the School of Informatics, The University of Edinburgh, Edinburgh, United Kingdom. Email: i.chatzinikolaidis@ed.ac.uk

explicitly using this rationale was presented in [10]. In their work everything was treated as part of an objective, *i.e.* all constraints were expressed as soft. Weighting the different constraints can be tedious and require much fine tuning. Furthermore, the inclusion of soft constraints and forces by distance can result in non-plausible motions.

Another way to express the intermittent nature of contact is by using complementarity constraints as shown in [11]. Complementarity problems are tackled with difficulty by continuous solvers due to the ill-posed nature of the constraints, so the authors resorted to relaxations to make them more amenable. Another important issue that plagues continuous optimization approaches is the trade-off between problem size and accuracy. In [12], the authors proposed methods from discrete mechanics to make integration accuracy better, without increasing significantly the problem size.

The phase-based parametrization introduced in [13] allowed the authors to avoid solving the complementarity problem explicitly, but with the trade-off of introducing the number of steps as parameter. Thus, the approach is less general than the complementarity formulation but computationally faster. But dynamic and kinematic constraints are enforced on regular intervals, which is not straightforward to select because more dynamic motions require finer computations. Also, quantities like force, limp position, and centroidal position and orientation are parametrised with a specific structure (*i.e.* splines), and as a result only smooth motions and contacts can be represented.

We propose a constrained nonlinear optimization framework which calculates centroidal motion, limb motion, contact forces, contact timings and locations in a unified manner, given initial state, desired final state, and information about the environment. The dynamics of our system are derived using discrete variational mechanics, with the associated geometric structure-preserving properties. The contributions of our work are the following:

1) The problem is formulated as one unified optimization, rather than a hierarchy of cascade optimizations that limits the solution space.
2) The use of discrete variational mechanics allows us to express the dynamics with a minimal representation, while maintaining good numerical integration accuracy.
3) Hard constraints are used to enforce physical plausibility while avoiding a piecewise parametric motion segmentation as splines, and as a result a wide range of *dynamically feasible* motions are computed.

The paper proceeds as follows. In Section II we elaborate on the proposed method, describing the model used in Section II-A, the optimization approach in Section II-B, and the derivation of the dynamics in Section II-C. Section III presents our results in 3 challenging situations and we conclude with some overall remarks in Section IV.

II. Formulation of the Nonlinear Optimization

A. Modelling approach

We use a single rigid body which is able to describe the principles and include all the quantities associated with locomotion; translational and angular momentum, contact forces, body torques, orientation, *etc.* are well defined. At the same time, the complexity does not reach the levels of the full dynamics that a model of a humanoid robot typically exhibit, with a large number of degrees of freedom. Furthermore, current approaches in both planning and control prevalently work with the centroidal dynamic [14], through the lens of which we can view the dynamic model as a single rigid body with mass equal to the total mass of the robot and configuration varying inertia.

We augment the kinematic model of the rigid body with a contact point. The contact point (or limb) should always lie inside simplified kinematic limits, as shown in Fig. 1, while its motion should exhibit continuity.

B. Formulation of the optimization problem

Ideally, we would like to compute state and control trajectories which satisfy at every instant all the constraints that we would wish to impose—both nonlinear equality and inequality constraints. As this is currently impossible for general formulations, we follow the paradigm of "first discretize then optimize" prevalent in the trajectory optimization literature. In specific, we discretize both the state and the control with respect to time. We refer to these discretization points as knots. In order to formulate the optimization problem, we strive in principle to express as many constraints as possible implicitly and not explicitly. That is, we try to include constraints in the formulation itself rather than explicitly enforcing them.

The problem can be described as:

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & o(\boldsymbol{x}) \\ \text{subject to} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) \\ & \boldsymbol{g}(\boldsymbol{x}) = 0 \\ & \boldsymbol{h}(\boldsymbol{x}) \preceq 0 \\ & \boldsymbol{lb} \preceq \boldsymbol{x} \preceq \boldsymbol{ub}, \end{aligned} \tag{1}$$

which in our case is a non-convex optimization problem. The symbols $\preceq$ and $\succeq$ denote a pointwise inequality between two vectors. The variables $\boldsymbol{x}$ of our optimization problem, denoted by capital letters, are the following:

(A) The global position $\boldsymbol{c} \in \mathbb{R}^3$ and the change of orientation $\boldsymbol{\beta} \in \mathbb{R}^3$ of the centre of mass (CoM).
(B) The duration $dt \in \mathbb{R}$ between two successive knots.
(C) The non-negative gains $\boldsymbol{\gamma} \in \mathbb{R}^+$ for obtaining the vertex form of the linearised friction cone.
(D) The limb's position in the global frame $\boldsymbol{p} \in \mathbb{R}^3$.
(E) The limb's velocity $\dot{\boldsymbol{p}} \in \mathbb{R}^3$.

All constraints are denoted by small letters. The nonlinear equality constraints are the following:

(a) The dynamics of the CoM. An in-depth discussion about this constraint is given in Section II-C.
(b) The initial translational velocity $\dot{\boldsymbol{c}}_0 \in \mathbb{R}^3$ of the CoM and the initial angular velocity $\boldsymbol{\omega}_0 \in \mathbb{R}^3$ in body coordinates. The initial position $\boldsymbol{c}_0 \in \mathbb{R}^3$, and the initial unit quaternion $\boldsymbol{\alpha}_0 \in \mathbb{H}, ||\boldsymbol{\alpha}_0|| = 1$ for the orientation are implicitly enforced.

(c) The desired final position $\boldsymbol{c}_f \in \mathbb{R}^3$ of the CoM. Final constraints for the rest of the quantities with initial values can be straightforwardly incorporated.

(d) We assume that we have access to a height map that describes the terrain's elevation. The contact point must be on the ground:

$$p^z = h^z_{map}. \tag{2}$$

(e) The contact point does not contribute to the dynamics of the system. Thus, its kinematics are enforced through a forward Euler approximation:

$$\boldsymbol{p}_{i+1} = \boldsymbol{p}_i + \dot{\boldsymbol{p}}_i dt_i. \tag{3}$$

(f) Unilateral and friction cone constraints for a contact force are enforced by the vertex form of the linearised friction cone [15]:

$$\boldsymbol{f} = \sum_{k=1}^{4} \gamma_k \boldsymbol{v}_k, \tag{4}$$

where $\boldsymbol{f} \in \mathbb{R}^3$ is the generated contact force, and $\boldsymbol{v}_k = \boldsymbol{n} + \mu \boldsymbol{t}_k$ are the extreme rays of the friction cone which are obtained from the surface normal $\boldsymbol{n} \in \mathbb{R}^3$, the surface tangents $\boldsymbol{t}_k = \{\pm \boldsymbol{t}_1, \pm \boldsymbol{t}_2\} \in \mathbb{R}^3$, and the friction coefficient μ.

The nonlinear inequality constraints are:

(g) The contact point should lie above the height map:

$$p^z \geq h^z_{\text{map}}. \tag{5}$$

(h) Kinematic box-type limits for the limb:

$$\boldsymbol{lb} \preceq {}^b\boldsymbol{p} - {}^b\boldsymbol{c} \preceq \boldsymbol{ub}, \tag{6}$$

where $\boldsymbol{lb}, \boldsymbol{ub}$ represent the lower and upper box bounds of the relative position between the limb and the CoM, and ${}^b\boldsymbol{p}$ and ${}^b\boldsymbol{c}$ are the limb and CoM position in the body frame, respectively.

Finally, the following lower bounds are defined:

(i) The time duration between successive knots should be non-negative, *i.e.* $\boldsymbol{dt} \succeq 0$.

(j) The non-negative gains $\gamma_i \geq 0$. We also use an upper bound for this gains to bound the contact force.

We start by defining the number of contacts per limb and the number of knots per phase. By phase we mean a situation when the limb is either in rigid contact with the terrain or not [13]. Other phases in our formulation do not exist. The number of knots per phase is equivalent to the accuracy that we wish to achieve. Subsequently, the knots are divided in three sets:

- Knots where the limb is in contact with the terrain belong to the contact set.
- Knots where the limb is not in contact belong to the flight set.
- Knots between a flight and a contact phase belong to the landing set.

This division is inspired by the different states that can describe two contacting rigid bodies, as explained in [16].

There exist quantities that are defined in all knots, but also quantities defined in knots of certain sets only. Specifically, quantities (A) and (B) are defined in all knots, (C) are defined at knots belonging to the contact set, (D) are defined for knots in the flight and landing sets, and (E) are defined during flight phases only. Furthermore, we assume that the position is fixed during contact—that is zero limb velocity—and this is implicitly enforced.

These quantities are accompanied by constraints in order to enforce physical plausibility. As in the case of quantities, different constraints are enforced between different sets of knots. Equality constraints (a), (b), (c), inequality constraint (h) and bounds (i) are enforced for quantities defined in all knots. Equality (d) is defined during landing phases, while (e) during the flight phases. Finally, inequality (g) is defined during flight phases and bound (j)—*i.e.* equality (f)—during contact phases.

C. Discrete variational mechanics

The main idea underlying discrete mechanics is to discretize the action and afterwards obtain the equations of motion for a mechanical system, rather than the "classical" approach of discretizing directly the equations of motion. The main advantage of this is that the obtained integrators automatically respect conservation of quantities like momentum and energy, and symplectic form, while exhibiting very good long term numerical behaviour. Here, only the necessary parts for our formulation will be presented while more information can be found in [17].

The starting point of the forced case is the discretization of the Lagrange—d'Alembert principle [18] that seeks discrete curves $\boldsymbol{q}|_{k=0}^{N}$, where $\boldsymbol{q}_i \in \mathbb{Q}$ is a discrete configuration in the configuration space, satisfying

$$\delta \sum_{k=0}^{N-1} L_d(\boldsymbol{q}_k, \boldsymbol{q}_{k+1}) + \sum_{k=0}^{N-1} \left(\boldsymbol{F}_k^- \cdot \delta \boldsymbol{q}_k + \boldsymbol{F}_k^+ \cdot \delta \boldsymbol{q}_{k+1} \right) = 0, \tag{7}$$

where $k = 0, \ldots, N$, $L_d : \mathbb{Q} \times \mathbb{Q} \to \mathbb{R}$ is the discrete Lagrangian, and $\boldsymbol{F}_k^+$, $\boldsymbol{F}_k^-$ are the right and left discrete forces, respectively, for all variations $\delta \boldsymbol{q}_k|_{k=0}^{N}$ vanishing at the endpoints. This is equivalent to the forced discrete EulerLagrange (FDEL) equations

$$D_{\boldsymbol{q}_k} L_d(\boldsymbol{q}_{k-1}, \boldsymbol{q}_k) + D_{\boldsymbol{q}_k} L_d(\boldsymbol{q}_k, \boldsymbol{q}_{k+1}) + \boldsymbol{F}_{k-1}^+ + \boldsymbol{F}_k^- = 0, \tag{8}$$

for $k = 1, \ldots, N-1$. In our case, the initial state is not two consecutive configurations but an initial configuration and an initial generalized velocity. We can use the left discrete Legendre transform $\mathbb{F}^{F^-} L_d : \mathbb{Q} \times \mathbb{Q} \to T^*\mathbb{Q}$ obtaining

$$D_{\dot{\boldsymbol{q}}_0} \mathcal{L}(\boldsymbol{q}_0, \dot{\boldsymbol{q}}_0) + D_{\boldsymbol{q}_0} L_d(\boldsymbol{q}_0, \boldsymbol{q}_1) + \boldsymbol{F}_0^- = 0, \tag{9}$$

where $\mathcal{L}$ is the system's Lagrangian, from which we must get the next state given our initial condition.

We now proceed to make the previous discussion more specific to our situation. The continuous Lagrangian $\mathcal{L} : TSE(3) \to \mathbb{R}$ of a single rigid body is

$$\mathcal{L} = \mathcal{T} - \mathcal{V} = \frac{1}{2} m \dot{\boldsymbol{c}}^T \dot{\boldsymbol{c}} + \frac{1}{2} \boldsymbol{\omega}^T I \boldsymbol{\omega} - \mathcal{V}, \tag{10}$$

where $\mathcal{T}$ is the kinetic energy, $\mathcal{V}$ is the potential energy, m is the mass of the rigid body and I is the inertia matrix. The translational and rotational dynamics can be decomposed, and as a result we study each part separately.

1) Translational dynamics: We follow a similar line of thought as presented in [19], the main differences being the inclusion of forces (*i.e.* the discretization of the Lagrange—d'Alembert principle rather than the principle of stationary action), and the use of non-constant time intervals due to the formulation of our optimization problem. In a similar fashion, we approximate the translational velocity as $\dot{\boldsymbol{c}} = \frac{\boldsymbol{c}_{k+1}-\boldsymbol{c}_k}{h_k}$ and the relevant integrals using the midpoint rule approximation. The left and right discrete forces are approximated then as

$$\boldsymbol{F}_k^- = \boldsymbol{F}_k^+ = \frac{h_k}{4}(\boldsymbol{f}_{k+1} + \boldsymbol{f}_k). \tag{11}$$

Thus, by virtue of (8) we have that

$$\frac{m}{h_{k-1}}(\boldsymbol{c}_k - \boldsymbol{c}_{k-1}) + \frac{m}{h_k}(\boldsymbol{c}_k - \boldsymbol{c}_{k+1}) - \frac{h_{k-1}+h_k}{2}\frac{\partial \mathcal{V}}{\partial \boldsymbol{c}}(\boldsymbol{c}_k) + \frac{h_{k-1}}{4}(\boldsymbol{f}_{k-1} + \boldsymbol{f}_k) + \frac{h_k}{4}(\boldsymbol{f}_k + \boldsymbol{f}_{k+1}) = 0, \tag{12}$$

while the initial condition is calculated by

$$m\dot{\boldsymbol{c}}_0 + \frac{m}{h_0}(\boldsymbol{c}_0 - \boldsymbol{c}_1) - \frac{h_0}{2}\frac{\partial \mathcal{V}}{\partial \boldsymbol{c}}(\boldsymbol{c}_0) + \frac{h_0}{4}(\boldsymbol{f}_0 + \boldsymbol{f}_1) = 0. \tag{13}$$

2) Rotational dynamics: The orientation is parametrized using unit quaternions due to the small number of parameters (4 contrary to rotation matrices that require 9) and the lack of gimbal lock (as opposed to Euler angles). In order to avoid explicitly enforcing the unit norm for the quaternions using Lagrange multipliers, we formulate the problem using a variational integrator that preserves the Lie group structure of the unit quaternions [19], [20]. Again, we use non-constant time steps which leads us to the constraint:

$$\frac{h_{k+1}}{4}(\boldsymbol{\tau}_k + \boldsymbol{\tau}_{k+1}) + \frac{2}{h_{k+1}}\left(\alpha^s_{k+1} I\boldsymbol{\alpha}^v_{k+1} + \boldsymbol{\alpha}^v_{k+1} \times I\boldsymbol{\alpha}^v_{k+1}\right) = \frac{2}{h_k}\left(\alpha^s_k I\boldsymbol{\alpha}^v_k - \boldsymbol{\alpha}^v_k \times I\boldsymbol{\alpha}^v_k\right) + \frac{h_k}{4}(\boldsymbol{\tau}_{k-1} + \boldsymbol{\tau}_k), \tag{14}$$

where $\boldsymbol{\alpha}_k = \begin{bmatrix}\alpha^s_k & \boldsymbol{\alpha}^v_k\end{bmatrix}^T$ is the unit quaternion of the relative orientation between knots k and $k+1$, and $\boldsymbol{\tau}_k$ is the body torque at knot k. In (14), it is assumed that $\boldsymbol{\alpha}$ is a unit quaternion. To implicitly enforce the unit norm, the authors in [20] used the parametrization $\boldsymbol{\alpha} = \begin{bmatrix}\sqrt{1-\boldsymbol{\phi}^T\boldsymbol{\phi}} & \boldsymbol{\phi}\end{bmatrix}^T$, which holds for $|\boldsymbol{\phi}| < 1$, which is not unconstrained and can lead the optimizer to compute complex values, while the authors in [19] used the exponential map, which has a singularity at zero. A more appropriate parametrization for our case is using the Cayley map, which is defined as:

$$\boldsymbol{\alpha} = \begin{bmatrix}\frac{2}{1+\boldsymbol{\beta}^T\boldsymbol{\beta}} - 1 \\ \frac{2}{1+\boldsymbol{\beta}^T\boldsymbol{\beta}}\boldsymbol{\beta}\end{bmatrix}, \tag{15}$$

where $\boldsymbol{\beta} \in \mathbb{R}^3$. Finally, as with the translational dynamics, the constraint for the initial condition is defined as:

$$I\boldsymbol{\omega}_0 + \frac{h_0}{4}(\boldsymbol{\tau}_0 + \boldsymbol{\tau}_1) = \frac{2}{h_0}\left(\alpha^s_0 I\boldsymbol{\alpha}^v_0 + \boldsymbol{\alpha}^v_0 \times I\boldsymbol{\alpha}^v_0\right) \tag{16}$$

TABLE I: Parameters used in all simulation scenarios

Type of rigid body	Cuboid
Dimensions (l_x, l_y, l_z)	$0.3 \times 0.3 \times 0.55(m)$
Mass (m)	$80(kg)$
Principal moments of inertia (I)	$2.6167, 2.6167, 1.2(kg \cdot m^2)$
Static friction coefficient (μ)	0.7
Height (h_b)	$1.1(m)$
Kinematic limits (l^k_x, l^k_y, l^k_z)	$0.6 \times 0.6 \times 0.2(m)$
Initial position $(\boldsymbol{c}_0)$	$[0, -1.4, 1.1](m)$
Initial velocity $(\dot{\boldsymbol{c}}_0)$	$[0, 0, 0](m/s)$
Initial orientation $(\boldsymbol{\alpha}_0)$	$[\sqrt{2}/2, 0, 0, \sqrt{2}/2]$
Initial angular velocity $(\boldsymbol{\omega}_0)$	$[0, 0, 0](rad/s)$
Final position $(\boldsymbol{c}_f)$	$[0, 0.9, 1.1](m)$
Maximum constraint violation	$5 \cdot 10^{-5}$
Objective	$\mathbf{0}^T$

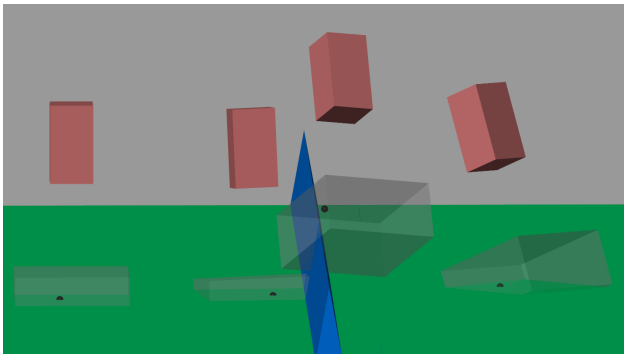


Fig. 2: Time-lapse snapshots of the solution in Case 1.

III. Results

In order to demonstrate the results of our formulation, we will focus on 3 different situations: jumping over an obstacle, leaping over a gap, and requiring a somersault. These are representative cases where classical approaches encounter large difficulties to generate dynamic motions. Table I summarises the parameters that are used across all simulation scenarios.

We validated the feasibility of the produced plans by accurate numerical simulation with MATLAB [21], using a very fine step size for integration. Also, MATLAB's nonlinear optimization solver `fmincon` (interior point algorithm) is used to solve the optimization problem. Finally, for constraints in which analytic gradients are not available, automatic differentiation using CasADi [22] is used.

Case 1: Jumping over an obstacle

In the first case, we place a triangular extrusion with height equal to half of the model's height and span of $0.2(m)$. Since the optimizer computes solutions at discrete knots, the interpolated motion afterwards might intersect the obstacle. In order to avoid such situations, we give a sketch of the desired solution by providing linear spaced positions of the CoM and limb as a starting point for the optimizer. We select 3 steps and 10 knots per phase.

As a quantitative measure of the difference between the optimizer solution and that of MATLAB for the CoM po-

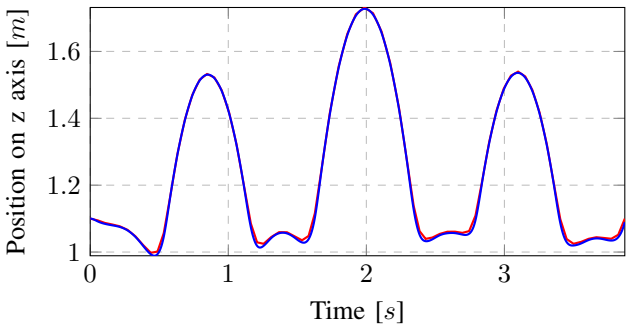


Fig. 3: Centre of mass position in Case 1: In blue is the position along the z-axis as computed by MATLAB, while in red is the linear interpolated result of our optimizer.

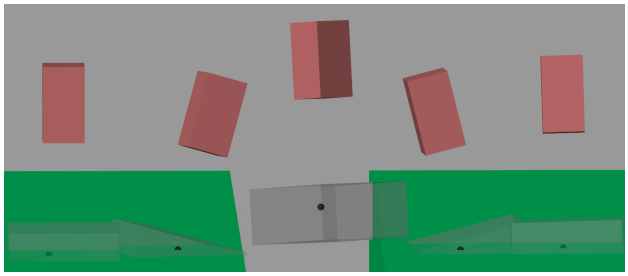


Fig. 4: Time-lapse snapshots of the solution in Case 2.

sition we use the root mean square error (RMSE), which is $[6.6216 \cdot 10^{-4}, 7.9697 \cdot 10^{-4}, 0.0066](m)$. In Fig. 3, we show the two outputs for the position in z axis only, since the other two axis have very small errors that make the plots almost indistinguishable. To measure the difference between the computed quaternions [23], we use the following metric

$$\theta = 2 \arccos\left(|\boldsymbol{\alpha}_1 \cdot \boldsymbol{\alpha}_2|\right), \tag{17}$$

with θ the angle required to get from one orientation to the other, and $\cdot$ the dot product between two quaternions. In this case, $\theta = 0.2941(rad)$ is the maximum angle. Snapshots of the resulting motion are shown in Fig. 2.

Case 2: Leaping over a gap

In this case, shown in Fig. 4, we place the model at a terrain with $3(m)$ height. At the same time, we place a gap of $1(m)$ span from $-0.5(m)$ to $0.5(m)$ along the y-axis.

Our scheme is only able to find local solutions. Thus, unless we model the gap in the formulation, it is very difficult to converge. Even though in the terrain representation we use a smooth surface interpolated using cubic splines, the gradients near the edges still change in a very abrupt manner. Furthermore, for points inside the gap, the optimizer is unable to find solutions because very large forces would be required in order to escape from it. As a result, we select the desired flight phase where the jump takes place. Then we add a lower bound for the position of the limb at the last knot of the landing set before the jump and an upper bound for the first knot of the landing set after the jump.

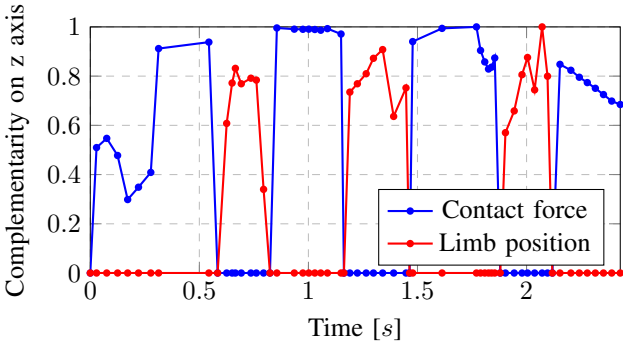


Fig. 5: Complementarity condition in Case 2: Force only exists when the limb is in contact with the terrain, while the limb moves freely only when force is 0. All quantities are normalized by their maximum values.

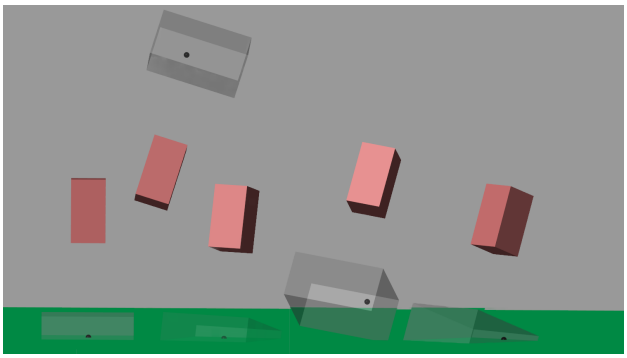


Fig. 6: Time-lapse snapshots for part of the solution in Case 3. The desired final position of the centre of mass, as specified in Table I, remains the same.

Here we select 3 steps and 8 knots per phase. The RMSE for the position of the CoM is $[0.0042, 0.0121, 0.0129](m)$, while the maximum angle is $\theta = 0.1633(rad)$. In order to stress out the complementarity inherent in the formulation, we show in Fig. 5 the contact forces and limb positions in the same graph. Since we are comparing different physical quantities, we normalize each one with their maximum value to get a qualitative comparison.

Case 3: Somersault

To show the modularity of the framework we include an orientation constraint. In specific, we desire an orientation π rad with respect to the x axis in the middle of the first flight phase. The situation is depicted in Fig. 6.

There are two potential ways to implement that: either via a suitable initialisation or via an equality constraint. Here, we implement the second approach.

We select 4 steps and we use a larger number of knots per phase, *i.e.* 15, to get a sufficiently accurate approximation of the orientation. This shows why being able to choose the number of knots per phase is an important factor; different motions may require different approximation.

The translational part, due to the fine mesh, has a RMSE of $[1.4259 \cdot 10^{-4}, 2.4917 \cdot 10^{-4}, 0.0015](m)$, while the rotational

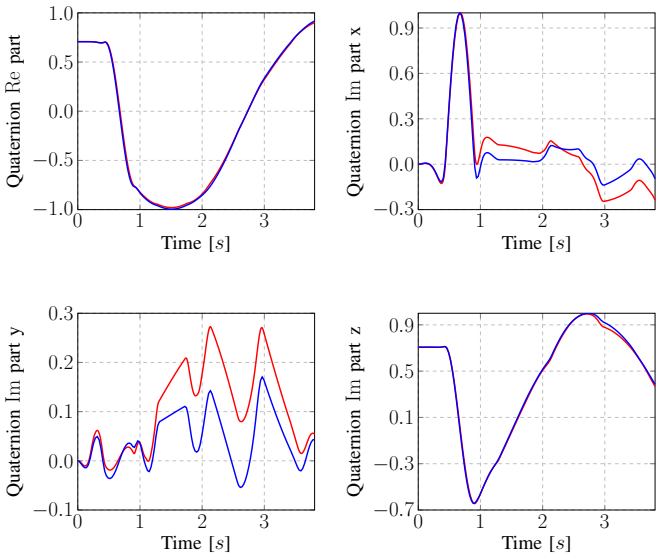


Fig. 7: Orientation for Case 3: In red is the MATLAB's output as baseline, while in blue is the output of our method.

part has a maximum angle of $\theta = 0.3127(rad)$. The rotational part has a larger error in this case due to the relative large time steps for a midpoint rule approximation. This error can be seen in the quaternion components presented in Fig. 7.

IV. Conclusion

Our study focuses on a unified nonlinear optimization formulation that is capable of producing a wide range of dynamic motions in challenging scenarios as demonstrated by our simulation study. The scope of this work focuses on the formulation of feasible solutions. Thus, there are still a number of important questions to be answered as future work: initialization, computation time, objective function, and the extension to multiple limbs.

The optimization problem in this work is able to converge to local solutions only. As a result, seeding with appropriate initializations is very important in terms of the quality of the solutions and computational speed, while the initialization requires specific terrain information. Besides, the convergence speed depends also on the characteristics of the terrain. Computation time can vary from seconds to minutes depending on these two factors.

A specific objective function is not yet given due to the additional computational cost, although it is necessary to quantitatively differentiate between feasible solutions. We plan to augment it by penalizing the magnitude of contact forces and large limb velocities. Finally, we plan to extend this formulation to multiple limbs by specifying different timings and enforcing the kinematic constraint separately and dynamic constraints for the aggregate contact force.

References

[1] M. B. Popović, A. Goswami, and H. Herr, "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications," *International Journal of Robotics Research*, vol. 24, no. 12, pp. 1013–1032, Dec. 2005.

[2] J. Englsberger, C. Ott, and A. Albu-Schäffer, "Three-dimensional bipedal walking control based on divergent component of motion," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 355–368, Mar. 2015.

[3] T. Koolen, T. de Boer, J. Rebula, A. Goswami, and J. Pratt, "Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models," *International Journal of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, July 2012.

[4] M. Vukobratović and B. Borovac, "Zero-moment point - thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157–173, Jan. 2004.

[5] K. Yuan and Z. Li, "An improved formulation for model predictive control of legged robots for gait planning and feedback control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2018.

[6] C. Zhou, X. Wang, Z. Li, and N. Tsagarakis, "Overview of gait synthesis for the humanoid coman," *Journal of Bionic Engineering*, vol. 14, no. 1, pp. 15–25, 2017.

[7] W. Hu, I. Chatzinikolaidis, K. Yuan, and Z. Li, "Comparison study of nonlinear optimization of step durations and foot placement for dynamic walking," in *IEEE International Conference on Robotics and Automation*, May 2018, pp. 433–439.

[8] S. Tonneau, A. D. 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.

[9] B. Aceituno-Cabezas, C. Mastalli, H. Dai, M. Focchi, A. Radulescu, D. G. Caldwell, J. Cappelletto, J. C. Grieco, G. Fernández-López, and C. Semini, "Simultaneous contact, gait, and motion planning for robust multilegged locomotion via mixed-integer convex optimization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2531–2538, July 2018.

[10] I. Mordatch, E. Todorov, and Z. Popović, "Discovery of complex behaviors through contact-invariant optimization," *ACM Transactions on Graphics*, vol. 31, no. 4, pp. 43:1–43:8, July 2012.

[11] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *International Journal of Robotics Research*, vol. 33, no. 1, pp. 69–81, Jan. 2014.

[12] Z. R. Manchester and S. Kuindersma, "Variational contact-implicit trajectory optimization," in *International Symposium on Robotics Research (ISRR)*, Puerto Varas, Chile, Dec. 2017.

[13] 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*, vol. 3, no. 3, pp. 1560–1567, July 2018.

[14] D. E. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonomous Robots*, vol. 35, no. 2, pp. 161–176, Oct. 2013.

[15] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *IEEE-RAS International Conference on Robotics and Automation*, May 2014, pp. 2589–2594.

[16] R. Featherstone, *Rigid body dynamics algorithms*. Springer US, 2008.

[17] J. E. Marsden and M. West, "Discrete mechanics and variational integrators," *Acta Numerica*, vol. 10, pp. 357–514, May 2001.

[18] O. Junge, J. E. Marsden, and S. Ober-Blöbaum, "Discrete mechanics and optimal control," *IFAC Proc. Vol.*, vol. 38, no. 1, pp. 538–543, Jan. 2005.

[19] X. Shen and M. Leok, "Lie group variational integrators for rigid body problems using quaternions," *ArXiv e-prints*, May 2017.

[20] Z. R. Manchester and M. A. Peck, "Quaternion variational integrators for spacecraft dynamics," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 1, pp. 69–76, Jan. 2016.

[21] The MathWorks Inc., "Matlab and simulink."

[22] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, July 2018.

[23] D. Q. Huynh, "Metrics for 3d rotations: comparison and analysis," *Journal of Mathematical Imaging and Vision*, vol. 35, no. 2, pp. 155–164, Oct. 2009.