In Proceedings of 16th RoboCup International Symposium, Mexico City, June 2012.
Best Paper Award.

Lateral Disturbance Rejection for the Nao Robot

Juan José Alcaraz-Jiménez[1], Marcell Missura[2], Humberto Martínez-Barberá[1], and Sven Behnke[2]

[1]Information and Communications Engineering, Computer Science,
Univ. of Murcia, Spain
{juanjoalcaraz,humberto}@um.es
http://robolab.dif.um.es/

[2]Autonomous Intelligent Systems, Computer Science, Univ. of Bonn, Germany
{missura,behnke}@cs.uni-bonn.de
http://ais.uni-bonn.de

Abstract. Maintaining balance in the presence of disturbances is crucial for bipedal robots. In this paper, we focus on the lateral motion component. In order to attain disturbance rejection and to quickly recover balance, we combine three different control approaches. As a principal building block, we generate center of mass trajectories with a linear model predictive controller that takes scheduled footsteps into account. Strong disturbances generate unexpected angular momenta that can compromise stability. A second control layer extends the underlying preview controller with two recovery strategies that modify the planned CoM trajectories to dampen the rotational velocity of the robot and adapt the timing of the steps according to the expected orbital energy of CoM trajectories at support exchange. Experiments with a real Nao robot show that the system is able to recover from lateral disturbances as long as the robot does not tip over the current support leg.

Keywords: Nao, Disturbances, Angular Momentum, Orbital Energy

1 Introduction

The Nao bipedal robot enjoys an increasing amount of scientific attention, especially since it has been selected to play humanoid soccer in the RoboCup standard platform league competitions. Several gaits that show reasonable performance on the soccer field have been presented for the Nao. The response to unexpected disturbances, however, remains a weakness of all gaits up to date.

Here, we are presenting a locomotion system based on the model predictive control framework (MPC), whose performance is improved by two additional controllers. The linear inverted pendulum model (LIPM) used by the MPC neglects the angular momentum of the robot, which is not a good assumption if the robot has been disturbed. To overcome this limitation, our first additional controller decreases the tipping moment around the outer border of the sole, ensuring a smooth recovery after disturbances. The second controller adjusts the

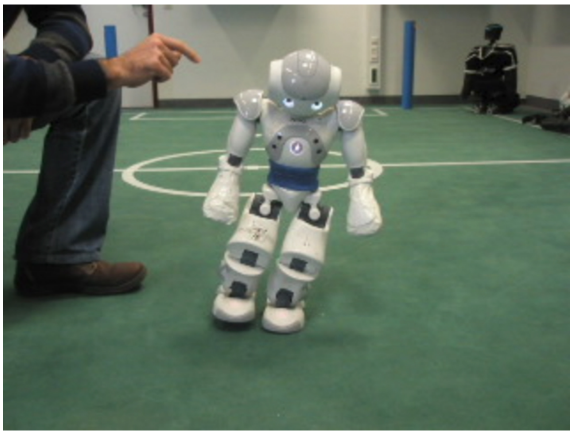

Fig. 1. Nao robot reacting to lateral disturbances.

timing of the steps to account for increased single-support durations while the robot is recovering from a lateral push. Both additional controllers are designed for the lateral motion component exclusively. With this configuration, the robot is able to reject relatively strong perturbations from the side—as long as it does not tip over the current support leg.

The importance of lateral stability is often overlooked. While in sagittal direction the swing foot can be flexibly placed virtually anywhere in front of or behind the robot, in lateral direction the location and the timing of footsteps are much more constrained. Most humanoid robots cannot cross their legs, therefore the locations on the outer side of the support leg are not available to maintain stability. Moreover, the rhythmic oscillation induced by the alternating role of support between the left and the right leg dictates a steady timing which is sensitive to disturbances and can quickly lead to a fall, if not adjusted on the fly.

This paper is structured as follows. After reviewing related work in Section 2, we describe the core of our walking engine in Section 3. Section 4 explains the feedback controllers that modify the MPC approach and Section 5 discusses the experimental results obtained.

2 Related Work

Numerous approaches have been proposed to implement dynamic walking for bipedal robots. For example, central-pattern generated omni-directional gaits proved to be an effective approach as they are used by leading teams [1, 2] in different leagues of the RoboCup competition.

On the other hand, locomotion systems based on the Linear Inverted Pendulum Model (LIPM) [3] and the Zero Moment Point (ZMP) [4] concepts have become more popular in recent years [5–8], because they provide a simpler set of

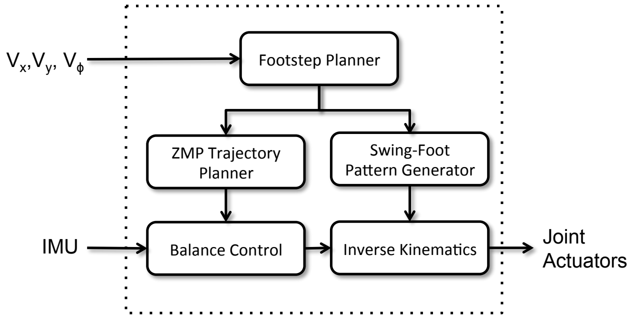

Fig. 2. Architecture of our locomotion module.

equations to generate Center of Mass (CoM) trajectories and reduce the number of parameters to tune.

In order to exploit the ZMP stability criterion, Kajita *et al.* proposed the use of Preview Control [9] to generate stable trajectories for the CoM. Following this approach, Wieber presented a slightly modified version with an analytical solution under certain constraints [10]. In this work, we utilize this solution for the generation of CoM trajectories that will be subsequently modified by another controller to reduce the angular momentum of the robot.

Kajita *et al.* described in [11] how the CoM trajectories that have been generated with a LIPM-based approach follow potential energy conserving orbits. In this work, we define a target energy level that is used to adapt the timing of the steps. The timing control is a key feature to recover the regular step frequency after strong disturbances. In a similar way, the use of potential energy conserving orbits to regulate the duration of single support stages has also been used in [12], where the focus is also set on the lateral component of the movement. However, in that work only the duration and size of the steps are adapted, but not the CoM trajectories.

3 Walking Pattern Generation

The balance controllers presented in this paper are embedded in the locomotion architecture sketched in Fig. 2. The input received from a higher behavior layer is used for the Footstep Planner to define the timing and position of future footsteps and the trajectory of the swing foot. Further details can be found in [13].

When the robot is walking, its feet swing alternately to reach the new positions of the footstep route. The trajectory that a foot follows in the air is calculated by the Swing-Foot Pattern Generator by means of Bezier curves. The output of this module is a sequence of Cartesian positions and a rotation matrix of the nonsupporting foot in the support-foot frame. These positions are delivered to the inverse kinematics module.

Additionally, to prevent the robot from falling, it is necessary to assure certain stability conditions. In this work, we will employ the LIPM model and the ZMP stability criterion. The LIPM involves two assumptions. First, the robot behaves like a single point mass concentrated at the center of the mass distribution of the body. And second, the motion of the point mass is restricted to a horizontal plane. The dynamic balance condition requires to keep the ZMP within the convex hull of the support polygons. The ZMP trajectories are generated in the ZMP Trajectory Planner module.

Given a certain state of the CoM, it is possible to utilize an optimal control strategy called *Model Predictive Control* to generate the future positions of the CoM that minimizes both, the tracking error of the ZMP trajectories, and the first derivative of the CoM acceleration. In this way, the CoM and ZMP trajectories are first discretized in constant time fragments of duration T, where a constant jerk ($\dddot{x}_k$) is applied to the CoM:

$$\begin{bmatrix} x_{k+1} \\ \dot{x}_{k+1} \\ \ddot{x}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{bmatrix} + \begin{bmatrix} T^3/6 \\ T^2/2 \\ T \end{bmatrix} \dddot{x}_k . \tag{1}$$

Following the approach described in [10], we can find an analytical solution to obtain the value of $\dddot{x}_k$

$$\dddot{x}_k = -e \left(\left(M_u^T M_u + \frac{R}{Q} I_{NxN} \right)^{-1} * M_u^T \left(M_x \hat{x} - P_k^{ref} \right) \right) . \tag{2}$$

The matrixes used in Equation (2), are defined in the expressions (3)-(6), where p_k is the reference position of the ZMP at the sample k, N is the number of reference samples and R/Q is a parameter to tune the trade-off between minimum reference tracking error and minimum jerk.

$$e = [1, 0...0], \tag{3}$$

$$M_u = \begin{bmatrix} \frac{T^3}{6} & 0 & 0 \\ \vdots & \ddots & 0 \\ (1+3N+3N^2)\frac{T^3}{6} & \dots & \frac{T^3}{6} - T\frac{z}{g} \end{bmatrix} , \tag{4}$$

$$M_x = \begin{bmatrix} 1 & T & \frac{T^2}{2} - \frac{z}{g} \\ \vdots & \vdots & \vdots \\ 1 & NT & \frac{N^2T^2}{2} - \frac{z}{g} \end{bmatrix} , \tag{5}$$

$$P_k = \left[p_k \dots p_{k+N-1} \right] . \tag{6}$$

The position of the CoM obtained in (1) is delivered to the inverse kinematics module that, together with the swing-foot pose, will generate the next position for the joint actuators. Although the open-loop execution of the locomotion approach described above is acceptable for low speeds, there are important

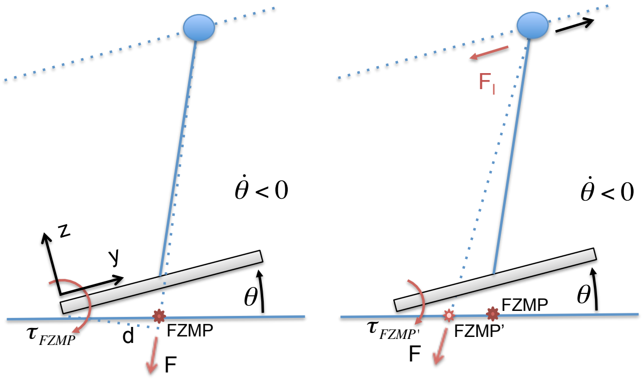

Fig. 3. On the left, the FZMP generates a moment that decreases θ. On the right, the effect of the angular velocity controller is shown. Additional acceleration of the CoM increases the inertial force and pushes the FZMP towards the axis of rotation, which decreases the restoring moment and avoids a too large rotational velocity when θ reaches the horizontal position.

deficiencies in its performance that prevent the robot from attaining a robust gait that rejects disturbances.

Since the system is based on the LIPM, the inertial effects due to rotations of the different parts of the robot, which are important in the case of strong disturbances, are neglected. In the next section, we propose a control approach that copes with this simplification.

4 Balance Control

To improve the performance of the gait pattern generation described previously, we modify the Balance Control module to include controllers that regulate the angular velocity of the CoM and the timing of the next footstep. Since this work focuses on the lateral component of the walking motion, we restrict the equations to the frontal plane.

4.1 Angular Velocity Control

The ZMP specifies the point on the ground where the tipping moment acting on the robot, due to gravitational and inertial forces, equals zero. This point can only exist within the limits of the convex hull of the support polygons. When the ground projection of gravitational and inertial forces lies outside the convex hull, this point is called Fictitious Zero Moment Point (FZMP) [14]. The FZMP involves the presence of a moment that causes a rotational acceleration of the CoM around the closest point of the convex support region.

Given an angle θ between the sole of the support foot and the ground, the total force F resulting from gravity and inertia generates a torque around the

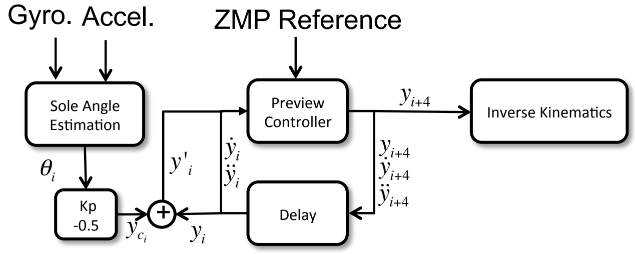

Fig. 4. The linear model predictive controller is complemented by the angular velocity controller.

contact point between sole and ground, as illustrated in Fig. 3. For a robot walking on a flat surface, ZMP-based gaits usually assume $\theta(t) = 0$, and place the target ZMP position approximately at the center of the sole. If a small disturbance occurs, θ will be different from zero and the support polygon reduces to a point at the edge of the sole. In this situation, the target ZMP should be placed theoretically at the edge of the foot to avoid applying any additional torque on the robot, which is the goal of the ZMP stability criterion. Nevertheless, it is common practice to neglect sole angles different from zero and to keep the projection of inertial and gravitational forces approximately at the same point, which is no longer a ZMP but a FZMP. This fact is generally beneficial for balance, because the torque generated by the FZMP will increase the rotational velocity of the sole such that θ decreases, as displayed in Fig. 3 (left).

When the sole reaches the horizontal position again, the robot is rotating with a nonzero angular velocity $\dot{\theta}$, and has therefore an angular momentum that forces the sole to keep rotating beyond the position $\theta = 0$. The rotational velocity of the sole $\dot{\theta}$ is then reduced by the torque that appears at the opposite side of the sole, but high magnitudes of $\dot{\theta}$ in this instant can directly lead to a fall or cause the landing of the swing-foot at an unexpected time and induce further instabilities.

Our strategy to mitigate this problem is to add an offset y_c to the position of the CoM proportional to the estimated angle of the sole

$$y_{c_i} = -K_c \theta_i, \tag{7}$$

where K_c is the positive proportional gain of the controller and i is the discrete time index.

In this way, when the CoM of the robot rotates around the edge of the sole of the supporting foot, the controller will accelerate the CoM to change the position of the FZMP. While the angle between the sole and the ground is growing, the FZMP is shifted away from the rotation axis to increase the torque that decelerates the rotation. On the other hand, when the angle of the sole is decreasing to recover the horizontal position, the FZMP is shifted towards the axis of rotation (Fig. 3 right) to reduce the torque and to reach the horizontal

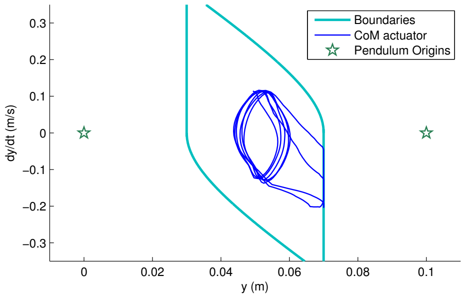

Fig. 5. The states of the CoM delivered by the preview controller are bounded in the phase space. A robot walking on the spot is pushed twice and the trajectory of the CoM reaches the limits.

position with a moderate angular velocity. The integration of the angular velocity controller into the Linear Model Predictive Controller is illustrated in Fig. 4.

Since the lag estimated for the system is four cycles of $10ms$, the actuator commands must be delayed by this amount of time before fusing the control signal of the angular velocity controller and the state of the CoM used by the linear model predictive controller.

In order to obtain an estimate of the sole angle, we estimate the orientation and angular velocity of the torso using the inertial sensors and subtract the delayed torso angle as it was commanded by the actuators four cycles before. The difference between the two angles is equal to the angle of the sole with respect to the floor.

The gyrometers provide accurate angular velocity measurements that can be integrated to obtain an estimate of the torso orientation. This orientation is fused with the angle estimated by the accelerometers to reduce the cumulative error generated by the integration of angular velocity. Since the use of the accelerometers is not sufficient to contain this drift, the FSR sensors in the feet of the robot are used to reset the zero position of the sole angle when a flat contact is detected.

When the robot is pushed from a side, the angular velocity controller will generate a yielding motion of the torso away from the pushing force to avoid the inclination of the sole. This absorption effect must be limited, however, because it can take the CoM to a position beyond the support foot, from where it is not possible to recover. To avoid this situation, the permitted CoM states are bounded in the phase space, as depicted in Fig. 5.

4.2 Step Timing Control

The combination of the linear model predictive controller and the angular velocity controller improves the balance of the robot significantly. Nevertheless, for external disturbances exceeding a certain magnitude, it is necessary to adapt the timing of the step.

Our strategy is to define a target orbital energy E_t in the frame of the next support foot and to calculate the remaining time to reach that orbital energy using the current pendulum origin. To determine the CoM trajectory with respect to the current pendulum origin, we use the 3D-LIPM [3] equations

$$y(t) = y_0 \cosh(kt) + \frac{\dot{y}_0}{k} \sinh(kt), \tag{8}$$

$$\dot{y}(t) = y_0 k \sinh(kt) + \dot{y}_0 \cosh(kt). \tag{9}$$

The position and velocity of the CoM at $t = 0$ are y_0 and $\dot{y}_0$, $k = \sqrt{g/h}$, where g is the gravitational acceleration and h the CoM height. Since we know that the CoM will "rebound" from the current support foot and accelerate towards the next support foot, we set y_0 to the apex of the trajectory and $t = 0$ at the time when the CoM is at y_0. Equations (8) and (9) can then be simplified to

$$y(t) = y_0 \cosh(kt), \tag{10}$$

$$\dot{y}(t) = y_0 k \sinh(kt). \tag{11}$$

The current time and the apex position are calculated as

$$t_n = \frac{\operatorname{atanh}\left(\frac{\dot{y}_n}{y_n k}\right)}{k}, \tag{12}$$

$$y_0 = \frac{y_n}{\cosh(k t_n)}, \tag{13}$$

where y_n and $\dot{y}_n$ are the current estimated position and velocity of the CoM with respect to the center of the current support foot.

Given the length of the step S_y, we can calculate the orbital energy relative to the frame of the next support foot:

$$E_{sw}(t) = \frac{1}{2}\left(\dot{y}^2(t) - \frac{g}{z_c}(y(t) - S_y)^2\right). \tag{14}$$

Our goal is to change the support foot when E_{sw} has the value of the target orbital energy E_t. Substituting (10), (11), and (13) in (14), we can calculate the optimal instant t_s for the the support exchange

$$t_s = \frac{\operatorname{acosh}\left(\frac{2E_t + k^2(S_y^2 + y_0^2)}{2 y_0 k^2 S_y}\right)}{k}. \tag{15}$$

The remaining time to the optimal exchange instant will be $\Delta t_s = t_s - t_n$.

We calculate a limit case where the pendulum origin is placed at the limit of the sole to estimate the minimum value for Δt_s. If the current scheduled time to exchange the support $\Delta t_{s_{sch}}$ is less than $\Delta t_{s_{min}}$, we add a delay of just one control cycle to $\Delta t_{s_{sch}}$. This way, the stepping motion is delayed as soon as the instability is detected and we achieve robustness against noisy estimations of the CoM position and velocity, since only large disturbances that are repeatedly detected in multiple control cycles will cause a significant delay of the step timing.

Finally, the support exchange is delayed until the CoM has reached at least 45% of the distance between the current and the next pendulum origin to enforce a symmetrical pose of the robot at support exchange with the CoM half way between the feet.

5 Experimental Results

In this section, we describe the experiments performed to validate our disturbance rejection approach. The platform used is the commercial humanoid robot Nao, developed by the French company Aldebaran Robotics. During the experiments, the robot is placed on a carpet similar to the ones used at RoboCup competitions.

The goal of the first experiment is to validate the performance of our angular velocity controller in combination with the preview control approach. During this experiment, feet motion is disabled so that the robot stands still, but the linear predictive controller is active and the ZMP is held fixed in the middle between the two feet. With this setup, the robot is tilted laterally by 45 degrees, so that it is standing on the outer edge of the sole. Then, the robot is released and allowed to freely swing back to the middle position. Fig. 6 illustrates the performance gained from the angular velocity controller. When the controller is disabled, the angular momentum accumulated by the robot during the time that the torso needs to recover the vertical position compels the robot to keep rotating, and thus the robot oscillates from one side to the other during the next four seconds. On the other hand, the angular velocity controller notably compensates the overshoot of the angular velocity and the equilibrium position is recovered in 1.25 seconds. When the angular velocity controller is disabled,

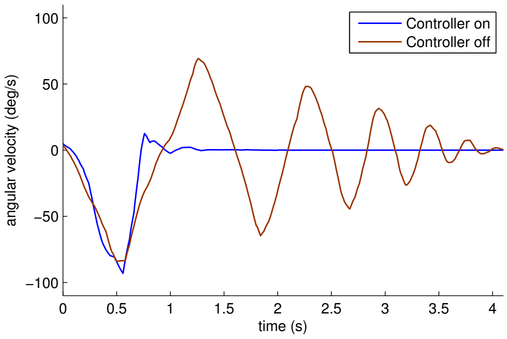

Fig. 6. Overshooting is reduced when the angular velocity controller is enabled.

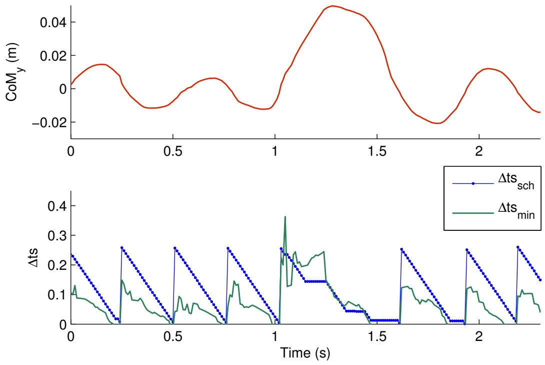

Fig. 7. The robot is pushed at t=1s while walking on the spot and adapts the duration of the step online. The scheduled time to change the support foot $\Delta t_{s_{sch}}$ is delayed during three phases. The first two times the delay happens because the estimated lower bound $\Delta t_{s_{min}}$ exceeds the scheduled time. The third time, $\Delta t_{s_{sch}}$ is delayed until the CoM covers 45% of the distance between the current and the next support foot.

the magnitude of the peak angular velocity in the first rebound is reduced only by 18%. Enabling the controller increases the reduction rate to 88%.

The goal of the second experiment is to demonstrate the performance of the disturbance rejection system while walking on the spot. The value used for the R/Q parameter of the linear model predictive controller is $1e-7$, the CoM height 0.255 m, the default distance between the feet is 0.1 m, and the K_c gain for the momentum controller is set to 0.5. The duration of every step is 0.25 ms and 5% of the time both feet are on the ground. The reference trajectories for the ZMP jump from one foot to the other at the support exchange time and have an offset of 0.01 m towards center of the robot in the lateral dimension and an offset of 0.005 m in forward direction with respect to the center of the foot.

An example for the effectiveness of the step timing control is shown in Fig. 7. Here one can observe that after a strong disturbance, the robot delays the next step in three phases and continues its normal walking rhythm afterwards.

Fig. 8 shows CoM trajectories during the pushing experiment. In the first row, the preview controller is working in open-loop mode. Before the robot is disturbed at t=2.5 s, the estimated position of the CoM follows a rhythmic pattern which is not synchronized with the CoM position sent to the inverse kinematics module. For example, at t=1.75 s, the measured and commanded CoM positions have opposite signs. After the disturbance, the CoM rebounds from the support leg, but the robot tips over on the opposite side.

In the second row, the angular velocity controller is enabled and the measured and commanded CoM trajectories stay synchronized. However, when the robot is pushed, the system is not able to recover the regular pace in time and tries to lift the foot that supports the robot. As a result, the robot needs two seconds to fully recover its balance.

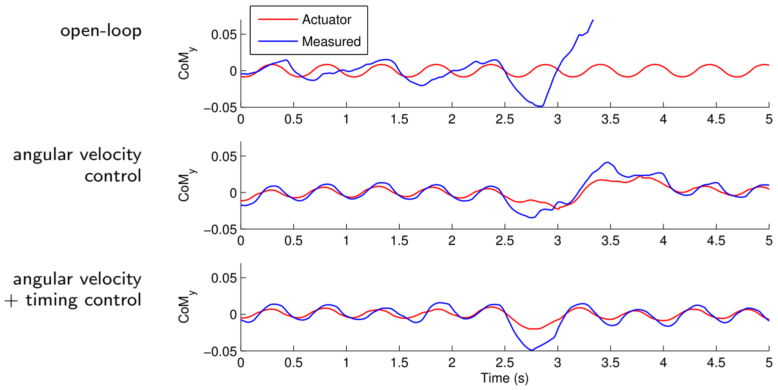

Fig. 8. Performance comparison of three different configurations for disturbance rejection.The robot is pushed at t=2.5 s while walking on the spot. On the top, the preview controller is working in open-loop mode. In the second row, the angular velocity controller is enabled. In the last row, the angular velocity controller and the timing control are both switched on.

This problem is solved when the timing controller is added to the configuration, as shown in the third row. In that case, the regular pace is recovered only 0.5 seconds after the disturbance.

The step timing controller alone (without angular velocity control) has worse performance than the open loop mode in the walking on the spot experiment. The open loop controller ignores the real position of the CoM and frequently supports the robot with the swing foot. In such situations, the LIPM is no longer a suitable model to describe the dynamics of the system because it does not take into account vertical oscillations. On the other hand, when the timing controller is enabled, the robot succeeds in using the scheduled foot to support the robot, but angular momentum cumulates through steps and causes the robot to tip over.

The accompanying video material [15] shows the Nao robot dealing with several disturbances while walking on the spot and recovering from states with high angular velocity.

6 Conclusions

We presented a bipedal locomotion system that combines three different approaches to reject disturbances and to rapidly recover the default posture and gait frequency. The base of the system is a model predictive controller that generates CoM trajectories based on footsteps scheduled for the future. The internal state of the CoM used by this controller is modified to reduce the angular momentum of the robot. Finally, the duration of every step is dynamically adapted to make sure that the orbital energy of the next step is above a minimal threshold.

In future work we will extend the concepts employed to control the lateral component of the walking motion to the sagittal dimension. The main difference in this case is that the velocity of the CoM does not change its sign in every step. On the other hand, the landing position of the feet can be freely modified, since it is not limited by self-collisions.

7 Acknowledgement

This work is supported by the Spanish Ministry of Education through its FPU program under grant AP2008-01816. Additionally, funding for the project is provided by Deutsche Forschungsgemeinschaft (German Research Foundation, DFG) under grant BE 2556/6.

References

1. S. Behnke, *Online trajectory generation for omnidirectional biped walking*, in IEEE Int. Conf. on Robotics and Automation (ICRA), 1597-1603, 2006.
2. C. Graf, A. Härtl, T. Röfer and T. Laue *A Robust Closed-Loop Gait for the Standard Platform League Humanoid*, in Workshop on Humanoid Soccer Robots of the IEEE-RAS Int. Conf. on Humanoid Robots, 2009.
3. S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi and H. Hirukawa, *The 3D Linear Inverted Pendulum Mode: A simple modeling for a biped walking pattern generation*, in Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2001.
4. M. Vukobratovic, A.A. Frank, and D. Juricic, *On the Stability of Biped Locomotion*, in IEEE Transactions on Biomedical Engineering, 17(1):25-36, 1970.
5. S. Czarnetzki, S. Kerner, O. Urbann, *Observer-based dynamic walking control for biped robots*, in Robotics and Autonomous Systems 57(8):839-845, 2009.
6. D. Gouaillier, D. Collette and C. Kilner, *Omni-directional closed-loop walk for NAO*, in IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids), 448-454, 2010.
7. F. Xue, X. Chen, J. Liu, and D. Nardi, *Real Time Biped Walking Gait Pattern Generator for a Real Robot*, in RoboCup 2011: Robot Soccer World Cup XV. Lecture Notes in Articial Intelligence, 2011.
8. C. Graf and T. Röfer, *A center of mass observing 3D-LIPM gait for the RoboCup Standard Platform League humanoid*, in RoboCup 2011: Robot Soccer World Cup XV. Lecture Notes in Articial Intelligence, 2011.
9. S. Kajitak, 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 Int. Conf. on Robotics and Automation, 2003.
10. P.-B. Wieber, *Trajectory Free Linear Model Predictive Control for Stable Walking in the Presence of Strong Perturbations*, in IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids), 2006.
11. S. Kajita, T. Yamaura and A.Kobayashi, *Dynamic walking control of a biped robot along a potential energy conserving orbit*, in Robotics and Automation, 1992.
12. M. Missura and S. Behnke, *Lateral Capture Steps for Bipedal Walking*, in IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids), 2011.
13. J.J. Alcaraz-Jiménez, D. Herrero-Pérez and H. Martínez-Barberá, *A Closed-Loop Dribbling Gait for the Standard Platform League*, in Workshop on Humanoid Soccer Robots of the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids), 2011.
14. M. Vukobratovic and B. Borovac, *Zero-moment point - Thirty five years of its life*, in International Journal of Humanoid Robotics 1(1), 157-173, 2004.
15. J.J. Alcaraz-Jiménez, M. Missura, H. Martínez-Barberá and S. Behnke. *Nao robot performs disturbance rejection and angular speed reduction tests.* http://www.ais.uni-bonn.de/movies/LateralControlNao.mp4.