Generalized Learning to Create an Energy Efficient ZMP-Based Walking

Nima Shafii[1,2,4], Nuno Lau[3,4], Luis Paulo Reis[1,5]

[1]LIACC - Artificial Intelligence and Computer Science Laboratory, Portugal
[2]Dept. of Informatics Engineering, Faculty of Engineering, University of Porto, Portugal
[3]Dept. of Electronics, Telecommunications and Informatics, University of Aveiro, Portugal
[4]IEETA - Institute of Electronics and Telematics Engineering of Aveiro, Aveiro, Portugal
[5]Dept. of Information Systems, School of Engineering, University of Minho, Portugal
`nima.shafii@fe.up.pt, nunolau@ua.pt, lpreis@dsi.uminho.pt`

Abstract. In biped locomotion, the energy minimization problem is a challenging topic. This problem cannot be solved analytically since modeling the whole robot dynamics is intractable. Using the inverted pendulum model, researchers have defined the Zero Moment Point (ZMP) target trajectory and derived the corresponding Center of Mass (CoM) motion trajectory, which enables a robot to walk stably. A changing vertical CoM position has proved to be crucial factor in reducing mechanical energy costs and generating an energy efficient walk [1]. The use of Covariance Matrix Adaptation Evolution Strategy (CMA-ES) on a Fourier basis representation, which models the vertical CoM trajectory, is investigated in this paper to achieve energy efficient walk with specific step length and period. The results show that different step lengths and step periods lead to different learned energy efficient vertical CoM trajectories. For the first time, a generalization approach is used to generalize the learned results, by using a programmable Central Pattern Generator (CPG) on the learned results. Online modulation of the trajectory is performed while the robot changes its walking speed using the CPG dynamics. This approach is implemented and evaluated on the simulated and real NAO robot.

Keywords: Humanoid Walking, Energy Efficiency, Zero Moment Point, Central Pattern Generators.

1. Introduction

In competitive, non-deterministic environments like RoboCup soccer humanoid robot leagues, stable omnidirectional biped walking is one of the keys to win a match. Such movements must accomplish different requirements. For example, in case the target is far away, the humanoid robot must be capable walking with minimum energy usage, since the energy resources of a humanoid robots is limited.

Biped walking can be defined as the modeling of the predefined Zero Moment Point (ZMP) references to the possible body swing or horizontal CoM trajectory. For approximating the ZMP dynamics, simple physical models have been used, such as Cart-on-a-table [2] and classical Inverted Pendulum Model [3]. Kajita has utilized an Cart-on-a-table model as an approach to generate stable CoM trajectory [4]. The

major drawback of the Cart-on-a-table model is its simplification, It models the walking by considering the CoM height is a fixed position, but biomechanical studies show that the CoM height is variant during walking [5]. The vertical CoM movement is also important for energy consumption [1].

A ZMP based walking with variable height can be generated by using inverted pendulum model [3]..Recently, Kormushev et al. have presented an approach for generating an energy efficient ZMP based walking [6]. They have used a vertical CoM trajectory generator, modeled by a Spline representation, and a policy search reinforcement learning. This approach was applied to minimize the energy consumption of a walk with only a specific and predefined step length and step period; however, the shape of an energy efficient vertical CoM trajectory is different for each walking characteristics, including step length and step period.

In this paper, a novel learning scenario is presented to achieve energy efficient ZMP-based walking with variable step lengths and step periods. Fourier basis functions are used as a policy representation for modeling the vertical CoM trajectory. In order to generate the horizontal CoM trajectory, first the position of the foot during a walking is planned. Then, the ZMP trajectory is generated based on the support foot polygon. In the next step, the position of the horizontal CoM trajectory is calculated by using the approach presented in [11], which is able to solve the differential equations of the inverted pendulum model numerically with respect to the input predefined ZMP and vertical CoM trajectories. The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [7] is used as a black-box optimizer to find an optimized and energy efficient vertical CoM trajectory policy. For different input walk speeds, including step lengths and step periods, different hip height trajectories are learned. The generated trajectories based on Fourier series are the input to the programmable Central Pattern Generators (CPGs) based on Hopf oscillator [8]. CPGs prepare online modulation of trajectories, in the face of changing the walk speed. The system overview of the proposed approach is provided in figure 1, which illustrates the role of each component.
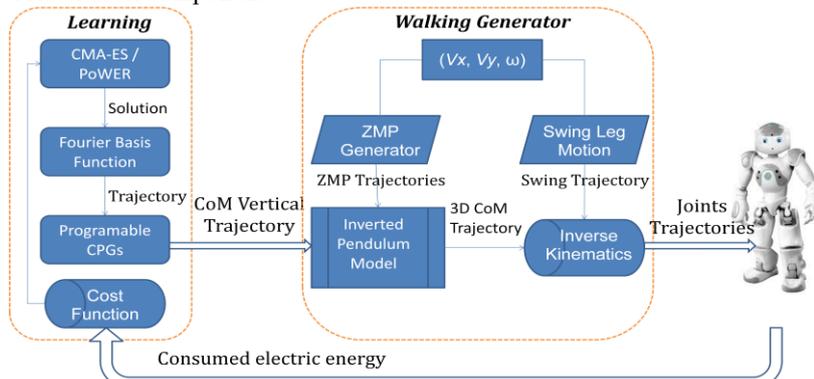

Fig.1. The interaction between each component of the proposed approach

The paper structure is as follows. First, A ZMP-based humanoid walking approach to control the walking stability is introduced. Then a CoM vertical trajectory generator based on Fourier basis functions is presented which can produce the

periodic vertical hip motion. This trajectory is the input signals of a CPGs approach, which is also introduced in Section 3. The learning scenarios with using the CMA-ES are explained in section 4. At the end of the article, the results of learning scenarios are presented and the efficiency of the method to generate energy efficient walking is shown by experiments.

2. Biped Locomotion Control Approach

A humanoid robot contains many degrees of freedom. It is difficult to maintain the balance of humanoid robot walking. The Zero Moment Point (ZMP) criterion [9] 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 support foot or the polygon containing the support legs, the given biped locomotion will be physically balanced.

ZMP based biped walking is assumed as a problem of balancing an inverted pendulum model, since in the single supported phase human walking can be represented as an inverted pendulum a predefined vertical CoM trajectory is the input of this model. Kagami et al. proposed an approach to generate the horizontal CoM trajectory by solving numerically inverted pendulum equations [10]. Figure 2-a shows the NAO humanoid robot and the inverted pendulum model. Figure 2-b shows a schematic view of the inverted pendulum model in XZ plane or in the sagittal plane. Two sets of inverted pendulum are used to model a 3D walking. One is for movements in the frontal plane; another is for movements in the sagittal plane.
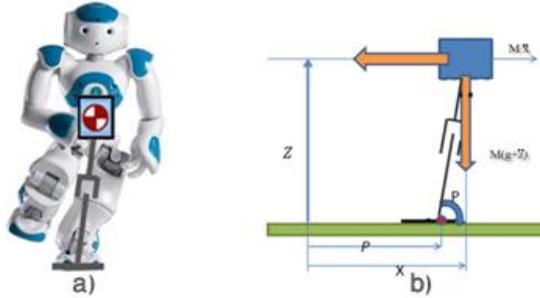

Fig.2. a) Frontal view of the NAO robot and the inverted pendulum model b) A schematic view of the inverted pendulum model

In the sagittal plane, the horizontal and vertical positions of CoM are denoted by x and z, respectively. Gravity g, horizontal CoM acceleration $\ddot{x}$, and vertical CoM acceleration $\ddot{z}$, create a moment T_p around the center of pressure (CoP) point P_x. The Equation (1) provides the moment around P.

$$T_p = M(g + \ddot{z})(x - P_x) - M\ddot{x}z \qquad (1)$$

We know from [9] 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 based on the position and acceleration of CoM. In order to generate a 3D walking, the CoM must also move in the frontal plane; hence, another inverted pendulum must be used in y direction. Using the same assumptions, equation (2) is given for movements in the frontal plane denoted by y.

$$P_x = x - \frac{z}{g + \ddot{z}}\ddot{x} \qquad\qquad P_y = y - \frac{z}{g + \ddot{z}}\ddot{y} \qquad (2)$$

In order to apply the inverted model in a biped walking problem, first the positions of the support foot during a walk must be determined. In a forward walk, the support foot positions are calculated based on the desired input step length. Then, the ZMP trajectory is designed based on support foot positions and the input step period. The vertical CoM position and acceleration trajectory must also be determined as the input of the inverted pendulum model, our approach to generate vertical CoM trajectories is explained in section 3. In the final step, the horizontal position of the CoM is calculated by solving the differential equations (2). The main issue of using the inverted pendulum is how to solve these differential equations. The solution is explained in section 2.1. Finally, an inverse kinematics method is used to find the angular trajectories of each joint based on the planned position of the feet and generated CoM position. We used and developed an inverse kinematic approach, which was applied on the NAO humanoid soccer robot, see details in [11].

2.1 Horizontal CoM Trajectory Generation

Kagami et al. proposed an approach to generate walking patterns by solving the ZMP equations numerically [10]. Kajita et al. used this numerical approach and the inverted pendulum model in order to generate the horizontal CoM trajectory of a ZMP based biped running [3].

In this numerical approach, in order to generate horizontal CoM, first the position and acceleration of CoM are discretized with a small time step Δt.

$$x(i\Delta t) \rightarrow x(i) \qquad (3)$$

$$\ddot{x}(i\Delta t) \rightarrow \frac{x(i - 1) - 2x(i) + x(i + 1)}{\Delta t^2}$$

Then, a tridiagonal system for the equation (2) is written as:

$$P_x = a_i x(i - 1) + b_i x(i) + a_i x(i + 1) \qquad (4)$$

Where,

$$a_i = -\frac{1}{\Delta t^2}\left(\frac{z(i\Delta t)}{g + \ddot{z}(i\Delta t)}\right) \qquad\qquad b_i = 1 + \frac{2}{\Delta t^2}\left(\frac{z(i\Delta t)}{g + \ddot{z}(i\Delta t)}\right) \qquad (5)$$

For generating CoM trajectory, the linear system is obtained, which is presented in equation (6). In order to solve this tridiagonal system Thomas algorithm can be applied. The solution can be obtained in O(n) operations. Here, $n = T_s/\Delta t$, in this study Δt is assumed to be 0.005 s, and T_s is the total time in which CoM is calculated.

$$
\overbrace{\begin{bmatrix} P_{(1)} \\ P_{(2)} \\ \vdots \\ P_{(n)} \end{bmatrix}}^{ZMP_x} = \overbrace{\begin{bmatrix} b_1 & a_1 & & & & \cdots & & 0 \\ a_2 & b_2 & a_2 & & & & \\ & a_3 & b_3 & a_3 & & & \vdots \\ & & \ddots & \ddots & \ddots & \\ \vdots & & & a_{n-1} & b_{n-1} & a_{n-1} \\ 0 & & & \cdots & & a_n & b_n \end{bmatrix}}^{K} \overbrace{\begin{bmatrix} x_{(1)} \\ x_{(2)} \\ x_{(3)} \\ \vdots \\ x_{(n)} \end{bmatrix}}^{CoM_x} \tag{6}
$$

Since *n* must be a finite value, therefore, boundary conditions must be used. Kagami et al. [10] assumed T_s to be a given time period of the walking step, and he presented boundary conditions for the beginning and the end of the walking step. The walk was assumed to be statically stable in the beginning and end of each walking step, and $a_1=a_n=0$. In this algorithm, the initial and final positions of CoM during a walk step need to be defined before the robot actually starts to walk. These are drawbacks of this approach, since the beginning of each walking step is not always statically balanced, i.e. the initial acceleration and velocity of the CoM have not been considered. In addition, before the robot starts to walk, it is not possible to define the exact position of CoM in the beginning and final of the walking step.

In order to remedy the aforementioned problems, we use the presented method by assuming $x_{(1)}$ equal to the middle of the feet at the beginning of the previous two walk steps, and $x_{(n)}$ is also equal to the middle of the feet in the predicted position of the following five steps. Consequently, the ZMP trajectory is given to the algorithm for a walk, which has been starting in the previous two steps and lasting in the future five steps. The CoM trajectory of the current walking step is extracted from the calculated CoM trajectory. At the beginning of each walking step, the presented procedure is repeated.

3. Vertical CoM trajectory Model

We consider the height trajectory as the periodic movement. In this study, vertical CoM trajectory is represented by the first five terms of the Fourier basis functions. Therefore, the equation of our vertical CoM trajectory generator is given in (7).

$$
F(t) = C + \sum_{i=1}^{i=2} \left(\beta_i \cos\left(\frac{i\pi t}{L}\right) + \alpha_i \sin\left(\frac{(i-1)\pi t}{L}\right) \right) \tag{7}
$$

The parameter L is equal to the step period, therefore the generator has five parameters such as C, β_1, β_2, α_1 and α_2. A black-box optimization approach can be applied in order to find the optimized hip height trajectory generator with respect to energy efficient walking, in the section 4 we describe our optimization scenario. The generated trajectory by Fourier basis functions is the input to the programmable CPGs.

In the CPGs implementation studies, nonlinear oscillators i.e. Hopf are interesting because of their synchronization properties when they are coupled with other oscillators or with an external drive signal. Most CPGs use phase-locking behavior for their coupling method [12]. If intrinsic frequency of the oscillator is close to

frequency component of the periodic input, phase-lock behavior will appear, and synchronization will be done perfectly. In 2006, Righeti et.al. designed an adaptive oscillator based on Hopf oscillator which was able to learn CPGs frequency from the frequency of periodic input signals [8]. They called their adaptive mechanism dynamic Hebbian learning because it shared similarities with correlation-based learning found in neural networks [13]. The structure of the network of adaptive Hopf oscillators is shown in figure 3.
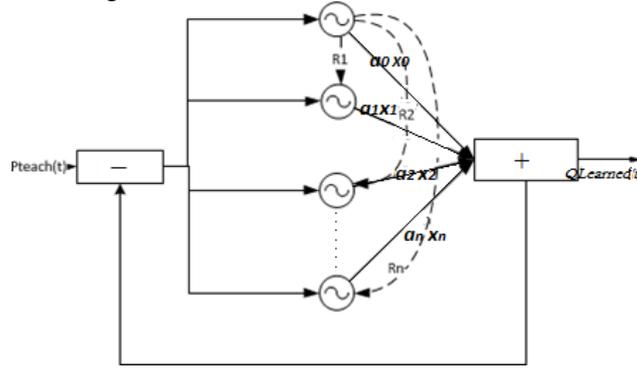
Fig. 3. A Schematic view of network of adaptive Hopf oscillators as a CPG block

In this study the external drive signal, or teaching trajectory, is the output trajectory of the presented Fourier based generator. Each oscillator is responsible for learning one frequency component of the signal. The network can be designed by four oscillators and each oscillator is denoted by i. The output of the system is the weighted sum of the output of the oscillators $Q_{learned}(t) = \sum_i a_i x_i$, here a_i is assumed the amplitude of each learned frequency. By using negative feedback loop, the already learned frequencies will be subtracted from the teaching signal $F(t) = P_{teach}(t) - Q_{learned}(t)$. It leads the system to adapt to remaining frequencies component which have not yet converged. According to the fact that each oscillator has its own phase shift, a variable encoding phase difference between the oscillator and the first oscillator of the network is associated with each of them. In order to reproduce any phase relationship between the oscillators, the Kuramoto coupling scheme [14] is used. The Equations describing the Total CPG's learning and dynamics are given as follow.

$$\dot{x}_i = \gamma(\mu - r_i{}^2)x_i - \omega_i y_i + \epsilon F(t) + \tau \sin(\theta_i - \emptyset_i) \tag{8}$$

$$\dot{y}_i = \gamma(\mu - r_i{}^2)y_i - \omega_i x_i \tag{9}$$

$$\dot{\omega}_i = \epsilon F(t)\frac{y_i}{r_i} \tag{10}$$

$$\dot{a}_i = \beta x_i F(t) \tag{11}$$

$$\dot{\emptyset}_i = sin(\frac{\omega_i}{\omega_0}\theta_0 - \theta_i - \emptyset_i) \tag{12}$$

$$\theta_i = sgn(x_i)cos^{-1}(-\frac{y_i}{r_i}) \qquad (13)$$

Equation 8, 9 and 10 are representing Hopf oscillator and its frequency learning, where γ controls the speed of recovery after perturbation. In equation 8, the Kuramoto coupling method is represented by $\tau \sin(\theta_i - \emptyset_i)$ in order to achieve phase synchronization between oscillators. Each adaptive oscillator is coupled with oscillator 0, with strength τ to keep correct phase relationships between oscillators. $\emptyset_i$ is the phase difference between oscillator i and 0.

Equation 12 and 13 shows how $\emptyset_i$ can converge to the phase difference between the instantaneous phase of oscillator $0, \theta_0$ scaled at frequency ω_i and the instantaneous phase of oscillator i, θ_i Learning rule for updating a_i is presented by equation 11, where β is learning rate. Learning rule shows how correlation between x_i and *F(t)* will be maximized. The correlation will be positive on average and will stop increasing when frequency component ω_i disappears from *F(t)* because of the negative feedback loop. The negative feedback is working like amount of the error, and learning rule is working like the perceptron rule and since the input signal is linearly separable, the above online algorithm will converge.

As conclusion, applying learning rules given as differential equations, parameters such as intrinsic frequencies, amplitudes, and weights of phase coupling can be automatically adapted to a teaching signal. One of its interesting aspects is that the learning is completely embedded into the dynamical system, and does not require external optimization algorithms.

Using Fourier basis functions together with presented CPG concept the proposed trajectory generator model , or policy representation, has the following advantages:

Smoothness: Using the CPGs increase basin of stability of walking. CPGs generate smooth and continuous trajectories without sudden accelerations, which enable the robot not fall and also reduce its energy consumption. By changing the step length and period during the walk, the robot may change its energy efficient vertical CoM trajectory; CPGs make able this change to be smoothly. CPGs also have the ability of frequency adaptation when walking step period and CoM trajectory period changed.

Periodicity: The Fourier basis function is easily able to represent periodic or cyclic movement. A biped walk often consists of periodic movements.

Convergence: the frequency of a walk is equal to the frequency of its vertical CoM trajectory. Using Fourier basis function the frequency parameters is eliminated, therefore the robot converges to the energy efficient walk faster compared to the approaches uses Spline basis function [6].

4. Learning Scenario

In this study, the vertical CoM trajectory is represented by the first five terms of the Fourier basis functions. The optimized energy efficient vertical CoM motion must be achieved for different step periods and step lengths. Since the step length and step period are continuous variables, they are discretized with a proper resolution. The

boundaries of the step lengths and step period resolutions used in this work are the following:

Step Length= [0.06…..0.18] *m*; *Resolution* = 0.04
Step Period= [0.4…. 0.8] *s*;   *Resolution* = 0.2

There are 12 possible combinations of the step periods and step lengths, and for each of them the energy optimization is performed. The optimized values of the Fourier basis functions terms must be found, with respect to minimization of actuator electrical power consumption. Bipedal walking is known as a complicated motion since many factors affect walking style and stability, such as robot's kinematics and dynamics, collision between feet and the ground. In such a complex motion, relation between gait trajectory and walking characteristic, e.g. energy consumption, is nonlinear. Stochastic optimization algorithms can be applied to find the optimized parameter values of the CoM vertical trajectory generator with respect to generate an energy efficient walk.

In this paper, Covariance Matrix Adaptation Evolution Strategy (CMA-ES) is used as a stochastic optimization algorithm for our gait optimization scenario. CMA-ES is a population-based stochastic, derivative-free method, which can be used in black-box optimization problems or direct policy search reinforcement learning. It has been successfully applied previously on gait optimization scenarios [15][16]. It is also reported that CMA-ES could achieve better results and faster convergence compared to other famous stochastic optimization techniques such as particle swarm optimization (PSO) and Genetic Algorithm (GA) [16].

CMA-ES generates a set of candidates, as the population, sampled from a multivariate Gaussian distribution. After generating the population, CMA-ES evaluates each candidate with respect to a fitness measure. After evaluating all the candidates in the population, the mean of the multivariate Gaussian distribution is recalculated as a weighted average of the candidates with the highest fitness. The covariance matrix of the distribution is also updated to bias the generation of the next set of candidates toward directions of previously successful search steps. In this study the population size is assumed to be 8.

For minimization of electrical power and energy consumption, the electrical power must be measured. The electrical power for a motor can be given in a simple form by $P_m = I^2 R$, where I is the current, R the resistance. The motor stall torque is calculated by $\tau = K_t I$, where K_t is the motor torque constant. By combining these expressions, the electrical power can be rewritten as $P_e = \frac{r}{K_t{}^2} \tau^2$. Therefore, in this study, the cost metric is measured by the sum of the joint-torques squared.

5. Results and Discussions

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 simulation is carried out using the RoboCup soccer simulator, rcsssever3d, which is the official 3D simulator released by the RoboCup community, in order to simulate humanoids soccer match. The simulator is based on Open Dynamic Engine (ODE). The ODE can report the produced torque of each joint in

each simulation step time, therefore the sum of the joint-torques squared can be calculated as the cost function.

Using the CMA-ES, after 30 iterations and 240 trails, the robot could reduce its energy usage by 25 percent, on average in all learning scenarios. The optimization is performed for 10 seconds walking with all the step lengths and step periods, which were presented in section 4. Figure 4, shows the convergence of the cost function of the learning scenario for the walk with step length 0.1 *m* and step period 0.4 *s*. The optimized vertical CoM trajectory for this walk is also shown in the figure 5.
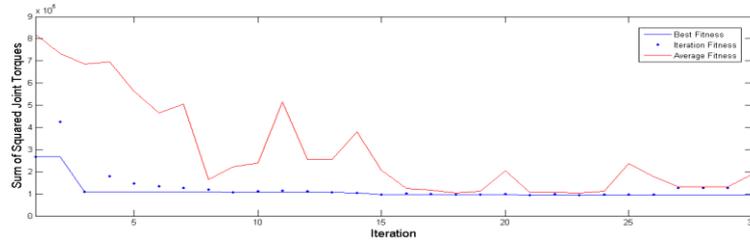
Fig 4. CMA_ES convergence for walk with step length 0.10 *m* and step period 0.4 *s*
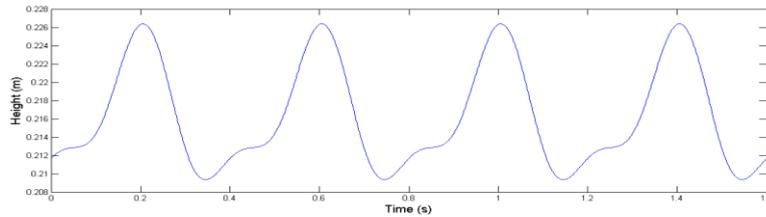
Fig. 5. CoM vertical trajectory for a walk during 1.6 s

The walking achieved by the above learning scenario presented in figure 4 is compared to the fixed height walking with the height assumed to be the offset of learned trajectory. The results show the sum of squared torques of the walk with variable height is reduced by 25 percent compared to the walk with fixed height. The same walking parameters with minor tuning are tested on a real NAO robot and the robot achieved to a more energy efficient walking. Please refer to https://www.dropbox.com/s/ft1en4blotnwcrx/Real_Robot_Experiment.wmv to watch the video of the fixed height walking versus the variable height walking. Figures 6 and 7, show the convergence of the cost function and the energy efficient vertical CoM vertical trajectory for the walk with step length 0.1 m and step period 0.8 s.
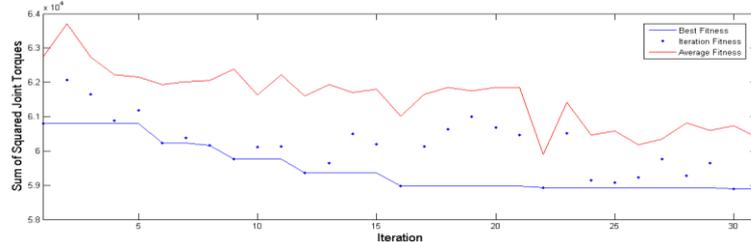

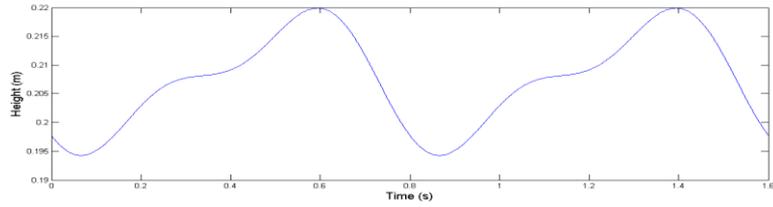**Fig 6.** Learning convergence for walking with step length 0.10 m and period 0.8 s

Fig. 7. Energy efficient CoM vertical trajectory during two step periods

Figures 8 and 9, also show the convergence of the cost function and the optimized vertical CoM trajectory of the walk with step length 0.14 *m* and step period 0.4 *s*.
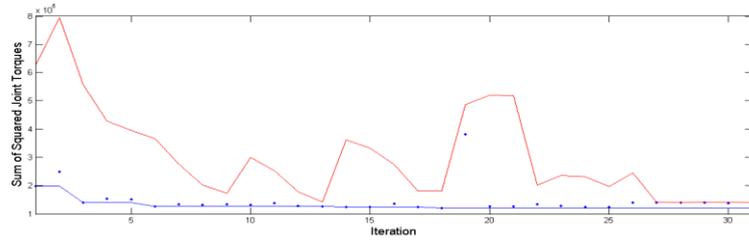

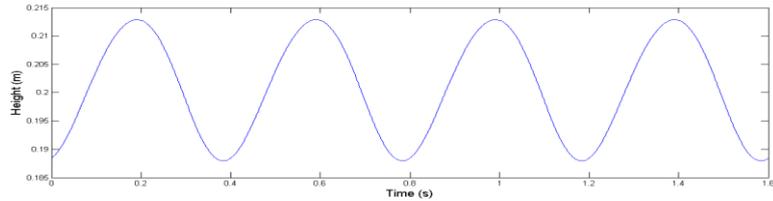**Fig. 8.** CMA_ES convergence for walk with step length 0.14 *m* and step period 0.4 *s*

Fig. 9. Optimized vertical CoM vertical trajectory for the above walking scenario

As shown in figures 5, 7 and 9, the optimized CoM vertical trajectory for different walking characteristics are different. By using programmable CPGs, the robot can change its walking speed, and modulation of the CoM trajectories can be done autonomously**.** Figure 10 shows the modulation of CoM trajectory of a walk when the robot, changes its walking characteristics from walk with step length 0.10 *m* and step period 0.8 *s* to the walk with step length 0.14 *m* and step period 0.4. This change is happening after the four seconds from the starting of the walk.
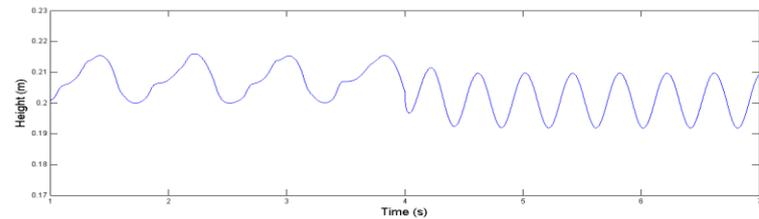

Fig. 10. Modulation of the vertical CoM trajectory by using the CPGs

For the same walking scenario shown in the figure 10, we test the change of the vertical CoM trajectories, this time with only use the Fourier basis function generator. Figure 11, shows this experiment. This figure also illustrates that, at the time when the change is happening, the CoM vertical trajectory has the sudden acceleration. In our experiment the robot in this scenario fell four times in 10 tests, this happens because of the explained sudden acceleration in vertical CoM trajectory. Nevertheless, by using CPGs for the same walking scenario the robot did not fall in 10 times tests, because of the smooth change in vertical CoM trajectory, as it is shown in figure 10.
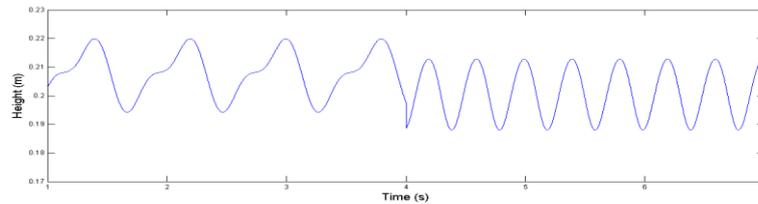

Fig. 11. Changing the vertical CoM trajectory by the Fourier based generator

7. Conclusions

This paper presented an approach to create an energy efficient walk. The walking controller approach is a ZMP based approach, which the ZMP dynamics is modeled by an inverted Pendulum model. A numerical approach is used to generate the horizontal CoM trajectory. The main contribution of this paper is the using of the CPG approach with Fourier based function in order to formulate the vertical CoM trajectory generator. By using the CMA-ES, an energy efficient walk is achieved for walking types with different characteristics, including the step lengths and periods.

The results show that by optimizing the vertical CoM trajectory, the energy consumption of the walk is reduced by as much as 25 percent compared to the walking with fixed height. For different step lengths and periods, the optimized CoM vertical trajectory is different in shape and characteristics. By using CPGs the online modulation and change of the vertical CoM trajectories is done smoothly and without jerk.

Since a ZMP-based approach is used and CPGs can generate smooth trajectories, the generated walking is stable, and the risk of hardware damage during the gait learning procedure is low. Therefore, Future work will be concerned with performing the gait learning directly on a real NAO robot. For improving the learning generalization, the linear regression may also be used to obtain the predicted values of the Fourier basis function terms based on new walk parameter values.

Acknowledgements

References

1. Gordon, K. E.,Ferris, D. P.,and Kuo, A. D.: Metabolic and Mechanical Energy Costs of Reducing Vertical Center of Mass Movement During Gait. , Arch. Phys. Med. Rehabil., vol. 90, pp. 136–144, (2009)
2. Kajita, S.,Kanehiro, F.,Kaneko, K.,Yokoi, K.,and Hirukawa, H.: 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, pp. 239–246, (2001)
3. Kajita, B. Y. S.,Nagasaki, T.,Kaneko, K.,and Hirukawa, H.: ZMP-Based Biped Running Control. in: IEEE/RSJ International Conference on Intelligent Robots and System, , (2007)
4. Kajita, S.,Kanehiro, F.,Kaneko, K.,and Fujiwara, K.: Biped walking pattern generation by using preview control of zero-moment point. , in: IEEE International Conference on Robotics and Automation, pp. 1620–1626, (2003)
5. Kuo, A. D.,Donelan, J. M.,and Ruina, A.: Energetic consequences of walking like an inverted pendulum: step-to-step transitions. , Exerc. Sport Sci. Rev., vol. 33, no. 2, pp. 88–97, (2005)
6. Kormushev, P.,Ugurlu, B.,Calinon, S.,Tsagarakis, N. G.,and Caldwell, D. G.: Bipedal walking energy minimization by reinforcement learning with evolving policy parameterization. , in: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 318–324, (2011)
7. Hansen, N.: The CMA evolution strategy: A tutorial. , (2005)
8. Righetti, L. and Ijspeert, A. J.: Programmable central pattern generators: an application to biped locomotion control. , in: IEEE International Conference on Robotics and Automation, 2006, pp. 1585–1590, (2006)
9. Vukobratović, M. and Juricić, D.: Contribution to the synthesis of biped gait. , IEEE Trans. Biomed. Eng., vol. 16, no. 1, pp. 1–6, (1969)
10. Kagami, S.,Nishivaki, K.,Inaba, M.,and Inoue, H.: A Fast Dynamically Equilibrated Walking Trajectory Generation Method of Humanoid Robot. , Auton. Robots, vol. 12, no. 1, pp. 71–82, (2002)
11. Domingues, E.,Lau, N.,Pimentel, B.,Shafii, N.,Reis, L.,and Neves, A.: Humanoid behaviors: from simulation to a real robot. .in: EPIA'11, Springer, pp. 352–364, (2011)
12. Pikovsky, A.,Rosenblum, M.,and Kurths, J.: Synchronization: A Universal Concept in Nonlinear Sciences. Cambridge University Press, (2003)
13. Righetti, L.,Buchli, J.,and Ijspeert, A.: Dynamic Hebbian learning in adaptive frequency oscillators. , Phys. D Nonlinear Phenom., vol. 216, no. 2, pp. 269–281, (2006)
14. Acebrón, J.,Bonilla, L.,Pérez Vicente, C.,Ritort, F.,and Spigler, R.: The Kuramoto model: A simple paradigm for synchronization phenomena. , Rev. Mod. Phys., vol. 77, no. 1, pp. 137–185, (2005)
15. MacAlpine, P.,Barrett, S.,Urieli, D.,Vu, V.,and Stone, P.: Design and optimization of an omnidirectional humanoid walk: A winning approach at the RoboCup 2011 3D simulation competition. , in: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence,,pp. 1047-1053, (2012)
16. Farchy, A.,Barrett, S.,MacAlpine, P.,and Stone, P.: Humanoid robots learning to walk faster: From the real world to simulation and back. , In: international conference on Autonomous agents and multi-agent systems, pp. 39-46, (2013)