

Optimization of a Humanoid Robot gait: multilocal optimization approach

José Lima*, Ana I. Pereira†, Paulo Costa‡, José Gonçalves*

* *INESC TEC (formerly INESC Porto) and Polytechnic Institute of Bragança, Portugal*
E-mail:{jllima,goncalves}@ipb.pt

† *Polytechnic Institute of Bragança, Portugal & Algoritmi, University of Minho, Portugal.*
E-mail:apereira@ipb.pt

‡ *INESC TEC (formerly INESC Porto) and Faculty of Engineering, University of Porto, Portugal*
E-mail:paco@fe.up.pt

Abstract

The humanoid robot gait planning presents a large number of unknown parameters that should be found to make the humanoid robot to walk. There are several approaches to achieve the gait but an accurate simulation can be used to compute it. A stable joint model of a humanoid robot is used in simulation to optimize the gait parameters. The optimization is based on the stretched simulated annealing with the multilocal algorithm approach. Final results prove the benefits of the presented optimization algorithm.

Keywords: Humanoid robot. Optimization. Simulation.

1 Introduction

Recently, humanoid robots have been enjoying great popularity and are now used as a research tool in many groups worldwide. They are a challenging task due to its hard control. Moreover, gait generation and optimization still remain a challenge for such a high-order highly-coupled nonlinear dynamical system [Hu et al., 2006]. The humanoid robot gait planning presents a large number of unknown parameters that should be found to make the humanoid robot to walk. It can be approached in two ways: the online (done in real-time that requires high computational effort) and the offline gait generation methods. This offline approach, brings some advantages such as the ability to use complex algorithms to find an optimal solution. This is the main topic of this paper.

The optimization of the humanoid robot gait is a good area to apply optimization methods over the simulated robot. It is prudent to avoid time-consuming optimization runs that wear out the robot hardware. Several approaches have been presented that aim at optimizing properties such as speed [Faber and Behnke, 2007, Hemker et al, 2007, Niehaus et al., 2007] or torso stability [Chalodhorn et al., 2007, Huang et al., 2008, Zhe et al, 2005] of a humanoid's walk.

In order to generate walking patterns for different locomotion kinematics, the common way of most existing approaches is to precompute reference trajectories [Sakka and Yokoi, 2005]. Similarly and based on previous authors works, using precomputed reference trajectory, this paper presents an overview to optimize the gait planning methods for humanoid robots [Lima et al., 2010]. For that purpose, the Stretched Simulated Annealing method was used for the optimization technique. It is a generic probabilistic method for the global optimization problem. It tries to find the global minimum of a given function in a large search space (as it happens in humanoid gait planning) and can escape from local minima. Besides, there must be used an optimizing method because it is unthinkable to perform a complete search once the search space is huge (i.e. dimension 100).

Before the optimization on the real robot, several iterations were evaluated using a simulated model of the humanoid robot. The simulations were conducted in *SimTwo* [SimTwo, 2009], a physical robot simulator that is capable to simulate user-defined robots in three-dimensional space since it includes a physical model based on rigid body dynamics (the *ODE - Open Dynamics Engine* [Smith, 2000]).

The paper is organized as follows: Section 2, shortly point out the developed Simulator (*SimTwo*) and the humanoid robot modeling with its parameters. Then, in Section 3, it is addressed the optimization technique based on the stretched simulated annealing well known algorithm where the gait optimization is applied. Section 4 presents a discussion of results and finally, Section 5 concludes this work and gives a future work on this topic.

2 Simulation Environment

Studying the robot's behaviour without real hardware is possible due to a physics based simulator implementation. The physics engine is the key to making simulations useful in terms of high performance robot control [Browning and Tryzelaar, 2003]. The dynamic behaviour of the robot (or multiple robots) is computed by the ODE (Open Dynamics Engine [Smith, 2000]), which is a free library for simulating rigid body dynamics.

There are several simulators with humanoid simulation capability. SimTwo, as a developed simulator, is a generic simulator that allows the access to the low level behaviour, such as dynamical model, friction model, servomotor model and sensors model in a way that can be mapped to the real robot, with a minimal overhead. The developed simulator, based on ODE, allows to build several robots. SimTwo was developed having in mind the full access to all control levels and the possibility of adding several sensors and its modeling. Besides, it owns a new precise and stable joint model that allows to simulate a robot with a high number of joints in a chain architecture without instability and noise (previously presented in [Lima et al., 2013]). SimTwo also allows to run simulations faster than real time: a very useful property to accelerate the research task. A snapshot of the developed simulator is presented in Figure 1.

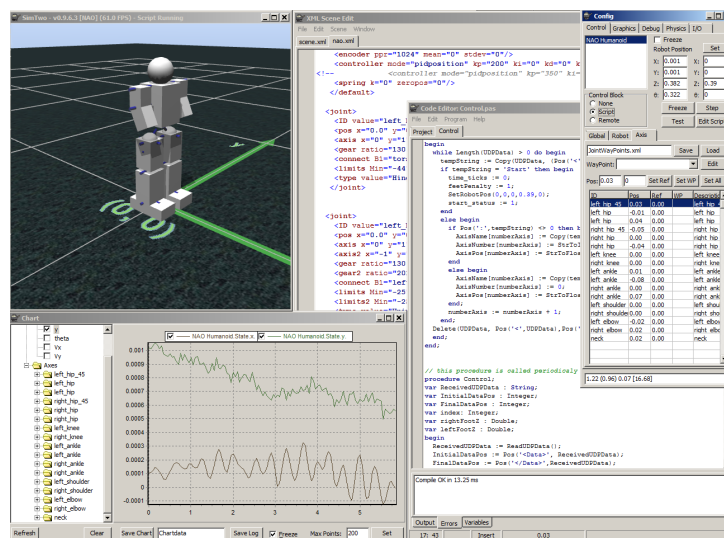


Figure 1: SimTwo simulator environment.

The dynamics realism in SimTwo is obtained by decomposing a robot in rigid bodies and electric motors (joints). Each body behaviour is numerically simulated using its physical characteristics such as shape, mass and moments of inertia, surface friction and elasticity. It is also possible to define standard joints such as socket, hinge and slider which can be coupled with an actuator or a sensor. Previous works validates the realism of the SimTwo simulator [Costa et al., 2011].

2.1 Servomotor Model

The servomotor can be based on a DC motor model with its main characteristics with an embedded state space feedback closed-loop controller. The block diagram of the servomotor model can be seen in Figure 2 where θ_{Ref}^i is the reference of the angle, ω_{Ref}^i is the reference of angular speed, Ua' is the Supply voltage of DC motor (with nonlinearities) and T_S is the available torque that will be applied to the gearbox (Electrical torque). The motor model parameter estimation has already been presented in previous work [Lima et al., 2005], as well as the friction constants that were computed based on an optimization method with validation of real hardware. The low level controller models the closed loop controller, in the real robot, which is implemented by the servomotor manufacturer (AX12 from Dynamixel).

2.2 Gearbox Model

A standard model would present some instability because an inertia moment of the motor shaft should appear multiplied by the square of gearbox ratio (in a chain of joints as it happens in a humanoid robot). The implemented model allows to increase the stability of the simulation with a spring damper

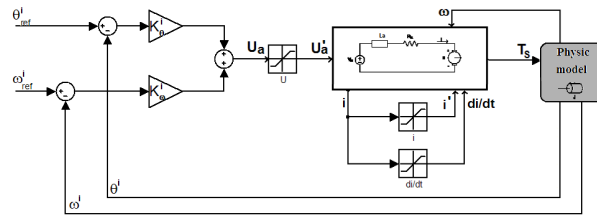


Figure 2: Motor model.

transmission approach. Moreover, the backlash of the gearbox can be approximated by this model. The improved model is implemented over the *physic model* presented in Figure 2, thus keeping the servo motor model and the controller unchanged.

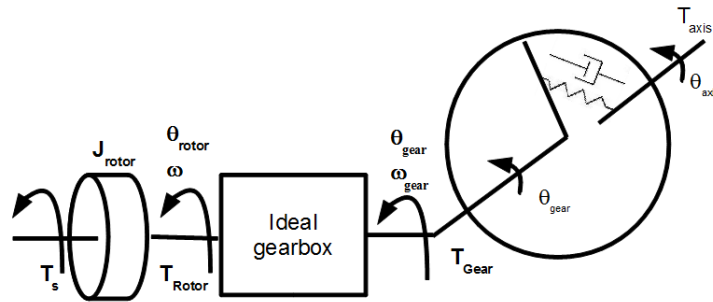


Figure 3: Gearbox model.

The motor inertia plus a spring damper gearbox model, *MIPSD* is detailed in [Lima et al., 2013] and presented in Figure 3 where the T_{axis} is the available torque at the gearbox output and T_{Rotor} is the available torque seen by the rotor.

2.3 Gait parameters

The gait-planning is one of the fundamental problems in humanoid mobile robots. The problem of gait planning for humanoid robots is fundamentally different from the path planning for wheeled robots due to the inherent characteristics of legged locomotion. The main challenge of gait planning is to find constraint functions and their associated gait parameters. However, finding repeatable gait when the constraint equations involve higher order differential equations still remained unsolved [Zhou et al., 2004]. There are the online and the offline generation methods [Zhou et al., 2004]. The first one, should be done in real time and requires a high computational effort. On the other hand, a popular way to solve this problem is to resort to offline optimization techniques. In this paper it is used the model parameters and actuator inputs that lead to fully open-loop stable walking motions (Figure 4).

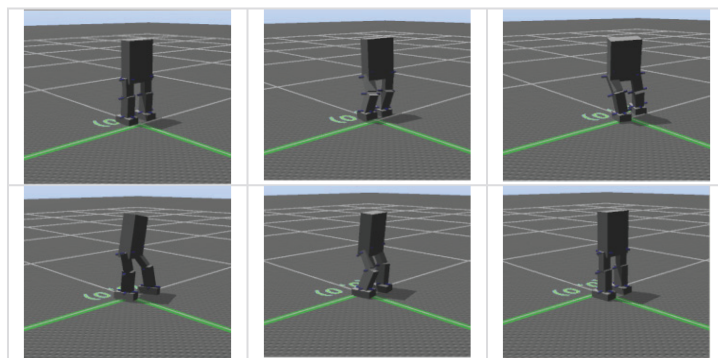


Figure 4: SimTwo Humanoid Simplified model walking

Having in mind the most basic locomotion of a humanoid robot (a simple step) it is desired to obtain the gait planning that allow joints to move the robot. It should be optimized according to the criteria:

maximizing the step distance. For the optimization technique, it was used the stretched simulated annealing. The search space in the present problem is composed by 10 degrees of freedom (each joint axis) and for 10 time instants: the optimization should be applied to a dimension (D) of 100. The optimization algorithm is implemented in Matlab that connects the simulator through network UDP packets and xml file. The simulator computes the walk distance and sends it to the Matlab so that to analyze the objective function, $f(x)$. Matlab optimization algorithm generates a new solution to be tested in simulator and shares it by a xml file and a start event of simulation by network, as presented in Figure 5.

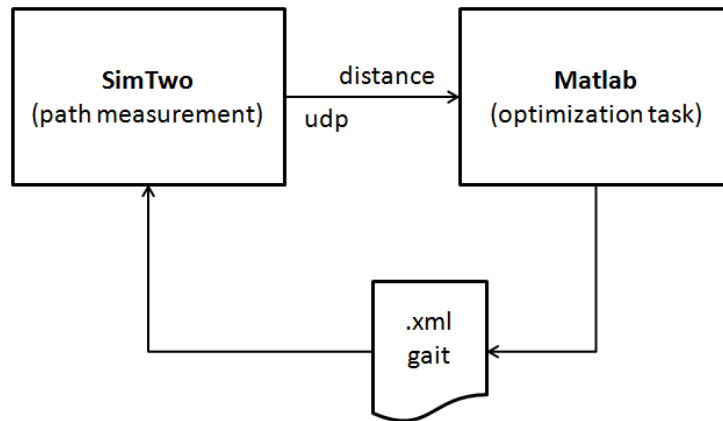


Figure 5: SimTwo and Matlab interface.

3 Stretched simulated annealing

The stretched simulated annealing (SSA) method belong to the class of the multilocal optimization methods. The SSA method solves, in each iteration, a global optimization problem using the simulated annealing (SA) algorithm. The SA is a point-to-point stochastic algorithm that does not require derivative information and is able to guarantee convergence to a global solution with probability one. In each iteration of the SSA method, the global problem is transformed using a function stretching technique [Pereira and Fernandes, 2009].

The function stretching technique aims to prevent the convergence of the SA algorithm to a previously computed global solution. Let x^j be that particular solution. Thus, the function stretching technique is applied only locally, in order to transform the objective function $f(x)$ in a neighbourhood of x^j , say $V_\varepsilon(x^j)$, $\varepsilon > 0$. So, $f(x)$ is reduced only on the region $V_\varepsilon(x^j)$ leaving all the other maxima unchanged. The mathematical formulation of the stretching function is given as follows.

$$\max_{l \leq x \leq u} f_j(x) \equiv \begin{cases} \bar{f}(x) - \frac{\delta_2 [\text{sign}(f(x^j) - f(x)) + 1]}{2 \tanh(\kappa(f(x^j) - f(x)))} & \text{if } x \in V_{\varepsilon^l}(x^j), l \in \{1, \dots, M\} \\ f(x) & \text{otherwise} \end{cases} \quad (1)$$

where $\bar{f}(x)$ is defined as

$$\bar{f}(x) = f(x) - \frac{\delta_1}{2} \|x - x^j\| [\text{sign}(f(x^j) - f(x)) + 1] \quad (2)$$

with δ_1 , δ_2 and κ positive constants and M is the number of solutions founded. The value of these parameters depend on the problem to solve. Transformations on the original objective function stretch the neighbourhood of x^j , with ray ε^j , downwards assigning smaller function values to those points to prevent the convergence of the global optimization method to that previously computed solution [Parsopoulos and Vrahatis, 2004, Pereira and Fernandes, 2009].

Each global optimization problem is solved using SA algorithm, see [Pereira and Fernandes, 2009] for details. The multilocal procedure terminates when for a predefined set of consecutive iterations no other solution is encountered [Pereira and Fernandes, 2008, Pereira and Fernandes, 2009].

The main steps of the SSA algorithm can be defined as follows.

Algorithm 1: (SSA algorithm)

Given: x^0 and the initial control parameter values for SA algorithm. Set $j = 1$.

While the stopping condition is not verified **do**

1. Find the solution of problem, x^j , using SA method.
2. Apply the stretching function in the point x^j .
3. Set $j = j + 1$

End while

End Algorithm

In the optimization of the humanoid robot gait problem the objective function $f : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}$ gives the distance, in meters, of the humanoid robot. The variable $x \in \mathbb{R}^{n \times n}$ represents the gait for the humanoid robot to walk.

4 Numerical Results

The numerical results were obtained using a Inter Core i7-2600 CPU 3.4 GHz with 8.0 GB of RAM.

For the optimization procedure it was considered $n = 10$ and the initial approximation is defined as

$$x_0 = \begin{bmatrix} -26 & 15 & 0 & -25 & 15 & 0 & 10 & 0 & 10 & 0 \\ -26 & 15 & -10 & -55 & 25 & -10 & 10 & 10 & 35 & 10 \\ -26 & 15 & -10 & -30 & -10 & -10 & 10 & 10 & 45 & 10 \\ -26 & 25 & 10 & -30 & 0 & 10 & 10 & -10 & 45 & -10 \\ -40 & 40 & 10 & -15 & 5 & 10 & 15 & -10 & 25 & -10 \\ -55 & 25 & 10 & -15 & 5 & 10 & 45 & -10 & 25 & -10 \\ -30 & -10 & 10 & -15 & 5 & 10 & 45 & -10 & 15 & -10 \\ -30 & -10 & 10 & -40 & 30 & 10 & 45 & -10 & 15 & -10 \\ -30 & -5 & -10 & -40 & 30 & -10 & 40 & 10 & 15 & 10 \\ -30 & 0 & -10 & -45 & 33 & -10 & 35 & 10 & 20 & 10 \\ -30 & 0 & -10 & -55 & 10 & -10 & 35 & 10 & 50 & 10 \end{bmatrix} \quad (3)$$

that was obtained through empirical analysis. The matrix (3) presents the initial gait for 10 joints (lines) and for 10 time instants (columns).

The upper and lower limits are defined using x_0 , and are defined as $l = x_0 - \alpha 1_{10}$ and $u = x_0 + \alpha 1_{10}$, where $1_{10} \in \mathbb{R}^{10 \times 10}$ is a matrix with coefficient one in all positions. The walking distance of the humanoid robot when considered the input variable x_0 is $f(x_0) = 0.089202$ m.

The parameters δ_1 , δ_2 , κ and α were fixed as 10^2 , 1, 10^{-3} and 2, respectively.

The SSA method was capable to identify only one solution for the proposed problem. The following figure presents optimization results. The solution was obtained after 2942 iterations, where it was identified the optimum value as 0.174153. This solution was obtained after 3030 seconds.

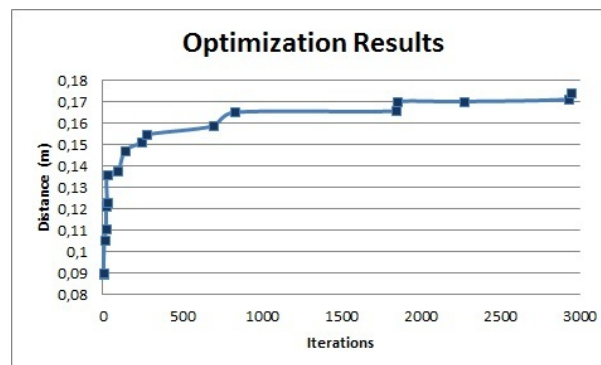


Figure 6: Optimization results

It is possible to observe that was possible to have 95% of improvement when it compared with the initial value.

5 Conclusions and future work

In this paper an optimization method based on the stretched simulated annealing method for the gait planning of humanoid robots was presented. For that purpose, tests were conducted on the personalized simulator SimTwo, a robot simulator that is capable to simulate user-defined robots in three-dimensional space with physical model based on rigid body dynamics (ODE). The new model of joints (motor inertia plus a spring damper gearbox model, *MIPSD*) was used since it allows to obtain more stable results. Using approximately 3000 iterations it was possible to find the optimum solution of the problem (maximize the walking distance), the stretched simulated annealing method was capable of increase 95%. These results validate the optimizations method even in a noisily system. As future work, the implementation of different optimization techniques could be tested and evaluated and the humanoid get up movements could be determined and optimized through this approach.

Acknowledgment

This work is financed by the ERDF - European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within projects FCOMP - 01 - 0124-FEDER-022701 and FCOMP-01-0124-FEDER-022674.

References

- [Browning and Tryzelaar, 2003] Browning, B., Tryzelaar, E. (2003) Ubersim: A realistic simulation engine for robotsoccer. *In Proceedings of Autonomous Agents and Multi-Agent Systems*, Australia.
- [Chalodhorn et al., 2007] Chalodhorn, R., Grimes, D., Grochow, K. and Rao, R. (2007) Learning to walk through imitation. *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, India.
- [Costa et al., 2011] Costa, P., Gonçalves, J., Lima, J. Malheiros, P. (2011) SimTwo Realistic Simulator: A tool for the development and validation of robot software. *Theory and Applications of Mathematics and Computer Science*, 1 (1): 17-33.
- [Faber and Behnke, 2007] Faber, F. and Behnke, S. (2007) Stochastic optimization of bipedal walking using gyro feedback and phase resetting. *Proc. of the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, Estados Unidos.
- [Hemker et al, 2007] Hemker, T., Stelzer, M., Stryk, O. and Sakamoto, H. (2009) Efficient walking speed optimization of a humanoid robot. *Int. Journal of Robotics Research*, 28 (2): 303-314.
- [Hu et al., 2006] Hu, L., Zhou, C., Sun, Z. (2006) Biped Gait Optimization Using Spline Function Based Probability Model. *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Estados Unidos.
- [Huang et al., 2008] Huang, Q., Yu, Z., Zhang, W., Duan, X., Huang, Y. and Li, K. (2008) Generation of humanoid walking pattern based on human walking measurement. *Proc. of the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, Coreia do Sul.
- [Lima et al., 2013] Lima, J., Gonçalves, J., Costa P., Moreira, A. (2013) Humanoid Gait Optimization Resorting to an Improved Simulation Model. *International Journal of Advanced Robotic Systems*,10 (67): 1-7.
- [Lima et al., 2010] Lima, J., Gonçalves, J., Costa, P., Moreira, A. (2010) Humanoid Robot Gait Planning Resorting to an Adaptive Simulated Annealing Algorithm. *Proceedings of 10th Conference on Autonomous Robot Systems and Competitions*, Portugal.
- [Lima et al., 2005] Lima, J., Gonçalves, J., Costa, P., Moreira, A. (2005) Humanoid Realistic Simulator: The servomotor joint modeling. *Proceedings of 6th International Conference on Informatics in Control, Automation and Robotics*, Italia.
- [Niehaus et al., 2007] Niehaus, C. and Röfer, T. and Laue, T. (2007) Gait optimization on a humanoid robot using particle swarm optimization. *Proceedings of the Second Workshop on Humanoid Soccer Robots*, Estados Unidos.

- [Parsopoulos and Vrahatis, 2004] Parsopoulos, K., Vrahatis, M. (2004) On the computation of all global minimizers through particle swarm optimization, *IEEE Transaction on Evolutionary Computation*, 8(3): 211–224.
- [Pereira and Fernandes, 2009] Pereira, A., Fernandes, E.M.G.P. (2009) A reduction method for semi-infinite programming by means of a global stochastic approach. *Optimization*, 58: 713–726.
- [Pereira and Fernandes, 2008] Pereira, A., Fernandes, E.M.G.P. (2008) On a reduction line search filter method for nonlinear semi-infinite programming problems. *Euro Mini Conference "Continuous Optimization and Knowledge-Based Technologies"*, L. Sakalauskas, G.W. Weber, E.K. Zavadskas (eds.), 174–179.
- [Sakka and Yokoi, 2005] Sakka, S., Yokoi, K. (2005) Humanoid Vertical Jumping based on Force Feedback and Inertial Forces Optimization. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Spain.
- [SimTwo, 2009] "SimTwo" (2009) <http://www.fe.up.pt/~paco/wiki>.
- [Smith, 2000] Smith, R. (2000) Open Dynamics Engine, <http://www.ode.org>.
- [Zhe et al, 2005] Zhe, T., Changjiu, Z. and Zengqitile, S. (2005) Humanoid walking gait optimization using GA-based neural network. *Advances in Natural Computation*, 3611: 252–261 .
- [Zhou et al., 2004] Zhou, C., Yue, P. K., Ni, J., Chan, S. (2004) Dynamically Stable Gait Planning for a Humanoid Robot to Climb Sloping Surface. *Proceedings of the 2004 IEEE Conference on Robotics, Automation and Mechatronics*, Singapore.