Kinodynamic Motion Retiming for Humanoid Robots

Stéphane CARON, Yoshihiko NAKAMURA, Quang-Cuong PHAM
Department of Mechano-Informatics, University of Tokyo, Japan

1. Introduction

In this paper, we advocate the use of Time-Optimal Path Parameterization (TOPP) to enable planning of dynamic motions for humanoid robots.

Consider a robot with n degrees of freedom. The Time-Optimal Path Parameterization (TOPP) problem is to find a time parameterization $s(t) : [0, T] \to [0, T_{\mathsf{ref}}]$ of a reference path $q(s) : [0, T_{\mathsf{ref}}] \to \mathbb{R}^n$ such that the retimed duration T is minimal. The optimization is constrained by physical limitations of the robot, *e.g.*, joint, velocity and torque limits. Additional constraints for balance or frictional contact can also be incorporated, as we will see.

A first solution to the TOPP problem based on the maximum velocity curve (MVC) was introduced as early as 1985 in seminal papers by Bobrow et al. [1] and Shin et al. [9]. Since then, it has been successively refined over the years to improve computation efficiency [10], deal with new kinds of constraints [7] or handle properly so-called "dynamic singularities" [8, 5]. This approach leads to fast computations, but dynamic singularities need to be treated with special care to avoid numerical instabilities.

More recently, convex optimization has been explored as an alternative to the MVC solution [12, 11, 2]. This approach allows for more general objective functions than the total duration T, *e.g.*, jerk or energy consumption. It is also said to be less sensitive to dynamic singularities, although the problem remains (see *e.g.*, Figure 4 in [12]). However, computation times with this approach are significantly slower than with the MVC approach (*e.g.*, [6] reported computation times one order of magnitude slower on n-dimensional point-mass problems).

TOPP has been applied to humanoid trajectory planning using convex optimization in [11] and [3]. In [11], the authors take into account balance and sliding constraints, but not the actuation redundancy that arises in double support configurations. In [3], the author deals with both actuator limits and frictional contacts, using a polytope projection algorithm for the latter case.

Throughout the paper, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ denote the derivatives of $\mathbf{q}$ with respect to the reparameterized time t, while $\mathbf{q}_s$ and $\mathbf{q}_{ss}$ denote the derivatives with respect to the path variable s. Both are linked by the chain-rule $\dot{\mathbf{q}} = \dot{s}\mathbf{q}_s$ and $\ddot{\mathbf{q}} = \ddot{s}\mathbf{q}_s + \dot{s}^2\mathbf{q}_{ss}$. As a consequence, constraints expressed as functions of $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ can be written equivalently as functions of the path derivatives $(s, \dot{s}, \ddot{s})$. For the humanoid, we are interested in three types of constraints:

- **ZMP balance:** if we denote by $x_{\mathsf{ZMP}}, y_{\mathsf{ZMP}}$ the floor coordinates of the ZMP and $\mathbf{p} := \begin{pmatrix} x_{\mathsf{ZMP}} & y_{\mathsf{ZMP}} & 1 \end{pmatrix}^{\mathrm{T}}$, the condition that the ZMP lies inside the support polygon can be written as
$$\mathbf{D}(s)\,\mathbf{p}(s, \dot{s}, \ddot{s}) \leq 0, \tag{1}$$
where $\mathbf{D}(s)$ is the matrix of the support polygon.
- **Frictional contact:** for a ground reaction force $\mathbf{f}_i$ exerted at a given contact point C_i, the linearized condition of frictional contact is
$$\begin{pmatrix} +1 & +1 & -\mu \\ +1 & -1 & -\mu \\ -1 & +1 & -\mu \\ -1 & -1 & -\mu \\ 0 & 0 & -1 \end{pmatrix} \mathbf{f}_i(s, \dot{s}, \ddot{s}) \leq 0, \tag{2}$$
- **Actuated torques:** given the stacked vector $\mathbf{f}$ of reaction forces at all contact points and the corresponding stacked matrix $\mathbf{J}_C$ of all contact jacobians, the equations of motions for the under-actuated humanoid are
$$\tau = \mathbf{S}\left[\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{q}}^{\mathrm{T}}\mathbf{C}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{g} - \mathbf{J}_C^{\mathrm{T}}\mathbf{f}\right] \tag{3}$$
with $\mathbf{M}(\mathbf{q})$ the inertia matrix, $\mathbf{C}(\mathbf{q})$ the Coriolis tensor and $\mathbf{g}(\mathbf{q})$ the gravity vector. Given that $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$ and $\mathbf{f}$ can be expressed as functions of $(s, \dot{s}, \ddot{s})$, the bounded torques condition $|\tau| \leq \tau_{\max}$ can be written as
$$\tau(s, \dot{s}, \dot{s}) - \tau_{\max} \leq 0 \tag{4}$$
$$\tau_{\max} - \tau(s, \dot{s}, \ddot{s}) \leq 0 \tag{5}$$

All three types of constraints boil down to a relation $f(s, \dot{s}, \ddot{s}) \leq 0$. We will show how each of them can be further factorized in:
$$\ddot{s}\mathbf{a}(s) + \dot{s}^2\mathbf{b}(s) + \mathbf{c}(s) \leq 0. \tag{6}$$

The MVC algorithm was developped to solve problems in this form. As such, with the ability to compute vectors $(\mathbf{a}(s), \mathbf{b}(s), \mathbf{c}(s))$ at each index $s \in [0, T_{\mathsf{ref}}]$ of the reference trajectory, we will be able to retime our humanoid motions with TOPP.

2. Balance constraints

Although it does not guarantee balance, the condition that the Zero-Moment Point (ZMP) lies inside the convex hull of ground contact points is physically consistent with the robot not tilting. In [7], the authors derived the TOPP form (6) of this constraint for rectangular support areas. We will now provide a similar derivation for any convex polygonal area.

For each link i of the robot, let us denote by $\mathbf{r}^i$ the position of its center of mass in the laboratory reference frame. We write $\mathbf{h}$ the resultant of contact forces applied to the ground:

$$\mathbf{h} := \sum_i m_i(\mathbf{g} - \ddot{\mathbf{r}}^i) \tag{7}$$

and τ the moment of $\mathbf{h}$ around the origin of the fixed world frame:

$$\tau = \sum_i \left(m_i \mathbf{r}^i \times (\mathbf{g} - \ddot{\mathbf{r}}^i) - \dot{\mathcal{L}}^i \right). \tag{8}$$

In this expression, $\dot{\mathcal{L}}^i$ denotes the angular momentum at the center of link i. It is given by

$$\dot{\mathcal{L}}^i = \mathbf{R}^i(\mathbf{I}^i \dot{\omega}^i + \omega^i \times (\mathbf{I}^i \omega^i))$$

with $\mathbf{R}^i$ (resp. $\mathbf{I}^i$) the rotation (resp. inertia) matrix associated with link i, and ω^i its angular velocity. In this work, we assumed that this angular momentum could be neglected, which was verified empirically in our experiments. We will thus omit it from now on.

The floor coordinates of the ZMP are given by:

$$\begin{pmatrix} x_{\mathsf{ZMP}} \\ y_{\mathsf{ZMP}} \end{pmatrix} = \frac{1}{\begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \mathbf{h}} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \tau \tag{9}$$

The condition that $(x_{\mathsf{ZMP}}, y_{\mathsf{ZMP}})$ belongs to the convex hull of the ground contact points can be expressed as the intersection of a set of half-planes, resulting in Equation (1). Let us focus on one line of this equation:

$$\alpha y_{\mathsf{ZMP}} + \beta x_{\mathsf{ZMP}} + \gamma \leq 0. \tag{10}$$

Introducing $\mathbf{u}_{\alpha\beta} := \begin{pmatrix} -\alpha & \beta & 0 \end{pmatrix}$ and $\mathbf{u}_\gamma := \begin{pmatrix} 0 & 0 & -\gamma \end{pmatrix}$, multiplying (10) by $-\begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \mathbf{h}$ yields

$$0 \geq \mathbf{u}_{\alpha\beta} \cdot \tau + \mathbf{u}_\gamma \cdot \mathbf{h}$$

Expanding $\mathbf{h}$ and τ, the constraint becomes:

$$\begin{aligned} 0 &\geq \sum_i m_i(\mathbf{u}_{\alpha\beta} \cdot \mathbf{r}^i \times (\mathbf{g} - \ddot{\mathbf{r}}^i) + \mathbf{u}_\gamma \cdot (\mathbf{g} - \ddot{\mathbf{r}}^i)) \\ &\geq \sum_i m_i(\mathbf{g} - \ddot{\mathbf{r}}^i) \cdot (\mathbf{u}_{\alpha\beta} \times \mathbf{r}^i + \mathbf{u}_\gamma) \end{aligned}$$

Given the jacobian $\mathbf{J}^i := \frac{\mathrm{d}\mathbf{r}^i}{\mathrm{d}\mathbf{q}}$ and hessian $\mathbf{H}^i := \frac{\mathrm{d}^2\mathbf{r}^i}{\mathrm{d}\mathbf{q}^2}$ of each link's COM, one can write:

$$\begin{aligned} \dot{\mathbf{r}}^i &= \mathbf{J}^i \dot{\mathbf{q}} = \mathbf{J}^i \mathbf{q}_s \dot{s} \\ \ddot{\mathbf{r}}^i &= \mathbf{J}^i \ddot{\mathbf{q}} + \dot{\mathbf{q}}^{\mathrm{T}} \mathbf{H}^i \dot{\mathbf{q}} = \mathbf{J}^i \mathbf{q}_s \ddot{s} + \dot{s}^2 (\mathbf{J}^i \mathbf{q}_{ss} + \mathbf{q}_s{}^{\mathrm{T}} \mathbf{H}^i \mathbf{q}_s) \end{aligned}$$

Using these expressions, (10) can finally be put in form (6) with:

$$\begin{aligned} \mathbf{a}_{\mathsf{ZMP}}(s) &= -\sum_i m_i \mathbf{J}^i \mathbf{q}_s (\mathbf{u}_{\alpha\beta} \times \mathbf{r}^i + \mathbf{u}_\gamma), \\ \mathbf{b}_{\mathsf{ZMP}}(s) &= -\sum_i m_i (\mathbf{J}^i \mathbf{q}_{ss} + \mathbf{q}_s^{\mathrm{T}} \mathbf{H}^i \mathbf{q}_s) \cdot (\mathbf{u}_{\alpha\beta} \times \mathbf{r}^i + \mathbf{u}_\gamma), \\ \mathbf{c}_{\mathsf{ZMP}}(s) &= \mathbf{g} \cdot (\mathbf{u}_{\alpha\beta} \times \mathbf{G} + M \mathbf{u}_\gamma), \end{aligned}$$

with $\mathbf{G}$ the COM of the robot and M its total mass.

3. Frictional Contact

As humanoid robots are not fixed to the ground, six additional degrees of freedom (DOF) are needed to account for the position and orientation of their mobile referential. These DOFs are not actuated: their movement results from interactions with the environment. We model these interations as a set of contact forces $\mathbf{f}_i$ applied at corresponding contact points C_i ($i = 1, \ldots, k$). Each foot in contact with the ground defines four contact points, one per corner of its rectangular contact surface. We denote by $\mathbf{J}_C^i = \frac{\mathrm{d}R_i}{\mathrm{d}\mathbf{q}}$ the contact jacobian at C_i. (Note that it only depends on $\mathbf{q}$.) The equations of motion are then given by Equation (3). Projecting on the last six coordinates of this equation gives:

$$\mathbf{P}(\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{q}}^{\mathrm{T}} \mathbf{C}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{g}) = \mathbf{P}\mathbf{J}^{\mathrm{T}}\mathbf{f} \tag{11}$$

with $\mathbf{P}$ the corresponding $6 \times n$ projector. Using the pseudo-inverse $(\mathbf{P}\mathbf{J}^{\mathrm{T}})^\dagger$, we can express a least-square solution $\mathbf{f}_0$ to (11):

$$\begin{aligned} \mathbf{f}_0 &= (\mathbf{P}\mathbf{J}^{\mathrm{T}})^\dagger \mathbf{P} \cdot (\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{q}}^{\mathrm{T}} \mathbf{C}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{g}) &(12) \\ \tau &= \mathbf{E}(\mathbf{q})\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{E}(\mathbf{q})\dot{\mathbf{q}}^{\mathrm{T}} \mathbf{C}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{E}(\mathbf{q})\mathbf{g} &(13) \end{aligned}$$

where $\mathbf{E}(\mathbf{q}) := \mathbf{S}\left(\mathbf{I}_n - (\mathbf{P}\mathbf{J}^{\mathrm{T}})^\dagger \mathbf{P}\right)$ is an actuated torques projection matrix. With this last expression, we can express torque constraints $\tau \leq \tau_{\max}$ as

$$\begin{aligned} \mathbf{a}_{\mathsf{lstsq}}(s) &= \mathbf{E}(\mathbf{q})\mathbf{M}(\mathbf{q})\mathbf{q}_s, \\ \mathbf{b}_{\mathsf{lstsq}}(s) &= \mathbf{E}(\mathbf{q})\left[\mathbf{M}(\mathbf{q})\mathbf{q}_{ss} + \mathbf{q}_s{}^{\mathrm{T}} \mathbf{C}(\mathbf{q})\mathbf{q}_s\right], \\ \mathbf{c}_{\mathsf{lstsq}}(s) &= \mathbf{E}(\mathbf{q})\mathbf{g}(\mathbf{q}) - \tau_{\max}. \end{aligned}$$

However, there are also frictional constraints on the contact forces $\mathbf{f}_i$ given by Equation (2) or, equivalently, $\mathbf{T} \cdot \mathbf{f} \leq 0$. One could *e.g.*, use an off-the-shelf Quadratic Programming (QP) solver to take into account these inequality constraints when solving for $\mathbf{f}$ at Equation (11). However, this approach would yield an actuated torques projector $\mathbf{E}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ with a non-linear dependency on $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$.

The rationale of our approach is to compute solutions to the QP problem *with inequality constraints* only for specific values of $\dot{s}$, and use a linear model to describe how $\mathbf{f}$ deviates from this solution when $\dot{s} < 1$ or $\dot{s} > 1$. All solutions to Equation (11) can be written:

$$\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \mathbf{f}_0(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z} \tag{14}$$

where the expression of a particular solution $\mathbf{f}_0(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is given by Equation (12), and $\widetilde{\mathbf{Q}}(\mathbf{q}) = (\mathbf{I} - (\mathbf{P}\mathbf{J}_C^{\mathrm{T}})^\dagger \mathbf{P}\mathbf{J}_C^{\mathrm{T}})$ is the projector on the nullspace of the solution space. We can use the vector $\mathbf{z} \in \mathbb{R}^{3k}$ to enforce the additional constraint $\mathbf{T}\mathbf{f} \leq 0$.

Using the chain-rule $\dot{\mathbf{q}} = \dot{s}\mathbf{q}_s$ and $\ddot{\mathbf{q}} = \ddot{s}\mathbf{q}_s + \dot{s}^2\mathbf{q}_{ss}$ in Equation (12), we can write the variation of $\mathbf{f}_0$ between the retimed state $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ and the initial

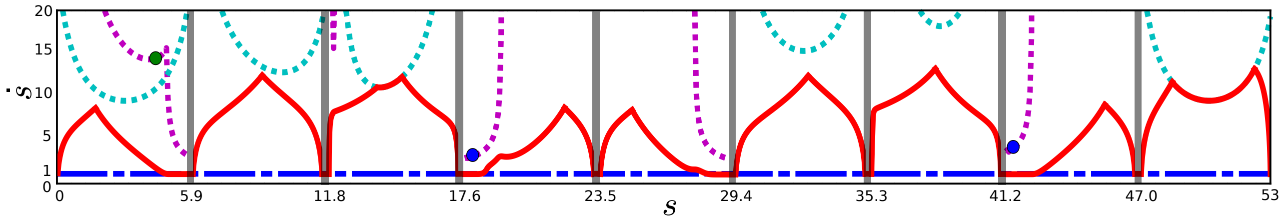

Fig.1 Retimed trajectory profiles in the $(s, \dot{s})$ space. Maximum Velocity Curves are depicted by dotted cyan and magenta lines. The dashed blue line represents the initial trajectory ($\dot{s} = 1$) while the red curve corresponds to the retimed trajectory. It may follow but never crosses the MVCs. Shaded grey areas show the intervals where we disabled retiming, forcing the retimed profile to go follow $\dot{s} = 1$. Blue and green dots indicate discontinuity of the MVC and singular points, respectively.

trajectory state $(\mathbf{q}, \mathbf{q}_s, \mathbf{q}_{ss})$ as follows:

$$\begin{aligned} \mathbf{f}_0(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) &= \dot{s}^2 \mathbf{f}_0(\mathbf{q}, \mathbf{q}_s, \mathbf{q}_{ss}) + (1 - \dot{s}^2)\widetilde{\mathbf{g}} + \ddot{s}\widetilde{\mathbf{m}}, \\ \widetilde{\mathbf{g}} &:= (\mathbf{P}\mathbf{J}_C^{\mathrm{T}})^{\dagger}\mathbf{P}\mathbf{g}(\mathbf{q}), \\ \widetilde{\mathbf{m}} &:= (\mathbf{P}\mathbf{J}_C^{\mathrm{T}})^{\dagger}\mathbf{P}\mathbf{M}(\mathbf{q})\mathbf{q}_s. \end{aligned}$$

We verify that $\mathbf{f}_0(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \mathbf{f}_0(\mathbf{q}, \mathbf{q}_s, \mathbf{q}_{ss})$ when $\dot{s} = 1$, which corresponds to the initial trajectory. Equation (14) becomes

$$\begin{aligned} \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \dot{s}^2 & \left[\mathbf{f}_0(\mathbf{q}, \mathbf{q}_s, \mathbf{q}_{ss}) + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_0\right] \\ &+ (1 - \dot{s}^2) \left[\widetilde{\mathbf{g}} + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_1\right] \\ &+ \ddot{s} \left[\widetilde{\mathbf{m}} + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_2\right] \end{aligned}$$

Under this formulation, solutions to the general problem "find $(\mathbf{z}_0, \mathbf{z}_1, \mathbf{z}_2)$ s.t. $\mathbf{T}\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \leq 0$" still depend on both $\dot{s}$ and $\ddot{s}$, but we make the approximation of choosing $\mathbf{z}_i$'s that only depend on $\mathbf{q}$. First, taking $(\dot{s}^2, \ddot{s}) = (1, 0)$, consider the following QP:

$$\begin{aligned} \min_{\mathbf{z}_0} \quad & \|\mathbf{z}_0\| \\ \text{s.t.} \quad & \mathbf{T}\left(\mathbf{f}_0(\mathbf{q}, \mathbf{q}_s, \mathbf{q}_{ss}) + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_0\right) \leq 0 \end{aligned}$$

This QP has a solution if and only if the input trajectory is feasible. To make the explanation easier, let us suppose for now that all QPs have feasible solutions. Using a solution $\mathbf{z}_0$, we derive $\mathbf{z}_1$ at, *e.g.*, $(\dot{s}^2, \ddot{s}) = (2, 0)$:

$$\begin{aligned} \min_{\mathbf{z}_1} \quad & \|\mathbf{z}_1\| \\ \text{s.t.} \quad & \mathbf{T}\left(\mathbf{u}_1 - \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_1\right) \leq 0 \end{aligned}$$

Where $\mathbf{u}_1 := 2\mathbf{f}_0(\mathbf{q}, \mathbf{q}_s, \mathbf{q}_{ss}) - \widetilde{\mathbf{g}} + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_0$. Similarly for $\mathbf{z}_2$ at, *e.g.*, $(\dot{s}^2, \ddot{s}) = (1, 1)$:

$$\begin{aligned} \min_{\mathbf{z}_2} \quad & \|\mathbf{z}_2\| \\ \text{s.t.} \quad & \mathbf{T}\left(\mathbf{v}_0 + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_2\right) \leq 0 \end{aligned}$$

Where $\mathbf{v}_0 := \mathbf{f}_0 + \widetilde{\mathbf{m}} + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_0$. Once $\mathbf{z}_0$, $\mathbf{z}_1$ and $\mathbf{z}_2$ are computed, the contact constraint (2) becomes (6) with

$$\begin{aligned} \mathbf{a}_{\mathsf{friction}}(s) &= \mathbf{T}\left(\widetilde{\mathbf{m}} + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_2\right) \\ \mathbf{b}_{\mathsf{friction}}(s) &= \mathbf{T}\left(\mathbf{f}_0 - \widetilde{\mathbf{g}} + \widetilde{\mathbf{Q}}(\mathbf{q})(\mathbf{z}_0 - \mathbf{z}_1)\right) \\ \mathbf{c}_{\mathsf{friction}}(s) &= \mathbf{T}\left(\widetilde{\mathbf{g}} + \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_1\right) \end{aligned}$$

Actuated torque limits are finally obtained from $\mathbf{f}$ using Equation (3), which gives

$$\begin{aligned} \mathbf{a}_{\mathsf{torque}}(s) &= \mathbf{S}^{\mathrm{T}}\left(\mathbf{M}(\mathbf{q})\mathbf{q}_s - \widetilde{\mathbf{m}} - \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_2\right) \\ \mathbf{b}_{\mathsf{torque}}(s) &= \mathbf{S}^{\mathrm{T}}\left(\mathbf{M}\mathbf{q}_{ss} + {\mathbf{q}_s}^{\mathrm{T}}\mathbf{C}(\mathbf{q})\mathbf{q}_s\right) \\ &\quad + \mathbf{S}^{\mathrm{T}}\left(\widetilde{\mathbf{g}} - \mathbf{f}_0 - \widetilde{\mathbf{Q}}(\mathbf{q})(\mathbf{z}_0 - \mathbf{z}_1)\right) \\ \mathbf{c}_{\mathsf{torque}}(s) &= \mathbf{S}^{\mathrm{T}}\left(\mathbf{g} - \widetilde{\mathbf{g}} - \widetilde{\mathbf{Q}}(\mathbf{q})\mathbf{z}_1\right) \end{aligned}$$

The previous method assumes that all QPs had feasible solutions, which allowed us to use points $(\dot{s}^2, \ddot{s}) \in \{(1,0), (2,0), (1,1)\}$ where computations are convenient. In the general case, the same method can be applied but the space $(\dot{s}^2, \ddot{s})$ needs to be searched for three feasible solutions. This search was straightforward in our experiments: random sampling in $(0, \dot{s}^2_{\max}) \times [0, \ddot{s}_{\max}]$ retuend feasible solutions after a few steps.

4. Experiment with HRP-4

In this experiment, we show how a slow and quasi-static motion can be accelerated by our solution while still maintaining ZMP balance, frictional contact and actuated torque limits.

The reference trajectory is a 53-second quasi-static motion. It starts by moving the COM to the left foot, then steps the right foot 15 cm forward, moves the COM to the right foot, steps the left foot 15 cm forward, and finally moves the COM to the center of the support area. These high-level instructions are translated into key-frames using inverse kinematics. Trajectory chunks are then interpolated between those key-frames using B-spline curves with a fixed duration $T_{\mathsf{ref}} = 5.9$ s. Foot contacts are maintained in the interpolated trajectories using the method of [4] for closure of kinematic chains. Figure 2 shows a timelapse of the input motion.

We used the TOPP solver [6] to retime this trajectory. We defined the support area for the ZMP constraints as a disc centered on the COM with radius 1 cm, resulting in conservative balance. The friction coefficient was set to 0.8 while the normal component of contact forces was enforced to a minimum 10 N (by changing the right-hand side of Equation (2)). Actuated torque limits were set to 50% of the robot's

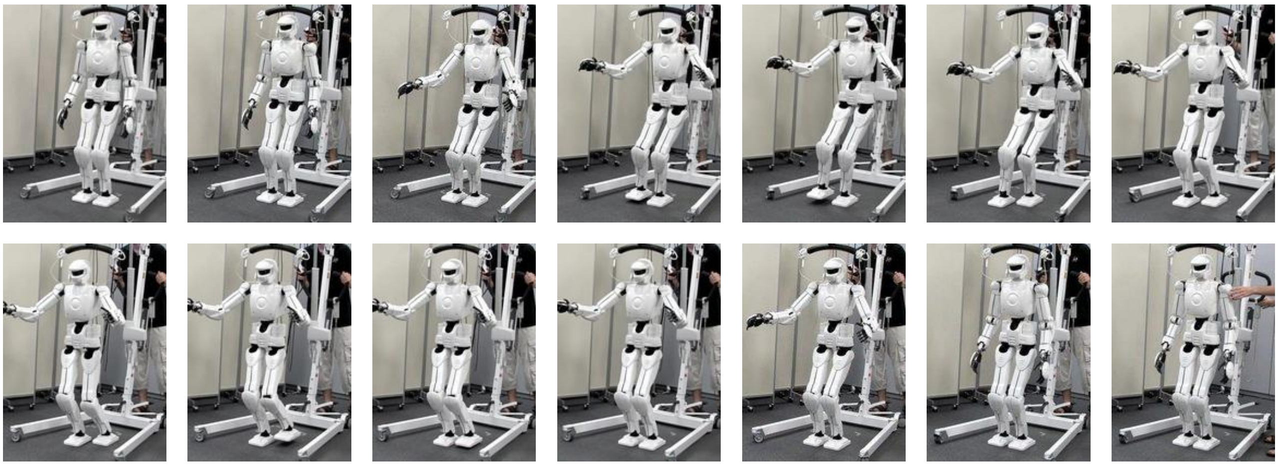

Fig.2 Timelapse of the original trajectory. The interval between two frames is 3.5 s (total duration: 53 s).

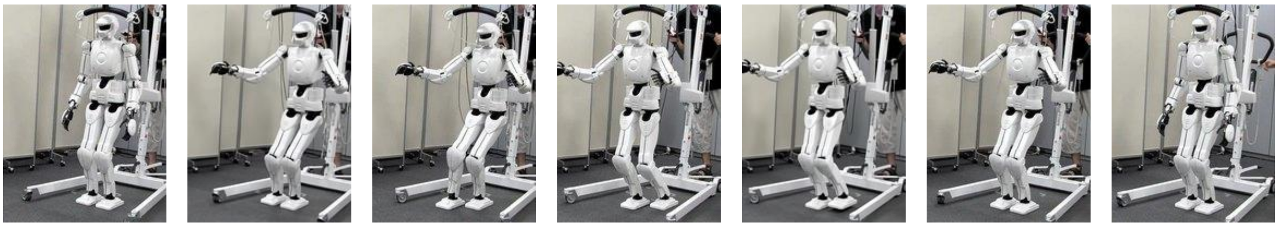

Fig.3 Timelapse of the retimed trajectory. The interval between two frames is 3.5 s (total duration: 24 s).

limits. Figure 3 shows a timelapse of the retimed motion after application of TOPP. The duration of the retimed trajectory is 24 s, *i.e.*, $2\times$ faster than its input.

We ran simulations with more aggressive values of the parameters in OpenHRP using full actuated torque limits and a 5 cm radius around the COM for the ZMP. Figure 1 shows the retimed trajectory profile and MVCs thus obtained. The most constrained part of the problem occurs when the COM gets close to the limits of the support area, which saturates the contact constraints. The duration of the retimed trajectory in simulation is 12.3 s, more than $4\times$ faster than its input.

Note that, for now, we do not retime the complete trajectory as a whole because of the discontinuities in $(\mathbf{a}(s), \mathbf{b}(s), \mathbf{c}(s))$ that happen when switching between single and double support. Rather, we set small time intervals around these transitions during which retiming was disabled (grey areas in Figure 1). The duration of these intervals was set to 1.5 s on the real robot and 0.2 s in the OpenHRP experiment.

5. Conclusion

In the present paper, we advocated the use of Time-Optimal Path Parameterization to enable planning of *dynamic* motions for humanoid robots. We extended the existing formulation of ZMP constraints to arbitrary polygonal areas and provided an original approach to incorporate frictional contact constraints in TOPP. We evaluated our solution experimentally on the HRP-4 platform.

References

[1] James E Bobrow, Steven Dubowsky, and JS Gibson. Time-optimal control of robotic manipulators along specified paths. *The International Journal of Robotics Research*, 4(3):3–17, 1985.

[2] Kris Hauser. Fast interpolation and time-optimization on implicit contact submanifolds. In *Robotics: Science and Systems*. Citeseer, 2013.

[3] Kris Hauser. Fast dynamic optimization of robot paths under actuator limits and frictional contact. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014.

[4] Yoshihiko Nakamura and Katsu Yamane. Dynamics computation of structure-varying kinematic chains and its application to human figures. *Robotics and Automation, IEEE Transactions on*, 16(2):124–134, 2000.

[5] Quang-Cuong Pham. Characterizing and addressing dynamic singularities in the time-optimal path parameterization algorithm. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013.

[6] Quang-Cuong Pham. A general, fast, and robust implementation of the time-optimal path parameterization algorithm. *arXiv preprint arXiv:1312.6533*, 2013.

[7] Quang-Cuong Pham and Yoshihiko Nakamura. Time-optimal path parameterization for critically dynamic motions of humanoid robots. In *IEEE-RAS International Conference on Humanoid Robots*, 2012.

[8] Zvi Shiller and Lu Hsueh-Hen. Computation of path constrained time optimal motions with dynamic singularities. *Journal of dynamic systems, measurement, and control*, 114(1):34–40, 1992.

[9] Kang G Shin and Neil D McKay. Minimum-time control of robotic manipulators with geometric path constraints. *Automatic Control, IEEE Transactions on*, 30(6):531–541, 1985.

[10] J-JE Slotine and Hyun S Yang. Improving the efficiency of time-optimal path-following algorithms. *Robotics and Automation, IEEE Transactions on*, 5(1):118–124, 1989.

[11] Wael Suleiman, Fumio Kanehiro, Eiichi Yoshida, J-P Laumond, and André Monin. Time parameterization of humanoid-robot paths. *Robotics, IEEE Transactions on*, 26(3):458–468, 2010.

[12] Diederik Verscheure, Bram Demeulenaere, Jan Swevers, Joris De Schutter, and Moritz Diehl. Time-optimal path tracking for robots: A convex optimization approach. *Automatic Control, IEEE Transactions on*, 54(10):2318–2327, 2009.