Advances in Autonomous Robotics Systems, Springer LNSC Volume 8717, (2014), pp 251-262

Humanoid Robot Gait Generator: Foot Steps Calculation for Trajectory Following.

Horatio Garton, Guido Bugmann[1], Phil Culverhouse, Stephen Roberts, Claire Simpson, Aashish Santana

Centre for Robotics and Neural Systems, Plymouth University, Plymouth, England.

Abstract. During bipedal gait, a robot falls from one foot to the other. This motion can be approximated with that of an inverted pendulum with discrete movements of the contact point. We detail here how to use the linear inverted pendulum model (LIPM) for selecting successive contact points in such a way that the centre of mass (COM) of the robot flexibly follows a predefined set of waypoints on a straight or curved trajectory, allowing it to move forward, stop and revert its direction of motion in stable way. The use of a fixed step cycle duration reduces the mathematical complexity and the computational load, enabling real-time updating of gait parameters in a microcontroller.

1 Introduction

1.1 Bipedal gait

Bipedal gait is composed of two cyclic movements. The frontal (left-right) oscillation moves the weight onto one foot (support foot) while the other (swing foot) travels in the air in order to touch the ground in a new, e.g. forward, position. The latter foot then becomes the support foot, allowing the other foot to swing too. The sagittal motion (forward-backward) is a forward or backward rotation around an axis that is approximately given by the position of the support foot. Observing humans shows numerous flexible ways of executing these two cycles. However, gait programming and control is still a difficult problem in humanoid robotics. Here, we are especially interested in gaits for small servo-driven humanoid robots. One approach used in the commercial Darwin robot [1] consists of a parameterized cyclic pattern generator, similarly to a gait developed at Plymouth University [2]. Such gaits are surprisingly effective and have set speed world records. They are also compact from the coding point of view and can be run on small microcontrollers. However, they lack flexibility and do not exploit the full potential of the robot. Another approach is exploiting the physics of the robot, modelled as a simple inverted pendulum, or its more constrained form, the Linear Inverted Pendulum Model (LIPM) [3,4]. Such models are also used

[1] corresponding author: gbugmann@plymouth.ac.uk

to model human gait. In this approach, the commanded leg movements “follow” the natural gravity-driven swing of the robot on the standing leg, leading to a more stable gait, as can be shown by analyzing the Zero Moment Point (ZMP) of the robot (see section 1.4).

The problem when using inverted pendulum models is to select successive feet positions such that the natural swing leads to the robot performing actions selected by the user, like turning, starting, stopping, reversing, etc. In this paper we describe in details an original approach that provides a flexible solution at a low computational cost. One of the key features is the use of a fixed gait cycle duration. This simplifies the solution significantly, obviating the need for iterative methods such as in [4].

1.2 Plymouth Humanoids

Plymouth Humanoids are Plymouth University’s robot football team. Using a team of five Drake robots, Plymouth competes in international robot competitions called RoboCup and FIRA. The Drake robots are designed, made and programmed by students and staff at Plymouth University. Figure 1 shows a drake robot. The gait for this robot must be dynamically controlled to produce forward, turning and sidestep motion at a given speed. These are complicated motions for a robot to accomplish; there is also the additional challenge of making the robot move at high speed.

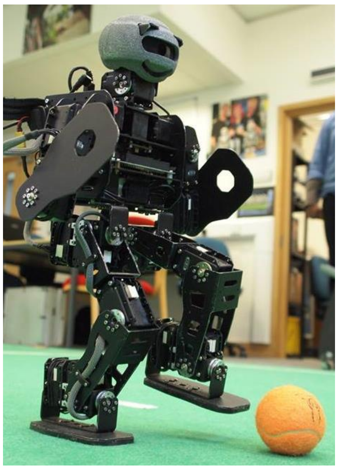

Fig. 1. Drake Robot

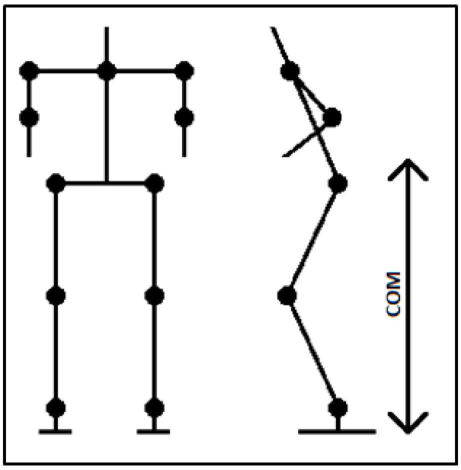

Fig. 2. Robot model in the walking position

At FIRA, the sprint event tests the robots ability to move forwards and backwards on a 3m track. The best robots are expected to complete the sprint in less than 25 seconds.

The drake robot is designed to conform to the RoboCup and FIRA specifications. The total height of the robot is 470mm, total weight is 3Kg. The height of the centre of mass (COM) in the walking position is 225mm. When the robot walks it has bent knees to allow the robot to extend its legs. The knee bends to shorten the total height of the robot by 25mm.

Due to the distribution of the mass of the upper body, it needs to be tilted forward to move the vertical projection of the centre of mass of the body above the hip joint (Figure 2). This reduces the torque on the hip pitch servos.

1.3 Linear inverted pendulum

The inverted pendulum model (IPM) of the robot consists of one large mass on the top of a massless pole. The walking motion of the robot is a combination of the frontal and sagittal motion cycles of the IPM. For an inverted pendulum of fixed length the equations of motions in the two planes are coupled and are difficult to resolve. A key finding by [3] is that, by introducing the constraint of a fixed height H of the mass, the two equations of motions become independent. This is the linear IPM (LIPM) (see Figure 3.) In practice, maintaining a fixed height is not a big constraint, and has the advantage of keeping the camera-carrying head more still. It is achieved by varying the length of the legs during the motion.

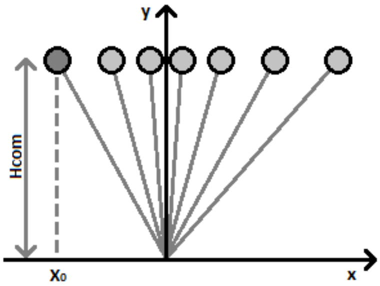

Fig. 3. Successive positions of the Linear Inverted Pendulum

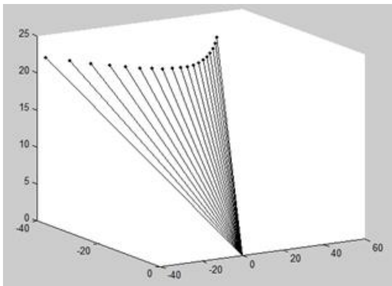

Fig. 4. Linear inverted pendulum motion in two axes.

The equation of motion is derived as follow. The mass undergoes a vertical gravitational force mg that exerts a torque $\tau = mgx$ on the pole, where x is the horizontal distance of the mass from the contact point. The torque exerts a horizontal force $F = \tau/H$ that accelerates the mass:

$$\ddot{x} = \frac{F}{m} = x\frac{g}{H} \qquad (1.3.1)$$

The same equation is used in the frontal and sagittal planes. Its solution is:

$$x(t) = x_0 * \cosh\left(\sqrt{\frac{g}{H}} * t\right) + \frac{v_0}{\sqrt{\frac{g}{H}}} * \sinh\left(\sqrt{\frac{g}{H}} * t\right) \qquad (1.3.2)$$

Where: x_0 = initial position of pendulum (m), measured from the foot position assumed to be x=0. v_0 = initial velocity of pendulum (ms-1). g = gravitational constant

(9.81ms-2). t = time (s). H = Height of COM (m). $\sqrt{\frac{g}{H}}$ is the time constant of the pendulum. It will be replaced by ω in this paper. The time derivative of (1.3.2) gives the velocity $v(t)$ of the mass:

$$v(t) = x_0 * \sqrt{\frac{g}{H}} * \sinh\left(\sqrt{\frac{g}{H}} * t\right) + v_0 * \cosh\left(\sqrt{\frac{g}{H}} * t\right) \quad (1.3.3)$$

The 3-D motion of the COM can be generated by using equation (1.3.2) for the each of the sagittal and frontal motion (see e.g. figure 4).

These are the equations of motion of a point like mass at the end of a pole. The physical robot has an extended mass. For such multi-body pendulums, the time constant is different from the point like model. In practice, we found multiplying the point-like time constant (ω) by 0.7 produces the best gait, qualitatively assessed as having a good long-term stability and small-amplitude head oscillations.

The zero moment point (ZMP) [5] of the LIPM has an interesting property. The ZMP is the point p on the ground where the foot should be placed so that no torque is exerted on the ankle, i.e. the configuration is stable. For a robot that is standing still, p is exactly below the COM. For a robot that is, for instance, pushed back, the ZMP (where one would place ones feet) needs to be moved backward so that the forward force generated by gravity compensates exactly for the external push. For a pole with an accelerated mass, the ZMP p (in one dimension) is defined by [6]:

$$p = x - \frac{H}{g}\ddot{x} \qquad (1.3.4)$$

Comparing (1.3.4) with (1.3.1), one can see that p=0 for the free-falling linear inverted pendulum. That is exactly the point where the foot of the pole touches the ground. Therefore, as long as one ensures that the motion of the robot follows its natural pendulum motion, stability conditions are automatically satisfied.

In reality a physical robot is not a LIPM, but a multi-body system made of pendulums that exert internal forces on each other. The ZMP of such system moves away from the centre of the foot during gait, but with a displacement normally smaller than the dimensions of the foot, thus preserving the stability of a robot actuated on the basis of a simple LIPM [7,8].

2 Gait Generator

2.1 Waypoints selection

There are many possible ways to design a gait based on equation 1.3.2 and 1.3.3. The approach selected here is to set target positions (waypoints) for the centre of mass to achieve after two steps, i.e. a full gait cycle. If the COM had only initial conditions and a goal position to move to, there would be an infinite number of possible solution trajectories, i.e. possible placements for the feet. To select one solution, firstly, the COM end velocity vector after the two steps must also be given, as described in section 2.2. Secondly, a time constraint is applied: One cycle of the gait, consisting of two footsteps, must be completed in a preset time T. This time is the same for all steps. This simplifies the inversion of the motion equations, because the sinh() and cosh() terms become constants. The robot walking speed is determined by T and by the distance between waypoints. We can achieve up to around 2 cycles/sec and distance of up to 15 cm per cycle. These values are close to the speed limits of the servo motors.

The robot is treated as a holonomic robot because it can move and rotate in two dimensions between two steps. At the start of each cycle of the gait, the robot is given a forward velocity a side velocity and a rotational velocity (forward speed FS, the side speed SS and turn speed TS). From these values a circular path is created and a new position and orientation for the COM can be calculated. Figure 5 shows how these commands are used to calculate the next COM position after one cycle time T. First, TS and SS define a motion direction V with unchanged robot orientation. TS then defines an arc or circle moving away from that direction and rotating the direction accordingly. The centre of the circle is placed on a line that is perpendicular to the velocity vector V (the vector sum of FS and SS). The length L of the arc from the start position to the next way point given by L = |V|*T. the radius r of the circle is defined by V and TS. TS is in rad s-1 (figure 5). It should be noted that the velocity V is the average velocity of the robot along its desired trajectory. This is different from the instantaneous velocity of the COM of the robot that has cyclic lateral and forward components vl and vf. The purpose of the gait generator in this paper is to control these components to make the COM pass through the desired waypoints.

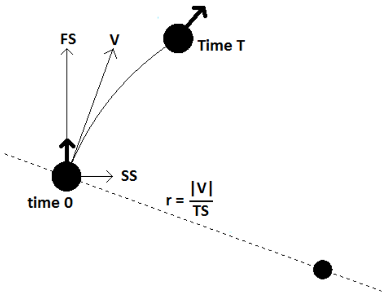

Fig. 5. COM positions before and after a cycle, including forward motion, rotation and side-stepping.

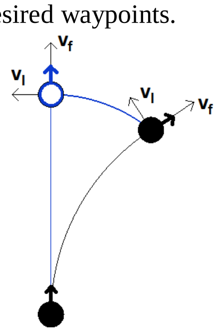

Fig. 6. Steps of the calculation of the COM velocity vector at the next waypoint (image for forward and turn only).

2.2 Calculating the end-of-cycle velocity vector

The velocity at the end of each cycle must be determined so that one set of left and right feet positions can be calculated. The end velocity V_{COM} is a vector in two dimensions.

The approach used here is to work out the V_{COM} for a straight line motion (TS=0), then to rotate the vector by how much the robot body would have turned at the end of the curve (figure 6).

The forward velocity at the end of the cycle is calculated on the basis of the average speed FS. The end-of-cycle lateral speed of the COM is based on the requested sidestep speed SS, as well as the desired separation of the feet. After the forward speed and side speed of the end of cycle COM position have been calculated, they are rotated by the angle the body of the robot will be facing.

2.2.1 Desired COM end-of-cycle lateral velocity

The gait is designed to swing on the left foot first then on the right, this results in the end velocity being negative (moving to the left side of the robot, figure 7). The cycle time (T) is separated into two halves. In the second half the robot is swinging on the right foot so the starting position of the pendulum is -D/2 where D is the separation of the feet (assuming that the foot switch takes place when the robot exactly in the middle between the two feet). At this moment, the initial velocity v_0 points towards the right. So, at the end of the swing on the left foot, the position is defined by equation 1.3.2, which can be solved for the initial velocity v_0 at the start of the swing on the left foot.

$$-\frac{D}{2} = -\frac{D}{2} * \cosh\left(\omega * \frac{T}{2}\right) + \frac{v_0}{\omega} * \sinh\left(\omega * \frac{T}{2}\right) \qquad (2.2.1.1)$$

$$v_0 = \frac{-\omega * D * \left(1 - \cosh\left(\omega * \frac{T}{2}\right)\right)}{2 * \sinh\left(\omega * \frac{T}{2}\right)} \qquad (2.2.1.2)$$

Of course, during the gait cycle, v_0 is also the initial velocity of the swing on the right foot. When a side-step velocity is specified, it must be added to equation 2.2.1.2 defining the full end velocity in the lateral direction v_l.

$$v_l = \frac{\omega * D * \left(1 - \cosh\left(\omega * \frac{T}{2}\right)\right)}{2 * \sinh\left(\omega * \frac{T}{2}\right)} + SS \qquad (2.2.1.3)$$

2.2.2 Desired COM end-of cycle forward Velocity

The required forward velocity of the COM at the end of a cycle is determined only by the requested forward speed FS of the robot. This corresponds to a required travel distance in one cycle of $F = * T$. If the robot is moving forward at speed FS, then the total distance it will move forward in one cycle is. We shall call this distance F. In the second half of the cycle, when the robot swings on the right foot, the COM will move from position –F/4 to F/4 in time T/2 (the positions are relative to the contact foot, assuming an equal distance travelled during each half-cycle). We can put this into equation 1.3.2 and solve for v_0 again.

$$\frac{F}{4} = -\frac{F}{4} * \cosh\left(\omega * \frac{T}{2}\right) + \frac{v_0}{w} * \sinh\left(\omega * \frac{T}{2}\right) \qquad (2.2.2.1)$$

$$v_0 = \frac{\omega F \left(1+\cosh\left(\omega \frac{T}{2}\right)\right)}{4 \sinh\left(\omega \frac{T}{2}\right)} \qquad (2.2.2.2)$$

If you used this v_0 in equation 1.3.3 and t = T/2 and x_0 = -F/4, the velocity at time T/2 would be the same as v_0. So the end velocity in the forward direction v_f is equal to v_0 in equation 2.2.2.2.

$$v_f = \frac{\omega * F * \left(1+\cosh\left(\omega * \frac{T}{2}\right)\right)}{4 * \sinh\left(\omega * \frac{T}{2}\right)} \qquad 2.2.2.3$$

Figure 8 shows the motion of the COM in the forward direction in relation to the right foot. Note that the velocity is nearly constant.

2.2.3 Applying the rotation

The vector for the end of cycle velocity (V_{COM}) must be rotated by the same angle the COM position is facing at the end of the cycle α_2 (figure 6).

$$v_{COM} = [v_l \quad v_f] * \begin{bmatrix} \cos(\alpha_2) & \sin(\alpha_2) \\ -\sin(\alpha_2) & \cos(\alpha_2) \end{bmatrix} \qquad (2.2.3.1)$$

α_2 is given by $\alpha_2 = \alpha_1 + T*TS$, where α_1 is the orientation of the robot at the start of the cycle.

2.3 Feet Placement Calculations

One cycle of the robot gait consists of a swing on the left foot then a swing on the right foot. Figure 7 shows the trajectory of the COM. The COM has an end position x_e and end velocity v_e. In each of the planes of motion the robot has an initial position x_0 and initial velocity v_0. (In what follows, the variable x is used for both planes, as the equations are exactly the same). The left foot and right foot placements must be selected to accomplish the cycle in time T and to reach the desired COM end state.

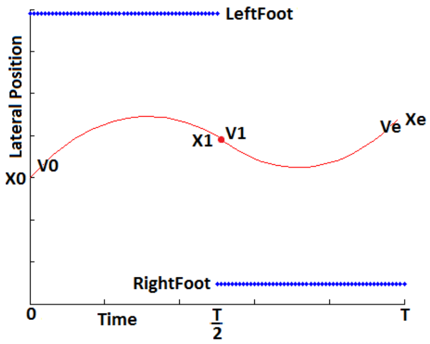

Fig. 7. Lateral foot positions and COM in time (T=0.5s, sidestep=20mm, lateral distance between feet = 90mm

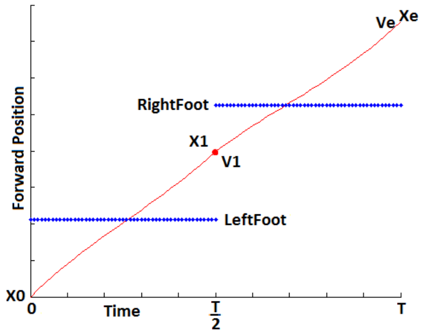

Fig. 8. Forward foot positions and COM in time – sagittal position during one step. The velocity is nearly constant. Stride length = 70mm.

At time T/2 the COM position is at x_1 with velocity v_1. This is the initial conditions for the swing on the right foot. The following equations can be created from Equation 1.3.2 and 1.3.3 for the mid cycle position and the mid-cycle velocity:

$$x_1 = (x_0 - L)\cosh\left(\omega\frac{T}{2}\right) + \frac{v_0}{w}\sinh\left(\omega\frac{T}{2}\right) + L \qquad (2.3.1)$$

$$v_1 = w(x_0 - L)\sinh\left(\omega\frac{T}{2}\right) + v_0\cosh\left(\omega\frac{T}{2}\right) \qquad (2.3.2)$$

And for the end cycle position and velocity:

$$x_e = (x_1 - R)\cosh\left(\omega\frac{T}{2}\right) + \frac{v_1}{\omega}\sinh\left(\omega\frac{T}{2}\right) + R \qquad (2.3.3)$$

$$v_e = \omega(x_1 - R)\sinh\left(\omega\frac{T}{2}\right) + v_1\cosh\left(\omega\frac{T}{2}\right) \qquad (2.3.4)$$

Where: L = Left foot position and R = Right foot position. Note that the dynamic part of the expressions operate with the pivot point (centre of ankle joint) at position x=0. This explains the subtractions and additions of L or R in the position expressions 2.3.1 and 2.3.3.

x_1 and v_1 can be substituted into equation 2.3.3 and 2.3.4. This gives two equations with two unknowns (L and R). The values of the sinh and cosh functions are constant, so this is just a linear simultaneous equation. This is the reason why a fixed frequency gait is useful. The equations can be rearranged to have L and R on the left hand side, then put into matrix form:

$$A\begin{bmatrix}L\\R\end{bmatrix} = B \qquad (2.3.5)$$

Where:

$$A = \begin{bmatrix}A_{11} & A_{12}\\A_{21} & A_{22}\end{bmatrix} \qquad (2.2.6)$$

$$A_{11} = -\cosh^2\left(\omega\frac{T}{2}\right) + \cosh\left(\omega\frac{T}{2}\right) - \sinh^2\left(\omega\frac{T}{2}\right) \qquad (2.3.7)$$
$$A_{12} = 1 - \cosh\left(\omega\frac{T}{2}\right) \qquad (2.3.8)$$
$$A_{21} = -2\,\omega\cosh\left(\omega\frac{T}{2}\right)\sinh\left(\omega\frac{T}{2}\right) + \omega\sinh\left(\omega\frac{T}{2}\right) \qquad (2.3.9)$$
$$A_{22} = -\omega\sinh\left(\omega\frac{T}{2}\right) \qquad (2.3.10)$$

and:

$$B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \qquad (2.3.11)$$

$$B_1 = x_e - x_0\left(\cosh^2\left(\omega\frac{T}{2}\right) + \sinh^2\left(\omega\frac{T}{2}\right)\right) - \frac{2v_0}{\omega}\cosh\left(\omega\frac{T}{2}\right)\sinh\left(\omega\frac{T}{2}\right) \quad (2.3.12)$$
$$B_2 = v_e - 2\,x_0\,\omega\cosh\left(\omega\frac{T}{2}\right)\sinh\left(\omega\frac{T}{2}\right) - v_0\left(\cosh^2\left(\omega\frac{T}{2}\right) + \sinh^2\left(\omega\frac{T}{2}\right)\right)$$
(2.3.13)

Both sides of equation 2.3.5 can be multiplied by A^{-1} to obtain the solution:

$$\begin{bmatrix} L \\ R \end{bmatrix} = A^{-1}B \qquad (2.3.14)$$

The same equation is used to compute Lx and Rx, and then Ly and Ry. The detailed resulting expressions are not shown here.

During the motion cycle, the COM is made to rotate with a constant yaw angular velocity (by applying a rotation to the contact foot). If α_1 is the yaw angle of the COM at the start of the cycle and α_2 is the angle at the end, t is the time during the cycle and T is the total time of the cycle then the COM angle $\alpha(t)$ is given by:

$$\alpha(t) = (\alpha_2 - \alpha_1) * \frac{t}{T} + \alpha_1 \qquad (2.3.15)$$

The angles of the swinging feet must also be calculated for when they reach the ground. The angles of the left foot, α_l, and the right foot, α_r, are given by equations 2.3.16 and 2.3.17.

$$\alpha_l = \frac{1}{4}\alpha_1 + \frac{3}{4}\alpha_2 \qquad (2.3.16)$$

$$\alpha_r = \frac{3}{4}\alpha_1 + \frac{1}{4}\alpha_2 \qquad (2.3.17)$$

This formulation is somewhat arbitrary and using ½ and ½ instead of ¼ and ¾ actually gives a better good looking motion.

As the robot needs to dynamically change speed and direction while walking, the feet positions cannot be calculated too far in advance. On Drake, at the beginning of each cycle, the feet positions are calculated for the next cycle. The feet positions of the current cycle are known because they were calculated in the previous cycle. The previous feet positions are also stored.

2.4 Feet lifting and ground speed matching

The vertical lift of the foot from the ground is a fourth order polynomial (at4+bt3+ct2+dt+e). The constants a to e are selected to make a curve that will have minimum impact on the ground. Figure 9 shows the vertical feet positions. The peak of the foot can be any value. On the drake robots the peak of the foot height is set to 30 mm.

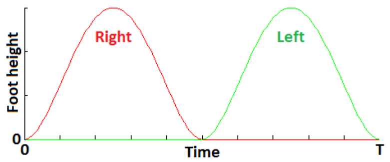

Fig. 9. Vertical feet positions.

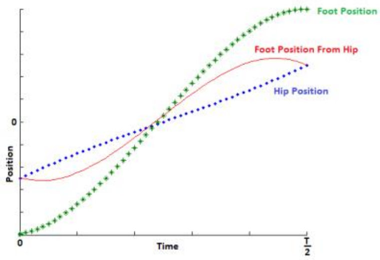

Fig. 10. Horizontal foot Movement during a half-cycle (start to end position**)**

Horizontally, each foot must move in the air from its previous position to its next position in time T/2. This is done smoothly using a third order polynomial equation (fig. 10, "foot position" curve).

In the world reference frame, the swinging foot moves forward. However, as the hip of the robot also moves forward, in the robot reference frame, the foot actually has to move backward before making contact with the ground (fig. 10. Curve "foot position from hip).

2.5 Start step

The start step consists of accelerating the COM to the right, then accelerating it back to the central position to give it the initial velocity required for a stable gait cycle. The acceleration curve is an elliptic curve which merges with the free pendulum curve at time Tm. From that time on, the robot is able to lift its left foot and start swinging on its right foot. In practice, the foot is made to lift at time 3T/4. An elliptic curve was chosen because of the flexibility to setting its slope (speed) at the merging point.

3 Results

Figure 11 shows four cycles of the robot COM motion and corresponding feet positions for a left curve. The green lines show the desired velocity at the end of each cycle. The red line shows the direction. Figure 12 illustrates an s-shaped trajectory through a superposition of video frames taken 1 second apart.

A qualitative observation of the robot motion suggests some mismatch between the pure LIPM and the physical robot, e.g. feet impacts seem to be compensating for a slight oscillation speed error in sideways motion.

Our robot does not have ankle torque meters and the quality of the stability cannot be measured in terms of ZMP. The robot recovers from brief sideways and forward pushes, but this is a purely qualitative assessment of its stability.

In straight line, the robot has achieved speeds in excess of 0.3ms/s.

It is noteworthy that all the calculations described in this paper are completed onboard the robot on an ATxmega256a3 microcontroller once per cycle, in addition to calculations done every 20ms of the COM position calculated using the LIPM formulas, and the inverse kinematic of the feet trajectories.

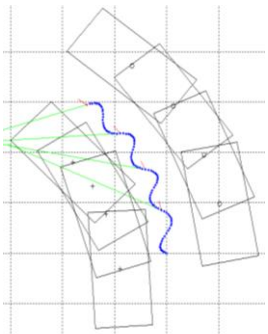

Fig. 11. Sequence of left and right foot positions corresponding to COM displacement curve shown in blue. The green lines show the direction of the COM velocity at the end of each cycle.

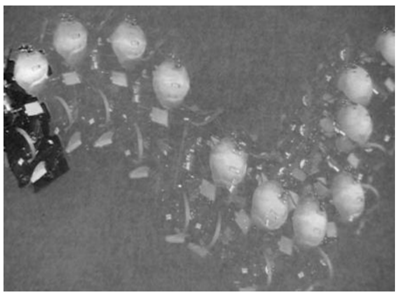

Fig. 12. S-shape trajectory executed by the robot. Superposition of snapshots taken 1 second apart.

4 Concluding comments

This paper described how to calculate the positions of feet to make a robot follow a predefined trajectory flexibly combining translation and rotation. The constraints are a constant height of the robot and a constant time between steps. The calculations are based on a simple LIPM model and produce working solutions that are used on a real robot. The gait actions include forward and backward, straight and curved gaits, and start and stop steps.

Future work will evaluate the fitness of the simple LIPM model for our robot in demanding conditions, such as high-speed motion on irregular surfaces, possibly leading to the development of an appropriately improved model to increase the operational envelope.

References:

1. Ha I., Y. Tamura Y. and Asama H.. Gait pattern generation and stabilization for humanoid robot based on coupled oscillators. In IEEE IROS, pages 3207–3212, 2011.
2. Gibbons P., Mason M., Vicente A, Bugmann G. and Culverhouse P. (2009). Optimization of Dynamic Gait for Bipedal Robots. Proc. 4th Workshop of Humanoid Soccer Robots held in Paris at IEEE Humanoids 2009.
3. Kajita, S., Kanehiro, F., Kaneko, K.,Yokoi, K., and Hirukawa, H. 2001a. The 3D Linear Inverted Pendulum Model: A simple modeling for a biped walking pattern generation. In Proceedings of the 2001 IROS, Maui, Hawaii, pp. 239–246.
4. Graf, C., Röfer, T.: A closed-loop 3D-LIPM gait for the RoboCup Standard Platform League humanoid. In: Zhou, C., Pagello, E., Behnke, S., Menegatti, E., Röfer, T., Stone, P. (eds.) Proceedings of the Fourth Workshop on Humanoid Soccer Robots in conjunction with the 2010 IEEE-RAS International Conference on Humanoid Robots (2010)
5. Vukobratovic M, Borovac B, Surla D and Stokic D, Biped Locomotion: Dynamics, Stability and Application. Springer-Verlag, 1990.
6. Choi Y, You B. J., and Oh S. R., On the stability of indirect ZMP controller for biped robot systems. Proceedings of International Conferenc on Intelligent Robots and Systems, pp: 1966 - 1971, vol.2, Sendal, Japan, June 2004.
7. Suleiman W., Kanehiro F., Miura K. and Yoshida E. Enhancing Zero Moment Point-Based Control Model: System Identification Approach Advanced Robotics 25 (2011) 427–446
8. Galdeano D., Chemori A., Krut S. and Fraisse P. Optimal Pattern Generator For Dynamic Walking in Humanoid Robotics", Transactions on Systems, Signals & Devices, To appear, 2014.