

Article

# Modeling, Simulation and Control of the Walking of Biped Robotic Devices—Part III: Turning while Walking

Giuseppe Menga <sup>\*,†</sup> and Marco Ghirardi <sup>†</sup>

Control and Computer Engineering Department, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10124 Torino, Italy; marco.ghirardi@polito.it

\* Correspondence: menga@polito.it; Tel.: +39-011-0907272

† These authors contributed equally to this work.

Academic Editor: Chien-Hung Liu

Received: 24 November 2015; Accepted: 11 March 2016; Published: 22 March 2016

**Abstract:** In part II of this group of papers, the control of the gait of a biped robot during rectilinear walk was considered. The modeling approach and simulation, using Kane's method with implementation leveraged by Autolev, a symbolic computational environment that is complementary, was discussed in part I. Performing turns during the walk is technically more complex than the rectilinear case and deserves further investigation. The problem is solved in the present part III as an extension of part II. The robot executes a rectilinear walk on a local reference frame whose progression axis is always tangent, and its origin performs the involute of the path curve. The curve is defined by its curvature (osculating circle) and center of curvature (evolute) along the path. Radius of curvature and center can change continuously (in practice at every sampling time). For postural equilibrium, Center of Gravity and Zero Moment Point (COG/ZMP) follow the same preview reference proposed for rectilinear walk ( $cog_{Ref_x}(t)$ ,  $cog_{Ref_x}(t)$ ,  $cog_{Ref_y}(t)$ ,  $cog_{Ref_y}(t)$ ). The effect of the turn on the sagittal plane is negligible and is ignored, while on the frontal plane it is accounted for by an offset on COG reference to compensate for the centrifugal acceleration. The body trunk and local frame rotation, and the generation of the references on this moving frame of the free foot trajectory during the swing deserve attention.

**Keywords:** gait; postural equilibrium; ZMP; preview control; biped robotics; exoskeletons; turn while walking

## 1. Introduction

This part III extends the results of the previous part II of this paper, where the control of a biped robot in rectilinear walk was described. As before, the modeling approach uses Kane's method [1], and the notation adopted is the one of Autolev [2], the symbolic computational environment originally developed by Kane to support his method.

The Jacobian matrix, used for the control as described in part II, is here referred to the local moving frame, instead of the inertial Cartesian space. The present extension has been motivated by the desire to maintain the control of the rectilinear walk as far as possible unchanged in transferring it to a walk with turns.

Literature basically follows two approaches for turning: omnidirectional turning, mostly related to techniques where joint motion is generated in real time, or extending with turning tracking of periodic preview signals. There are few studies about direction control. In [3], motion stability during turning is ensured by adjusting the swing leg center of mass (COM) and hip position trajectories in a trial and error fashion. Kajita *et al.* [4] proposed a 3D Linear Inverted Pendulum Model (LIPM)

for bipedal walking pattern generation, but no information was revealed about the complete gait generation or about the experimental validation.

[5,6] and more recently [7,8] studied the turning motions for biped with a Zero Moment Point (ZMP) based footstep planning.

Besides the standard turning motion, Shi *et al.* [9] proposed a method for online omni-directional walking pattern generation, but no information was revealed about the complete gait generation or about the experimental validation.

Chevallerau *et al.* [10] propose a control law to regulate the walking direction through the net yaw motion about the stance foot over a step.

Lu *et al.* [11] also use LIPM for postural equilibrium and two parameters for controlling direction independently for the body and swinging the foot:  $\theta$  for body rotation during double stance only, and  $K$  for smoothing COM trajectory for feet stamp position.

Our approach belongs to this second category of preview tracking, but unlike previous works, it is based on a running local reference frame with respect to which the robot maintains rectilinear walk, to rotate this frame and move its origin with respect to the world coordinates to follow an involute of the path curve during turns. The postural and gait control of the robot is obtained defining and tracking Cartesian trajectories on this moving frame, adopting the appropriate (*i.e.*, from the joint space to the local moving frame—this is simple using Autolev, as it offers efficient mathematical expressions to be used in real time control for all the necessary kinematic relationships) Jacobian matrix to transfer these trajectories to the joint space.

The central point of the approach is the preview of the Center of Gravity (COG) of a biped robot proposed for a rectilinear walk. The radius and center of curvature of the curve change continuously for a completely arbitrary path (in practice at every sampling period). Then, ignoring interaction between the frontal and sagittal planes, and radius changes, the control scheme remains almost unchanged, with respect to the rectilinear walk. Only the COG trajectory on the frontal plane of the robot has to be corrected in order to take into account the centrifugal acceleration during the turn.

Trunk rotation and swing foot rotation around the vertical axis are solved simplifying the approach of [11], by needing only one variable: the angle of rotation of the moving reference  $\theta_z$ . In fact, the trunk angle around the vertical axis is programmed to maintain continuously, during double and single stance, the frontal plane of the robot orthogonal to the walk path, and the swing foot angle follows the trunk using a feedback, similar to the one proposed by [10]. For the swing foot, different to the other approaches where the joint trajectories were computed with alternative techniques, or were not well documented, the rectilinear walk in a moving local frame defines a trajectory that can be, at each step, interpolated with simple polynomial functions in real time and corrected by feedback.

## 2. Modelling and Degrees of Freedom

In the modeling approach discussed in part I, applied to the rectilinear walk, the rotational velocities of the swing foot ( $\dot{\theta}_{z_{foot_2}}$ ), and of the trunk ( $\dot{\theta}_{z_{hat}}$ ) were constrained to zero. In this case, these angles are controlled, so, in single stance, the model has two more degrees of freedom ( $\theta_{z_{ankle_1}}$  and  $\theta_{z_{ankle_2}}$ ), and in double stance one more ( $\theta_{z_{ankle_1}}$ ) with respect to the rectilinear walk. The control of these new degrees of freedom requires two new reference trajectories.

## 3. Gait with Turns

A planar walk is assumed. The reference COG trajectory coordinates given in the rectilinear walk and their time derivatives, identical to those adopted in part II, are

$$COG_{x_{preview}}(t), \dot{COG}_{x_{preview}}(t), COG_{y_{preview}}(t), \dot{COG}_{y_{preview}}(t). \quad (1)$$

Indicate with  $N$  the inertial reference frame with origin  $N_o$  and unit vectors  $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$ , according to the right-handed convention,  $\mathbf{n}_3$  being the vertical axis pointing upward.  $L$  is a local moving reference

frame defined by its origin  $L_o$  and unit vectors  $l_1, l_2, l_3$ .  $l_1$  are aligned at each instant to the direction of walk tangent to the path curve,  $l_2$  is orthogonal to  $l_1$  on the horizontal plane and  $l_3$  vertical, aligned with  $n_3$ . The plane  $l_2 - l_3$  is parallel to the frontal plane of the robot. The vector from  $N_o$  to  $L_o$  is indicated with  $l_o$ .

The point  $P$  is the current position at time  $t$  of the COG projection on the walk path, parametrized by its arc length  $s(t)$ .  $p(t)$  is the vector from the  $N_o$  to  $P$ ,  $x_L = s(t), y_L = 0$  are the coordinates of  $P$  on the local frame, and  $x(t), y(t)$ , detailed later, on the inertial frame.

The walk path (see Figure 1) is a regular plane curve defined by its curvature (osculating circle) and its center of rotation (involute) as function of  $s(t)$  [12]. In planning the trajectory the arc length will coincide with the preview of COG along  $x: s = COG_{x_{preview}}$ . To generate an arbitrary path, radius and center of rotation change continuously. In practice, they are updated at every sampling time with radius  $R_i, i = 1, \dots, n$ . The rotation around  $n_3$  of the local frame  $L$ , coincident with that of the robot frontal plane, is defined during one sampling period by

$$\dot{\theta}_z(t) = \dot{s}(t) / R_i. \tag{2}$$

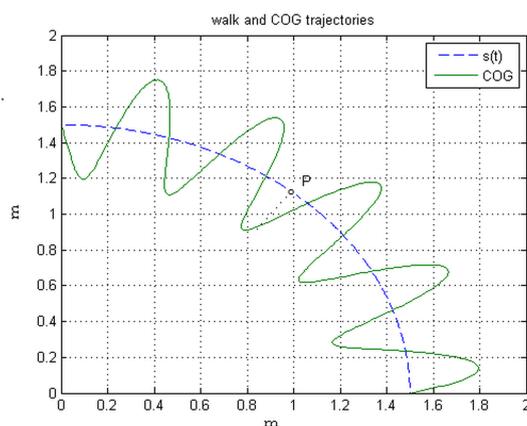


Figure 1. Walk trajectory and COG trajectory during a turn.

After  $n - 1$  sampling periods, at instant  $t_{n-1}$ ,  $s_{n-1}, x_{n-1}, y_{n-1}$  and  $\theta_{z_{n-1}}$  indicate, respectively, the arc length, the world coordinates  $x, y$  and the orientation angle. The significant elements to define the kinematics of the local frame during the next  $n$ th period, shown in Figure 2, are detailed as follows.

The rotation angle, from Equation (2) with radius  $R_n$ , is

$$\theta_z(t) = \theta_{z_{n-1}} + \int_{t_{n-1}}^t \dot{\theta}_z(\tau) \cdot d\tau. \tag{3}$$

The center of rotation  $C_n$  with respect to  $N_o$  is defined by the vector

$$c_n = (x_{n-1} - R_n \sin(\theta_{z_{n-1}}))n_1 + (y_{n-1} + R_n \cos(\theta_{z_{n-1}}))n_2, \tag{4}$$

using the last sampling position.

The current walk position  $P(t)$  with respect of  $N_o$  is expressed using the center  $C_n$  and the local reference frame  $L$ :

$$p(t) = c_n - R_n l_2(t). \tag{5}$$

Given  $p(t)$ , the origin of frame  $L$ , through the vector from  $N_o$  to  $L_o(t)$ , is

$$l_o(t) = p(t) - s(t)l_1(t), \tag{6}$$

i.e.,  $L_0(t)$  executes the involute of the path curve.

The current coordinates of  $P(t)$  in  $N$  are obtained from the scalar products

$$x(t) = p(t) \cdot n_1, y(t) = p(t) \cdot n_2. \tag{7}$$

In Appendix A, a fragment of the Autolev code used to generate the expressions for the real time computation of the necessary functions is presented.

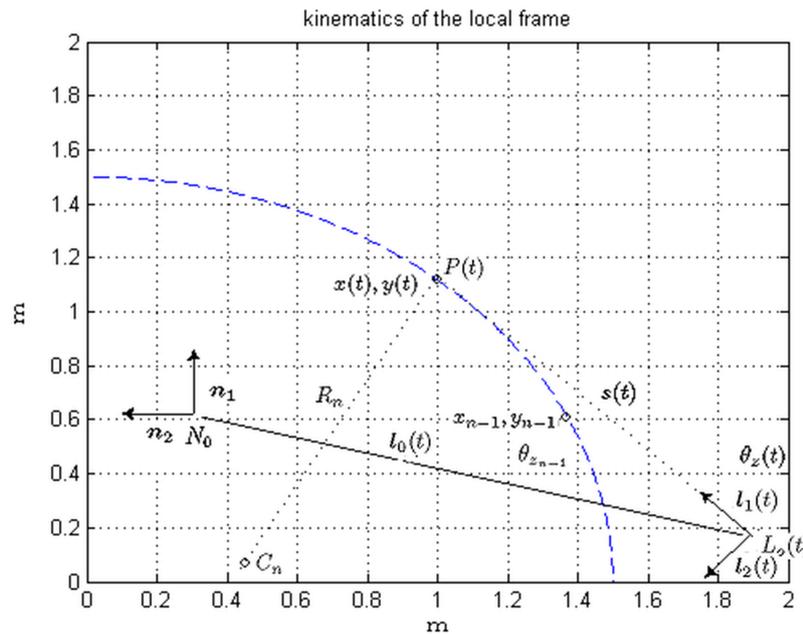


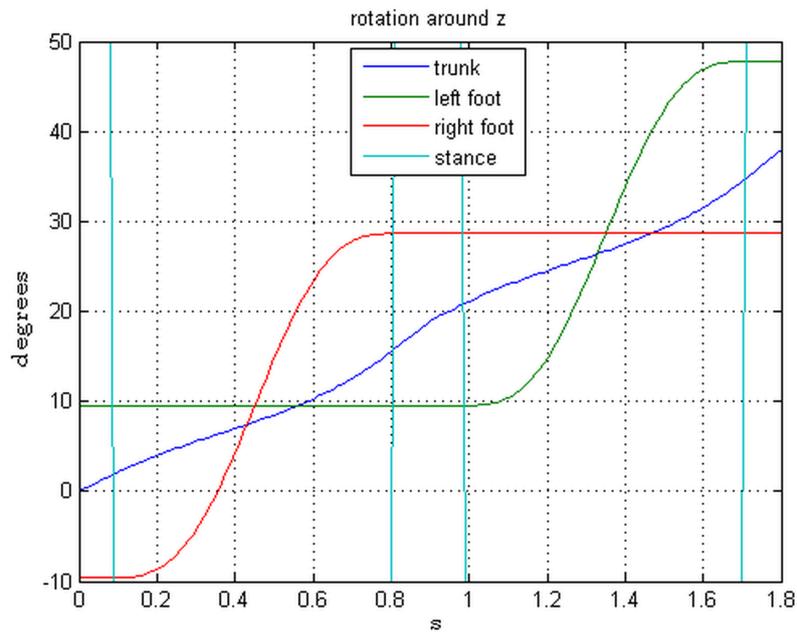
Figure 2. The kinematics of the moving reference frame during one sampling period.

#### 4. Trunk and Feet Rotations

The trunk rotation simply follows continuously the angle  $\theta_z$  of rotation of the local reference frame as defined by Equation (3). This maintains the frontal plane orthogonal to the direction of walk during the whole stride period.

The feet, *vice versa*, follow a different rotation pattern, as they can rotate during the single stance period only, one at the time when they swing. Their velocity starts from and goes to zero during the acceleration and deceleration at *Toe<sub>1</sub> off* and *Heel<sub>2</sub> strike* (see part I for the definition of the gait), i.e., when they leave and touch the ground at each step.

To maintain the pace of turn, a standard bell shaped pattern is adopted during the period of swing for the angle velocity of each foot, and its amplitude is controlled in real time by feedback, to ensure during a step that the mean of the angles of the two feet  $(\theta_{foot_1} + \theta_{foot_2}(t))/2 \cong \theta_{trunk}(t) = \theta_z(t)$  follows the trunk angle. The result is shown in Figure 3. The small misalignment between trunk and feet angles is due to the feedback sampled at each step period.



**Figure 3.** Behavior of rotational angles of trunk (blue) and feet (right green, left red) around the vertical axis.

### 5. COG and Feet Reference Trajectories in Local Frame

In the local reference frame that accompanies the turn of the robot,  $COG_{x_{preview}}$  and  $COG_{y_{preview}}$  along  $l_1$  and  $l_2$  remain almost unchanged with respect to the rectilinear walk. Only  $COG_{y_{preview}}$  has to be corrected by an offset so as to compensate for the centrifugal acceleration influencing  $ZMP_y$ . In fact, during the turn, the  $ZMP_y$  is not only affected by the accelerations and decelerations of the  $COG_y$  but also by the centrifugal acceleration due to the speed of rotation. Adopting the standard linearized inverted pendulum approximation, the actual  $ZMP_y$  is given by

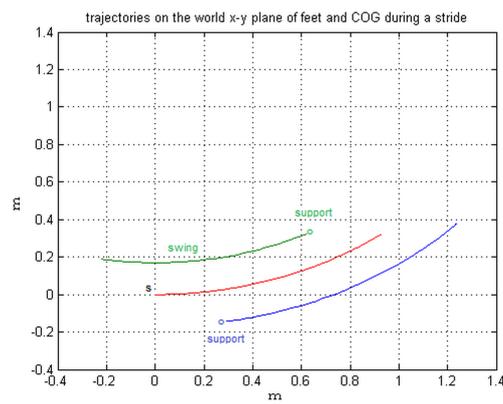
$$ZMP_y = COG_y - COG_z/g \cdot (\ddot{COG}_y + \dot{COG}_x^2 \cdot (R_n - COG_y)/R_n^2) \tag{8}$$

Thus, in order to guarantee the desired position of the  $ZMP$  on  $L$ , the following *a priori* correction term must be added to the original  $COG_{y_{preview}}$  for rectilinear walk:

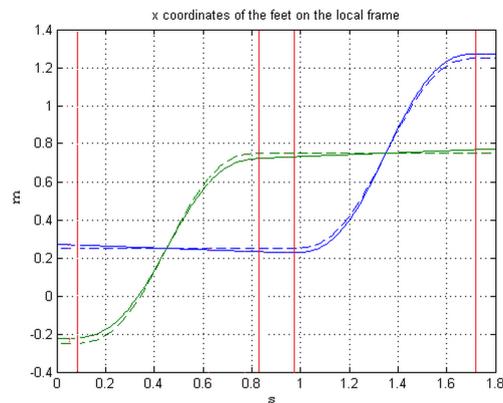
$$COG_z/g \cdot \dot{COG}_x^2 \cdot (R_n - COG_{y_{preview}})/R_n^2 \tag{9}$$

Moreover, using the extended  $COG/ZMP$  estimator outlined in the Appendix A of Part II, this acceleration is detected as an external disturbance and a further *a posteriori* correction is introduced.

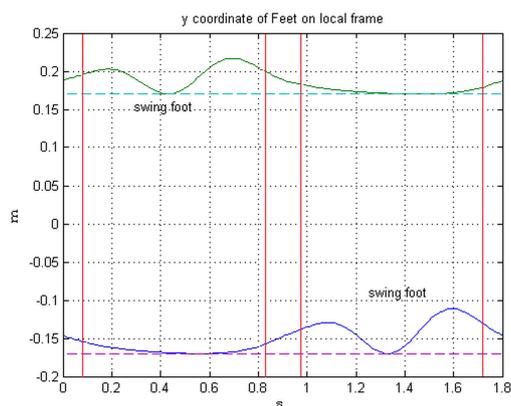
The reference trajectories (along  $x$  and  $y$ ) of the swing foot, also, require a modification, to take into account the relative motion of the local frame with respect to the world frame. To easily understand and as an example, one stride following an ideal circular path is performed in Figure 4, and in Figures 5 and 6, the trajectories along  $x_L$  and  $y_L$  are compared with those of the rectilinear walk, *i.e.*,  $R = \infty$ , drawn in dashed lines.



**Figure 4.** Trajectories of feet (green and blue) and COG (red) on the plane  $x - y$  for an ideal circular stride.



**Figure 5.** Comparison of the trajectory along  $x_L$  of rectilinear walk (dashed) and walk with turn (continuous).



**Figure 6.** Comparison of the trajectory along  $y_L$  of rectilinear walk (dashed) and walk with turn (continuous).

Circularity during a step in reality is not a stringent requirement, as these trajectories are generated in real time to adapt to the actual path, as described in the next subsections.

### 5.1. Implementation of the Swing Foot Reference along $x_L$

In the rectilinear walk, the velocity of the swing foot along  $x$  is similar to a bell shaped function with an integral equal to twice the step length  $step_L$  on a time interval equal to the single stance period. In fact, in the rectilinear walk, a symmetrical or skewed standard bell function is taken as reference of the velocity, and its amplitude is adaptively corrected in real time to maintain the feet aligned with the  $ZMP_x$ .

In the walk with turns, this will not be sufficient, and a correction must be added to the original trajectory to be computed step by step in order to zero the forward velocity of the foot on the world reference at contact with the ground.

Figure 4 shows the center path of the COG and the trajectories of the two feet during a stride; the forward foot (blue line) will initially support the step. In Figure 5, the  $x$  coordinates of the feet on the moving frame  $L$  are plotted, the difference with respect to the rectilinear walk, drawn with dashed lines, is evident, and analyzed more closely.

The position and velocity of the center of the supporting foot during the period of time it remains on the ground, *i.e.*, a step plus a double stance period, is computed with the technique of generating the involute of an arc of a circle described in Section 3. Indicating the arch length with  $s$ , and considering that at the end of the double stance the foot that will support the next swing advances the COG position of one step length, the following equations result:

$$\begin{aligned} x_{Foot1_L}(s) &= s + (R + L_{feet}/2 \cdot phase) \cdot \sin((0.5 \cdot step_L \cdot (step_P + ds_P)/step_P - s)/R) \\ \dot{x}_{Foot1_L}(s) &= (1 - (1 + L_{feet}/2 \cdot phase) \cdot \cos((0.5 \cdot step_L \cdot (step_P + ds_P)/step_P - s)/R)) \cdot step_L/step_P \\ 0 &\leq s \leq step_L \cdot (step_P + ds_P)/step_P. \end{aligned} \tag{10}$$

On the rectilinear walk, those equations are simply:

$$\begin{aligned} x_{Foot1_L} &= step_L/2 \cdot (step_P + ds_P)/step_P \\ \dot{x}_{Foot1_L} &= 0 \end{aligned} \tag{11}$$

$L_{feet}$  is the distance between the two feet along the  $y$  axis,  $phase = \pm 1$  according to whether the center of rotation is at the right or left of the supporting foot,  $step_L$  and  $step_P$  are the step length and period,  $ds_P$  is the period of double stance.

With  $x_0, x_1, \dot{x}_0, \dot{x}_1$  the position and velocity of the correction term for the swing foot at the beginning and end of a single stance period is indicated. Note that the beginning of the swing period corresponds in Equation (10) to  $s = step_L \cdot (step_P + ds_P)/step_P$  and the end to  $s = 0$ . From the difference between Equations (10) and (11), it results that positions and velocities of the correction term at the beginning and end of the swing interval are:

$$\begin{aligned} x_0 = -x_1 &= (R + L_{feet}/2 \cdot phase) \cdot \sin((0.5 \cdot step_L \cdot (step_P + ds_P)/step_P - s)/R) \\ \dot{x}_0 = \dot{x}_1 &= (1 - (1 + L_{feet}/2 \cdot phase) \cdot \cos(0.5 \cdot step_L \cdot (step_P + ds_P)/step_P - s)/R) \cdot step_L/step_P \end{aligned} \tag{12}$$

*i.e.*, at the extremes, positions have opposite values and velocities are identical.

Measuring  $x_0$  and  $\dot{x}_0$ , and assuming as initial position  $x_0$ , initial velocity  $\dot{x}_0$  and final position  $-x_0 + \delta_x$ , a third order polynomial for the correction during the next single stance period is derived at each step.  $\delta_x$  has been introduced to adapt, by feedback in real time at each step, the correction to the real path.

To obtain the new reference trajectory, this correction is added to the original one of the rectilinear walk.

## 5.2. Implementation of the Swing Foot Reference along $y_L$

Without turns, the feet motion along  $y$  is zero, as it does not deviate from a baseline at a constant distance from the walk path. *Vice versa*, when the moving frame rotates, a correction has to be considered.

In the ideal condition of Figure 4, the trajectory of the swing foot on  $L$  to connect two successive support periods is shown in Figure 6. It starts and ends approximately at the same value, with the opposite derivative, reaching near the mid-distance the baseline of the rectilinear walk. To interpolate this function, a fourth order polynomial is necessary.

Let  $y_0$  and  $\dot{y}_0$  be the position and velocity of the foot center along  $l_2$  at the end of the previous double stance period. Choose  $y_1 = y_0 + \delta_y$ ,  $\dot{y}_1 = -\dot{y}_0$  as values of the position and its velocity at the end of the next single stance period, and  $y_m = L_{feet}/2 \cdot phase$  the position at mid-distance. With these five values, a fourth order polynomial is computed in real time for the reference trajectory to be used for the next single stance period. As in the previous section,  $\delta_y$  is used in feedback, step by step, to adaptively bring the feet trajectory at the end of the single stance to  $ZMP_y$ .

## 6. A Simulation Example

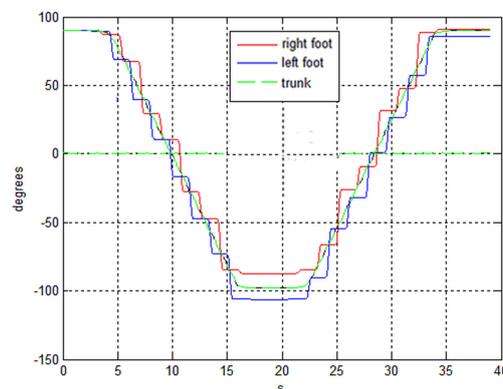
The following figures show the results of the dynamical simulation of the control of a model identical to the one in the previous part II, *i.e.*, a man 1.78 m tall of 75 kg wearing an exoskeleton of 30 kg during an “S” shaped path, with a distance between the feet of 0.34 m, a step length of 0.5 m, a step period of 0.9 s, and a double stance interval of 0.18 s. The control and the example are identical to those of part II, and the description of the simulator built at Politecnico di Torino is contained in part I. There, in particular, the critical issues related to the contact between feet and ground are discussed.

The path was obtained as described in Section 3 with a sampling period of 10 ms, a radius changing at a rate of 7.5 during the time of a step, with a minimum value of 2 m.

In Figure 7, the rotational angles of trunk and feet are shown. The non-perfect symmetry of the feet angles is a consequence of the feedback with sampling interval of the step period. However, it is argued that this will be of no consequence.

The  $y_L$  coordinates of the feet are shown in Figure 8. During the swing period, the reference and its tracking can be seen. The effect of the feedback in maintaining the feet on the baseline is evident.

Similarly, Figure 9 shows the tracking of the swing feet  $x_L$  coordinates. The reference during single stance is shown as a dashed line.



**Figure 7.** Behavior of rotational angles of trunk (green), right (red) and left (blue) feet along the vertical axis.

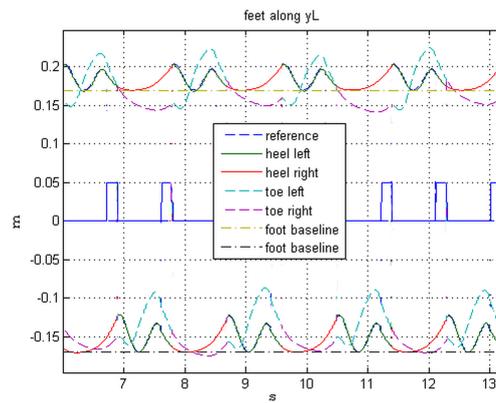


Figure 8. Trajectories in time during a left turn, along  $l_2$ , of toes and heels on the local frame.

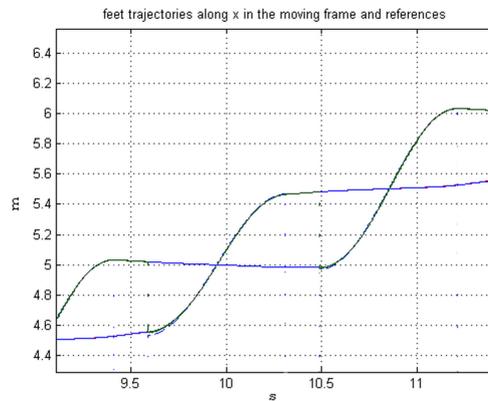


Figure 9. Tracking of the feet trajectories along  $l_1$  during the turn (reference and trajectory).

The trajectories of toes, heels, COG and ZMP on the horizontal plane of the world frame are presented in Figure 10, with details in Figure 11. Note the position of the center of the supporting foot and the average of the point cluster of the ZMP on the support. The behavior of the radius that generates the path is in Figure 12.

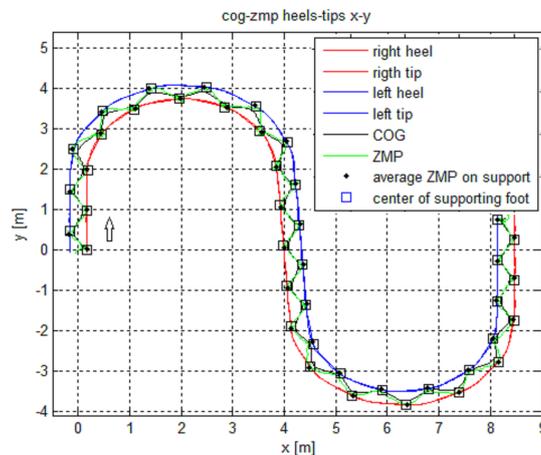
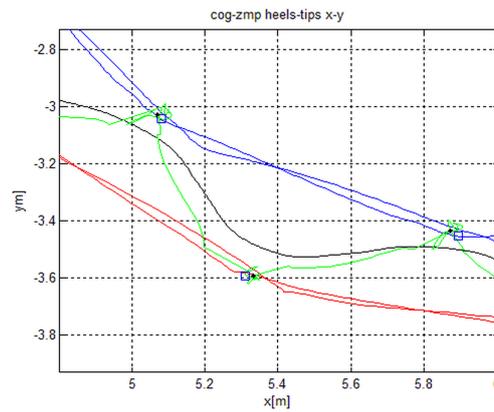
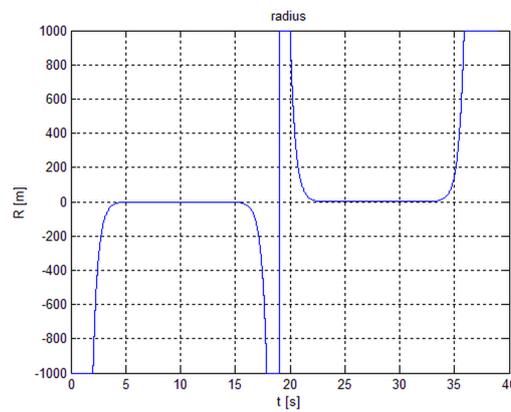


Figure 10. Trajectories on the horizontal plane of the world frame of COG, ZMP, tips and heels.



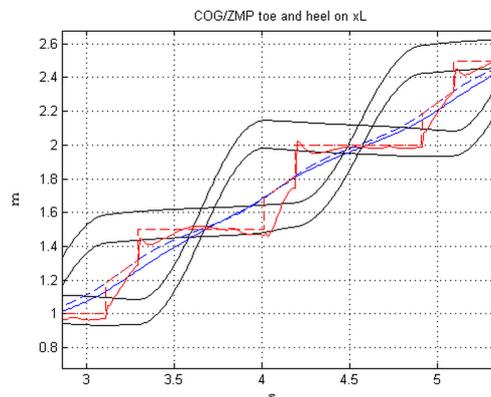
**Figure 11.** Details of the trajectories on the horizontal plane of the world frame of COG, ZMP, tips and heels (legend is as in Figure 10)



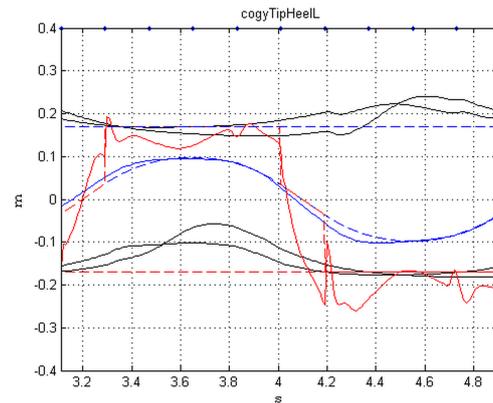
**Figure 12.** The behavior of the radius that generates the “S” shaped path.

Figures 13 and 14 show the behavior of COG, ZMP, tips and heels in the local frame during one stride.

The blue line is the COG behavior (the dashed line is the reference), the red line is the ZMP, the black lines are heel and toe. The rectilinear dashed lines in Figure 14 indicate the baselines of the two feet.



**Figure 13.** Trajectories along  $l_1$ , of COG reference (blue dashed), COG (blue), ZMP reference (red dashed), ZMP (red), tips and heels (black) on the local reference frame.



**Figure 14.** Trajectories along  $l_2$ , of COG reference (blue dashed), COG (blue), ZMP reference (red dashed), ZMP (red), tips and heels (black) on the local reference frame.

## 7. Conclusions

An extension of the rectilinear walk of a biped to the walk with turns has been presented. It is obtained by transferring a rectilinear walk and the tracking of reference trajectories on a Cartesian moving frame that accompanies the turns of the curvilinear path in addition to controlling the joints using the appropriate Jacobian matrix from joint space to this moving space. The path curve of the walk can be arbitrary and is generated in real time by changing the radius of curvature at each sampling time.

A minimalistic approach has been adopted in the definition of the reference trajectories, with the aim to generate and adapt them in real time. This accounts for bell shaped functions, third and fourth order polynomials, whose parameters are corrected by feedback in order to maintain a coherent walking pattern.

The inconsistency of these (somewhat arbitrary) trajectories with the multichain kinematics of the biped was accommodated, as shown by the simulations, by the adoption of the “extended Jacobian” and weighted least squares of part II. With the objective of adaptively providing a reference pattern for the biped, a further extension toward a completely free walk, is the on line change of the gait parameters  $L_{feet}$ ,  $step_P$ ,  $step_L$ ,  $ds_P$ . This is actually under investigation and will be the object of future works.

**Acknowledgments:** This research has been partially supported by MIUR, the Italian Ministry of Instruction, University and Research and the Region Piedmont.

**Author Contributions:** The authors contributed equally to this work.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A: Autolev Code Fragment—The Kinematics of the Local Frame $L$

```

% during one sampling period
% inertial frame
Frames N
% local frame
Frames L
% current nominal position and center of rotation
Points , P, C % corrent postion on the path, center of rotation

% arc length, coordinate of the path and vertical rotation angle at time t
Variables s', x, y, thetaz'
Specified cogRefx'
Constants radius, s0, x0, y0, thetaz0 % initial condition for a sampling period
s = s0
thetaz = thetaz0
s' = cogRefx' % the arc length velocity is the same than the COG velocity along x
thetaz' = s'/radius

% simple rotation of the local frame around z of thetaz
simprot(N,L,3, thetaz)

% position of the center of rotation on the inertial frame
P_No_C> = (x0 - radius*sin(thetaz0)*N1> + (y0 + radius*cos(thetaz0)*N2>

% current nominal position from the center of rotation
P_C_P> = -radius*L2>

% origin of the local reference
P_P_Lo> = -s*L1>

% coordinates of the nominal position on the inertial frame
x = Dot(P_No_P>, N1>)
y = Dot(P_No_P>, N2>)

```

At the end of each sampling period to prepare for the next

```

x0 = x
y0 = y
thetaz0 = thetaz
s0 = cogRefx

```

## References

1. Kane, T.R.; Levinson, D.A. *Dynamics: Theory and Applications*; McGraw-Hill: New York, NY, USA, 1985.
2. Mitiguy, P. MotionGenesis: Advanced Solutions for Forces, Motion, and Code-Generation. Available online: <http://www.motiongenesis.com/> (accessed on 17 March 2016).
3. Yagi, M.; Lumelsky, V. Synthesis of turning pattern trajectories for a biped robot in a scene with obstacles. In Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Takamatsu, Japan, 30 October–5 November 2000.
4. Kajita, S.; Kanehiro, F.; Kaneko, K.; Yokoi, K.; Hirukawa, H. The 3D Linear Inverted Pendulum Model: A simple modeling for a biped walking pattern generation. In Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, HI, USA, 29 October–3 November 2001.
5. Nishiwaki, K.; Kagami, S.; Kuffner, J.; Okada, K.; Kuniyoshi, Y.; Inaba, M.; Inoue, H. Online Humanoid Locomotion Control by using 3D Vision Information. *Exp. Robot. VIII* **2003**, *5*, 85–94.
6. Kurazume, R.; Hasegawa, T.; Yoneka, K. The sway compensation trajectory for a biped robot. In Proceedings of the IEEE International Conference on Robotics & Automation (ICRA), Taipei, Taiwan, 14–19 September 2003.
7. Nishiwaki, K.; Kagami, S. Online walking control system for humanoids with short cycle pattern generation. *Int. J. Robot. Res.* **2009**, *28*, 729–742.

8. Engelsberger, J.; Ott, C.; Albu-Schaffer, A. Three-dimensional bipedal walking control based on divergent component of motion. *IEEE Trans. Robot.* **2015**, *31*, 355–368.
9. Shi, G.; Wang, H.; Fang, B. Online omnidirectional walking patterns generation for biped robot. In Proceedings of the 9th International Conference on Electronic Measurement and Instruments, Beijing, China, 16–19 August 2009.
10. Wang, T.; Chevallereau, C.; Rengifo, C. Walking and steering control for a 3d biped robot considering ground contact and stability. *Robot. Auton. Syst.* **2012**, *60*, 962–977.
11. Lu, J.; Chen, J.; Lin, P. Turning in bipedal robot. *J. Bion. Eng.* **2013**, *10*, 292–304.
12. Do Carmo, M.P. *Differential Geometry of Curves and Surfaces*; Prentice-Hall: Upper Saddle River, NJ, USA, 1976.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).