Multiobjective Optimization of a Quadruped Robot Locomotion using a Genetic Algorithm

Miguel Oliveira, Lino Costa, Ana Rocha, Cristina Santos and Manuel Ferreira

Abstract In this work, it is described a gait multiobjective optimization system that allows to obtain fast but stable robot quadruped crawl gaits. We combine bio-inspired Central Patterns Generators (CPGs) and Genetic Algorithms (GA). A motion architecture based on CPGs oscillators is used to model the locomotion of the robot dog and a GA is used to search parameterizations of the CPGs parameters which minimize the body vibration, maximize the velocity and maximize the wide stability margin. In this problem, there are several conflicting objectives that leads to a multiobjective formulation that is solved using the Weighted Tchebycheff scalarization method. Several experimental results show the effectiveness of this proposed approach.

1 Introduction

Robot locomotion is a challenging task that involves the control of a large number of degrees of freedom (DOF's). Several previous works, [9, 12] proposed biologic approaches to generate and modulate gait locomotion of quadruped robots, combining biometric sensory information with motion oscillators such as Central Pattern Generators (CPGs).

There are still many open questions in the quadruped locomotion, considering learning gaits or gait optimization. The problem of finding the best possible locomotion is a problem currently addressed in the literature. In [2] it is presented a Genetic Algorithm (GA) robust to the noise in the parameters evolution and that

Miguel Oliveira, Cristina Santos and Manuel Ferreira
Industrial Electronics Department, School of Engineering, University of Minho, 4800-058 Guimaraes, Portugal, e-mail: {mcampos,cristina,mjf}@dei.uminho.pt

Lino Costa, Ana Rocha
Department of Production and Systems , School of Engineering, University of Minho, 4800-058 Guimaraes, Portugal, e-mail: {lac, arocha}@dps.uminho.pt

also avoids premature local optima. The evaluation criterion is to maximize robot velocity. A comparison between several gait learning algorithms, including Genetic and Policy Gradient algorithms, is presented in [11]. The optimization goal is to determine the best 12 parameters of an elliptical locus scheme of locomotion, such that the robot takes less time to walk a certain distance.

In [6] it is presented an evolutionary algorithm based on a GA, in which genetic operators are chosen by an adaptation mechanism. Locomotion is implemented in real time and is evaluated by analysing the forward-backward, side-walk and rotation motion as well as the vibration. In [8] it is presented an evolutionary algorithm to optimize a vector of parameters for locomotion of an ERS110 robot. In [10] it is presented an optimization system for the locomotion of an ERS110 based on the Powell's method. It online optimizes 12 parameters of a locus locomotion scheme.

In this work, we propose a multiobjective approach to optimize a quadruped slow crawl gait, using Central Pattern Generators (CPGs) and a GA. CPGs are neural networks located in the spine of vertebrates, that generate coordinated rhythmic movements, namely locomotion [7]. In this work, a locomotion controller, based on CPGs, generates trajectories for hip robot joints [12]. Speed, vibration and stability are the evaluated criterions used to explore the parameter space of the network of CPGs to identify the best crawl pattern.

In order to achieve the desired crawl gait, it is necessary to appropriately tune these parameters by means of a optimization procedure. The resulting optimization problem has multiobjective nature since there are several conflicting objectives. This multiobjective problem was solved by a GA [5] applied to the corresponding Weighted Tchebycheff scalarized formulation [3]. Optimization is done online in a simulated ers-7 AIBO robot using Webots [13].

We have already addressed a slightly different but related problem in a preliminary experience using a genetic algorithm [16] and the electromagnetism-like algorithm [17]. In these works, we noticed that solving this problem requires a considerable computational effort. Notably because several constraints are imposed in this optimization problem. Thus, alternative techniques for handling constraints can make the search more efficient.

This article is structured as follows. In Sect. 2, we introduce several multiobjective optimization concepts. In Sect. 3 the optimization problem is formulated. Sect. 4 presents the optimization system details. Simulated results are presented in Sect. 5. The paper ends with a discussion and conclusions in Sect. 6.

2 Multiobjective Optimization

Mathematically, a multiobjective optimization problem with s objectives and n decision variables can be formulated as, without loss of generality:

$$\min \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_s(\mathbf{x}))$$
$$\text{subject to} \quad \mathbf{g}(\mathbf{x}) \geq 0 \text{ and } \mathbf{h}(\mathbf{x}) = 0$$

where $\mathbf{x} \in \mathbb{X}^n$, $\mathbf{g}(\mathbf{x})$ and $\mathbf{h}(\mathbf{x})$ are the inequality and equality constraints, respectively.

Solving multiobjective problems is a very difficult task due to, in general, for this class of problems, the objectives conflict across a high-dimensional problem space and the computational complexity of the problem (NP-hardness). Thus, the interaction between the multiple objectives gives rise to a set of efficient solutions, known as Pareto-optimal solutions.

For a multiobjective minimization problem, a solution $\mathbf{a}$ is said to dominate a solution $\mathbf{b}$, if and only if, $\forall i \in \{1, ..., s\} : f_i(\mathbf{a}) \leq f_i(\mathbf{b})$ and $\exists j \in \{1, ..., s\} : f_j(\mathbf{a}) < f_j(\mathbf{b})$. A solution $\mathbf{a}$ is said to be non-dominated regarding a set $\mathbb{Y}^n \subseteq \mathbb{X}^n$ if and only if, there is no solution in $\mathbb{Y}^n$ which dominates $\mathbf{a}$. The solution $\mathbf{a}$ is Pareto-optimal if and only if $\mathbf{a}$ is non-dominated regarding $\mathbb{X}^n$.

The main goal of a multiobjective algorithm is to find a good and balanced approximation to the Pareto-optimal set. Multiobjective problems can be addressed by scalarization methods such as the Weighted Tchebycheff function [3]:

$$\min f(\mathbf{x}) = \max\{W_i \sum_{i=1}^{s} |F_i - f_i(\mathbf{x})|\} \tag{1}$$

where $W_i \geq 0$ and $\sum_{i=1}^{s} W_i = 1$. The weighted distance is measured to an utopian objective vector $\mathbf{F}$ with components F_i. Different combinations of the weights W_i can produce different (weakly) Pareto optimal solutions. The problem defined in (1) is non differentiable and must be solved by a derivative free algorithm such as genetic algorithms [3].

3 Problem Formulation

The proposed network of CPGs generates trajectories for the robot limbs. Different combinations of these trajectories for each joint in terms of amplitude, offset and frequency, result in different gait patterns.

The proposed CPGs are based on Hopf oscillators and one intrinsic property is the possibility to smoothly modulate the generated trajectories according to explicit changes in the CPG parameters: amplitude, offset and the stance knee value. Therefore, in order to tune the CPG parameters, we use a GA to search for an optimal combination of these parameters. Speed, vibration and stability are the objectives to optimize in order to define different walking pattern solutions.

Robot trajectories are generated and modulated by the proposed network of CPGs, by explicitly changing the CPG parameters: amplitude (A), Offset(O), and the stance knee value (K), for each limb. Further, the parameter swing frequency (ω_{sw}) is common for the overall network. This means a total of 13 parameters we need to tune to modulate trajectories. However, left and right fore and left and right hind limb trajectories have the same amplitude, offset and frequency but a relative phase of π among them. These considerations enable us to reduce the number of CPG parameters required to optimize, as follows: amplitude of the fore and hind

limbs (A_{FL}, A_{HL}); fore and hind limbs knee stance angle (K_{FL}, K_{HL}); fore and hind limbs offset (O_{FL}, O_{HL}) and swing frequency (ω_{sw}). Thus, the problem has 7 decision variables corresponding to the 7 CPG free parameters.

The goal is the minimization of the body vibration and maximization the velocity and wide stability margin. We have used a scalarization method based on the minimization of the Weighted Tchebycheff function:

$$\min f(\mathbf{x}) = \max\{W_a|F_a - f_a(\mathbf{x})|, W_v|F_v - f_v(\mathbf{x})|, W_{WSM}|F_{WSM} - f_{WSM}(\mathbf{x})|\} \quad (2)$$

where W_a, W_v, W_{WSM} are weights satisfying $W_a, W_v, W_{WSM} \geq 0$ and $W_a + W_v + W_{WSM} = 1$, and $\mathbf{x} = (A_{FL}, A_{HL}, K_{FL}, K_{HL}, O_{FL}, O_{HL}, \omega_{sw})$ is the vector of decision variables.

The weighted distance is measured to a reference point, e.g., an utopian objective vector with components F_a, F_v and F_{WSM}. Different combinations of the weights W_a, W_v and W_{WSM} can produce different (weakly) Pareto optimal solutions that represent different locomotion compromises.

In (2), $f_a(\mathbf{x})$, $f_v(\mathbf{x})$ and $f_{WSM}(\mathbf{x})$ are, respectively, the robot body vibration, robot forward velocity and wide stability margin (WSM) computed for CPG parameterization given by $\mathbf{x}$. We consider that a good gait should have less vibration, because the robot is subjected to less strain.

In order to calculate the total vibration we sum the standard deviation of the measures of the (a_x, a_y, a_z) accelerometers built-in onto the robot, i.e., $std(a_x) + std(a_y) + std(a_z)$, similarly to [14, 6, 15].

We calculate forward velocity using the traveled distance of the robot during 12 seconds. A gait is considered better if it achieves higher velocities.

For stability, we calculate the wide stability margin [18]. This is a measure of the locomotion stability that provides the shortest distance between the projection of the center of mass in the ground and the polygon formed by the vertical projection in the ground of robot feet contact points. A gait is considered better for higher WSM values.

The search range of the CPG network parameters directly depend on the Aibo Ers-7 robot. The values of A_{FL} and A_{HL} are limited by the maximum range that the AIBO Hip joints may have. Offset values O_{FL} and O_{HL} for the hips are limited by the same ranges and the calculated amplitude values, A_{FL} and A_{HL}, respectively.

Maximum and minimum values for each knee stance angle are calculated in order to avoid leg collision during locomotion. Thus, the problem has several simple boundary constraints (for A_{FL}, A_{HL}, O_{FL}, O_{HL} and ω_{sw}), as follows

$$
\begin{aligned}
0.01 \leq A_{HL}, A_{FL} \leq 60 \\
-40 \leq O_{FL} \leq 20 \\
-20 \leq O_{HL} \leq 40 \\
1 \leq \omega_{sw} \leq 12
\end{aligned}
\quad (3)
$$

Moreover, several inequality constraints were imposed (for O_{FL}, O_{HL}, K_{FL} and K_{HL}), given by

$$-40 + \frac{A_{\text{FL}}}{2} \leq O_{\text{FL}} \leq 20 - \frac{A_{\text{FL}}}{2}$$

$$-20 + \frac{A_{\text{HL}}}{2} \leq O_{\text{HL}} \leq 40 - \frac{A_{\text{HL}}}{2}$$

$$K_{\text{FL}} \leq \max\{-O_{\text{FL}} - \frac{A_{\text{FL}}}{2} + 50, -O_{\text{FL}} + \frac{A_{\text{FL}}}{2} + 50\}$$

$$K_{\text{FL}} \geq \max\{-O_{\text{FL}} - \frac{A_{\text{FL}}}{2} + 20, -O_{\text{FL}} + \frac{A_{\text{FL}}}{2} + 20\} \quad (4)$$

$$K_{\text{HL}} \leq \max\{-O_{\text{HL}} - \frac{A_{\text{HL}}}{2} + 40, -O_{\text{HL}} + \frac{A_{\text{HL}}}{2} + 40\}$$

$$K_{\text{HL}} \geq \max\{-O_{\text{HL}} - \frac{A_{\text{HL}}}{2} - 5, -O_{\text{HL}} + \frac{A_{\text{HL}}}{2} - 5\}$$

4 Optimization System

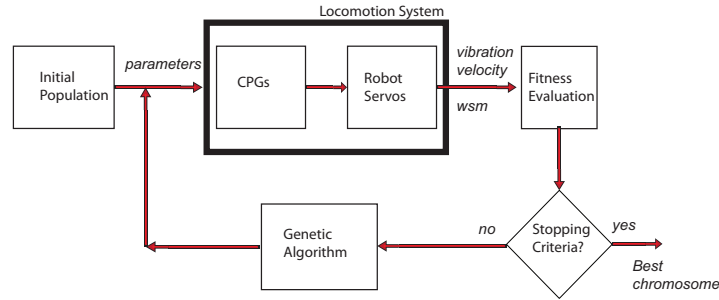A scheme of the optimization system is depicted in fig 1. In order to tune the CPG

Fig. 1 Optimization Locomotion System

parameters, we use a GA to search the optimal combination of the CPG parameters. GAs are population based algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover [5]. GAs work with a population of points that represent potential optimal solutions to the problem being solved, usually referred to as chromosomes.

In this work, real representation of the variables was considered. So, each vector consists of a vector of 7 real values representing the decision variables of the problem. In our optimization system, we begin the GA search by randomly generating an initial population of chromosomes. The chromosomes were evaluated according to the fitness function defined in (2), in terms of robot body vibration, robot forward velocity and stability.

In order to handle the simple boundary constraints, each new generated point is projected component by component in order to satisfy boundary constraints (for all

components of $\mathbf{x}$, see (3)) as follows:

$$x_i = \begin{cases} l_i & \text{if } x_i < l_i \\ x_i & \text{if } l_i \leq x_i \leq u_i \\ u_i & \text{if } x_i > u_i \end{cases} \qquad (5)$$

where l_i, u_i are the lower and upper limit of i component, respectively. A repairing mechanism is implemented to handle the inequality constraints. Therefore, any infeasible solution is repaired exploring the relations among variables expressed by the inequality constraints (see (4)), i.e., the values O_{FL}, O_{HL}, K_{FL} and K_{HL} are repaired in order to satisfy the constraints. The application of this repairing mechanism to all infeasible solutions in the population, guarantees that all solutions become feasible.

We implement a tournament selection that guarantees that better chromosomes are more likely to be selected. Although selection assures that in the next generation the best chromosomes will be present with a higher probability, it does not search the space, because it just copies the previous chromosomes. The search results from the creation of new chromosomes from old ones by the application of genetic operators.

The crossover operator takes two randomly selected chromosomes; one point along their common length is randomly selected, and the characters of the two parent strings are swapped, thus generating two new chromosomes.

The mutation operator, randomly selects a position in the chromosome and, with a given probability, changes the corresponding value. This operator introduces diversity in the population since selection and crossover, exclusively, could not assure the exploration of new regions in the search space.

In order to recombine and mutate chromosomes, the Simulated Binary Crossover (SBX) and Polynomial Mutation were considered, respectively. These operators simulate the working of the traditional binary operators [4].

5 Simulation Results

In this section, we describe the experiment done in a simulated ers-7 AIBO robot using Webots [13]. Webots is a software for the physic simulation of robots based on ODE, an open source physics engine for simulating 3D rigid body dynamics.

The ers-7 AIBO dog robot is a 18 DOFs quadruped robot made by Sony. The locomotion controller generates trajectories for the hip and knee joint angles, that is 8 DOFs of the robot, 2 DOFs in each leg.

At each sensorial cycle (30 ms), sensory information is acquired. For each chromosome, the evaluation time for locomotion was 12s. We apply the Euler method with 1ms fixed integration step, to integrate the system of equations. At the end of each chromosome evaluation the robot is set to its initial position and rotation, such that initial conditions are equal for the evaluation of all chromosomes of all populations.

In all experiments, the optimization system ends when the number of generations exceeds 50 generations. We depict results when a population was established with 30 chromosomes. The SBX crossover and polynomial mutation probabilities were, respectively, 0.9 and 1/7. An elite size of 3 chromosomes was implemented.

Table 5 presents the results obtained for several combinations of weights in the scalar function. As reference point considered has the following components: $F_a = 0$, $F_v = -150$ and $F_{WSM} = -65$. Combinations of weights defining multiobjective problems with different number of objectives were considered:

- $T1$ to $T3$ correspond to the optimization of each one of the objectives;
- $T3$ to $T15$ are different combinations of weights of two objectives being simultaneously optimized;
- finally, $T16$ to $T19$ are combinations of weights in which the three objectives are simultaneously optimized.

In this table, the solutions obtained for each weights combinations are also presented, in terms of f_a, f_v and f_{WSM}. In the last column, the value of the scalar function is also presented ($f(\mathbf{x})$).

Table 1 Multiobjective results for different combinations of weights.

	W_a	W_v	W_{WSM}	f_a	f_v	f_{WSM}	$f(\mathbf{x})$
$T1$	1	0	0	0.0135	21.0471	14.6093	0.0136
$T2$	0	1	0	0.1213	138.5492	6.0067	11.4508
$T3$	0	0	1	0.0384	4.9457	64.6248	0.3752
$T4$	0.2	0.8	0	0.1058	125.1315	6.1785	19.8948
$T5$	0.4	0.6	0	0.1215	127.7335	7.2684	7.3599
$T6$	0.6	0.4	0	0.1181	134.967	6.3661	4.0242
$T7$	0.8	0.2	0	0.1118	129.3525	4.5230	4.1295
$T8$	0	0.8	0.2	0.1210	135.7304	7.6250	11.4750
$T9$	0	0.6	0.4	0.1125	114.7752	11.0080	21.5968
$T10$	0	0.4	0.6	0.1292	89.3642	19.1294	27.5223
$T11$	0	0.2	0.8	0.0307	17.9153	63.8483	26.4169
$T12$	0.8	0	0.2	0.0206	29.8379	54.8389	2.0322
$T13$	0.6	0	0.4	0.0144	3.3254	50.2399	5.9040
$T14$	0.4	0	0.6	0.0340	2.9505	63.7906	0.7256
$T15$	0.2	0	0.8	0.0249	0.5827	63.7955	0.9635
$T16$	1/3	1/3	1/3	0.1013	103.5880	11.3905	17.8680
$T17$	1/2	1/4	1/4	0.0698	95.7004	11.5376	27.1498
$T18$	1/4	1/2	1/4	0.1081	121.8231	9.1177	27.1498
$T19$	1/4	1/4	1/2	0.1259	66.8218	19.2712	22.8644

These solutions are also depicted in Figures 2a)-c). As expected, it can be observed that solutions $T1$, $T2$ and $T3$ are the extreme solutions of the Pareto front in Figures 2a), 2b) and 2c). The other solutions of the front represent different compromises of the objectives. Taking into account the information provided by these graphs, it is possible to choose a compromise solution that represent a different locomotion gait of the robot. Moreover, it possible to inspect the relationships between objectives.
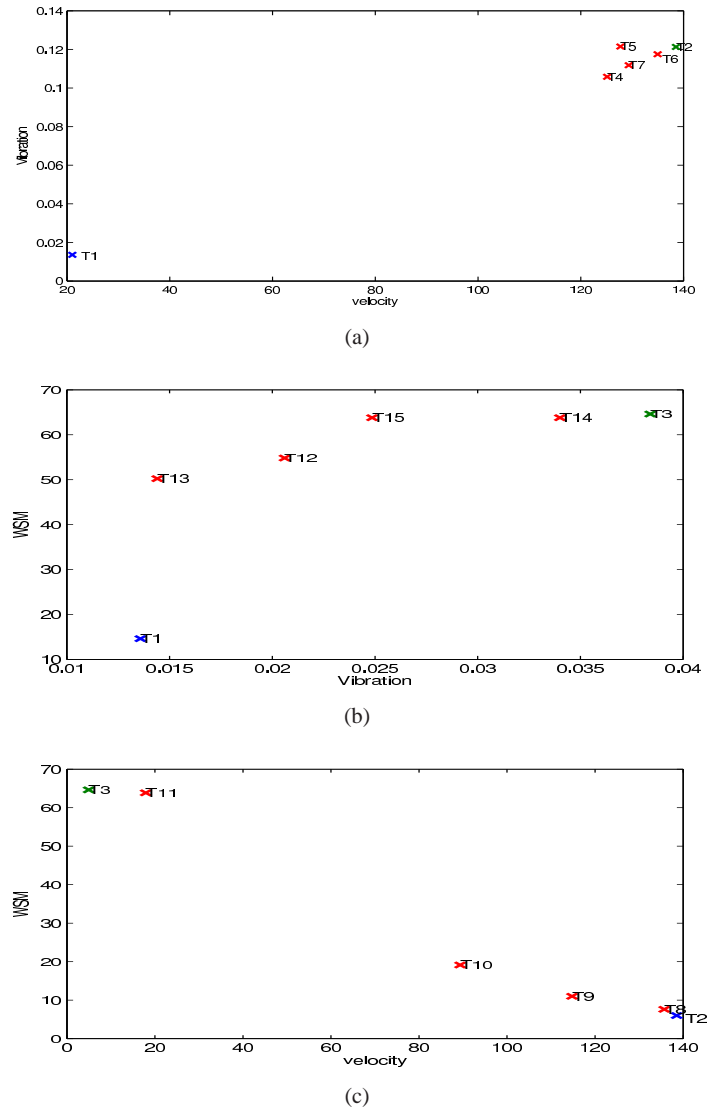
Fig. 2 Representation of the solutions in objective space. a) velocity versus vibration ($T1$, $T2$ and $T4$ to $T7$ combinations of weights). b) vibration versus WSM ($T1$, $T3$ and $T12$ to $T15$ combinations of weights). c) velocity versus WSM ($T2$, $T3$ and $T8$ to $T11$ combinations of weights).

In Figure 3, the 2D projections of 3D objective space is presented. Since the 3 objectives are conflicting, these solutions define a 3D Pareto surface. This multi-objective approach allows to select solutions that achieve the highest velocity for a slow, crawl gait and perceive the tradeoff in terms of vibration and WSM.
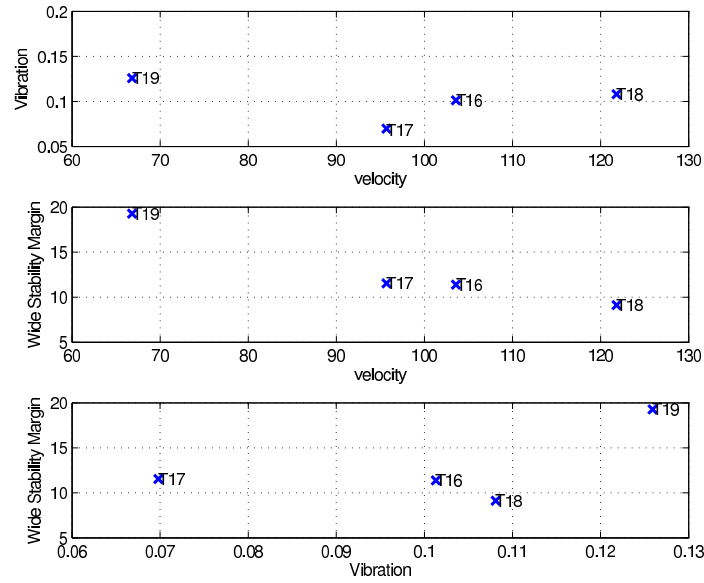
Fig. 3 Representation of the solutions in objective space: velocity, vibration and WSM ($T16$ to $T19$ combinations of weights).

6 Conclusions and Future Work

In this article, we have addressed the locomotion optimization of a quadruped robot. A locomotion controller based on dynamical systems to model CPGs, generates quadruped locomotion. These CPG parameters are tuned by an optimization system. This optimization system combines CPGs and a genetic algorithm which solves a multiobjective formulation of the problem. The goal is to optimize simultaneously, three conflicting objectives, namely the vibration, velocity and WSM.

Experiments were performed in the Webots robotics simulator. The multiobjective optimization was formulated considering a scalarization function based on the Weighted Tchebycheff method for different combinations of weights. The solutions obtained represent locomotion strategies that are different compromises of the objectives.

We also plan to use multi-objective optimization algorithms such as MEES [1] or NSGAII [3]. We will extend this optimization work to address other locomotion related problems, such as: the generation and switch among different gaits according to the sensorial information and the control of locomotion direction.

7 Aknowledgments

References

1. L. Costa, and P. Oliveira, An Adaptive Sharing Elitist Evolution Strategy for Multiobjective Optimization. *Evolutionary Computation*, **11**(4), 417–438, (2003).
2. Sonia Chernova and Manuela Veloso, An evolutionary approach to gait learning for four-legged robots,*In Proceedings of IROS04*, (September 2004).
3. K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley and Sons, Ltd, Chichester, (2001).
4. R.B. Deb, K. Agrawal, Simulated binary crossover for continuous search space., *Complex Systems*, 9(2), 115-149, (1995).
5. D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.
6. Dragos Golubovic and Huosheng Hu, Evolving locomotion gaits for quadruped walking robots, *Industrial Robot: An International Journal*, 32, 259-267, (2005).
7. S. Grillner, Neurobiological bases of rhythmic motor acts in vertebrates, *Science*, 228(4696), 143-149, (1985).
8. Gregory S. Hornby, Seichi Takamura, Takashi Yamamoto, and Masahiro Fujita, Autonomous evolution of dynamic gaits with two quadruped robots, *IEEE Transactions on Robotics*, 21, 402-410, (2005).
9. Ig Mo Koo, Tae Hun Kang, Gia Loc Vo, Tran Duc Trong, Young Kuk Song and Hyouk Ryeol Choi, Biologically inspired control of quadruped walking robot, *International Journal of Control, Automation and Systems*, 7(4), 577-584, (2009).
10. Min Sub Kim and William Uther, Automatic gait optimisation for quadruped robots, *In Australasian Conference on Robotics and Automation*, (2003).
11. Nate Kohl and Peter Stone, Machine learning for fast quadrupedal locomotion, *in The Nineteenth National Conference on Artificial Intelligence*, pp. 611-616, (2004).
12. Vítor Matos, Cristina P. Santos, and Carla M. A. Pinto, A brainstemlike modulation approach for gait transition in a quadruped robot, *in IROS*, pp. 2665-2670, (2009).
13. O. Michel, Webots: Professional mobile robot simulation, *Journal of Advanced Robotics Systems*, 1(1), 39-42, (2004).
14. Thomas Röfer, Evolutionary gait-optimization using a fitness function based on proprioception,*in RobuCup*, pp. 310-322, (2004).
15. Manish Saggar, Nate Kohl, and Peter Stone, Autonomous learning of stable quadruped locomotion, *in RoboCup2006: Robot Soccer World Cup X*. Springer Verlag, (2007).
16. C. Santos, M. Oliveira, A.M.A.C. Rocha and L. Costa, Head Motion Stabilization During Quadruped Robot Locomotion: Combining Dynamical Systems and a Genetic Algorithm, *IEEE International Conference on Robotics and Automation (ICRA 2009)*, May 12-17, Kobe, Japan 2009.
17. C.P. Santos, M. Oliveira, V. Matos, A.M. A.C. Rocha and L. Costa, Combining Central Pattern Generators with the Electromagnetism-like Algorithm for Head Motion Stabilization during Quadruped Robot Locomotion, *2nd International Workshop on Evolutionary and Reinforcement Learning for Autonomous Robot Systems (ERLARS 2009)*, St. Louis, Missouri, USA, October 15, 2009.
18. S. Song and K.Waldron, Machines that Walk: The Adaptive Suspension Vehicle, *MIT Press*, Cambridge, 1989.