Fast foot prints re-planning and motion generation during walking in physical human-humanoid interaction

Olivier Stasse, Paul Evrard, Nicolas Perrin, Nicolas Mansard, Abderrahmane Kheddar

***Abstract*— In this paper a system allowing real-time interaction between a human and a humanoid robot while walking is presented. The aim of this work is to integrate humanoid robots into collaborative working environment. Co-located realization of a task is one instance of such collaboration. To achieve such task whole-body motion generation while keeping balance is mandatory. This is obtained using a real-time pattern generator allowing on-line foot-print modification integrated in a stack of controllers. Several experiments of direct interaction between a human and a HRP-2 humanoid robot illustrates the results.**

I. Introduction

The main context of this work is to introduce humanoid robots in collaborative working environments. As multi-purpose robotic systems they would be able to manipulate product, carry tools, inspect product lines and so on. To separate the different context of collaborative work in nowadays companies IT-infrastructure, we have introduced a taxonomy of *contexts* to realize a collaborative task. This taxonomy is depicted in Fig.1. More precisely it consists in four different class of situations: (1) An *autonomous* context realization when the robot is directly interacting with a human to perform a task, and particularly during physical interaction. (2) A *local* context realization when the robot is using the surrounding network and computer capabilities to expand its functional space. (3) A *semi-local* context realization when the robot is interacting with a collaborative working application targeted for an application or for a structure such as a company. (4) A *global* context realization when the robot is interacting with services external to its *semi-local* structure for instance Google Images services, manufacturer product specification, etc.

In [1] we have presented how it is possible to integrate HRP-2 in such a collaborative environment to perform autonomously an inspection task. In [2] we have demonstrated how it is possible to teleoperate a walking HRP-2 while having the robot interacting with a local operator. This paper focus on an *autonomous* context were the robot is in direct contact with the person to collaborate.

II. Problem statement

The goal of this paper is to propose a whole body controller including the walking modality able to interact with the environment while deciding in real-time which next step to perform. More precisely it concerns:

1) Balancing of the robot,
2) Interacting with the human operator,
3) Avoiding self-collision.

Controlling the balance of the robot in real-time is the first needed stage in order to prevent the robot from falling on the human operator. The difficulty in doing so lies mostly in the need for a very short reaction time. The scientific difficulty is explained in more details in the following paragraph. The problem related to the interaction with the human operator is the need for a force-based control law with a robot which is not controlled in torque. Finally, after integrating those controllers the final difficulty is to include a very fast collision detection scheme inside the system which will prevent the robot to self-collide while walking.

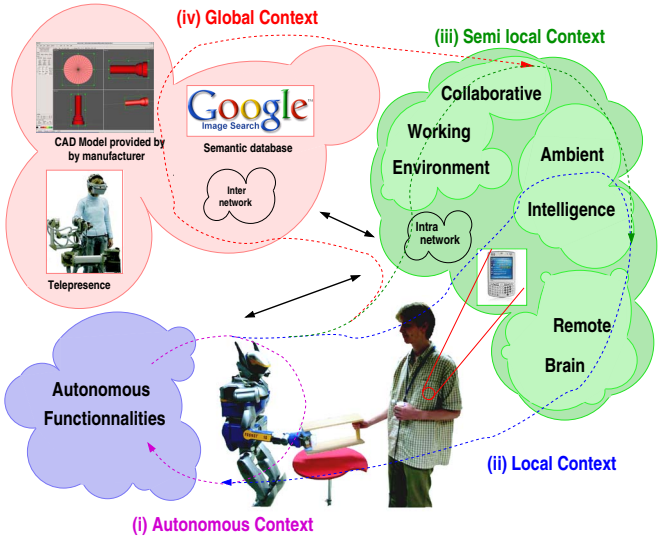

Fig. 1. Four contexts of task realization in the physical common workspace

A. Balance

Considering a humanoid robot with a state vector $\mathbf{x} = [\mathbf{r}, \mathbf{q}]^\top$, where $\mathbf{r}$ is the free floating parameters and $\mathbf{q}$ the position of the joints, the main problem related with balancing is to be able to generate in real-time a trajectory $\mathbf{q}(t)$ such that the robot's Zero Momentum Point (ZMP), $\mathbf{p} = [p_x p_y]$,is always in the support polygon defined by the contact points of the robot. In general the method used to find the trajectory $\mathbf{q}(t)$ is the following:

O. Stasse, P. Evrard, N. Perrin and A. Kheddar are with CNRS-AIST, JRL UMI 3218/CRT, Intelligent Systems Research Institute, AIST, Japan. N. Mansard and N. Perrin are also with the Gepetto Group, LAAS, CNRS University of Toulouse, France. P. Evrard and A. Kheddar are also with the CNRS-UMR, LIRMM, CNRS, France. {olivier.stasse,evrard.paul,n.perrin}@aist.go.jp, kheddar@lirmm.fr, nmansard@laas.fr

1) find a ZMP trajectory called $\mathbf{p}^{\text{ref}}(t)$ which comply with the support polygon constraints,
2) from the ZMP trajectory find a trajectory $\mathbf{q}(t)$ which realizes $\mathbf{p}^{\text{ref}}(t)$.

Finding the appropriate $\mathbf{q}(\mathbf{t})$ for an articulated multi-body model of a robot is the main difficulty, because it involves a complex second order differential equation. In order to make the problem more tractable, a single-point-mass model is often consider. The simplified system is still difficult to solve. A famous assumption to further simplify the problem is to set the CoM height to a constant value, which provides the 3D linearized inverted pendulum [3]. To solve this equation, it is possible to use techniques such as the preview control which is based on a receding window of future ZMP constraints as proposed for instance by Kajita in [3]. In addition [3] takes into account, to some extend, the multi-body model by using a dynamic filter. We have tried this method for the targeted application in [2]. The resulting delay of having two preview window (one for the LIPM and one for the dynamic filter) induces a 3.2s delay which reduces considerably the reactivity of the system. The cost of the dynamic filter is too high to change in real-time the feet positions inside the receding window, when the CoM reference trajectory is generated every 5ms. In Nishiwaki et al. [4] a method is proposed to change on-line, and inside a step-cycle, the ZMP and CoM trajectories while including a dynamic filter. This smart method has been used for vision based reactive planning [5] and reactive interaction with heavy objects [6]. The method relies mostly onto two main concepts: increasing the time period to generate the reference CoM trajectory with the dynamic filter (from 5ms to 20ms) and increasing the frequency of the ZMP controller (from 200Hz to 1kHz). The technique chosen in the present paper is to force the humanoid to behave like an inverted pendulum, and to use the commercially available ZMP controller of HRP-2. To have an extremely fast ZMP and CoM reference trajectories generation able to modify its preview window, we used the method proposed in Morisawa *et al.* [7]. This method assumes in addition to constant CoM height that the ZMP trajectory is a third order polynomial. Once the CoM trajectory is found, $\mathbf{q}(t)$ is then computed by a controller in charge of following the whole body CoM trajectory such as the one described by Sugihara *et al.* [8]. The results in the experimental section show that our control architecture allows a very stable walking. Our implementation takes 1ms *every one step cycle* and 30μs at each time cycle.

B. Interaction with a human

Force control based whole-body motion generation have been proposed previously to push a table in [6]. This method relies mostly on a predictive model of the table which mostly resist to the force applied by the robot. When interacting with a human, the scheme is quite different, and in this specific work the robot is mostly acting as a follower. Therefore it will be shown in the experimental part that with an admittance controller, and a proper foot generation, a human-humanoid interaction can be performed without perturbating the robot's balancing. The most relevant previous work to this paper is [9] which is using a similar admittance controller to compensate for the forces induces by the interaction with the user. However the relationship between the hand displacement and the walking pattern generator is only specified through a walk velocity vector, without more details regarding the foot print generation, and the timing of the foot generation.

C. Self-collision

In order to avoid self-collision when generating on-line steps, a common technique is to use a set of safe discrete steps [5] for which the transitions are known to be free of any collision. The disadvantage of this kind of approach is the limited feasible sets of actions inducing a lot of intermediate steps. This is problematic when interacting with a human because a reactive following of the human is desirable. An extension using an informed local search has been proposed by Chestnutt [5]. This extension expands the robot's action set locally from the discrete steps found by the previous method. In the present work, we use a representation similar to the one proposed by Chestnutt in [5] for HRP-2, and explain in more detail in paragraph V. The upper body self-collisions are currently avoided by setting appropriate joint limitations.

III. Real-time walking pattern generator

This section recalls briefly the method proposed by Morisawa *et al.* [7]. In the following it is assumed that the single support phase and the double support phase of the steps correspond to intervals indexed by j.

A. Analytical trajectory for the robot's ZMP and CoM

Let us assume that the ZMP can be written:

$$p_x^{(j)} = \sum_{i=0}^{n} a_i^{(j)} (t - t_{j-1})^i,\ t_{j-1} \leq t \leq t_j,\ j = 1, \cdots, m, \tag{1}$$

where $a_i^{(j)}$ $(i = 0, \cdots, n, j = 1, \cdots, m)$ are scalar coefficients, and m is the number of intervals. Then it can be shown that the CoM trajectory can be obtained from a composition of hyperbolic cosines and sinus as well as polynomials:

$$\begin{aligned} c_x^{(j)} = V^{(j)}&\cosh\left(T_c(t - t_{j-1})\right) + W^{(j)}\sinh\left(T_c(t - t_{j-1})\right) \\ &+ \sum_{i=0}^{n} A_i^{(j)} (t - t_{j-1})^i, \quad j = 1, \cdots, m \end{aligned} \tag{2}$$

where $A_i^{(j)}$ and $a_i^{(j)}$ can be computed from one to the other [7].

To connect intervals together the following constraints are considered:

1) Initial condition for the CoM position and velocity;
2) Connection of two intervals for the CoM position and velocity;
3) Terminal condition for the CoM position and velocity;

4) Initial condition for ZMP position and velocity at each interval;
5) Terminal condition for ZMP position and velocity at each interval.

Assuming that we consider a trajectory of 3 steps, with the robot stopping at the third step, then only the coefficents $A_i^{(j)}$ of the first interval and the coefficient $V^{(j)}$ and $W^{(j)}$ of all the intervals must be computed. This is realized by solving a linear system. A more detailed description of this linear system can be found in [7] and [10].

B. Measurement of ZMP instability

Modifying the foot length of the swing leg amounts to connect the previous ZMP and COM trajectories to the new ones. The constraints of the current state and on the new connection can induce a deviation of the ZMP which can be computed. It is realized by considering a fourth order polynomial for the ZMP trajectory first interval with the new constraints. In order to compensate for the deviation of the ZMP due to the step length modification, it has been proposed in [7] to add a transition time.

C. Correction in orthogonal direction

Applying this algorithm gives the results depicted in Fig.2. The foot landing position is changed $15cm$ forward along the X axis at $t = 2.3s$. The trajectory along the ZMP and the CoM are correctly reinitialized by computing an appropriate time shift. However this time shift implies a fluctuation of the ZMP shape in the orthogonal direction as depicted in Fig. 3. The origin of this fluctuation is the fourth order polynomial at the beginning of the new trajectory. Morisawa proposed to used a preview control to compensate for this fluctuation. The preview control takes the difference between the ideal ZMP (ZMP_{ref}) and the polynomial ZMP (ZMP_{poly}). The output is a correction of the ZMP (ZMP_{corr}) which should be added to ZMP_{poly} to finally get the correct one (ZMP_{final}). The result is depicted in Fig. 3. The importance of this filtering is highlighted in the experimental section.

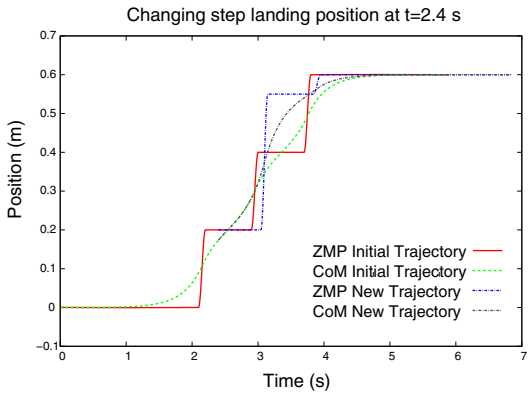

Fig. 2. Changing the foot landing position at time $t = 2.4$ seconds $15\ cm$ forward.

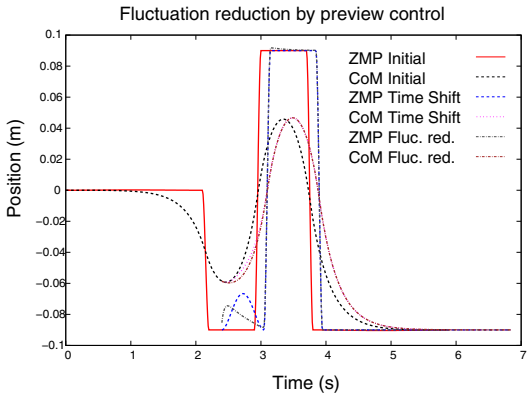

Fig. 3. ZMP fluctuation in the orthogonal direction.

IV. Whole Body Motion Generation

A. Stack of Tasks (SOT)

The task function approach consists of control laws in the task space. Under well defined conditions the stability properties on the smaller task space can be extended to the whole body of the robot [11]. Using the redundancy formalism [12] it is then possible to compose several tasks. Only the tasks in the stack are taken into account in the control law. The task at the bottom level has priority over all the others, and the priority decreases as the stack level increases. Any new task added in the stack does not disturb the tasks already in the stack. Let $(\mathbf{e_1}, \mathbf{J_1}, \mathbf{e_1^*}) \cdots (\mathbf{e_n}, \mathbf{J_n}, \mathbf{e_n^*})$ be n tasks, where $\mathbf{e_i}$ denotes the task space i, $\mathbf{J_i}$ the Jacobian that is used as a differential link between the task space and the state vector $\mathbf{x}$, and $\mathbf{e_i^*}$ is the reference control law in the task space. The control law computed from these n tasks ensures that the task $\mathbf{e_i}$ does not disturb the task $\mathbf{e_j}$ if $i > j$. A recursive computation of the joint velocity is proposed in [12]:

$$\dot{\mathbf{q}}_\mathbf{i} = \dot{\mathbf{q}}_{\mathbf{i-1}} + (\mathbf{J_i P^A_{i-1}})^{\#}(\dot{\mathbf{e}}_\mathbf{i}{}^* - \mathbf{J_i}\dot{\mathbf{q}}_{\mathbf{i-1}}), \quad i = 1 \cdots n \quad (3)$$

where $\mathbf{P_i^A}$ is the projector onto the null-space of the augmented and $\dot{\mathbf{q}}_\mathbf{0} = 0$. $^{\#}$ is any pseudo-inverse operator, typically a weighted inverse. The robot joint velocity realizing all the tasks in the stack is $\dot{\mathbf{q}} = \dot{\mathbf{q}}_\mathbf{n}$.

B. Force-Based Control

The interaction with the human operator is here performed mostly through an admittance controller with an apparent mass. Using the foregoing control scheme, it is straightforward to realize many tasks in free space. However, for force-based control and tasks involving contact, the redundancy formalism does not directly apply. A solution to this problem is the *dynamic inverse* as proposed in [13], [14] (rather than using the *kinematic inverse* [12]). However, this solution requires the robot to be torque controlled, which is not the case for our robot. We rather propose to modify the *kinematic inverse* control scheme in order to obtain a similar behavior. Two parameters can used to realize this modification: the reference control law $\mathbf{e_i^*}$, the norm $^{\#}$ to be used for the pseudo-inverse.

The space in which the control is designed is the operational space (6D position) of the contact point denoted by $\mathbf{r}$. We impose the following dynamics to this point:

$$\mathbf{M}\ddot{\mathbf{r}} + \mathbf{B}\dot{\mathbf{r}} = \mathbf{f} \tag{4}$$

where $\mathbf{r}$ is a reference position and orientation in the task space, $\mathbf{M}$ and $\mathbf{B}$ are arbitrary masses and damping matrices of the equivalent virtual system, and $\mathbf{f}$ are forces and torques exerted on the equivalent virtual point. Typically $\mathbf{f}$ can be set to the real forces measured at the contact point. This gives a first choice for the control law $\mathbf{e_i^*}$ set to a first order integrator given that $\mathbf{M}$ and $\mathbf{B}$ have been set.

Using a pseudo-inverse weighted by the inertia matrix $\mathbf{A}$ of the robot and the virtual mass $\mathbf{M}$ as $\mathbf{\Lambda} = \left(\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^{\mathbf{T}}\right)^{-1}$, the apparent inertia of the end-effector of the robot, it is obtained:

$$\mathbf{A}\ddot{\mathbf{q}} = \mathbf{J}^{\mathbf{T}}\mathbf{f} - \mathbf{B_q}\dot{\mathbf{q}} \tag{5}$$

where $\mathbf{B_q} = \mathbf{J}^{\mathbf{T}}\mathbf{B}\mathbf{J}$ is the friction factor of the whole body structure. This last equation corresponds to a simplified version of the dynamic equation of the whole-body compliant robot with gravity compensation, with forces $\mathbf{f}$ acting on $\mathbf{r}$ and a friction $\mathbf{B_q}$ that may be tuned by selecting $\mathbf{B}$ to stabilize the system.

Consequently, if the control parameters are chosen as described above, the obtained control is equivalent to the real dynamics of the robot. The control represents a generalization of (3), which means that both force-based contact tasks and position-based free space tasks can be realized within this control structure. No specific values have to be chosen or tuned, except for the damping gain $\mathbf{B}$, that has been experimentally verified to be very robust [15].

This solution generalizes the use of the kinematics inverse for collaborative tasks.

V. Foot step generation

A. Foot position

The foot step generation is decided from the force sensor in the wrist of the robot when the human operator is holding the robot hands. Let us consider the *rest* pose which corresponds to the robot standing still in front of the human operator. The corresponding position of the right hand is respectively noted $\mathbf{Rh}_r^{lf}$ in the left foot reference frame, and $\mathbf{Rh}_r^{rf}$ in the right foot reference frame. At time t, when deciding where to place the next foot print ($nfp \in \{lf, rf\}$), the right hand is at position $\mathbf{Rh}^{sp}(t)$ in the support foot reference frame with $sp \in \{lf, rf\}$. Let us note the next foot print position in the support reference frame $\mathbf{Nfp}^{sp}$. Then

$$\mathbf{Nfp}^{sp} = \mathbf{Rh}^{sp}(t)(\mathbf{Rh}_r^{nfp})^{-1} \tag{6}$$

The next foot-step is chosen to minimize the difference between the reference 6D pose of the right hand where both feet are in support, and the current position of the right hand in the support foot frame. This new foot position is computed at the beginning of each single support phase. The new foot position is clipped into a safe area as explained in the next paragraph. Finally the clipped step position is send to the pattern generator as described in section III. The next date at which a new step should be computed is deduced from the time shift returned by the walking pattern generator. It is important to stress here that all the feet positions are generated on-the-fly from the force information extracted from the sensor. There is no *planning per se*, and if no step is provided the robot is simply stepping on the spot.

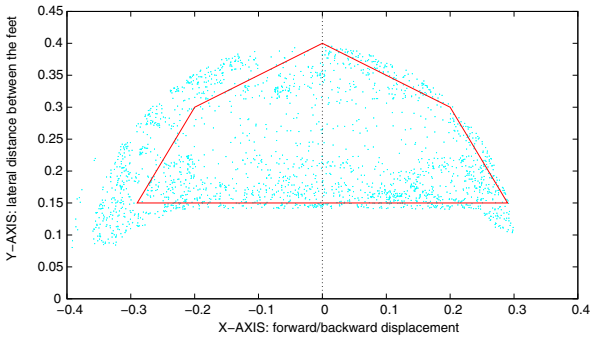

Fig. 4. Safe region for a transition between two positions of the right foot

B. Foot feasibility

The foot feasibility is realized by clipping the foot position in a sub-region which is considered to be safe. To build this region, an intensive generation of trajectories based on different steps configuration has been realized. Along those trajectories the self-collision, dynamic stability and joint limits have been tested. Let us consider two positions of the right foot $(x_1^R, y_1^R, \theta_1^R), (x_2^R, y_2^R, \theta_2^R) \in SafeArea$ where $SafeArea$ is the red polygon depicted in Fig.4. Our method is built in such way so that $\forall (x_1^R, y_1^R, \theta_1^R), (x_2^R, y_2^R, \theta_2^R) \in SafeArea$ the control algorithm is highly likely to provide an articular trajectory $\mathbf{q}(t)$ without collision and dynamically stable which makes the robot moves its right foot from $(x_1^R, y_1^R, \theta_1^R)$ to $(x_2^R, y_2^R, \theta_2^R)$.

If the orientation of the feet is always 0, the symmetry along Y-axis of the red support polygon implies that once a step is performed (say for example with the right foot), the left foot necessarily lies into the actualized support polygon. Since in our experiments we constraint the change of orientation to be very small, we can assume (thanks to some margins that we put in our feasibility estimations) that the initial configuration before performing a step is always such that the swing foot lies in the "SafeArea" support polygon. Hence, clipping steps in the SafeArea polygon defines a stable strategy for footprints decision in the sense that the steps performed will always belong to a set which has been extensively tested offline. For more details on the problem of footsteps clipping, the interested reader is kindly invited to read [16].

VI. Experiments

A. Experimental setup

The pattern generator described in section III provides the CoM reference trajectory and the feet trajectories for

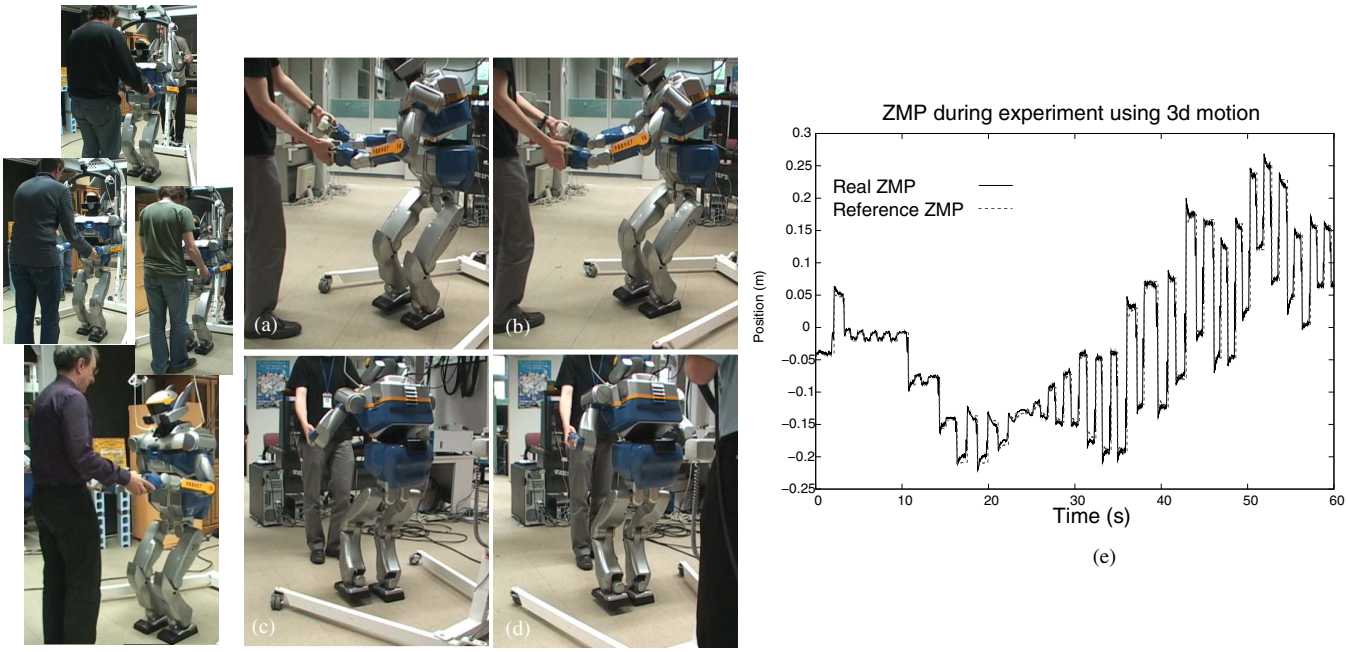

Fig. 5. Experimental results

two controllers: a CoM trajectory tracker, and a relative foot position controller. In order to impose the linearized inverted pendulum model (LIPM), a waist controller is added which maintain the waist height to a constant. In addition we have to make sure that the momentum generated by the multi-body model of the robot is also compatible with the LIPM. The linear momentum is already provided by the CoM controller, and the angular momentum is approximated by forbidding any roll and pitch for the waist. The two last controllers implement the force-based control described in paragraph IV-B.

B. Results

We have tested two different experimental contexts for this direct manipulation. The first one constrained the motion in the sagittal plane. In this setup there are no sideways walking, only frontal direction. Therefore as the kinematic constraints become less activated, it is possible to use foot step length of 20cm in the sagittal direction. The robot is quite reactive, and it was possible for several people to test the system with no particular training as depicted in the left part of Fig.5. A video of the corresponding experiments can be found at [17] The second context released the constraint on the sagittal plan, and allow 3D foot motion: (x, y, θ). It is the one depicted in Fig.5-(a-d). The sub-figures (a) and (b) depicts the robot turning with the human operator. Sub-figure (c) demonstrate a side-way walk, and (d) a backward walk. Those motions can be seen in the companion video of this paper at [18]. At the end the last segment presents a preliminary result on holding an object. Compare to the work presented in [9] it appears clearly that our work generates whole body motion, when [9] is mostly keeping its upper body still. Unfortunately the lack of information in [9] forbid to perform further comparisons.

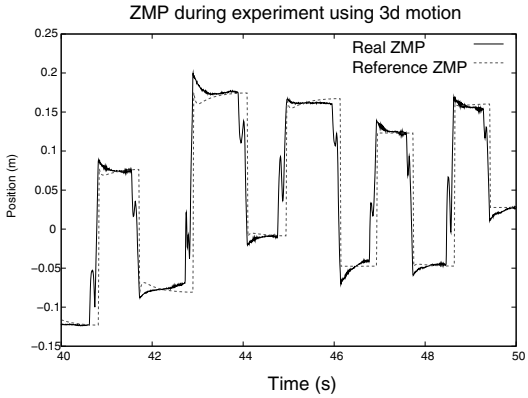

Fig. 6. Zoom of Fig. 5-(e)

Fig.5-(e) and Fig.6 depict the ZMP reference along the X-axis computed by the pattern generator described in section III, and obtained during the experience in the companion video. The real ZMP is also represented. One can see the deformation induced by the change of the swinging foot position when this one start its flying phase. This occurs when the time-shift is applied according to the Y-axis selected because it has the biggest ZMP reference perturbation. The ZMP real is deviating 2cm far from the reference trajectory which is quite similar to the result obtain by [4]. Most of our current fall are due to the constraints of the robot forbidding to execute fully the tasks of the system. When going backward the robot reaches more quickly its chest joint limit, and therefore in this direction the CoM task is more difficult to realize.

Although there is one SVD decomposition to be computed for each task, this control scheme fits within the 5ms control loop of the HRP-2. It takes less than 1ms to compute the new weights of the ZMP and CoM trajectories. Between those peaks the pattern generator merely computes the polynomials given by (1) and (2).

C. Discussion

The current limitations of the system are the size of the steps when allowing turning motion. The robot in the first context, i.e. sagittal motions, has shown faster capabilities. The current problem is to build a more complex representation of the safe area depicted in Fig.4. Indeed there are some transitions between foot position and the robot's CoM speed which are not possible. A possible extension of this work is to build a probabilistic function that will give us an estimate of feasible transitions, coupled with protection mechanisms ensuring the safety of the human operator and the integrity of the robot.

Another limitation is the speed variation induced by the pattern generator proposed in [7]. One possibility is to modify the position of the feet instead of the time interval as proposed by Diedam *et al* [19].

VII. Conclusions and future works

The overall whole body control is performed by a generic inverse-kinematic based structure. This enables at the same time force-based control for interaction by decoupling the tasks along the generalized inertia of the robot and following of the reference of the CoM and the feet.

These reference trajectories are computed by a real-time pattern generator that allows real-time modification of the CoM and the ZMP trajectories as the interaction with the user changes.

A critical issue is then to properly choose the next foot print to provide as an input to the pattern generator, in order to avoid unfeasible steps or auto-collision. It has been solved by using a generic decision algorithm to compute the safe subspace off-line and answer on-line in a blink of an eye.

As a case study real interaction with humans have been performed with experimented and non-experimented users.

VIII. Acknowledgments

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

References

[1] O. Stasse, F. Lamiraux, R. Ruland, A. Kheddar, K. Yokoi, and W. Prinz, "Integration of humanoid robots in collaborative working environment: A case study on motion generation," *Intelligent Service Robotics*, p. accepted, 2009.

[2] P. Evrard, N. Mansard, O. Stasse, A. Kheddar, T. Schauß, C. Weber, A. Peer, and M. Buss, "Intercontinental, multimodal, wide-range tele-cooperation using a humanoid robot," in *IEEE/RSJ Intelligent Conference on Intelligent RObots and Systems*, 2009.

[3] S. Kajita, K. 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*, 2003, pp. 1620–1626.

[4] K. Nishiwaki and S. Kagami, "Online walking control system for humanoids with short cycle pattern generation," *International Journal of Robotics Research*, vol. 28, no. 6, pp. 729–742, 2009.

[5] J. Chestnutt, "Navigation planning for legged robots," Ph.D. dissertation, The Robotics Institute, Carnegie Mellon University, 2007.

[6] M. Stilman, K. Nishiwaki, S. Kagami, and J. J. Kuffner, "Planning and executing navigation among movable obstacles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 820–826.

[7] M. Morisawa, K. Harada, S. Kajita, S. Nakaoka, K. Fujiwara, F. Kanehiro, K. Kaneko, and H. Hirukawa, "Experimentation of humanoid walking allowing immediate modification of foot place based on analytical solution," in *IEEE Int. Conf. on Robotics and Automation*, 2007, pp. 3989–3994.

[8] T. Sugihara and Y. Nakamura, "Whole-body cooperative balancing of humanoid robot using cog jacobian," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, pp. 2575–2580.

[9] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, "Cooperative works by a human and humanoid robot," in *IEEE/RAS International Conference on Robotics and Automation (ICRA)*, 2003, pp. 2985–2991.

[10] K. Harada, S. Kajita, K. Kaneko, and H. Hirukawa, "An analytical method for real-time gait planning for humanoid robots," *Internation Journal of Humanoid Robotics*, vol. 3, no. 1, pp. 1–19, 2006.

[11] N. Mansard and F. Chaumette, "Task Sequencing for Sensor-Based Control," *IEEE Trans. on Robotics*, vol. 23, no. 1, pp. 60–72, February 2007.

[12] B. Siciliano and J.-J. Slotine, "A General Framework for Managing Multiple Tasks in Highly Redundant robotic Systems," in *IEEE Int. Conf. on Advanced Robotics*, 1991, pp. 1211–1216.

[13] O. Khatib, "A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation," *Int. J. of Robotics Research*, vol. 3, no. 1, pp. 43–53, 1987.

[14] J. Park and O. Khatib, "Contact Consistent Control Framework for Humanoid Robots," in *IEEE Int. Conf. on Robotics and Automation*, 2006.

[15] N. Mansard, O. Stasse, P. Evrard, and A. Kheddar, "A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks," in *International Conference on Advanced Robotics (ICAR)*, no. 119, 2009.

[16] N. Perrin, O. Stasse, F. Lamiraux, P. Evrard, and A. Kheddar, "On the problem of online footsteps correction for humanoid robots," in *National Conference of the Robotic Society of Japan*, 2009, pp. 301–04.

[17] O. Stasse, P. Evrard, N. Perrin, N. Mansard, and A. Kheddar, "Direct interaction between non-experimented users and hrp-2." [Online]. Available: http://staff.aist.go.jp/olivier.stasse/DirectInteractionPublic.wmv

[18] ——, "Direct interaction between hrp-2 and a human." [Online]. Available: http://staff.aist.go.jp/olivier.stasse/Humanoids2009.wmv

[19] H. Diedam, D. Dimitrov, P.-B. Wieber, K. Mombaur, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *IEEE/RSJ Intelligent Conference on Intelligent RObots and Systems*, 2008.