Omnidirectional Walking and Active Balance for Soccer Humanoid Robot

Nima Shafii[1,2,3], Abbas Abdolmaleki[1,2,4] , Rui Ferreira[1], Nuno Lau[1,4], Luis Paulo Reis[2,5]

[1]IEETA - Instituto de Engenharia Eletrónica e Telemática de Aveiro, Universidade de Aveiro
[2]LIACC - Laboratório de Inteligência Artificial e Ciência de Computadores, Universidade do Porto
[3]DEI/FEUP - Departamento de Engenharia Informática, Faculdade de Engenharia, Universidade do Porto
[4]DETI/UA - Departamento de Eletrónica, Telecomunicações e Informática, Universidade de Aveiro
[5]DSI/EEUM - Departamento de Sistemas de Informação, Escola de Engenharia da Universidade do Minho
nima.shafii@fe.up.pt, abbas.a@ua.pt, rui.ferreira@fe.up.pt, nunolau@ua.pt, lpreis@dsi.uminho.pt

Abstract. Soccer Humanoid robots must be able to fulfill their tasks in a highly dynamic soccer field, which requires highly responsive and dynamic locomotion. It is very difficult to keep humanoids balance during walking. The position of the Zero Moment Point (ZMP) is widely used for dynamic stability measurement in biped locomotion. In this paper, we present an omnidirectional walk engine, which mainly consist of a Foot planner, a ZMP and Center of Mass (CoM) generator and an Active balance loop. The Foot planner, based on desire walk speed vector, generates future feet step positions that are then inputs to the ZMP generator. The cart-table model and preview controller are used to generate the CoM reference trajectory from the predefined ZMP trajectory. An active balance method is presented which keeps the robot's trunk upright when faced with environmental disturbances. We have tested the biped locomotion control approach on a simulated NAO robot. Our results are encouraging given that the robot has been able to walk fast and stably in any direction with performances that compare well to the best RoboCup 2012 3D Simulation teams.

Keywords: Bipedal Locomotion, Gait Generation, Foot Planner, Active Balance

1. Introduction

In robotics, bipedal locomotion is a form of locomotion where robot moves by means of its two legs. This movement includes any type of omnidirectional walking or running. A humanoid robot is a robot with its overall appearance is based on the human body, allowing the robot to move on its two legs.

Humanoid robots must have capability to adjust speed and direction of its walk to perform its tasks. Due to having a huge amount of controller design space, as well as being an inherently nonlinear system, it is very difficult to control the balance of the humanoid robot during walking. The ZMP [1] criterion is widely used as a stability measurement in the literature. For a given set of walking trajectories, if the ZMP trajectory keeps firmly inside the area covered by the foot of the support leg or the convex hull containing the support legs, this biped locomotion will be physically feasible and the robot will not fall while walking.

A Biped walking trajectory can be derived from a predefined ZMP reference by computing feasible body swing or CoM trajectory. The CoM trajectory can be calculated by a simple model, approximating the bipedal robot dynamics, such as Cart-table model or linear inverted pendulum model [2].

There is not a straightforward way to compute CoM from ZMP by solving the differential equations of the cart-table model. The approaches presented previously, on how to tackle this issue, are organized into two major groups, optimal control approaches and analytical approaches. Kajita has presented an approach to find the proper CoM trajectory, based on the preview control of the ZMP reference, which makes the robot able to walk in any direction [3]. This is a dominant example of the optimal control approach. Some analytical approaches were also developed based on the Fourier series approximation technique, which can create straight walking reference [4].

Although making a humanoid robot walk in a straight or curved line is very important and motivating for researchers, generating other types of walking, such as side and diagonal walking, and being able to change the direction of the walk, can improve the ability of a humanoid robot to maneuver with more agility in a dynamic environment as, for example, a soccer field. Recently, several researches focused on the locomotion of humanoid soccer robots have been published, which are mainly based on the ZMP control approaches [5][6][7]. Although many of them have a brief explanation of their walk engine, the foot planner and ZMP controller parts have not been explained in detail. In addition, most of them do not include an active balance method.

In this paper, we present an omnidirectional walk engine to develop an autonomous soccer humanoid robot, which mainly consist of a Foot planner, a ZMP and Center of Mass (CoM) generator and an Active balance loop. The Foot Planner generates footprint based on desired walk speed and direction. The ZMP generator generates the ZMP trajectory, which is the position of the center of the support foot in each walk step. The CoM reference trajectory is obtained by using a ZMP preview controller on the predefined ZMP trajectory. Position trajectories of the swing foot are generated through Bézier curves based on predefined footsteps. The swing foot orientation is also kept parallel to the ground to reduce the effect of the contact force. An active balance method is used to modify the foot position in order to keep the robot stable during the walk in face of external perturbations. Finally, leg joint angles are calculated based on swing foot positions, CoM references and support foot position by using inverse kinematics, and then joints are controlled by simple independent PID position controllers.

The reminder of the paper is organized as follows. The next section outlines our proposed omnidirectional walk engine. Then, the foot planner algorithm, which is used to plan the next feet positions and generate the ZMP trajectory, will be explained. Section 2.2 explains the preview controller applied to the biped walking to generate the feasible CoM reference trajectory from predefined ZMP trajectory. Our approach for the active balance, in order to keep the robot stable during walking, in the face of the external perturbation, is described in section 2.3. Experimental results on simulated NAO robot are presented and discussed in Section 3. General conclusions and future works are discussed in the last section.

2. Omnidirectional Walking Engine

This section, presented the design of the walking controllers to enable the robot with an omnidirectional walk. Developing an omnidirectional walk gait is a complex task made of several components. In order to get a functional omnidirectional walk, it is necessary to decompose it in several modules and address each module independently. Figure 1, shows the general architecture of walk engine modules and how they interact with each other to achieve stable walking.

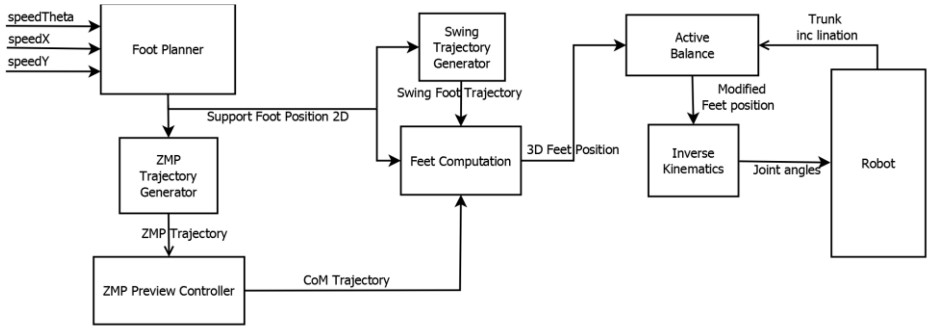

Fig. 1: Architecture of the Walk engine modules

In the following sections, the main modules such as foot planner, active balance and ZMP preview controller will be explained in detail.

2.1 Foot Planner

The footstep planner generates the support foot future positions that the robot will use. During the walking process, we need to plan future steps based on current state of feet, desired walk vector and the preview controller look ahead duration. At the initial state, the robot is in double support and CoM is located at the center of the line that connects the feet centers. After changing the CoM to the area of the support foot, the robot lifts its swing foot and moves it to next footprint. The CoM moves from its initial location to a new position. Figure 2 shows how we generate the next position and orientation of right (swing) foot based on velocity vector (V_x, V_y, w) in XY plane. We first calculate the next position of CoM by multiplying the time duration of one step by the input velocity. This gives the linear position and orientation change that can be used to determine the next position and orientation of CoM (x,y,θ). Then, we calculate a reference point, which is used to compute the next swing foot position. The reference point has a 55mm distance (half distance of two legs in NAO robot), at 90 degree, from the support foot. From the previous reference point and the CoM position, the next reference point can be determined, from which the next support position can be derived. In order to rotate CoM θ degrees we rotate the target footstep θ degrees relative to the CoM frame.

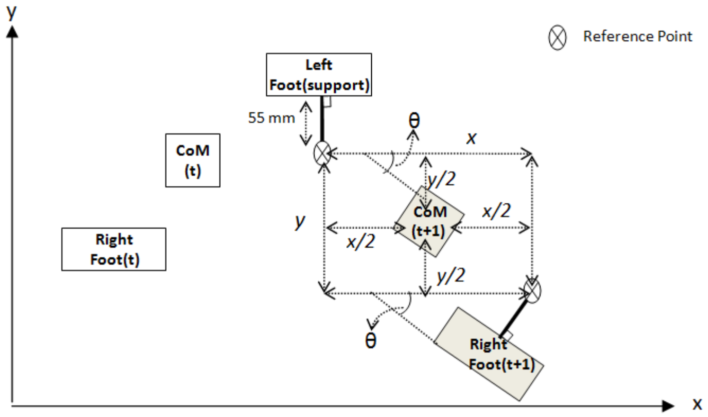

Fig. 2: Next step Calculation of foot planner for walk vector (x,y,θ).

We also take into consideration two constraints when calculating target foot:

1. Foot reachability
2. Self-Collision of feet

Foot reachability means whether the robot is capable to move its foot to the calculated target footprint or not, and self-collision means whether the feet is colliding with each other or not. We consider these constraints by defining maximum and minimum distances of feet in x and y axis, which should not be violated by new planned step and can be shown as follows.

$$A < Rightfoot.x - Leftfoot.x < B$$
$$C < Rightfoot.y - Leftfoot.y < D$$

A, B, C and D are constant parameters that represent the maximum and minimum relative feet positions in the global frame.

2.2 Cart-Table Model and CoM Reference Generator

Many popular approaches used for bipedal locomotion are based on the ZMP stability indicator and cart-table model. ZMP cannot generate reference walking trajectories directly but it can indicate whether generated walking trajectories will keep the balance of a robot or not. Nishiwaki proposed to generate walking patterns by solving the ZMP equation numerically [7]. Kajita assumed that biped walking is a problem of balancing a cart-table model [2], since in the single supported phase, human walking can be represented as the Cart-table model or linear inverted pendulum model [3].

Biped walking can be modeled through the movement of ZMP and CoM. The robot is in balance when the position of the ZMP is inside the support polygon. When the

ZMP reaches the edge of this polygon, the robot loses its balance. Biped walking trajectories can be derived from desired ZMP by computing the feasible CoM. The possible body swing can be approximated using the dynamic model of a Cart-on-a-table.

Cart-table model has some assumptions and simplifications in its model [3]. First, it assumes that all masses are concentrated on the cart. Second, it assumes that the support leg does not have any mass and represents it as a massless table. Although these assumptions seem to be far from reality, modern walking robots usually have heavy trunks with electronics circuits and batteries inside. Therefore, the effect of leg mass is relatively small. Figure 3 shows how robot dynamics is modeled by a cart-on-a-table and its schematic view.

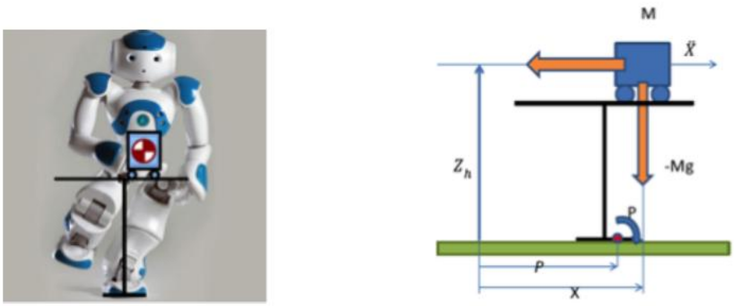

Fig. 3: Schematic view of Cart-table model and a humanoid robot

Two sets of cart-table are used to model 3D walking. One is for movements in frontal plane; another is for movements in coronal plane. The position of Center of Mass (CoM) M is x and Z_h defined in the coordinate system O. Gravity g and cart acceleration create a moment T_p around the center of pressure (CoP) point P_x. The Equation (1) provides the moment or torque around P.

$$T_p = Mg(x - P_x) - M\ddot{x}z_h \tag{1}$$

We know from [8] that when the robot is dynamically balanced, ZMP and CoP are identical, therefore the amount of moment in the CoP point must be zero, $T_p=0$. By assuming the left hand side of equation (1) to be zero, equation (2) provides the position of the ZMP. In order to generate proper walking, the CoM must also move in coronal plane, hence another cart-table must be used in y direction. Using the same assumption and reasoning equation (3) can be obtained. Here, y denotes the movement in y.

$$P_x = x - \frac{Z_h}{g}\ddot{x} \tag{2}$$

$$P_y = y - \frac{Z_h}{g}\ddot{y} \tag{3}$$

In order to apply cart-table model in a biped walking problem, first the position of the foot during walking must be planned and defined, then based on the constraint of ZMP position and support polygon, the ZMP trajectory can be designed. In the next step, the position of the CoM must be calculated using differential equations (2) (3). Finally, inverse kinematics is used to find the angular trajectories of each joint based on the planned position of the foot and calculated CoM. Two different inverse kinematic approaches, which were applied on the NAO humanoid soccer robot can be found in [8] [9].

The main issue of applying Cart-table model is how to solve its differential equations. Even though theoretically CoM trajectory can be calculated by using the exact solution of the Cart-table differential equations, applying calculated trajectory is not straightforward in a real biped robot walking because the solution consists of unbounded *cosh* functions, and the obtained CoM trajectory is very sensitive to the time step variation of the walking.

An alternative robust CoM trajectory generation method can be found in [4], in which the solution of the Cart-pole model differential equation is approximated based on Fourier representation of the ZMP equation. Kajita et. al also presents a very applicable approach to calculate the position of the CoM from the cart-table model. This approach is based on the preview control of the ZMP reference.

2.2.1 ZMP Preview Controller

In this section, the ZMP Preview Control approach proposed by Kajita et al [3] and an extended explanation of the method by Park [10] is presented. The jerk x of the cart areas of the system is assumed as input u of the cart table dynamics ($\frac{d\ddot{x}}{dt}$=u). Considering this assumption, the ZMP equations (2) (3) can be converted to a strongly appropriate dynamical system which is presented in equation (4), where P is the position of the ZMP and $X = (\mathrm{x}, \dot{\mathrm{x}}, \ddot{\mathrm{x}})$ is the state of CoM.

$$\frac{\mathrm{d}}{\mathrm{dt}}\begin{pmatrix}\mathrm{x}\\ \dot{\mathrm{x}}\\ \ddot{\mathrm{x}}\end{pmatrix} = \begin{pmatrix}0\ 1\ 0\\ 0\ 0\ 1\\ 0\ 0\ 0\end{pmatrix}\begin{pmatrix}\mathrm{x}\\ \dot{\mathrm{x}}\\ \ddot{\mathrm{x}}\end{pmatrix} + \begin{pmatrix}0\\ 0\\ 1\end{pmatrix}\mathrm{u} \qquad (4)$$

$$\mathrm{P} = \begin{pmatrix}1 & 0 & -\frac{\mathrm{Z_h}}{\mathrm{g}}\end{pmatrix}\begin{pmatrix}\mathrm{x}\\ \dot{\mathrm{x}}\\ \ddot{\mathrm{x}}\end{pmatrix}$$

Using this Cart table model, a digital controller is designed which allows the system output to follow the input reference. The discretized system of the equation (5) is given below

$$x(k+1) = AX(k) + Bu(k) \qquad (5)$$
$$P(k) = CX(k)$$

Where

$$A = \begin{bmatrix}1 & \Delta t & \Delta t^2/2\\ 0 & 1 & \Delta t\\ 0 & 0 & 1\end{bmatrix}, B = \begin{bmatrix}\Delta t^3/6\\ \Delta t^2/2\\ \Delta t\end{bmatrix}, C = \begin{bmatrix}1 & 0 & -\frac{z_h}{g}\end{bmatrix}$$

By assuming the incremental state $\Delta X(k) = X(k) - X(k-1)$, the state is augmented as $\tilde{X} = \begin{bmatrix} p(k) \\ \Delta X(k) \end{bmatrix}$, and consequently the equation (6) is rewritten as

$$\begin{aligned} \tilde{X}(k+1) &= \tilde{A}\tilde{X}(k) + \tilde{B}u(k) \\ P(k) &= \tilde{C}\tilde{X}(k) \end{aligned} \tag{6}$$

Where

$$\tilde{A} = \begin{bmatrix} 1 & CA \\ 0 & A \end{bmatrix}, \tilde{B} = \begin{bmatrix} CB \\ B \end{bmatrix}, \tilde{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$

Calculation of digital controller input and error signal are presented in equation (7). Figure (4) shows the block diagram of the system.

$$u(k) = -G_i \sum_{i=0}^{k} e(i) - G_x x(k) \tag{7}$$

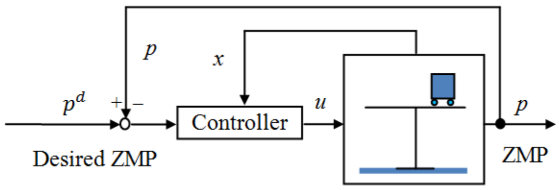

Fig. 4: ZMP Preview control Diagram

Here, G_i and G_x are assumed as the gain for the ZMP tracking error and the gain for state feedback respectively. The ZMP tracking error is $e(i) = P - P^d$ and k denoted the k^{th} sample time. It was reported that the controller of (7) was not able to follow the reference ZMP sufficiently. The main cause of this issue is the inherent phase delay. For addressing this problem, the original digital controller is redesigned in equation (8).

$$u(k) = -G_i \sum_{i=1}^{k} e(i) - G_x x(k) - \sum_{j=1}^{NL} G_p P^d(k+j) \tag{8}$$

The third term consists of the planed ZMP reference up to NL samples in future. Since this controller uses future information, it is called a preview controller and the gain $G_p(i)$ is called the preview gain. Experience shows that one second of future desired walking is sufficient for the Preview controller to generate a smooth trajectory, therefore, parameter NL can be calculated based on the incremental time step, $NL = \frac{1}{\Delta t}$.

A detailed explanation of the preview control approach as an optimal controller technique can be found in [11]. The optimal gain, G_i, G_x, are calculated by solving discrete algebraic Riccati equation.

$$\tilde{P} = \widetilde{A^T}\tilde{P}\tilde{A} - \widetilde{A^T}\tilde{P}\tilde{B}(R + \widetilde{B^T}\tilde{P}\tilde{B})^{-1}\widetilde{B^T}\tilde{P}\tilde{A} + \tilde{Q} \tag{9}$$

Where $\widetilde{Q} = diag\{Q_e, Q_x\}$. Then, the optimal gain is defined by

$$\tilde{G} = (R + \widetilde{B^T}\tilde{P}\tilde{B})^{-1}\widetilde{B^T}\tilde{P}\tilde{A} = [G_i \quad G_x] \qquad (10)$$

The optimal preview gain is recursively computed as follows.

Considering $\widetilde{A_c} = \tilde{A} - \tilde{B}\tilde{G}$

$$G_p(i) = (R + \widetilde{B^T}\tilde{P}\tilde{B})^{-1}\widetilde{B^T}\tilde{X}(i-1) \qquad (11)$$

$$\tilde{X}(i) = \widetilde{A_c^T}\tilde{X}(i-1)$$

Where $G(1) = -G_i \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $\tilde{X}(1) = -\widetilde{A_c^T}\tilde{P}\begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $Q_e = 1, Q_x = 0$ and $R = 1 \times 10^{-6}$, are assumed. Figure (5) shows G_p profile towards the future. We can observe that the magnitude of the preview gain quickly decreases, thus the ZMP reference in the far future can be neglected.

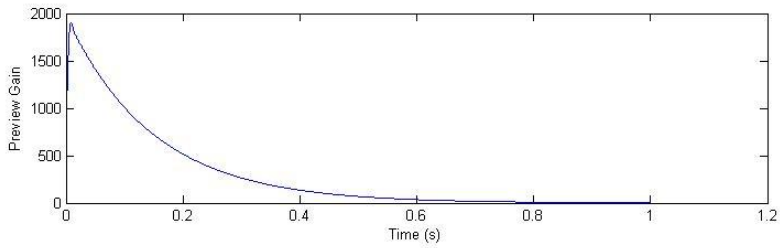

Fig. 5: Gain calculated based on the preview controller

2.3 Active Balance

The cart-table model has some simplifications in biped walking dynamics modeling; in addition, there is inherent noise in leg's actuators. Therefore, keeping walk balance generated by cart-table model cannot be guaranteed. In order to reduce the risk of falling during walking, an active balance technique is applied.

The Active balance module tries to maintain an upright trunk position, by reducing variation of trunk angles. One PID controller is designed to control the trunk angle to be Zero. Trunk angles, pitch and roll, can be derived from inertial measurement unit of the robot. When the trunk is not upright, instead of considering a coordinate frame attached to the trunk of biped robot, position and orientation of feet are calculated with respect to a coordinate frame center, which is attached to the CoM position and makes the Z axes always perpendicular to the ground plane.

The transformation is performed by rotating this coordinate frame with the rotation angle calculated by the PID controller. This makes the Z axes always perpendicular to the ground plane. The feet orientation is kept constantly parallel to ground. The Transformation formulation is presented in equation (11).

$$Foot = T_{Foot}^{CoM}(pitchAng, rollAng) \times Foot \qquad (11)$$

The *pitchAng* and *rollAng* are assumed to be the angles calculated by the PID controller around *y* and *x* axis respectively. Figure 6 shows the architecture of the active balance unit.

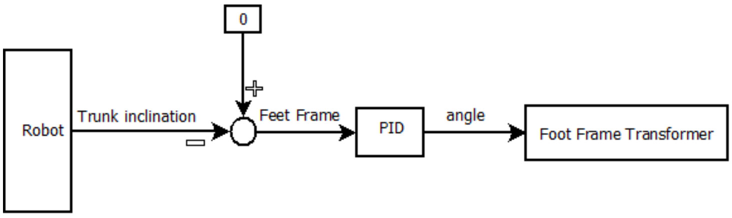

Fig. 6: Active Balance controller

3. Results and Discussion

In this study, a simulated NAO robot is used in order to test and verify the approach. The NAO model is a kid size humanoid robot that is 58 cm high and 21 degrees of freedom (DoF). The link dimensions of the NAO robot can be found in [12]. The simulation is carried out using the RoboCup soccer simulator, rcsssever3d, which is the official simulator released by the RoboCup community, in order to simulate humanoids soccer match. The simulator is based on Open Dynamic Engine and Simspark [13]. The Robocup simulator was studied and compared whit another humanoid robot simulator in [14].

In order to test the behavior and evaluate its performance, several walking scenarios were designed. The parameters of walking engine used in the walking scenarios are presented in Table 1.

Table 1: Walking scenarios parameters

Parameters	**Value**
Step Period	0.2 s
Step Height of the swing foot	0.02 m
Speed in X direction	0.5,0,-0.5 m/s
Speed in Y direction	0.3, 0,-0.3 m/s
Percentage of the Double Support Phase (DSP) to the whole step time	15 %
Height of the inverted pendulum (Z_h)	0.21 cm
Incremental sample time used in ZMP preview controller (Δt)	0.002 s

In the first scenario, the robot must walk forward with speed of 0.5 m/s for 10 seconds, then change to a sidewalk that lasts for 10 seconds with the speed of 0.3 cm/s, then he must do a diagonal back walk for another10 seconds with the speed of -0.5 m/s and -0.3 m/s in x and y directions, respectively. At the end, the robot must stop at the starting point. In order to show the ability of the developed behavior to execute the scenario, generated CoM trajectory and footprint projected on the ground plane are shown in Figure 7. Figure 8 also shows a generated curved walking, in which the robot walks with the speed of 0.2 m/s and rotates 6 degree in each walk step (w=30 deg/s).

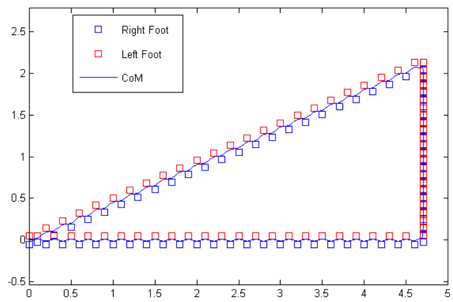

Fig. 7: CoM and footprint of the proposed walking scenario

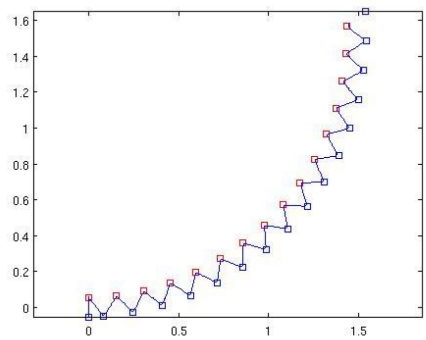

Fig. 8: CoM and footprint of a curved walk

The planned ZMP trajectory based on the first 5 walk steps footprints of the proposed walking scenario (forward walking) and the followed ZMP trajectory, achieved using the preview controller, are shown in figure 9.

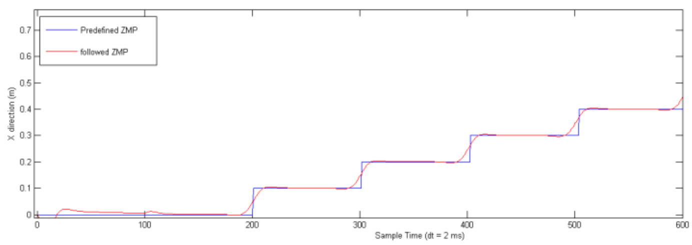

Fig. 9: ZMP trajectories for proposed the forward walk

Table 2 lists the values of some of the relevant results for the developed gait implemented on our simulated NAO compared with corresponding behavior from other RoboCup 3D simulation teams. We collect times and distances/angles for forward, backward, side and diagonal (45 degrees) walk and rotate in place. For walk without rotation the agent walked for 20 seconds and recorded its position and time at each cycle. For rotate in place, the agent completed 10 full turns and recorded its time. By analyzing the results of the last RoboCup competition, the reported results are approximated based on the average speeds of the locomotion skills for teams participating in the final round of the RoboCup 2012 competition. It is also impressive that the robot can change walking direction without reducing its speed, which for other teams is a very difficult task.

Table 2: Comparison between the performance of the locomotion generated by the proposed walk engine, with the skills performance of other teams that participating in final round of the RoboCup 2012 competition

Motion	Our Agent	UT Austin Villa 3D	RoboCanes	Bold Hearts	magmaOffenburg
Forward walk (m/s)	0.66	0.7	0.69	0.59	0.54
Backward Walk(m/s)	0.59	-	0.61	0.47	0.48
Side Walk (m/s)	0.51	-	0.47	0.42	-
Turn in Place (deg/s)	125	100	110	-	110

4. Conclusions and Future Work

Creating a good walk engine in RoboCup soccer competitions is a critical and important task for all participant teams. The walk engine should enable robots to walk stably and fast in any direction. This paper presents an implementation of an omnidirectional walk engine augmented with an active balance method for the NAO humanoid robot. The cart-table model and preview control are used to generate the CoM reference trajectory from the predefined ZMP trajectory. And the active balance method, keeps the robot's trunk upright in case of environmental disturbances. The biped locomotion control approach has been tested on a simulated NAO robot. Our results show that the robot has been able to walk fast and stably in any direction with performances that compare well with the best RoboCup 2012 teams. It is a proper approach in order to be applied on the humanoid soccer robot.

In future work, the proposed method will be tested and implemented on a real humanoid robot. Our aim is to develop a fully autonomous soccer humanoid team, in which the proposed walk engine will be used for their locomotion. Although the experimental results show the robot capability to perform omnidirectional walk in simulation environment, there is still a small gap between simulation and real soccer environment.

Acknowledgements

The first and second authors are supported by FCT under grant SFRH/BD/66597/2009 and SFRH/BD/81155/2011, respectively. This work was supported by project Cloud Thinking (funded by the QREN Mais Centro program, ref. CENTRO-07-ST24-FEDER-002031).

References

1. M. Vukobratovic, D. Stokic, B. Borovac, and D. Surla, *Biped Locomotion: Dynamics, Stability, Control and Application*. Springer Verlag, 1990, p. 349.
2. S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, “The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 239–246.
3. S. Kajita, F. Kanehiro, K. Kaneko, and K. Fujiwara, “Biped walking pattern generation by using preview control of zero-moment point,” in *IEEE International Conference on Robotics and Automation ,ICRA 2003*, 2003, pp. 1620–1626.
4. K. Erbatur and O. Kurt, “Natural ZMP Trajectories for Biped Robot Reference Generation,” *IEEE Transactions on Industrial Electronics*, vol. 56, no. 3, pp. 835–845, Mar. 2009.
5. C. Graf and T. Rofer, “A closed-loop 3D-LIPM gait for the RoboCup Standard Platform League humanoid,” in *Fourth Workshop on Humanoid Soccer Robots in IEEE conference of Humanoid Robots*, 2010, pp. 15–22.
6. D. Gouaillier, C. Collette, C. Kilner, and A. Robotics, “Omni-directional Closedloop Walk for NAO,” in *IEEE-RAS International Conference on Humanoid Robots*, 2010, pp. 448–454.
7. J. Strom and G. Slavov, “Omnidirectional walking using zmp and preview control for the nao humanoid robot,” in *RoboCup 2009: Robot Soccer World Cup,*, 2010, pp. 125–137.
8. R. Ferreira, L.P. Reis, A. P. Moreira, N. Lau, "Development of an Omnidirectional Kick For a NAO Humanoid Robot", 13th Ibero-American Conference on AI , LNAI 7637, pp 571-580 ,2012.
9. E. Domingues, N. Lau, B. Pimentel, N. Shafii, L. P. Reis, A. J. R. Neves, "Humanoid Behaviors: From Simulation to a Real Robot", 15th Port. Conf. Artificial Intelligence, EPIA 2011, LNAI 7367, pp 352-364, 2011.
10. J. Park and Y. Youm, “General ZMP preview control for bipedal walking,” in *2007 IEEE International Conference on Robotics and Automation*, 2007, no. April, pp. 2682–2687.
11. T. Katayama, T. Ohki, T. Inoue, and T. Kato, “Design of an optimal controller for a discrete-time system subject to previewable demand,” *International Journal of Control*, vol. 41, no. 3, pp. 677–699, 1985.
12. D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, “Mechatronic design of NAO humanoid,” in *Proceedings of the IEEE International Conference on Robotics and Automation (2009)*, 2009, pp. 769–774.
13. J. Boedecker and M. Asada, “SimSpark – Concepts and Application in the RoboCup 3D Soccer Simulation League,” *Autonomous Robots*, pp. 174–181, 2008.
14. N. Shafii, L. P. Reis and R. J. Rossetti, "Two humanoid simulators: Comparison and synthesis," *in 2011 6th Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 1-6, 2011.