

Nature-Inspired Optimization for Biped Robot Locomotion and Gait Planning

Shahriar Asta and Sanem Sariel-Talay

Computer Engineering Department
Istanbul Technical University, Istanbul, Turkey
{asta,sariel}@itu.edu.tr

Abstract. Biped locomotion for humanoid robots is a challenging problem that has come into prominence in recent years. As the degrees of freedom of a humanoid robot approaches to that of humans, the need for a better, flexible and robust maneuverability becomes inevitable for real or realistic environments. This paper presents new motion types for a humanoid robot in coronal plane on the basis of Partial Fourier Series model. To the best of our knowledge, this is the first time that omni-directionality has been achieved for this motion model. Three different nature-inspired optimization algorithms have been used to improve the gait quality by tuning the parameters of the proposed model. It has been empirically shown that the trajectories of the two specific motion types, namely side walk and diagonal walk, can be successfully improved by using these optimization methods. The best results are obtained by the Simulated Annealing procedure with restarting.

Keywords: Biped robot locomotion, gait planning, coronal plane, sagittal plane, omnidirectional locomotion, simulated annealing, genetic algorithms.

1 Introduction

Gait planning is essential for a robot to navigate in an environment. Especially humanoid robot gait planning problem has come into prominence due to the recent research trend toward human-like robots and the difficulty of generating stable motions for biped locomotion. Several different gait planning models have been developed for humanoid robots. These models rely on various mathematical and kinematic approaches. ZMP based motion model which was originally introduced by Vukobratović and further developed and extended by others [1-3], is one of the most popular gait generation models in mobile robotics. In this approach, the main objective is to design robot's motion in such a way that the zero moment point (the point where total inertia force equals to zero), does not exceed a predefined stability region. Another popular gait generation model considers the Center Of Mass (COM) to evaluate the quality of the generated gait [4]. Central Pattern Generator (CPG) method [5] generates gaits by considering a lattice of neurons for the production of repetitive patterns.

As a recent work, the Partial Fourier Series (PFS) method [6] was introduced for generating a forward walk motion in the sagittal plane. The parameters of the proposed

model were optimized using a Genetic Algorithm. The PFS method is easy to implement on humanoid robots, and promising results has been observed for a forward walk motion in [6].

We extend the work by [6] by applying the PFS model for different types of biped motions. The main objective of this paper is to introduce omni-directionality to PFS motion model. The first step toward this goal is to show that this motion model is suitable for biped locomotion in coronal plane in addition to the sagittal plane. For that purpose, the models of two important motions for a humanoid robot, namely, side walk and diagonal walk are introduced in this plane. These motions are constructed based on the physical and kinematic models. However, their parameters are needed to be optimized. Therefore, an optimization step is needed. In this paper, the parameters of PFS model are optimized by using nature-inspired methods: variations of Simulated Annealing and Genetic Algorithms. This paper presents both the optimized parameters and a comparative analysis of these methods for humanoid gait planning problem.

The rest of the paper is organized as follows. First, we present a brief review on the humanoid robot mode used in the experiments and its basic motion model. Then, the selected Simulated Annealing and Genetic Algorithms are reviewed. After this review, the two new motion types in coronal plane are introduced. It has been shown that the robot is able to perform a side walk and a diagonal walk using the Partial Fourier Series model. After presenting the model, the optimization processes are discussed. Finally, the experiments and the results are presented followed by the conclusion.

2 Motion Model

The PFS model was first introduced by [6] for the simulated Aldebaran Nao robot model in Simspark simulation environment. Simspark is the official simulator for RoboCup competitions and uses ODE (Open Dynamics Engine) for physics simulation [7]. We use the same robot model for presenting our proposed motions.

Nao robot has 22 degrees of freedom of which only 12 have been used in this motion model. The height of the robot is 57 cm and its weight is 4.3 kg. Since the simulated Nao robot (Fig. 1(b)) is a realistic representation of the real Nao humanoid robot, its joint structure (Fig. 1(c)) is the same with the real one.

The human walking motion can be modeled by a smoothed rolling polygonal shape and a periodic function accordingly. In this sense, one can use the principle of Partial Fourier Series (PFS) in order to decompose the bipedal walk's periodic function into a set of oscillators [9]. Assigning these oscillators to the joints of a humanoid robot enables one to develop a gait for the robot. According to [4], the main periodic function for the bipedal walk can be formulated as following:

$$f(t) = C + \sum_{i=1}^N A_i \sin(i \frac{2\pi}{T} t + \phi_i) \quad (1)$$

Where N is the number of frequencies (degrees of freedom which are used in gait definition), C is the offset, $A_{1...N}$ are amplitudes, T is the period and $\phi_{1...N}$ are phases.

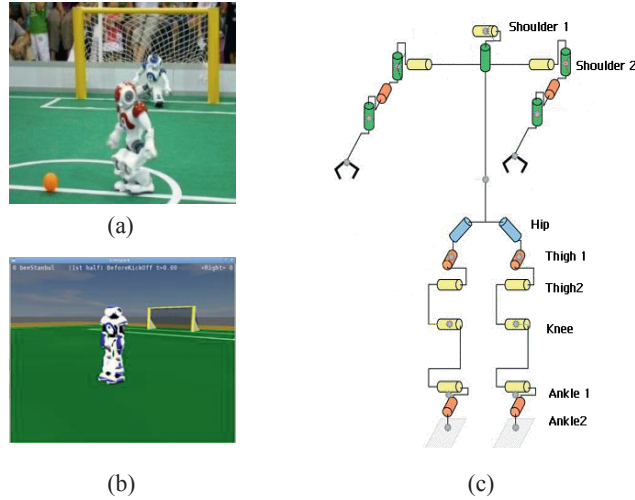


Fig. 1. a) The Aldebaran Nao robot at RoboCup Standard Platform League [8] b) Simulated Nao Robot at Simspark environment, c) Nao Robot Structure [7]

This PFS model ensures the right and left feet of the robot alternately perform swing and support roles. Note that, this alternation can be achieved by adding a shift of π for pitch joints of the right foot. The above mentioned oscillators give the desired angle of joints in a specific time. In order to control the robot with these joint angles, we use a simple proportional control method:

$$Speed * (\theta_{Target} - \theta_{Current}) \quad (2)$$

Sending the outcome to the robot's motion interface and then to Simspark causes the joints to move with the desired speed and value.

3 Motions in Coronal Plane

There are three major planes in which humanoid robot locomotion is considered: Sagittal plane, Coronal plane and Transverse plane. Forward and backward walk motions moves the robot in the sagittal plane. Motions in Coronal plane move the robot toward the sides. And finally, Transverse plane motions change the orientation of the robot. In this paper, we have extended the PFS model which is proposed for forward walk [6] into a model for coronal plane motions: side walk and diagonal walk.

3.1 Side Walk

The side walk motion is a coronal plane motion which does not change the orientation of the robot. The side walk motion involves the oscillation of two types of joints, namely, the pitch and the roll joints. We use the pitch joints to produce single support locomotion. This is based on a simple intuition rather than a detailed kinematics analysis. The idea is closing the swinging foot's pitch joints, that is, assigning values to these joints so that the foot is detached from the ground and moved toward the

robot's torso. Opening pitch joints has exactly the reverse meaning. Closing pitch joints opens space for the swinging foot to move along the coronal plane and prevents from collision with the ground. At the same time, the support foot's pitch joints are opened to expand the space created for the swinging foot. For roll joints however, opening means, assigning values to the joints so that the foot moves away from the torso side ward and closing means the reverse action.

Both feet's roll joints have to be opened to opposite sides simultaneously to their respective leg's pitch joints. When the swinging foot touches the ground, the pitch joints should be close to their offset value to ensure sideward stability. This is where the roll joints have to be at their maximum values in order to maximize the force applied to the ground. In other words, there should always be a $\pi/2$ phase difference between the roll and pitch joints of the same leg. At the same time, in order to ensure that roll joints are closed and opened in opposite directions there should also be a phase difference of π between the corresponding roll joints of the two legs. Note that, in order to maintain a support-swing sequence, a phase difference of π is applied between the corresponding pitch joints of the two legs.

Table 1. The joint oscillators for a side walk motion in the coronal plane

Left Leg Joints	Right Leg Joints
$f_{LShoulder2}(t) = C_1 + A_1 \sin\left(\frac{2\pi t}{T} + \frac{\pi}{2} + \varphi_1\right)$	$f_{RShoulder2}(t) = -C_1 + A_1 \sin\left(\frac{2\pi t}{T} - \pi/2 + \varphi_1\right)$
$f_{LThigh1}(t) = C_2 + A_2 \sin\left(\frac{2\pi t}{T} + \pi/2 + \varphi_2\right)$	$f_{RThigh1}(t) = -C_2 + A_2 \sin\left(\frac{2\pi t}{T} - \pi/2 + \varphi_2\right)$
$f_{LThigh2}(t) = C_3 + A_3 \sin\left(\frac{2\pi t}{T} + \varphi_3\right)^-$	$f_{RThigh2}(t) = C_3 + A_3 \sin\left(\frac{2\pi t}{T} + \pi + \varphi_3\right)^-$
$f_{LKnee}(t) = C_4 + A_4 \sin\left(\frac{2\pi t}{T} + \pi + \varphi_4\right)^+$	$f_{RKnee}(t) = C_4 + A_4 \sin\left(\frac{2\pi t}{T} + \varphi_4\right)^+$
$f_{LAnkle1}(t) = C_5 + A_5 \sin\left(\frac{2\pi t}{T} + \varphi_5\right)^-$	$f_{RAnkle1}(t) = C_5 + A_5 \sin\left(\frac{2\pi t}{T} + \pi + \varphi_5\right)^-$
$f_{LAnkle2}(t) = C_6 + A_6 \sin\left(\frac{2\pi t}{T} + \pi/2 + \varphi_6\right)$	$f_{RAnkle2}(t) = -C_6 + A_6 \sin\left(\frac{2\pi t}{T} - \frac{\pi}{2} + \varphi_6\right)$

The set of equations in Table 1 presents the coronal plane motion gait planning equations for each joint. One has to note that, in the kinematic design of the simulated Nao robot, the roll joints' y axes (horizontal) point to the opposite directions [7]. That is the reason for using negative coefficients for *RShoulder2*, *RThigh1* and *RAnkle2* joints. The offset and amplitude values of the corresponding *Thigh2* and *Ankle2* joints are initially set as the same for the simplicity of the model ($A_3 = A_5$, $C_3 = C_5$). The + and - superscripts in the formulation of some joints represent positive or negative half of their sinusoids respectively. The formulation in Table 1 results in a side walk motion with an average speed at 40 cm/sec. However, the parameters are needed to be optimized for a faster and straight sidewalk.

3.2 Diagonal Walk

Diagonal walk is another useful motion in the coronal plane. The robot is supposed to plan a straight trajectory line at a 45 degree angle with the horizon, without changing its orientation. The same intuition for the side walk motion also applies to the

diagonal walk. One has to note that, the diagonal walk motion is a variation of a side walk motion for which both forward and sideward force vectors are applied. This is why an additional joint, the hip joint in Fig1(c), is used to create the required forward force. Therefore, the diagonal walk motion model can be given as an extended side walk model with the following two PFS formulations (one for each leg's hip joint).

$$\begin{aligned} f_{LHip}(t) &= -C_{Hip} + A_{LHip} \sin\left(\frac{2\pi t}{T} + \pi/2 + \varphi_{Hip}\right) \\ f_{RHip}(t) &= -C_{Hip} + A_{RHip} \sin\left(\frac{2\pi t}{T} - \pi/2 + \varphi_{Hip}\right) \end{aligned} \quad (3)$$

Note that, unlike other joints, amplitudes for the right and left hips are not necessarily the same. Just like side walk, we assign parameters to C_{Hip} , A_{LHip} and A_{RHip} based on a simple reasoning. The resulting diagonal walk motion does not follow the desired trajectory. Snapshots of these initial side walk and diagonal walk motions are shown in Fig. 2.

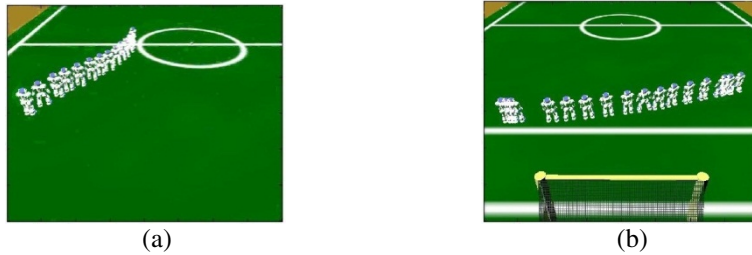


Fig. 2. The motion patterns before optimization (a) side walk (b) diagonal walk

4 Optimization Methods

We have investigated Simulated Annealing and Genetic Algorithm optimization methods in order to optimize the parameters of the motion model of a humanoid robot.

Simulated Annealing (SA) is a global optimization method which applies a variation of Hill Climbing search to improve the solution quality but performing probabilistic random moves to escape from local optima. The probability value decreases exponentially according to a function of the temperature (T) value as in the original annealing process [10]. At higher values of T bad moves are more likely to be allowed. As T closes to zero, bad moves are less likely to happen, until T is zero in which case, the algorithm behaves like Hill Climbing. In this paper, we have tested two configurations of Simulated Annealing procedure. One configuration is the simple SA algorithm as mentioned above. The other includes restarting after a cooling procedure.

In SA with restarting (SAR), the simulated annealing algorithm is iterated inside a main loop which repeatedly resets the temperature to its initial value. A small cooling factor which causes sharp temperature decrease after the evaluation of each individual is used in the annealing process. This reduces the number of annealing iterations and the number of times that SAR considers to replace the current individual with the next one. In such a case, the probability of replacing the current individual is less than that of SA without restart.

```

Initialize (Curr , CurrentFitness = Fitness(Curr) , T)
for (iteration) do
  reset temperature
  while (T >= End Temperature)
    Next ← ComputeNext ()
    ΔE ← Fitness(Next) – CurrentFitness
    if(ΔE < 0)
      Curr ← Next
      CurrentFitness ← CurrentFitness + ΔE
    else
      p = e-ΔE/T
      if(p > 0.5)
        Curr ← Next
        CurrentFitness ← CurrentFitness + ΔE
      if (Fitness < ΔE)
        CurrentFitness ← Fitness
      T ← T * CoolingRate

```

Fig. 3. Pseudo code for the simulated annealing with random restart

In Genetic Algorithms (GAs), a population of individuals is evolved through generations. Each individual contains a possible solution to the problem in the form of numerical values. In order to generate the next population, individuals undergo several operations [10]. There are three major operations in a simple Genetic Algorithm: selection, crossover and mutation. A standard genetic algorithm approach is used in this research. The corresponding parameters for these optimization methods are presented in the next section.

5 Optimization of Motion Models

As discussed in the previous sections, the constructed PFS model produces the specified motion patterns. However, the parameters are needed to be tuned to obtain the motion trajectories in their desired sense (e.g., the speed of the side walk motion needs to be fastened and the orientation should be kept constant during a diagonal walk). Nature-inspired optimization methods are suitable for addressing these types of problems due to their size and complexity. In this work, three optimization methods, namely, Simulated Annealing (SA), Simulated Annealing with restarts (SAR) and Genetic Algorithms (GA) have been utilized to optimize the parameters and a comparative analysis is provided for humanoid gait planning problem. The following three subsections present the settings of the algorithms. Different fitness functions are constructed for each motion type.

The fitness function for the side walk is expressed as :

$$fitness = d_{target} + \bar{\theta}, \quad \bar{\theta} = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x}) + \sum_{i=1}^N (y_i - \bar{y}) + \sum_{i=1}^N (z_i - \bar{z})}{N}} \quad (4)$$

This fitness function has been adapted from [6]. Here d_{target} is the final distance to the target at the end of each simulation trial. The target is fixed and defined as the location of the ball at the center of the field. $\bar{\theta}$ is the average oscillation of the torso

(in radians per second). The oscillation values are received from robot's gyro which is installed on the torso. Note that, the Simspark is a realistic simulation environment in which actions are nondeterministic and noisy. That is, the outcome of an action is not completely predictable as desired. Therefore, the fitness value of a parameter set is determined as an average of 3 simulation trials.

Similarly the fitness function for the diagonal walk is designed as :

$$fitness = d_{target} + \bar{\varphi} \quad (5)$$

Where d_{target} is the distance to the target from the initial location of the robot. The target is simulated as the fixed location of the ball at the center of the field. $\bar{\varphi}$ is the average absolute value of the horizontal angle detected by the robot's camera during a trial.

$$\bar{\varphi} = \frac{\sum_t |\varphi_t|}{N} \quad (6)$$

In both equations (4) and (6), N is the number of simulation cycles in each trial. A simulation cycle is 0.02 sec. Duration of each trial is 10 and 7 seconds for the diagonal walk and the side walk respectively. The initial location of the robot at the start time of a trial is (-8,0) for side walk and (-5,5) for diagonal walk. For the side walk experiments, the robot's orientation is set to an angle such that torso's x axis is parallel to the midfield line. At the start of each trial, the robot's head is rotated 90 and 45 degrees toward the target(i.e., the ball) for the side walk and diagonal walk, respectively. Corresponding angle for the head is kept constant during a trial.

The selected methods and the parameter values for GA, SA and SAR during the optimization of side walk and diagonal walk gait planning are shown in Table 2.

Table 2. The experimental settings for the optimization algorithms

Genetic Algorithm				
P_{mutation}	P_{crossover}	Crossover Type	Population Size	Stop Criteria
0.55	0.85	Uniform	50	Similarity of population
Elitism	Selection Policy	Mutation Type	Fitness Function	# of Parameters
10%	Proportional	Random from the range of joint values	Minimization	8 (SW)
Simulated Annealing				
Initial Temperature	Cooling Rate	End Temperature	Fitness Function	# of Parameters
4000	0.99	0.001	Minimization	11 (DW)
Simulated Annealing with Restart				
Initial Temperature	Cooling Rate	End Temperature	Fitness Function	# of Parameters
400	0.5	0.001	Minimization	11(DW)/8(SW)

In GA, each individual represents the parameters $\varphi_1 \dots \varphi_6$ and T in equations of table 1 along with the speed in Equation (2). All the represented values in an individual are double. SA in its original form has been used for diagonal walk optimization. Each individual in SA represents the parameters $\varphi_1 \dots \varphi_6$ and T in equations of table 1, and the φ_{Hip} , A_{LHip} , A_{RHip} , C_{Hip} in Equation (3). Note that, diagonal walk has 3 more parameters as compared to side walk, while speed is not considered as a parameter for Diagonal Walk. SAR has been tested and applied on both motions with the same setting. In SAR, the cooling rate is set to a lower value than that of the simple SA and the annealing is performed fewer times (20 times).

6 Experimental Results

The selected optimization algorithms have been tested for optimizing the parameters for the sidewalk and diagonal walk motions. Each algorithm on each respected motion type was run for 10 times. Since the simulation environment is non-deterministic, the fitness value for each individual is determined as an average of 3 evaluations.

Table 3. The overall best fitness values averaged over 10 runs for the two motion models

	Side walk			Diagonal Walk			
	μ	σ	Confidence Interval (99%)		μ	σ	Confidence Interval (99%)
GA	7.14	0.35	6.79 - 7.49	SA	3.17	0.18	2.99 - 3.35
SAR	5.83	0.39	5.44 - 6.15	SAR	2.29	0.13	2.16 - 2.42

The overall results (with the 99% confidence interval) are presented in Table.3. As can be seen, SAR performs better than SA and GA. It results in a lower mean fitness value among 10 runs and there is no overlap between the interval values.

SAR outperforms GA for the side walk motion. However at some points, the SAR shows a bad performance which closes it to the performance of GA. One other interesting fact is that sometimes the GA shows a better performance than its mean performance and closes down to the SAR's worst performance. This is due to the random nature of GA which is much more emphasized than SAR

A single run's comparative result is presented in Fig4 (a). As can be seen from these results, SAR can find the improved parameters in less than 250 iterations for the diagonal walk in one single run. This is equivalent to running the simple SA algorithm with the parameters in Table 2 for 10 loops with restarts. This comparative study illustrates that SAR outperforms the simple SA which is not able to find a reasonable result even in more iterations. When the performance of SAR is analyzed for two different motions, the convergence to the improved parameters for the diagonal walk has been observed in later iterations compared to that of the side walk. This is due to the structure of the diagonal walk which uses an additional joint (hip joint).

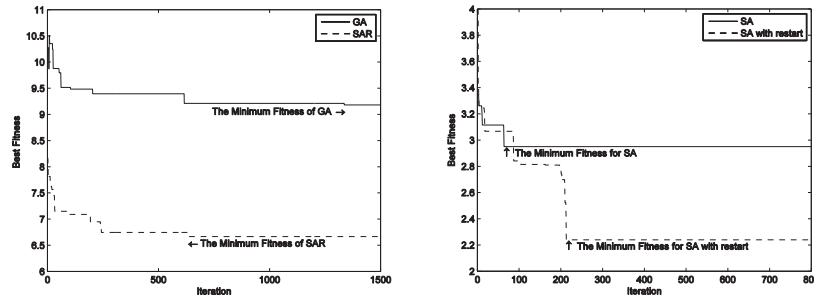


Fig. 4. Single run results of (a) GA, SAR for side walk. (b) SA, SAR for diagonal walk.

The improved side walk motion (resulting from SAR optimization) has a speed of 55 cm/sec (which is slightly better than its original version at 40 cm/sec) and the deviation from a straight baseline is considerably small. The diagonal walk motion with the optimized parameters ensures the robot to keep its orientation at a 45 degree straight line toward the target, and its speed is 35 cm/sec. The snapshots of the resulting motions are shown in Fig 5.



Fig. 5. (a) improved diagonal walk (b) improved side walk

7 Conclusions

In this work, we have shown that the PFS model can be successfully extended for omni-directional motions on humanoid robots. However, an optimization step is needed to specify the tuned parameters for the PFS model for different motions. It has been empirically shown that the optimization can be performed by using nature-inspired optimization techniques. A comparative analysis of three optimization methods is given for gait planning of two motion models for the simulated Nao robot in Simspark environment. As the experimental results illustrate, the best parameters are obtained by using Simulated Annealing with restarts. These best parameters produce the desired motion trajectories. The future work includes a general omni-directional gait planning based on the introduced PFS model for all motion directions.

Acknowledgement. Authors would like to thank Ismail Hakki Yadigar for his contribution in the implementation of the nature-inspired methods.

References

1. Vukobratović, M., Borovac, B.: Zero-moment point—Thirty five years of its life. *International Journal of Humanoid Robotics* 1(1), 157–173 (2004)
2. Yanjun, Z.: Fuzzy Omnidirectional Walking Controller for the Humanoid Soccer Robot. In: *IEEE-RAS International Conference on Humanoid Robots* (2009)
3. Gao, Z., Wu, J., Chen, H., Pan, B., Zheng, H., Liang, D., Liu, X.: Omnidirectional Motion Control for the Humanoid Soccer Robot. In: Cai, Z., Li, Z., Kang, Z., Liu, Y. (eds.) *ISICA 2009. CCIS*, vol. 51, pp. 1–8. Springer, Heidelberg (2009)
4. Graf, C., Härtl, A., Röfer, T., Laue, T.: A Robust Closed-Loop Gait for the Standard Platform League Humanoid. In: *IEEE-RAS International Conference on Humanoid Robots* (2009)
5. Carla, M.A.P., Golubitsky, M.: Central Pattern Generators for Bipedal Locomotion. *Journal of Mathematical Biology* 53(3), 474–489 (2006)
6. Picado, H., Gestal, M., Lau, N., Reis, L.P., Tome, A.M.: Automatic Generation of Biped Walk Behavior Using Genetic Algorithms. In: *Proceedings of the 10th International Conf. on Artificial Neural Networks: Part I: Bio-Inspired Systems: Computational and Ambient Intelligence* (2009)
7. Boedecker, J., Dorer, K., Rollman, M., Xu, Y., Xue, F., Buchta, M., Vatankhah, H.: *Simspark Manual* (2010)
8. Aldebaran Official Web Site, <http://www.aldebaran-robotics.com/>
9. Shafii, N., Javadi, M.H., Kimiaghdam, B.: A Truncated Fourier Series with Genetic Algorithm for the control of Biped Locomotion. In: *Proceeding of the 2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 1781–1785 (2009)
10. Russell, S.J., Norvig, P.: *Artificial Intelligence, A Modern Approach*, 2nd edn. (2003)