

On Humanoid Motion Optimization

Wael Suleiman, Eiichi Yoshida, Jean-Paul Laumond, André Monin

▶ **To cite this version:**

Wael Suleiman, Eiichi Yoshida, Jean-Paul Laumond, André Monin. On Humanoid Motion Optimization. IEEE-RAS 7th International Conference on Humanoid Robots, Nov 2007, Pittsburgh, Pennsylvania, United States. pp.CD ROM. hal-00164004v3

HAL Id: hal-00164004
https://hal.science/hal-00164004v3

Submitted on 29 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On Humanoid Motion Optimization

Wael Suleiman*, Eiichi Yoshida†, Jean-Paul Laumond* and André Monin*
IS/AIST-ST2I/CNRS Joint Japanese-French Robotics Laboratory (JRL)
* LAAS - CNRS, university of Toulouse
7 Avenue du Colonel Roche, 31077 Toulouse, France
{suleiman, jpl, monin}@laas.fr
† Intelligent Systems Research Institute, National Institute
of Advanced Industrial Science and Technology (AIST)
AIST Central 2, Umezono 1-1-1, Tsukuba, 305-8568 Japan
{e.yoshida}@aist.go.jp

***Abstract*—In this paper, we present a recursive method for the optimization of humanoid robot motions. The method is based on an efficient dynamics algorithm, which allows the calculation of the gradient function with respect to the control parameters analytically. The algorithm makes use of the theory of Lie groups and Lie algebra. The main objective of this method is to smooth the pre-calculated humanoid motions by minimizing the efforts, and at the same time improving the stability of the humanoid robot during the execution of the planned tasks. Experimental results using HRP-2 platform are provided to validate the proposed method.**

***Index Terms*—Humanoid robot; Motion planning; Recursive dynamics; Optimization**

I. Introduction

Few years ago, talking about humanoid robots was some-kind of science fiction. The recent technological advancement has made this dream a reality. Actually the ability of humanoid robots to execute complex tasks increases rapidly.

The latest trends in humanoid research are to increase their autonomous behavior as well as improving the stability and smoothness of the planned motions.

Optimizing motions to improve their performance was an active research subject in recent years. In virtual reality, Lo and Metaxas [1] have proposed a method based on optimal control theory within a recursive dynamics framework. The objective of their work is to simulate a dynamically-correct astronaut motions by minimizing joint torques.

In humanoid research, Ruchanurucks *et al* [2] have proposed a method to optimize upper body motion of humanoid robot in order to imitate a recored human motion. Their optimization objective function preserves the salient characteristics of the original motion, and at the same time it respects the physical constraints of the humanoid robot. However, the notion of minimizing an energy function is not introduced because the objective is only to guarantee the feasibility of the motions. Furthermore, the authors have mentioned that the resulting trajectories would meet the latter limits while the former limits are often violated. This is because their method consider the velocity and force constraints separately,

Safonova *et al* [3] use also a pre-recorded human motion to generate optimal motion of the upper body of Sarcos humanoid robot. The objective is also to imitate human motions, so the function to be minimized is the difference between the recorded and the executed motion by the robot. Guilamo *et al* [4] consider the optimization of manipulability trajectories. The optimal solution is in kinematic sense and does not take into account the dynamic constraints.

Sentis and Khatib [5] have proposed a whole-body control framework for humanoids. This framework integrates task-oriented dynamic control while complying with humanoid physical constraints. The controllers are calculated in the operational space [6] at multiple levels. As it is known, these controllers provide the torques which should be applied on each joint, that means the humanoid robot should be controlled by joint torques. On the contrary, HRP-2 platform and many humanoid robots are controlled by joint positions. Although the use of inverse dynamics methods can give the associated joint positions with the calculated joint torques, this method is time consuming and is not numerically efficient.

In this paper, our objective is to smooth pre-calcuted humanoid motions. These motions can be provided by kinematic planning methods, which can take into account the limits of joint angles. On the other hand, they can not guarantee that the calculated motions do not violate the torques limits.

Studying human movements [7], [8] has brought out a connection between minimizing energy dissipation and forces, and the smoothness of human movements. On account of the complexity of calculating the energy dissipation, one can use an approximative prediction of it. A good predictor of human's metabolic energy is proven to be the joint torques [9]. As the humanoid robot is supposed to realize human-like motions, our goal can be achieved by minimizing the joints torques during the planned motion.

The contributions of this paper are:

- Developing an optimization framework for humanoid robot motions. This framework takes as input a pre-calculated motions, which are provided by motion planning techniques. The output is an optimized and stable motions.
- The proposed method uses the forward dynamic formulation and the quantities to be optimized are the joint positions. As a consequence, the humanoid robot can be controlled directly in the joint space and not in torque

control space. Therefore the method is well adapted for a position controlled humanoid like HRP-2 platform.
- By using an efficient dynamic algorithm, we can calculate the derivative of joint torques with respect to joint position, velocity and acceleration $(q, \dot{q}, \ddot{q})$ analytically. This procedure is similar to the procedure used in [1]. However, the problem of virtual human stability and ground reaction forces modeling are not considered in [1].
- The validation of the proposed method pointed out that the optimized motion is smoother than the pre-calculted one.

The remainder of this paper is organized as follows. In Section II the kinematic structure of HRP-2 humanoid robot, and the definition of active and passive parts are introduced. An overview of the algorithm of recursive multibody dynamics is given in Section III. In Section IV the optimization problem is formulated. In Section V the discretization of configuration space and solving the optimization problem are pointed out. Experimental results are given in Section VI and Section VII concludes the paper.

II. Humanoid robot: kinematic structure

The kinematic structure of the humanoid robot HRP-2 [10] is given in the Fig. 1. In this structure the degree of freedoms are presented by cylinders. The structure contains 30 degree of freedoms.

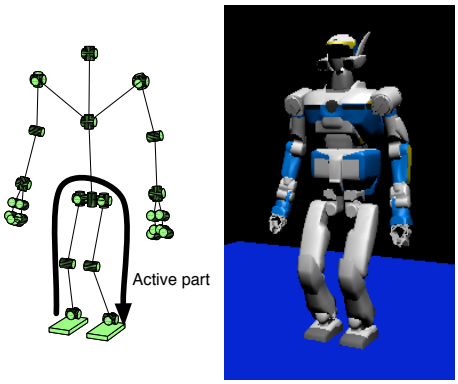


Fig. 1. Description of HRP2 kinematic structure.

Our objective is to optimize pre-calculated humanoid motions and improve their stability. The pre-calculated motions are the output of motion planning algorithms [11], [12]. These motions are generally complex, e.g. transporting an object and guaranteeing the collision avoidance [13]. Such motions usually use the upper part of humanoid robot to execute the desired task and the lower part is mainly responsible of the locomotion and maintaining the stability of humanoid robot.

For that, we divide the kinematic structure into two parts. The first part is the active part, which is the lower part. The upper body is regarded as passive part.
In order to achieve our objective, we optimize only the values of the degree of freedoms of the active part. The passive part will be taken into account in the calculation of the dynamic equations without modifying their angular values. The active part consists of 12 degree of freedoms.

III. Recursive multibody dynamics

Park *et al* [14] have proposed to write the recursive multibody dynamics for serial open or branched kinematic chains using Lie group and Lie algebra. The main advantage of this formulation is to relate the joint torques and joint angles explicitly. Therefore the differentiation of joint torques with respect to joint angles can be done analytically.

Let us define the Lie groups $SO(3)$ and $SE(3)$, which denote the orthonormal matrix Θ in $R^{3\times 3}$ and the homogeneous transformation group respectively. The Lie algebra of $SO(3)$ and $SE(3)$ are denoted $so(3)$ and $se(3)$ respectively. The operators defined on these groups are: skew, matrix exponential, adjoint map $Ad_G(.)$, dual adjoint $Ad^*_G(.)$, Lie bracket $ad_g(.)$ and dual Lie bracket $ad^*_g(.)$. For more details on Lie group, Lie algebra and the operators definitions see Appendix.

A. Forward Kinematics

The kinematics of an open chain can be modeled as a sequence of homogeneous transformation between consecutive joint frames. Let $T_{i-1,i} \in SE(3)$ be the transformation matrix between the frame of link i and the frame of link $i-1$.
The matrix $T_{i-1,i}$ can be written using matrix exponential notation as follows

$$T_{i-1,i} = M_i e^{S_i q_i} \tag{1}$$

where $S_i \in se(3)$ is the joint screw written in the coordinate of link $i-1$, q_i is the current position of joint i and M_i is the coordinate transformation between link i and link $i-1$.
Using the above definition of transformation matrix, the end-effector of a kinematic chain can be calculated by the product

$$\begin{aligned} T_{0,n} &= T_{0,1} T_{1,2} \cdots T_{n-1,n} \\ &= M_1 e^{S_1 q_1} M_2 e^{S_2 q_2} \cdots M_n e^{S_n q_n} \end{aligned} \tag{2}$$

Note that by expressing the matrix of transformation in exponential form, we can calculate its derivative with respect to q_i analytically.

B. Recursive inverse dynamics of branched chains

Branched chains are serial open chains with two or more branches leading to two or more tip links [14], [15]. In the branched chains two definitions arise :
- *Parent link*: the link inward (towards the base) from a given link.
- *Child link*: the link or links which are outward (towards the tips) from a given link.

Spatial velocity of branched chains:
- **Initialization**: Given V_0.
- **Outward recursion**: loop over all links in depth manner:

$$\begin{aligned} T_{P,i} &= M_i e^{S_i q_i} \\ V_i &= Ad_{T^{-1}_{P,i}}(V_P) + S_i \dot{q}_i \\ a_i &= -ad_{S_i \dot{q}_i}(V_i) \\ b_i &= -ad^*_{V_i}(J_i V_i) \end{aligned} \tag{3}$$

where the index P denotes the parent link of link i, $T_{P,i}$ designs the mapping from the link i to its parent P and V_P denotes the spatial velocity of link P.

Applied torques on the branched chains: In order to calculate the inward recursion of forces and torques, we define the external forces applied on a link j by $\hat{F}_j$.

- **Initialization:** Given the external applied forces on each link $\hat{F}_j$, $\dot{V}_0$ and $\hat{J}_j = 0$ for each tip link.
- **Inward recursion**: loop over all links in reversed breadth

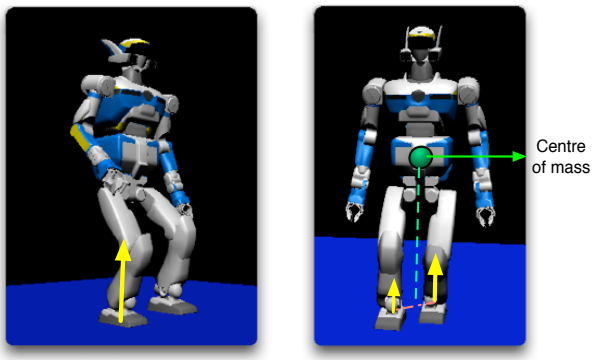
$$\begin{aligned}
\dot{V}_i &= Ad_{T_{P,i}^{-1}}\left(\dot{V}_P\right) + S_i\ddot{q}_i + a_i \\
\hat{J}_i &= J_i + \sum_{j\in C} Ad^*_{T_{i,j}^{-1}} \hat{J}_j Ad_{T_{i,j}^{-1}} \\
B_i &= b_i + \sum_{j\in C} Ad^*_{T_{i,j}^{-1}} z_j \\
z_i &= \hat{J}_i\left(S_i\ddot{q}_i + a_i\right) + B_i + \sum_{j\in C} Ad^*_{T_{i,j}^{-1}} \hat{F}_j \\
F_i &= \hat{J}_i\, Ad_{T_{P,i}^{-1}}\left(\dot{V}_P\right) + z_i \\
\tau_i &= S_i^T F_i
\end{aligned} \tag{4}$$

where C denotes the child links for link i, and τ_i is the torque applied on the joint i.
The matrix J_i is called the spatial inertia and it is defined as follows

$$J_i = \begin{bmatrix} I_i - m_i[r_i]^2 & m[r_i] \\ -m[r_i] & m.\mathbf{1} \end{bmatrix} \tag{5}$$

where I_i is the inertia of the link i about its centre of mass and m is its mass. r_i is the vector from the point of application of the force and the centre of mass of the link i. Recall that $[r_i]$ is the skew operator see Appendix for more details.

C. Ground reaction forces



(a) Single support. (b) Double support.

Fig. 2. Ground reaction forces represented by the yellow arrows.

In the absence of external forces applied on the robot, the only forces are the ground reactions forces. In order to estimate the ground reaction forces, we distinguish two cases

1) Single support: The ground reaction force is applied on the support foot and its magnitude is equal to $(Mg - a_G)$, where M is the total mass of the humanoid robot, g is the gravitational acceleration, and a_G is the acceleration of the humanoid robot centre of mass.
2) Double support: The ground reaction forces are applied on the two feet. Their magnitudes are proportional to the distance from the projection of the center of mass on the floor and the centers of the support feet.

A demonstration of the ground reaction forces in single and double support is shown in Fig. 2.

IV. Optimization problem formulation

The cost function to be minimized is the integral of the Euclidean norm of joint torques.The stability of the humanoid robot can be assured by guaranteeing that the foot, which is in contact with the ground, will be immobile. Such condition can be satisfied by assuring that the spacial velocity and acceleration of the support foot are null.

Let $\mathscr{Q}_t$ be the vector of angular values of joints in the configuration space defined as follows

$$\begin{aligned}
\mathscr{Q}_t &= \begin{bmatrix} \mathbf{q_t} \\ \mathbf{q_t^p} \end{bmatrix} \\
&= \begin{bmatrix} q_{1,t} & q_{2,t} & \cdots & q_{n,t} & q^p_{1,t} & q^p_{2,t} & \cdots & q^p_{n_p,t} \end{bmatrix}^T
\end{aligned} \tag{6}$$

where $\mathbf{q_t}$ and $\mathbf{q_t^p}$ denote the vectors of angular values of active and passive joints respectively.
The optimization problem can be formulated as follows

$$\min_{\mathbf{q_t},\dot{\mathbf{q}}_\mathbf{t},\ddot{\mathbf{q}}_\mathbf{t}} \int_{t_0}^{t_f} {\tau_t}^T \tau_t dt$$

subject to

$$\begin{aligned}
&\text{Permanent constraints} \begin{cases} S^T F_t = \tau_t \\ \mathbf{q_{t_0}} = \mathbf{q_0}, \dot{\mathbf{q}}_{\mathbf{t_0}} = \mathbf{0}, \ddot{\mathbf{q}}_{\mathbf{t_0}} = \mathbf{0} \\ \mathbf{q_{t_f}} = \mathbf{q_f}, \dot{\mathbf{q}}_{\mathbf{t_f}} = \mathbf{0}, \ddot{\mathbf{q}}_{\mathbf{t_f}} = \mathbf{0} \\ \tau^- \le \tau_t \le \tau^+ \\ \mathbf{q}^- \le \mathbf{q_t} \le \mathbf{q}^+ \end{cases} \\
&\text{Temporary constraints} \begin{cases} V_{support\ foot} = 0 \\ \dot{V}_{support\ foot} = 0 \\ \mathscr{G}_{foot} = \mathscr{G}^{ref}_{foot} : \mathrm{t_{c1}}, \cdots, \mathrm{t_{cp}} \end{cases}
\end{aligned} \tag{7}$$

where τ_t, F_t and S are defined as follows

$$\tau_t = \begin{bmatrix} \tau_{1,t} \\ \tau_{2,t} \\ \vdots \\ \tau_{n,t} \end{bmatrix}, F_t = \begin{bmatrix} F_{1,t} \\ F_{2,t} \\ \vdots \\ F_{n,t} \end{bmatrix}, S = \begin{bmatrix} S_1 & 0 & \cdots & 0 \\ 0 & S_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & S_n \end{bmatrix} \tag{8}$$

τ_t and F_t are the vectors of the applied torques and forces on the joints of active part respectively. $\tau_{i,t}$ and $F_{i,t}$ denote the value of the applied torque and force on the joint i respectively. x^- and x^+ denote the minimal and the maximal values of vector x respectively.
$\mathscr{G}_{foot}$ denotes the configuration of the foot in the euclidean space as shown in Fig. 3

$$\mathscr{G}_{foot} = \begin{bmatrix} Pos_{front} \\ Pos_{back} \end{bmatrix} \tag{9}$$

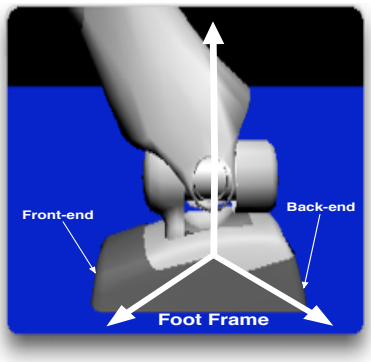


Fig. 3. Front and back-ends of the support foot.

where Pos_{front} and Pos_{back} are the positions of the front and back ends of the foot respectively, and $\mathscr{G}_{foot}^{ref}$ is the planned reference configuration. t_{c^k} denotes the instant of foot contact with the ground. Note that assuring $\mathscr{G}_{foot}$ guarantees the orientation of the rigid foot as well.
Using Eq. (2), the $\mathscr{G}_{foot}$ can be written as a function of $\mathbf{q_t}$ as follows

$$\mathscr{G}_{foot} = \begin{bmatrix} T_{0,foot} \begin{bmatrix} p_{front} \\ 1 \end{bmatrix} \\ T_{0,foot} \begin{bmatrix} p_{back} \\ 1 \end{bmatrix} \end{bmatrix} \tag{10}$$

where $T_{0,foot} = M_1 e^{S_1 q_1} M_2 e^{S_2 q_2} \cdots M_n e^{S_n q_n}$ denotes the transformation matrix of the foot expressed in the global fixed frame, and q_1 is the angular value of the ankle associated to the immobile support foot. p_{front} and p_{back} denote the positions of the front and back ends of the foot in its frame.

In order to transform this optimization problem into a classical optimization problem, let us define

$$X_t = \begin{bmatrix} \mathbf{q_t}^T & \dot{\mathbf{q}}_\mathbf{t}^T & \ddot{\mathbf{q}}_\mathbf{t}^T \end{bmatrix}^T, \quad L(X_t) = \int_{t_0}^{t_f} {\tau_t}^T \tau_t dt$$

$$G(X_t) = \begin{bmatrix} \tau_t - \tau^+ \\ -\tau_t + \tau^- \\ \mathbf{q_t} - \mathbf{q}^+ \\ -\mathbf{q_t} + \mathbf{q}^- \end{bmatrix}, \quad H(X_t) = \begin{bmatrix} \tau_t - S^T F_t \\ \mathbf{q_{t_0}} - \mathbf{q_0} \\ \dot{\mathbf{q}}_{\mathbf{t_0}} \\ \ddot{\mathbf{q}}_{\mathbf{t_0}} \\ \mathbf{q_{t_f}} - \mathbf{q_f} \\ \dot{\mathbf{q}}_{\mathbf{t_f}} \\ \ddot{\mathbf{q}}_{\mathbf{t_f}} \\ V_{support\ \ foot} \\ \dot{V}_{support\ \ foot} \\ \mathscr{G}_{foot} - \mathscr{G}_{foot}^{ref} \end{bmatrix} \tag{11}$$

Thus the optimization problem (7) can be transformed to the following classical form

$$\begin{aligned} &\min_{X_t} \ L(X_t) \\ &\text{subject to} \\ &\qquad H(X_t) = 0 \\ &\qquad G(X_t) \leq 0 \end{aligned} \tag{12}$$

The above optimization problem has been extremely studied in the literature of optimization theory. To solve this optimization problem, one can use the augmented Lagrange multiplier method, which is a very efficient and reliable method [16], [1]. Using the augmented Lagrange multiplier method transforms the optimization problem (12) to the minimization of the following function

$$\min_{X_t,\lambda} \tilde{L}(X_t,\lambda) = L(X_t) + \lambda_\psi^T \psi + \frac{1}{2}\sigma \psi^T \psi + \lambda_H^T H + \frac{1}{2}\sigma H^T H \tag{13}$$

where $\lambda = \begin{bmatrix} \lambda_\psi^T & \lambda_H^T \end{bmatrix}^T$, $\psi = \max\left\{G(X_t), \frac{-1}{\sigma}\lambda_\psi\right\}$. Then there exist λ^* such that X_t^* is an unconstrained local minimum of $\tilde{L}(X_t,\lambda^*)$ for all σ smaller than some finite $\bar{\sigma}$.
To solve the unconstrained optimization problem of $\tilde{L}(X_t,\lambda)$ with respect to X_t, one can use Gauss-Newton method. Note that the function $\tilde{L}(X_t,\lambda)$ is differentiable in X_t if and only if $L(X_t)$, $H(X_t)$ and $G(X_t)$ are differentiable in X_t, and in this case we can write

$$\begin{aligned} \frac{\partial \tilde{L}(X_t,\lambda)}{\partial X_t} = &\frac{\partial L(X_t)}{\partial X_t} + (\lambda_H + \sigma H)^T \frac{\partial H(X_t)}{\partial X_t} + \\ &\max\left\{0, \lambda_\psi + \sigma G(X_t)\right\}^T \frac{\partial G(X_t)}{\partial X_t} \end{aligned} \tag{14}$$

As λ^* is unknown, an update rule is used

$$\begin{aligned} \lambda_H^{k+1} &= \lambda_H^k + \sigma H(X_t^k) \\ \lambda_\psi^{k+1} &= \lambda_\psi^k + \sigma \psi(X_t^k) \end{aligned} \tag{15}$$

where X_t^k is the unconstrained minimum of $\tilde{L}(X_t,\lambda^k)$. Such updating rule will generate a sequence λ^k converges to λ^* [17]. In practice, a good schedule is to choose a moderate σ^0, and increase it as follows

$$\sigma^{k+1} = \alpha \sigma^k \tag{16}$$

where α is between 5 and 10. A threshold $\bar{\sigma}$ is chosen and the update rule of σ stops when σ^k becomes higher than $\bar{\sigma}$.

For more details on the algorithm of augmented Lagrange multiplier method see [18], [16], [17].

Approximating the gradient function $\frac{\partial \tilde{L}(X_t,\lambda)}{\partial X_t}$ by a numerical difference method is usually used in practice. However, this approach not only a time consuming method on account of the evaluation of the gradient calculation, but also may not converge well because of the approximation.

As we have mentioned the main advantage of using the recursive dynamic algorithm explained in Section III-B is calculating the gradient function analytically in a recursive way.

A. Gradient calculation

The objective is to calculate the gradient of the dynamic quantities, such as τ_t, V_t and $\dot{V}_t$.
By considering the vector of parameters $X_t = \begin{bmatrix}\mathbf{q_t}^T & \dot{\mathbf{q}}^T & \ddot{\mathbf{q}}^T\end{bmatrix}^T$, let us start by calculating the derivatives of the operators with respect to an element x of X_t

$$\begin{aligned}
\frac{\partial T_{0,n}}{\partial x} =& T_{0,i}\left(S_i\delta_{x,q_i}\right)T_{i,n}\\
\frac{\partial Ad_{T_{i-1,i}^{-1}}(Y)}{\partial x} =& ad_{Ad_{T_{i-1,i}^{-1}}(Y)}\left(S_i\delta_{x,q_i}\right)+Ad_{T_{i-1,i}^{-1}}\left(\frac{\partial Y}{\partial x}\right)\\
\frac{\partial Ad^*_{T_{i,i+1}^{-1}}(Y)}{\partial x} =& ad^*_{Ad_{M_{i+1}}\left(S_{i+1}\delta_{x,q_{i+1}}\right)}\left(Ad^*_{T_{i,i+1}^{-1}}(Y)\right)+\cdots\\
& Ad^*_{T_{i,i+1}^{-1}}\left(\frac{\partial Y}{\partial x}\right)\\
\frac{\partial ad_Z(Y)}{\partial x} =& ad_{\frac{\partial Z}{\partial x}}(Y)+ad_Z\left(\frac{\partial Y}{\partial x}\right)\\
\frac{\partial ad^*_Z(Y)}{\partial x} =& ad^*_{\frac{\partial Z}{\partial x}}(Y)+ad^*_Z\left(\frac{\partial Y}{\partial x}\right)
\end{aligned} \tag{17}$$

where δ_{x_1,x_2} is the Kronecker's delta function defined as follows

$$\delta_{x_1,x_2} = \begin{cases} 1 & \text{if } x_1 = x_2\\ 0 & \text{otherwise}\end{cases} \tag{18}$$

The calculation of the gradient with respect to X_t can be done in a recursive way analogously to the recursive dynamic calculation.
Forward recursion:

- **Initialization**: Given $\frac{\partial V_0}{\partial X_t}$.
- loop over all links in depth manner:

$$\begin{aligned}
\frac{\partial V_i}{\partial X_t} &= \frac{\partial Ad_{T_{P,i}^{-1}}(V_P)}{\partial X_t}+S_i\frac{\partial \dot{q}_i}{\partial X_t}\\
\frac{\partial a_i}{\partial X_t} &= -\frac{\partial ad_{S_i\dot{q}_i}(V_i)}{\partial X_t}\\
\frac{\partial b_i}{\partial X_t} &= -\frac{\partial ad^*_{V_i}(J_iV_i)}{\partial X_t}
\end{aligned} \tag{19}$$

Backward recursion:

- **Initialization:** Given $\frac{\partial \hat{F}_j}{\partial X_t}$, $\frac{\partial \dot{V}_0}{\partial X_t}$.
- loop over all links in reversed breadth

$$\begin{aligned}
\frac{\partial \dot{V}_i}{\partial X_t} =& \frac{\partial Ad_{T_{P,i}^{-1}}(\dot{V}_P)}{\partial X_t}+S_i\frac{\partial \ddot{q}_i}{\partial X_t}+\frac{\partial a_i}{\partial X_t}\\
\frac{\partial \hat{f}_i}{\partial X} =& \sum_{j\in C}\frac{\partial Ad^*_{T_{i,j}^{-1}}}{\partial X_t}\hat{f}_jAd_{T_{i,j}^{-1}}+\cdots\\
& Ad^*_{T_{i,j}^{-1}}\hat{f}_j\frac{\partial Ad^*_{T_{i,j}^{-1}}}{\partial X_t}+Ad^*_{T_{i,j}^{-1}}\frac{\partial \hat{f}_j}{\partial X_t}Ad_{T_{i,j}^{-1}}\\
\frac{\partial B_i}{\partial X_t} =& \frac{\partial b_i}{\partial X_t}+\sum_{j\in C}\frac{\partial Ad^*_{T_{i,j}^{-1}}z_j}{\partial X_t}\\
\frac{\partial z_i}{\partial X_t} =& \frac{\partial \hat{f}_i}{\partial X_t}\left(S_i\ddot{q}_i+a_i\right)+\hat{f}_i\left(S_i\frac{\partial \ddot{q}_i}{\partial X_t}+\frac{\partial a_i}{\partial X_t}\right)+\cdots\\
& \frac{\partial B_i}{\partial X_t}+\sum_{j\in C}\frac{\partial Ad^*_{T_{i,j}^{-1}}\hat{F}_j}{\partial X_t}\\
\frac{\partial F_i}{\partial X_t} =& \frac{\partial \hat{f}_i}{\partial X_t}Ad_{T_{P,i}^{-1}}(\dot{V}_P)+\hat{f}_i\frac{\partial Ad_{T_{P,i}^{-1}}(\dot{V}_P)}{\partial X_t}+\frac{\partial z_i}{\partial X_t}\\
\frac{\partial \tau_i}{\partial X_t} =& S_i^T\frac{\partial F_i}{\partial X_t}
\end{aligned} \tag{20}$$

where as we mentioned, earlier, C denotes the child links for link i.

V. Discretization of configuration space

It is well known that the space of the admissible solutions of the minimization problem (7) is very large. In order to transform this infinite dimensional space to a finite one, we can use a basis of shape functions.
Let us consider a basis of shape functions B_t that is defined as follows

$$B_t = \begin{bmatrix}B_t^1 & B_t^2 & \cdots & B_t^l\end{bmatrix}^T \tag{21}$$

where B_t^i denotes the value of shape function number i at the instant t, the dimension of B_t is l defines the dimension of the basis of shape functions.

The projection of the vector of angular values $\mathbf{q_t}$ into the basis of shape functions B_t can be given by the following formula

$$\mathbf{q_t} = Q_BB_t \tag{22}$$

where Q_B is a constant matrix.
The derivative $\dot{\mathbf{q}}_\mathbf{t}$ and $\ddot{\mathbf{q}}_\mathbf{t}$ can be written as follows

$$\begin{aligned}
\dot{\mathbf{q}}_\mathbf{t} =& Q_B\dot{B}_t\\
\ddot{\mathbf{q}}_\mathbf{t} =& Q_B\ddot{B}_t
\end{aligned} \tag{23}$$

In this case, the derivative with respect to each element $Q_B(i,j)$ of the matrix Q_B can be computed using the following formula

$$\begin{aligned}
\frac{\partial Y_t}{\partial Q_B(i,j)} &= \frac{\partial Y_t}{\partial X_t}\times\frac{\partial X_t}{\partial Q_B(i,j)}\\
&= \frac{\partial Y}{\partial X_t}\times\left(e_i\otimes\begin{bmatrix}B_t^j\\ \dot{B}_t^j\\ \ddot{B}_t^j\end{bmatrix}\right)
\end{aligned} \tag{24}$$

where $e_i \in \mathscr{R}^n$,

$$e_i = \begin{matrix}[0 & \dots & 0 & \underset{\uparrow}{1} \\ & & & i \end{matrix} \quad 0 \quad \dots \quad 0]^T$$

and $\otimes$ denotes Kronecker's product operator.

By using the discretization of the configuration space, the optimization problem transforms into the problem of finding the optimal matrix Q_B, which minimizes the function $\tilde{L}(X_t, \lambda)$ in Eq. (13).

It remains to define the shape functions B_t^i. In our case, the shape functions should verify the following properties:

1) They are continuous.
2) Their first and second derivatives are continuous.

Therefore, we use the *quartic B-spline functions*.

VI. Experimental results

The experimental scenario that we have tested to validate the proposed method is the following

1) The robot carries a bar with its right hand.
2) The robot starts walking and depressing the vertical position of its pelvis. At the same time, it lifts up the carried bar.

The characteristics of the carried bar are: length= 2 m, weight= 0.7 Kg, cylindric form with uniform density distribution. The robot grasps the bar at 0.35 m from its end.

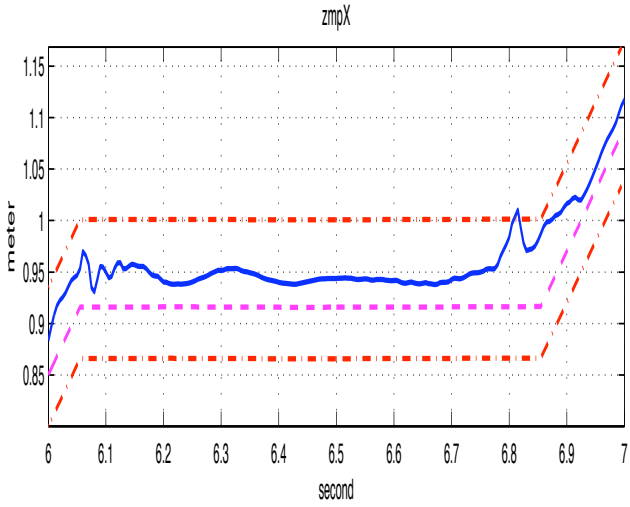
Snapshots of the conducted motion[1] are presented in Fig. 4. Note that the objective of using the carried bar is to generate a non-symmetry in the kinematic structure, and also to perturb dynamically the motion by the movement of the bar.

To compare the obtained results with the results obtained by the method presented in [19], we use the real quantities measured by the sensors of the humanoid robot HRP-2.
Fig. 5 shows the x coordinate of the Zero Moment Point (ZMP). The ground reaction force applied on the left foot are presented in Fig. 6. Fig. 7 shows the applied torque on the left knee.

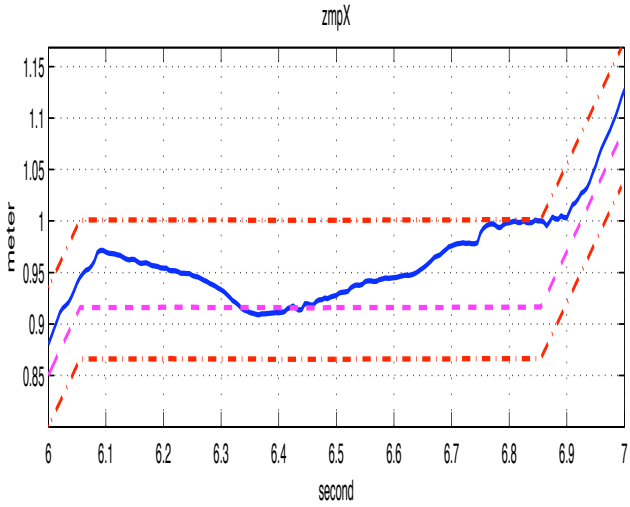
From Figures 5, 6 and 7 we conclude that

1) The optimization method smoothes the shape of ZMP, and the oscillations have been avoided. Note that the ZMP trajectory of the optimized motion does not follow the designed ZMP trajectory. This is because the optimization method assures the stability of humanoid robot by guaranteeing that the spacial velocity and acceleration of the support foot are null, and it does not consider the trajectory of ZMP.
2) Using the optimization method avoids the surges in the ground reaction forces applied on the foot.
3) Using the optimization method not only minimizes the joint torques, but also it avoids the surges.

[1]Videos of experimental results are available at http://www.laas.fr/~suleiman/Humanoids07/index.htm



(a) Before optimization.



(b) After Optimization.

Fig. 5. x coordinate of ZMP: the solid blue line is the measured ZMP_x, the red dash-dotted lines design the safe stability zone and the magenta dashed line denotes the designed reference. The oscillations around 6.1 and 6.8 seconds disappear after optimization as well as the surge at 6.81 second.

VII. Conclusion

In this paper, we have presented an optimization method for humanoid robot motions. The objective of this method is to smooth and improve the stability of the humanoid motions. To achieve this goal, the integral of Euclidean norm of the applied torques on the joints is minimized, and the dynamical stability conditions are transformed into guaranteeing that the spatial velocity and accelerations of the support foot are null.

The experimental results have pointed out that the proposed method not only smoothes the motion but also yields a dynamically stable motion. Future work will focus on the improvement of the proposed method to take into account all degree of freedoms of the humanoid robot. In this case deforming the planned trajectory is required, therefore a



Fig. 4. Snapshots of the conducted motion.

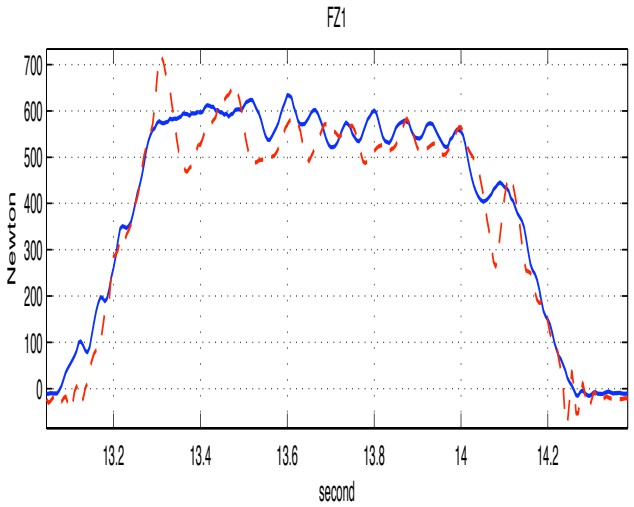


Fig. 6. Applied ground reaction forces on the left foot: solid line for the optimization method and dashed line for the classical method. The surge at 13.3 second disappears after optimization.

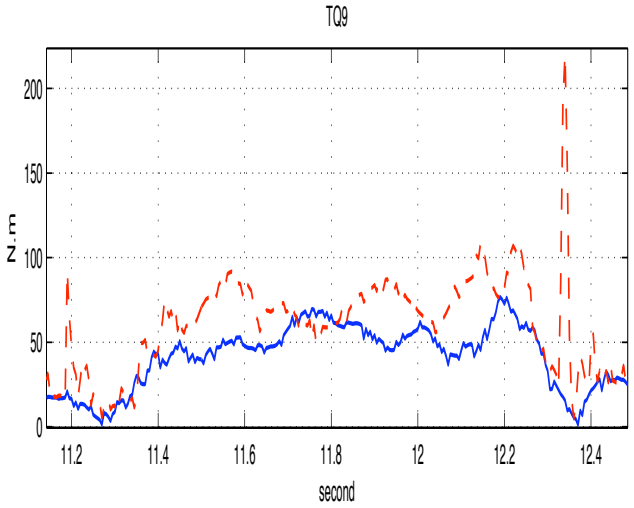


Fig. 7. Applied torques on the left knee: solid line for the optimization method and dashed line for the classical method. The surges at 11.2 and 12.35 seconds disappear after optimization.

function to calculate the distance between the robot and the obstacles should be developed and considered as an additional constraint.

APPENDIX

A *Lie group* is a differentiable manifold. An example of Lie group is the orthonormal matrix Θ in $R^{3\times 3}$, which is called $SO(3)$. Note that this group consists of the rotation matrices in Euclidean space. Another example of Lie group is the group of homogeneous transformation which is the special Euclidean group or $SE(3)$. Given a rotation $\Theta \in SO(3)$ and translation $b \in R^3$, the homogeneous matrix is defined as follows

$$G = \begin{bmatrix} \Theta & b \\ 0 & 1 \end{bmatrix} \tag{A-1}$$

An important concept associated with each Lie group is the notation of *Lie algebra*. The tangent space at the identity element of a Lie group is called the Lie algebra for that group. The Lie algebra of $SO(3)$ and $SE(3)$ are denoted $so(3)$ and $se(3)$ respectively.
Let us define some notations and operations on Lie groups and Lie algebra:

1) Skew operator:

$$[.] : \omega \in R^3 \rightarrow so(3)$$
$$[\omega] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{A-2}$$

2) $(.,.)$ operator:

$$(.,.) : \{\omega, v\} \in R^3 \rightarrow se(3)$$
$$(\omega, v) = \begin{bmatrix} [\omega] & v \\ 0 & 0 \end{bmatrix} \tag{A-3}$$

3) Matrix exponential:

$$e^{(\omega,v)} = \exp \begin{bmatrix} [\omega] & v \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \exp([\omega]) & Av \\ 0 & 1 \end{bmatrix} \tag{A-4}$$

where

$$\exp([\omega]) = I + \frac{\sin\phi}{\phi}[\omega] + \frac{1-\cos\phi}{\phi^2}[\omega]^2,\ \phi = \|\omega\|$$
$$A = I + \frac{1-\cos\phi}{\phi^2}[\omega] + \frac{\phi-\sin\phi}{\phi^3}[\omega]^2 \tag{A-5}$$

4) Adjoint map on $SE(3)$:

$$Ad_G(h) : se(3) \to se(3)$$
$$Ad_G(h) = \begin{bmatrix} \Theta & 0 \\ [b]\Theta & \Theta \end{bmatrix} \begin{bmatrix} h_\omega \\ h_v \end{bmatrix} \tag{A-6}$$

where $G \in SE(3)$ is defined as in (A-1), and $h = (h_\omega, h_v) \in se(3)$.

5) Dual adjoint operator:

$$Ad^*_G(h^*) : se(3)^* \to se(3)^*$$
$$Ad^*_G(h^*) = \begin{bmatrix} \Theta^T & \Theta^T[b]^T \\ 0 & \Theta^T \end{bmatrix} \begin{bmatrix} M \\ F \end{bmatrix} \tag{A-7}$$

where $G \in SE(3)$, and $h^* = (M,F) \in se(3)^*$.

6) Lie bracket operator:

$$ad_g(h) = [g,h] = \begin{bmatrix} [g_\omega] & 0 \\ [g_v] & [g_\omega] \end{bmatrix} \begin{bmatrix} h_\omega \\ h_v \end{bmatrix} \tag{A-8}$$

where $g,h \in se(3)$. $h = (h_\omega, h_v)$ and $g = (g_\omega, g_\omega)$.

7) Dual Lie bracket operator:

$$ad^*_g(h^*) = [g,h^*] = \begin{bmatrix} [g_\omega]^T & [g_v]^T \\ 0 & [g_\omega]^T \end{bmatrix} \begin{bmatrix} M \\ F \end{bmatrix} \tag{A-9}$$

where $g = (g_\omega, g_v) \in se(3)$ and $h^* = (M,F) \in se(3)^*$.

REFERENCES

[1] J. Lo and D. Metaxas, "Recursive dynamics and optimal control techniques for human motion planning," *Computer Animation*, pp. 220–234, 1999.

[2] M. Ruchanurucks, S. Nakaoka, S. Kudoh, and K. Ikeuchi, "Humanoid robot motion generation with sequential physical constraints," in *Proc. IEEE International Conference on Robotics and Automation*, pp. 2649–2654, 2006.

[3] A. Safonova, N. Pollard, and J. Hodgins, "Optimizing human motion for the control of a humanoid robot," in *Proc. Applied Mathematics and Applications of Mathematics*, 2003.

[4] L. Guilamo, J. Kuffner, K. Nishiwaki, and S. Kagami, "Manipulability optimization for trajectory generation," in *Proc. IEEE International Conference on Robotics and Automation*, pp. 2017–2022, 2006.

[5] L. Sentis and O. Khatib, "A whole-body control framework for humanoids operating in human environments," in *Proceedings of the IEEE Internactional Conference in Robotics and Automation*, May 2006.

[6] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal of Robotics and Automation*, vol. 3, no. 1, pp. 43–53, 1987.

[7] D. Winter, *Biomechanics and Motor Control of Human Movement*. John Wiley, 2nd ed., 1990.

[8] G. Yamaguchi, *Performing Whole-body simulations of Gait with 3D*, ch. Multiple Muscle Systems: Biomechanics and Movement Organization. Springer-Verlag, 1990.

[9] A. Skrinar, R. Burdett, and S. Simon, "Comparison of mechanical and metabolic energy consumption during normal gait," *Journal of orthopedic research*, vol. 1, no. 1, pp. 63–72, 1983.

[10] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi, "Humanoid robot HRP-2," in *Proc. IEEE International Conference on Robotics and Automation*, pp. 1083–1090, 2004.

[11] E. Yoshida, I. Belousov, C. Esteves, and J.-P. Laumond, "Humanoid motion planning for dynamic tasks," in *Proc. IEEE-RAS International Conference on Humanoid Robots*, 2005.

[12] E. Yoshida, O. Kanoun, C. Esteves, and J.-P. Laumond, "Task-driven support polygon humanoids," in *Proc. IEEE-RAS International Conference on Humanoid Robots*, 2006.

[13] E. Yoshida, C. Esteves, T. Sakaguchi, J.-P. Laumond, and K. Yokoi, "Smooth collision avoidance: Practical issues in dynamic humanoid motion," in *Proc. IEEE International Conference on Intelligent Robotics and Systems*, 2006.

[14] F. Park, J. Bobrow, and S. Ploen, "A Lie group formulation of robot dynamics," *International Journal of Robotics Research*, vol. 14, no. 6, pp. 1130–1135, 1995.

[15] G. Sohl and J. Bobrow, "A recursive multibody dynamics and sensitivity algorithm for branched kinematics chains," tech. rep., Department of Mechanical Engineering, University of California, June 2000.

[16] R. Rockafellar, "Augmented lagrange multiplier functions and duality in nonconvex programming," *SIAM J.Control*, vol. 12, May 1974.

[17] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1995.

[18] R. Rockafellar, "Penalty methods and augmented lagrangians in nonlinear programming," in *Proc. 5th IFIP Conference on Optimization techniques*, 1973.

[19] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *Proc. IEEE International Conference on Robotics and Automation*, pp. 1620–1626, 2003.