



HAL
open science

Humanoid feet trajectory generation for the reduction of the dynamical effects

Paolo Pierro, Olivier Stasse, Abderrahmane Kheddar, Kazuhito Yokoi, Carlos Balaguer

► **To cite this version:**

Paolo Pierro, Olivier Stasse, Abderrahmane Kheddar, Kazuhito Yokoi, Carlos Balaguer. Humanoid feet trajectory generation for the reduction of the dynamical effects. Humanoids, Dec 2009, Paris, France. pp.454-458, 10.1109/ICHR.2009.5379542 . lirmm-00796738

HAL Id: lirmm-00796738

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00796738v1>

Submitted on 5 Mar 2013

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.

Humanoid feet trajectory generation for the reduction of the dynamical effects

Paolo Pierro, Olivier Stasse, Abderrahmane Kheddar, Kazuhito Yokoi and Carlos Balaguer

Abstract—In this paper we present a different strategy for generating the trajectory of the swinging leg for a walking humanoid robot which takes into account the effects due to acceleration and velocities of the joints onto the center of mass of the robot. The trajectory of the leg is chosen to be constituted by two forth order polynomials interlaced by a via-point which satisfies the optimality criterium. This approach is validated on a humanoid robot HRP-2.

I. INTRODUCTION

Most of the human size humanoid robot includes in their ankle a compliant material to absorb the force resulting from the impact when the swinging foot is landing on the floor. A controller is then generally provided to compensate the effect of this compliant material [1]. Often on top of this controller a walking pattern generator provides articular (or torque) trajectories ensuring that the robot is balanced. Current real-time walking pattern generator assumes a simplified “inverted pendulum” model to simplify the problem of finding a CoM trajectory following a given ideal ZMP trajectory or satisfy the constrained related with the ZMP [2]. This simplified model does not take into account the compliant material, and let the underlying controller compensating the passivity of this material. Although this controller works effectively for moderate walking speeds, when performing fast motion or extended stepping over an object it is not sufficient. The compliant material deformation is such that the robot is leaning forward and might hit the floor sooner than expected and created large impact forces. Classically, this is addressed by adding masses to the model to take into account the inertia effect of the legs. However instead of modifying the CoM trajectory and fixing a 3rd order polynomial trajectory for the feet, we want to modify the foot trajectory itself. Indeed for a service robot, the upper body part is used to perform task such as grasping, holding, while the swinging foot trajectory main task is to move towards the next support foot position. Therefore modifying the swinging foot trajectory seems to be a rational choice if the upper body is constrained by other tasks. In this paper we propose a method to plan the swinging foot trajectory in order to minimize the inertia effect on the compliant material.

P. Pierro and C. Balaguer are with Universidad Carlos III de Madrid, Avda Universidad 30, 28911, Leganés, Madrid, Spain. {ppierro, balaguer}@ing.uc3m.es

O. Stasse, A. Kheddar and K. Yokoi are with the Joint Japanese-French Robotics Laboratory, AIST/CNRS, AIST Tsukuba Central 2, Umezono 1-1-1, Tsukuba 305-8568m Japan. {olivier.stasse, abderrahmane.kheddar, kazuhito.yokoi}@aist.go.jp

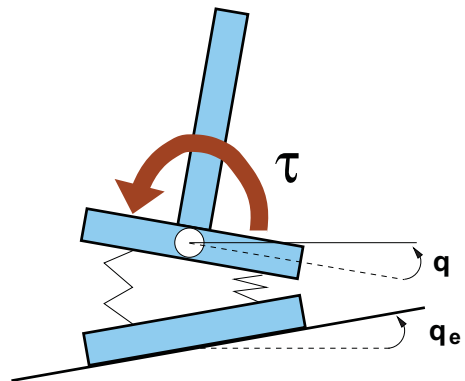


Fig. 1. The flexible material is located between the sole and the force sensor.

II. PROBLEM STATEMENT

Let us consider a foot embedding a compliant material between the sole and a force sensor as depicted in Fig.1. Following [1] the floor reaction torque τ is given by:

$$\tau = -k_e(q - q_e) \quad (1)$$

When this foot is the support foot and the robot is moving forward, a desired behavior is to avoid having $q_e > 0$ when the swinging foot is currently landing, because it has a non-null speed. Although controller such as the one described in [3] are designed to provide a direct-drive torque control, they also control the body posture assuming that the robot is behaving like an inverted pendulum. If the robot has not a mass distribution compatible with the assumption of the inverted pendulum, like for instance Johnnie [4], or while performing fast motion this disturb the controller. For instance in [5], HRP-2 is stepping over an obstacle and is performing a step which twice longer than normal standard mode. The use of the classical controller to compensate for the compliant material, the inverted pendulum for the CoM trajectory, and a 3rd polynomial for the feet trajectory generate impact forces about twice the weight of the robot as depicted in Fig.3. In the case of the step-over the CoM trajectory is specifically designed to avoid an obstacle, in other case it can be the result of a planning process such as in [6]. In such situation, the swinging foot trajectory can be modified to minimize its inertia effect relative to the CoM, and minimize the perturbation. In [5] the foot trajectory has been changed to a Clamped-Spline to minimize the deformation of the compliant material. But this modification was mostly heuristically motivated. In the present paper we give a more well-grounded way of finding swinging foot

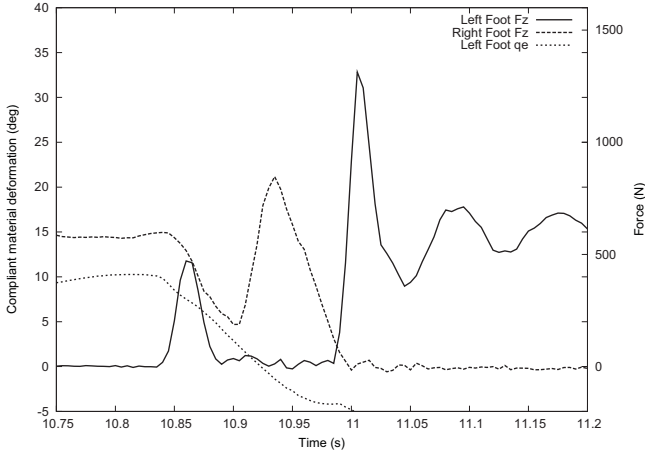


Fig. 2. Landing with $q_e > 0$ makes the robot hits the floor with a strong impact.

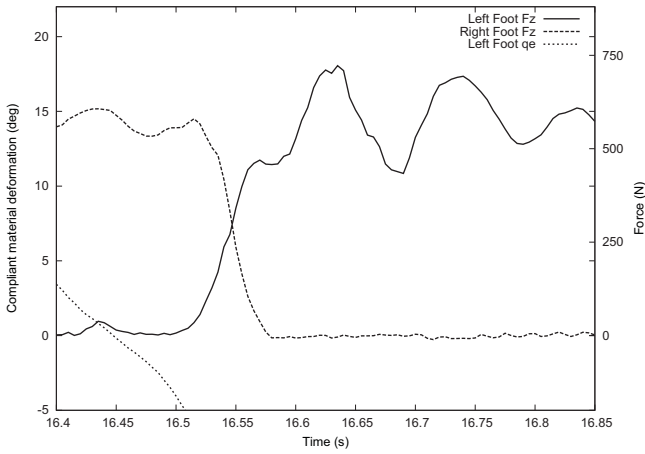


Fig. 3. Foot trajectory modified to get $q_e < 0$ when landing.

trajectories which minimizes the angular momentum relative to the CoM.

III. THE MODEL FOR THE FLOATING LEG

The floating leg may be modeled in a simplified way as a two link kinematic chain (double pendulum). Within the interest of the paper, this model is sufficient for modeling the robot leg in the sagittal plane during the single support phase. In fact, the joint 1 represents the waist and joint 2 the knee and their movements have to be considered relative with respect to the CoM (Center of Mass).

In our model, we resume the mass of the whole link 1 with m_1 and regarding the foot and the leg with m_2 . The lengths of the links are l_1 and l_2 . The system is free to move in a vertical plane. The local acceleration of gravity is g .

The equations of motion for the double pendulum are well documented in literature [7], and can be calculated using the Lagrange formulation. The joint space dynamic model can be written in a compact matrix form:

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \quad (2)$$

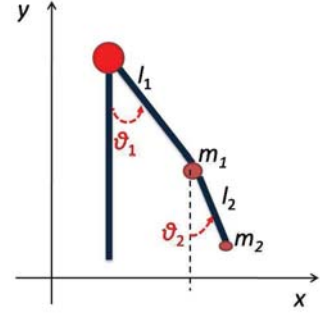


Fig. 4. Double pendulum

where $\mathbf{B}(\mathbf{q})$ represents inertia matrix which is:

$$\mathbf{B}(\mathbf{q}) = \begin{bmatrix} (m_1 + m_2)l_1^2 & m_2l_1l_2 \cos(\theta_1 - \theta_2) \\ m_2l_1l_2 \cos(\theta_1 - \theta_2) & m_2l_2^2 \end{bmatrix} \quad (3)$$

$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ represents the quadratic velocity terms:

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} 0 & m_2l_1l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_2 \\ -m_2l_1l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_1 & 0 \end{bmatrix} \quad (4)$$

And $\mathbf{g}(\mathbf{q})$

$$\mathbf{g}(\mathbf{q}) = \begin{bmatrix} (m_1 + m_2)l_1g \sin(\theta_1) \\ m_2l_2g \sin(\theta_2) \end{bmatrix} \quad (5)$$

The joints variables are represented by the vector \mathbf{q} , their velocities by the vector $\dot{\mathbf{q}}$ and the accelerations by $\ddot{\mathbf{q}}$.

IV. STRATEGY FOR THE SELECTION OF THE TRAJECTORY OF THE FLOATING LEG

In order to find a trajectory for the double pendulum, we can suppose to use a via point through which the trajectory should pass. This way, the trajectory will be defined by a piece-wise polynomial of the desired order, depending on the initial and final conditions we want to specify.

A. Trajectory definition in the joint space

As a first approximation to the problem, the trajectories for both joint variables can be supposed to follow a parabolic equation:

$$\begin{cases} \mathbf{q}_1(t) = \mathbf{a}t^2 + \mathbf{b}t + \mathbf{c}, & \text{for } 0 \leq t \leq t_c \\ \mathbf{q}_2(t) = \mathbf{d}(T-t)^2 + \mathbf{e}(T-t) + \mathbf{f}, & \text{for } t_c \leq t \leq T \end{cases} \quad (6)$$

where t_c represents the instant of the via point through which the trajectory should pass.

So, 7 vectorial equations are needed for calculating the trajectory for \mathbf{q} univocally. Six equations come from the initial and final conditions, and from imposing position and velocity continuity in t_c :

$$\begin{cases} \mathbf{q}_1(0) = \mathbf{q}_0 \\ \dot{\mathbf{q}}_1(0) = \dot{\mathbf{q}}_0 \\ \mathbf{q}_2(T) = \mathbf{q}_T \\ \dot{\mathbf{q}}_2(T) = \dot{\mathbf{q}}_T \\ \mathbf{q}_1(t_c) = \mathbf{q}_2(t_c) \\ \dot{\mathbf{q}}_1(t_c) = \dot{\mathbf{q}}_2(t_c) \end{cases} \quad (7)$$

which can be solved as a function of t_c .

B. Trajectory definition in the task space

In this model, it is considered only the movement of the leg in the sagittal plane, where the two components can be chosen to follow a fourth-order polynomial:

$$\begin{cases} \mathbf{x}_1(t) = \mathbf{a}_1 t^4 + \mathbf{b}_1 t^3 + \mathbf{c}_1 t^2 + \mathbf{d}_1 t + \mathbf{e}_1, & \text{for } 0 \leq t \leq t_c \\ \mathbf{x}_2(t) = \mathbf{a}_2 t^4 + \mathbf{b}_2 t^3 + \mathbf{c}_2 t^2 + \mathbf{d}_2 t + \mathbf{e}_2, & \text{for } t_c \leq t \leq T \end{cases} \quad (8)$$

where t_c represents the instant of the control point through which the trajectory should pass.

So, 9 vectorial equations are needed for calculating the trajectory for \mathbf{x} univocally. Eight equations come from the initial and final conditions (position, velocity and acceleration), and from imposing position and velocity continuity in t_c :

$$\begin{cases} \mathbf{x}_1(0) = 0 \\ \dot{\mathbf{x}}_1(0) = 0 \\ \ddot{\mathbf{x}}_1(0) = 0 \\ \mathbf{x}_1(t_c) = \mathbf{x}_2(t_c) \\ \dot{\mathbf{x}}_1(t_c) = \dot{\mathbf{x}}_2(t_c) \\ \mathbf{x}_2(T) = \delta \mathbf{x} \\ \dot{\mathbf{x}}_2(T) = 0 \\ \ddot{\mathbf{x}}_2(T) = 0 \end{cases} \quad (9)$$

This solution guarantees initial and final velocities and accelerations as zero. The ninth condition comes from imposing the condition on the dynamical conditions.

C. Cost function

If it is desired to reduce the torque to the CoM due to the movement of the foot, the following vector from Eq. (2) should be considered:

$$\tau_d(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} 1 & 0 \end{bmatrix} [\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}] \quad (10)$$

The function we want to minimize is

$$f^* = \frac{1}{T} \left| \int_0^{t_c} \tau_{d,1} dt + \int_{t_c}^T \tau_{d,2} dt \right| \quad (11)$$

This cost function has to be minimized given the constraints (7) in case of a specified trajectory in the joint space or the constraints (9) if the trajectory in the task space is given.

The minimum of the cost function (11) with respect to t_c can be calculated by finding the solution for the function:

$$\psi(t_c) = [\tau_{d,1}(t_c) - \tau_{d,2}(t_c)] \left[\int_0^{t_c} \tau_{d,1} dt + \int_{t_c}^T \tau_{d,2} dt \right] \quad (12)$$

This equation can be solved using the Newton-Raphson method in order to find iteratively successively better approximations to the roots of the real-valued function $\psi(t_c)$. Iteratively:

$$t_c(n+1) = t_c(n) - \frac{\psi(t_c(n))}{\psi'(t_c(n))} \quad (13)$$

In the case of eq. (6), finding this solution is almost easy, but in the case of the trajectory in the task space, we have to reformulate the eq. (2) in order to write it in the task space.

If \mathbf{x} , $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$ represent, respectively, the position, velocity and acceleration in the task space, it is possible to write [7]:

$$\ddot{\mathbf{x}} = \frac{\partial \dot{\mathbf{x}}}{\partial t} = \frac{\partial(\mathbf{J}\dot{\mathbf{q}})}{\partial t} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} \quad (14)$$

So,

$$\ddot{\mathbf{q}} = \mathbf{J}^{-1}\ddot{\mathbf{x}} - \mathbf{J}^{-1}\dot{\mathbf{J}}\mathbf{J}^{-1}\dot{\mathbf{x}} \quad (15)$$

Here, $\dot{\mathbf{J}}$ is the time derivative of the jacobian \mathbf{J} . So, it is possible to rewrite the equations of motions 2 like so:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{x}} + \mathbf{K}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{x}} + \mathbf{g}(\mathbf{q}) = \tau \quad (16)$$

where:

$$\begin{aligned} \mathbf{M}(\mathbf{q}) &= \mathbf{B}(\mathbf{q})\mathbf{J}^{-1}(\mathbf{q}) \\ \mathbf{K}(\mathbf{q}, \dot{\mathbf{q}}) &= \left(\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{B}(\mathbf{q})\mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{J}}(\mathbf{q}, \dot{\mathbf{q}}) \right) \mathbf{J}^{-1}(\mathbf{q}) \end{aligned} \quad (17)$$

The mean effect of the inertia over the joint 1 along the trajectory is:

$$\begin{aligned} \bar{\tau}_1 &= \frac{1}{T} \left| \int_{t=0}^{t_c} \mathbf{M}_1(\mathbf{q}(\mathbf{x}_1(t)))\ddot{\mathbf{x}}_1 + \mathbf{K}_1(\mathbf{q}(\mathbf{x}_1(t)), \dot{\mathbf{q}}(\mathbf{x}_1(t), \dot{\mathbf{x}}_1(t)))\dot{\mathbf{x}}_1 dt \right. \\ &\quad \left. + \int_{t=t_c}^T \mathbf{M}_1(\mathbf{q}(\mathbf{x}_2(t)))\ddot{\mathbf{x}}_2 + \mathbf{K}_1(\mathbf{q}(\mathbf{x}_2(t)), \dot{\mathbf{q}}(\mathbf{x}_2(t), \dot{\mathbf{x}}_2(t)))\dot{\mathbf{x}}_2 dt \right| \end{aligned} \quad (18)$$

where,

$$\begin{aligned} \mathbf{M}_1 &= \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \mathbf{M} \\ \mathbf{K}_1 &= \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \mathbf{K} \end{aligned} \quad (19)$$

Considering the constraints (9), after a few calculations, it is possible to show that the equation (12) becomes:

$$\begin{aligned} \psi(t_c) &= \left[\cos(\theta_1) - \frac{\sin(\theta_1)}{\tan(\theta_1 - \theta_2)} m_1 \right] (x_{1,a} - x_{2,a}) + \\ &\quad + \left[\sin(\theta_1) + \frac{\cos(\theta_1)}{\tan(\theta_1 - \theta_2)} m_1 \right] (x_{1,b} - x_{2,b}) \end{aligned} \quad (20)$$

being a and b the two components of the vectors \mathbf{x} .

V. SIMULATION RESULTS

In this section, the simulation results of the proposed method are shown. The simulations refer to the model of the robot HRP-2. The simulated step length is 40 cm and the leg is lifted 15 cm during the floating phase. We first present the results using Matlab® for the trajectory both in the joint space and in the task space. Then, we present the results given by the simulator Open-HRP3.

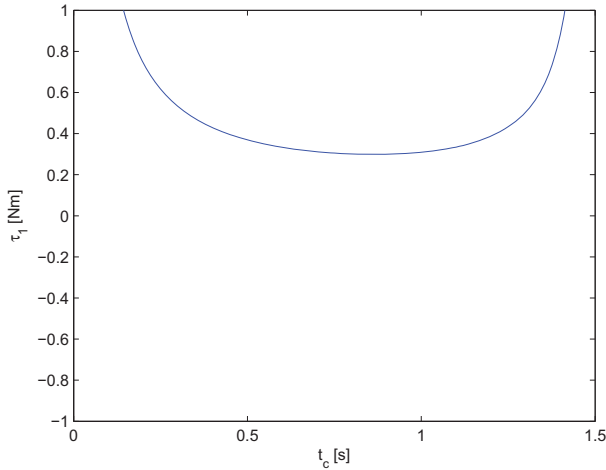


Fig. 5. Dynamical effects for a different values of t_c over a trajectory in the joint space

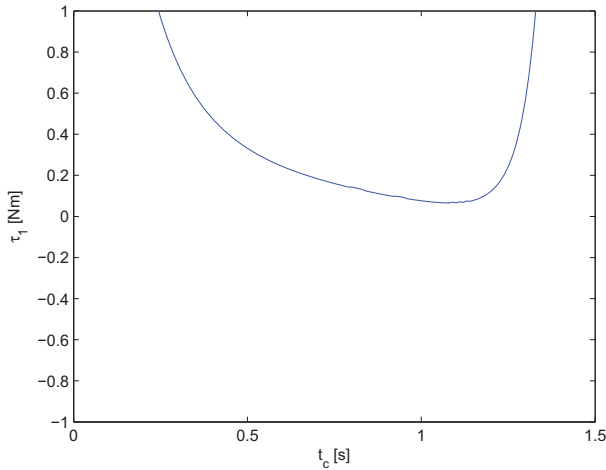


Fig. 6. Dynamical effects for a different values of t_c over a trajectory in the task space

A. Evaluation of the influence of the via point

In Fig. 5, the results of different via points along a trajectory in the joint space belonging to the constraints (7). It is possible to see that, for the given conditions for the step trajectory, a minimum exists for $t_c = 0.86s$

Given the same conditions, we found the values for τ_d in case of specifying the trajectory for the constraints (9). The results are depicted in Fig. 6, showing that in this case the minimum occurs at $t_c = 1.08s$.

B. Simulated experiments

In this section, the results of the simulation using a trajectory specification in the task space are presented. We have chosen to use such a solution since the approximation of a given trajectory in the joint space seems not to be worth. In fact, it makes sense to specify the trajectory in the task space.

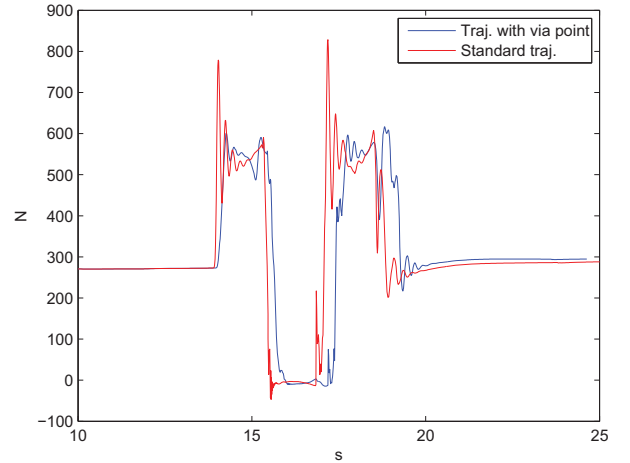


Fig. 7. Force read at the left foot

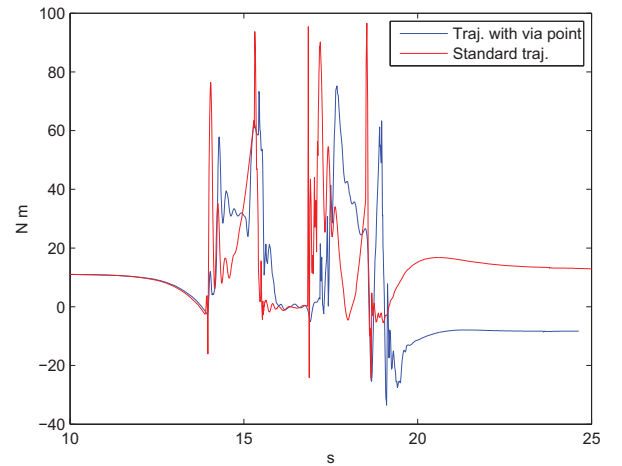


Fig. 8. Torque reading at the left foot

In Fig. 7, we show the force in the z direction (i.e. the direction perpendicular to the ground) representing the impact effect on the ground. As it can be seen from the graph, the impact is considerably reduced using the via point. This result is almost straightforward since the trajectory is specified in order to start and arrive with zero acceleration.

The results in Fig. 8 show the validity of the presented work. In fact, the torque acting on the left foot during the right foot floating is actually related to the torque acting on the CoM. It is possible to see that this torque is greatly reduced with respect to a standard trajectory.

VI. EXPERIMENTAL RESULTS

The proposed algorithm has been tested over the robotic platform HRP-2. In Fig. 10 we present the results of the impact on the ground based on [5]. In this case, huge impact of about 1200 N has been reduced down to about 650 N (as shown in fig. 9). Also, the torque measured by the F/T sensor shows the validity of the proposed work.

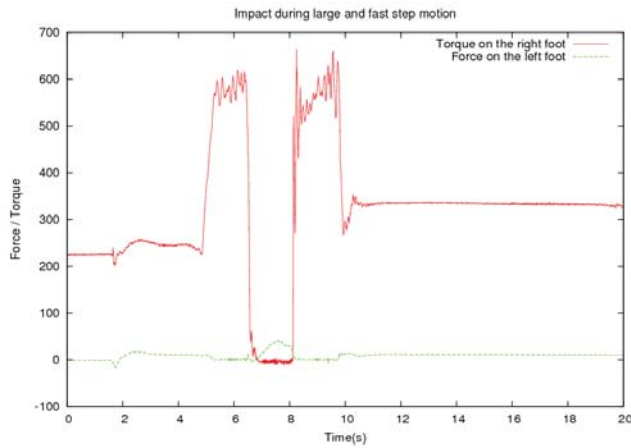


Fig. 9. Force/Torque sensor read in case of the trajectory with via point

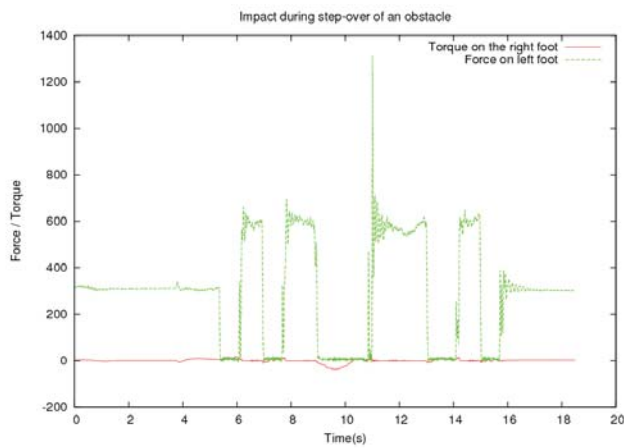


Fig. 10. Force/Torque sensor read in case of a standard trajectory

VII. CONCLUSIONS AND FUTURE WORKS

In this work, we presented a novel technique in order to generate the flying leg trajectory based on dynamical generation.

We have built a simplified model of the robot HRP-2 which we used in order to find a via point for the flying foot. The trajectory of the flying foot follows a piece-wise fourth order polynomial which permits to specify initial and final acceleration putting them to a zero value. Another condition comes from reducing the effects to the CoM of the velocities and accelerations due to the movement of the leg. This way, the torque applied to the CoM is reduced improving the global stability of the robot.

The simulations have shown the validity of the proposed work. In fact, using the Open-Hrp3 platform the impacts of the foot on the ground are really small.

Moreover, the algorithm has tried directly on the real

platform, giving good results compared to a standard foot trajectory. In fact, the torque on the ankle of the supporting foot is greatly reduced together with the force measured when the foot touches the ground.

It should be noted that using a more complex model of the flying leg, it is possible to further improve the results.

In addition, we are working on some optimization techniques which permit to get the same result in a reduced computational time, improving the performance while working on the online generation.

VIII. ACKNOWLEDGMENTS

This work was supported by grants from the ROBOT@CWE EU CEC project, Contract No. 34002 under the 6th Research program. The second author is also supported by grants from the R-BLINK Project, Contract ANR-08-JCJC-0075-01.

REFERENCES

- [1] S. Kajita, K. Yokoi, M. Saigo, and K. Tanie, "Balancing a humanoid robot using backdrive concerned torque control and direct angular momentum feedback," in *IEEE/RAS International Conference on Robotics and Automation (ICRA)*, 2001, pp. 3376–3382.
- [2] 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 *IEEE/RAS, International Conference on Robotics And Automation (ICRA)*, 2003, pp. 1620–1626.
- [3] S. Kajita, T. Nagasaki, K. Kaneko, and K. Tanie, "A running controller of humanoid biped hrp-2r," in *IEEE/RAS International Conference on Robotics and Automation (ICRA)*, 2003, pp. 618–624.
- [4] T. Buschmann, S. Lohmeier, M. Bachmayer, H. Ulbrich, and F. Pfeiffer, "A collocation method for real-time walking pattern generation," in *7th IEEE-RAS International Conference on Humanoid Robots*, 2007, pp. 1–6.
- [5] O. Stasse, B. Verrelst, B. Vanderborght, and K. Yokoi, "Strategies for humanoid robots to dynamically walk over large obstacles," p. accepted, 2009.
- [6] E. Yoshida, C. Esteves, I. Belousov, J.-P. Laumond, T. Sakaguchi, and K. Yokoi, "Planning 3-d collision-free dynamic robotic motion through iterative reshaping," *Robotics, IEEE Transactions on*, vol. 24, no. 5, pp. 1186–1198, 2008.
- [7] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. London, Great Britain: Springer-Verlag, 2009.