Evolution Strategies Combined with Central Pattern Generators for Head Motion Minimization during Quadruped Robot Locomotion

Cristina P. Santos, Miguel Oliveira, Hermínia Mendes, Manuel Ferreira and Lino Costa

Abstract— **In autonomous robotics, the head shaking induced by locomotion is a relevant and still not solved problem. This problem constraints stable image acquisition and the possibility to rely on that information to act accordingly.**

In this article, we propose a movement controller to generate locomotion and head movement. Our aim is to generate the head movement required to minimize the head motion induced by locomotion itself. The movement controllers are biologically inspired in the concept of Central Pattern Generators (CPGs). CPGs are modelled based on nonlinear dynamical systems, coupled Hopf oscillators. This approach allows to explicitly specify parameters such as amplitude, offset and frequency of movement and to smoothly modulate the generated oscillations according to changes in these parameters. Based on these ideas, we propose a combined approach to generate head movement stabilization on a quadruped robot, using CPGs and an evolution strategy. The best set of parameters that generates the head movement are computed by an evolution strategy.

Experiments were performed on a simulated AIBO robot. The obtained results demonstrate the feasibility of the approach, by reducing the overall head movement.

I. INTRODUCTION

Visually-guided locomotion is important for autonomous robotics. However, there are several difficulties, for instance, the head shaking that results from the robot locomotion itself that constraints stable image acquisition and the possibility to rely on that information to act accordingly. The motion of quadruped, biped and snake-like robots, for instance, with cameras mounted in their heads, causes head shaking.This kind of disturbances, generated by locomotion itself, makes it difficult to keep the visual frame stable and, therefore, to act according to the visual information. Head stabilization is very important for achieving a visually-guided locomotion, a concept which has been suggested from a considerable number of neuroscientific findings in humans and animals [16].

In this article, we aim to build a system able to minimize the head motion of a quadruped robot that walks with a walking gait. We propose a motion stabilization system for the head of an ers-7 AIBO quadruped robot. Basically, head motion is set such to generate the movement opposed to the one induced by the locomotion itself.

Several similar works have been proposed in literature [4], [7], [6], [5]. But these methods consider that the robot moves according to a scheduled robot motion plan, which imply that space and time constraints on robot motion must be known before hand as well as robot and environment models. As such, control is based on this scheduled plan. Other works have successfully achieved gaze stabilization [5], that consists on image stabilization during head movements in space. The overall of the gaze stabilization approaches can be divided into two types of techniques. One uses specific hardware, like accelerometers and gyroscope to estimate the 3D posture of the head, and complex control algorithms to compensate the oscillations. The use of inertial information was already proposed by several authors [5], [14], [15]. Typically this kind of techniques is used in binocular robot heads, where gaze is implemented through the coordination of the two eye movements. Most of the approaches are inspired in biological systems, specifically in the human Vestibular-Ocular Reflex (VOR). In robots with fixed eyes, the fixation point procedure is achieved by compensatory head or body movements, based on multisensory information of the head.

In this work, we propose a combined approach to generate head movement stabilization on a quadruped robot, using Central Pattern Generators (CPGs) and Evolution Strategies (ESs) [17], [18]. We intend to use a head controller, based on Central Pattern Generators (CPGs), that generates trajectories for tilt, pan and nod head joints. CPGs are neural networks located in the spine of vertebrates, able to generate coordinated rhythmic movements, namely locomotion [11]. These CPGs are modelled as coupled oscillators and solved using numeric integration. These CPGs have been applied in drumming [1] and postural control [3]. This dynamical systems approach model for CPGs presents multiple interesting properties, including: low computation cost which is well-suited for real time; robustness against small perturbations; the smooth online modulation of trajectories through changes in the dynamical systems parameters and phase-locking between the different oscillators for different DOFs.

In order to achieve the desired head movement, opposed to the one induced by locomotion, it is necessary to appropriately tune the CPG parameters. This can be achieved by optimizing the CPG parameters using an optimization method. The optimization process is done offline according to the head movement induced by the locomotion when no stabilization procedure was performed.

Cristina Santos, Miguel Oliveira, Herminia Mendes and Manuel Ferreira are with Industrial Electronics Department, School of Engineering, University of Minho, 4800-058 Guimaraes, Portugal `cristina@dei.uminho.pt`, `mcampos@dei.uminho.pt`, `hmendes@dei.uminho.pt`, `mjf@dei.uminho.pt`

Lino Costa is with Production Systems Department, School of Engineering, University of Minho, 4710-057 Braga, Portugal `lac@dps.uminho.pt`

Some algorithms for solving this type of problem require substantial gradient information and aim to improve the solution in a neighborhood of a given initial approximation. When the problem has more than one local solution, the convergence to the global solution may depend on the provided initial approximation. Thus, searching for a global optimum is a difficult task that could be done by using stochastic-type algorithms. The stochastic methods can be classified in two main categories, namely, the point-to-point search strategies and the population-based search techniques like evolutionary algorithms. Several evolutionary approaches have been applied to global optimization problems with success, namely Evolution Strategies (ES) [17], [18], Genetic Algorithms [2] and Particle Swarm Optimization [12]. Moreover, in the past, ESs proved to be powerful global optimizers [19] which are easy to implement and computationally inexpensive in terms of memory requirement. The GA is well suited and has already been applied to solve this optimization problem because it can handle both discrete and continuous variables, nonlinear objective and constrain functions without requiring gradient information [13]. Since, in general, ESs are the most efficient in terms of function evaluations [19], they are suitable to apply in the optimization of the CPG parameters of amplitude, offset and frequency of each head oscillator to head motion stabilization during quadruped robot locomotion.

The remainder of this paper is organized as follows. In Section II, the system architecture and how to generate locomotion and head movement is described. The main ideas concerning the optimization system, namely the problem statement that evaluates the head movement, the ES algorithm used to optimize the CPG parameters and some experimental results, are described in Section III. Simulated results are described in Section IV. Conclusions are made in Section V-A.

II. SYSTEM ARCHITECTURE

Our aim is to propose a control architecture that is able to generate both locomotion for a quadruped robot and to generate head motion such as to minimize the head movement induced by the the locomotion itself.

The overall system architecture is depicted in Figure 1.

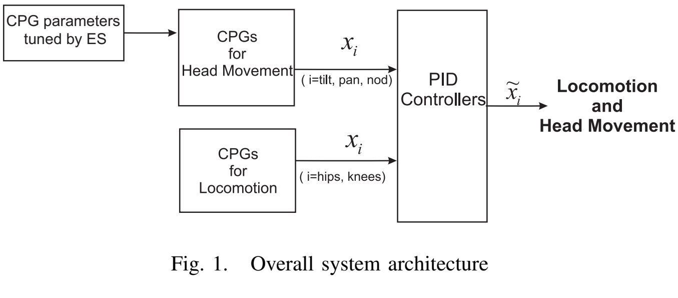

Fig. 1. Overall system architecture

The proposed movement controllers are biologically inspired in the concept of CPGs, modeled by dynamical equations. Hip and knee trajectories are generated by a locomotion controller. A head controller generates trajectories for the neck tilt, pan and nod joint values. These trajectories are used as input for the PID controllers of these joints.

The head controller parameters have to be tuned such that the resultant movement is as desired. Using our CPG approach allows us to assign explicit parameters for each of the nonlinear oscillators, independently controlling the amplitude, offset and frequency of the movement. We apply a stochastic optimization method, the ES algorithm, in order to determine the best set of CPG control parameters that results in, or close to the desired movement.

A. Locomotion Generation

In this section we present the network of CPGs used to generate locomotion. A CPG for a given degree-of-freedom (DOF) is modelled as coupled Hopf oscillators, that generate a rhythmic movement.

1) Rhythmic Movement Generation: The rhythmic locomotor movements for a robot joint are generated by the x variable of the following Hopf oscillator

$$\begin{aligned} \dot{x} &= \alpha\left(\mu - r^2\right)(x - O) - \omega z, \\ \dot{z} &= \alpha\left(\mu - r^2\right) z + \omega (x - O), \end{aligned} \quad (1)$$

where $r = \sqrt{\left((x-O)\right)^2 + z^2}$, peak-to-peak amplitude of the oscillations is given by $A = \sqrt{\mu}$ for $\mu > 0$, ω specifies the oscillations frequency (in rad s^{-1}) and relaxation to the limit cycle is given by $\frac{1}{2\alpha\mu}$.

This oscillator contains an Hopf bifurcation from a stable fixed point at $x = O$ (when $\mu < 0$) to a structurally stable, harmonic limit cycle, for $\mu > 0$. The fixed point x has an offset given by O.

It generates smooth trajectories due to the stable solutions of the dynamical solutions, despite small changes in the parameters. It exhibits limit cycle behaviour and describes a stable rhythmic motion where parameters μ, ω and O control the desired amplitude, frequency and offset of the resultant oscillations.

The generated x solution of this nonlinear oscillator is used as the control trajectory for the hip swing and knee joints of the robot limbs. These trajectories encode the values of the joint's angles and are sent online for the lower level PID controllers of each hip swing joint.

2) Locomotion Controller Architecture: Fig. 2 depicts the network structure used to generate locomotion for a quadruped robot. Interlimb coordination is achieved by bilaterally coupling the dynamics of the four hip swing oscillators (illustrated by right-left arrows in Fig. 2). Intralimb coordination is achieved by unilaterally coupling each hip swing oscillator to the corresponding knee oscillator. These couplings ensure that the limbs stay synchronized. For the hip joints, identified by subscript [1], this is achieved by modifying (1) as follows:

$$\begin{aligned} \begin{bmatrix} \dot{x}_{\mathrm{i}[1]} \\ \dot{z}_{\mathrm{i}[1]} \end{bmatrix} &= \begin{bmatrix} \alpha\left(\mu - r_{i[1]}^2\right) & -\omega \\ \omega & \alpha\left(\mu - r_{i[1]}^2\right) \end{bmatrix} \begin{bmatrix} x_{\mathrm{i}[1]} - O_{\mathrm{i}[1]} \\ z_{\mathrm{i}[1]} \end{bmatrix} \\ &+ \sum_{j \neq i} \mathbf{R}(\theta_{\mathrm{i}[1]}^{\mathrm{j}[1]}) \begin{bmatrix} x_{\mathrm{j}[1]} - O_{\mathrm{j}[1]} \\ z_{\mathrm{j}[1]} \end{bmatrix} \end{aligned}$$

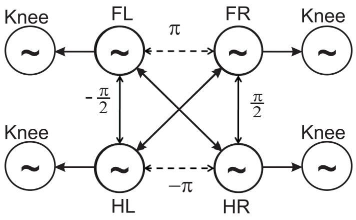

Fig. 2. Locomotion controller architecture depicting coupling structure among the CPGs for a walking gait. The footfall sequence is: HL-FL-HR-FR, with each foot lagging a quarter of a cycle from the previous.

For the knee joints, identified by subscript [3], we modify (1) as follows:

$$\begin{bmatrix} \dot{x}_{\mathrm{i}[3]} \\ \dot{z}_{\mathrm{i}[3]} \end{bmatrix} = \begin{bmatrix} \alpha\left(\mu - r_{i[3]}^2\right) & -\omega \\ \omega & \alpha\left(\mu - r_{i[3]}^2\right) \end{bmatrix} \begin{bmatrix} x_{\mathrm{i}[3]} - O_{\mathrm{i}[3]} \\ z_{\mathrm{i}[3]} \end{bmatrix} + \frac{1}{2}\mathbf{R}(\psi_{\mathrm{i}[3]}^{\mathrm{j}[1]}) \begin{bmatrix} x_{\mathrm{j}[1]} - O_{\mathrm{j}[1]} \\ z_{\mathrm{j}[1]} \end{bmatrix}$$

where i, j = Fore Left (FL), Fore Right (FR), Hind Left (HL) and Hind Right (HR) limbs. The linear terms are rotated onto each other by the rotation matrices $\mathbf{R}(\theta_{\mathrm{i}[1]}^{\mathrm{j}[1]})$ and $\mathbf{R}(\psi_{\mathrm{i}[3]}^{\mathrm{j}[1]})$, where $\theta_{\mathrm{i}[1]}^{\mathrm{j}[1]}$ is the relative phase among the i[1]'s and j[1]'s hip oscillators and represents bidirectional couplings between these oscillators such that $\theta_{\mathrm{i}[1]}^{\mathrm{j}[1]} = -\theta_{\mathrm{j}[1]}^{\mathrm{i}[1]}$ and $\psi_{\mathrm{i}[3]}^{\mathrm{j}[1]}$ is the required relative phase among the i[3]'s and j[1]'s oscillators (see Fig. 2). We assure that closed-loop interoscillator couplings have phase biases that sum to a multiple of 2 π.

Each hip oscillator lags a quarter of a cycle from the previous. The relative phases between hips and knees, $\psi_{\mathrm{i}[3]}^{\mathrm{j}[1]}$, were all set to 180.

The final result is a network of oscillators with controlled phase relationships, able to generate more complex, synchronized behavior such as locomotion. It generates coordinated rhythmic movements in a stable and flexible way. The generated trajectories are smooth, stable and robust to perturbations.

Due to the properties of this type of coupling among oscillators, the generated trajectories are stable and smooth and thus potentially useful for trajectory generation in a robot.

3) Generating a walking gait: A gait event sequence is specified using the duty factors and the relative phases, where the first event, and the start of the stride, is chosen as the event when the fore left leg (reference leg) is set down. Parameters were chosen in order to respect feasibility of the experiment. We set the frequency to $\omega = 2.044\ rads^{-1}$ in regards with the motor limitations. Speed of convergence of rhythmic systems were set to 0.08s ($\frac{1}{2\alpha\mu_{\mathrm{i}}}$), in regard to stability during the integration process and to feasibility of the desired trajectories. The μ_{i} parameters of the front and hind limbs were set to 6.25 and 25, respectively. This yield a non-singular, regular and symmetric gait with a FL-HR-FR-HL gait even sequence, a duty factor of 0.73 and a velocity of 19mms^{-1} (measured in the Z direction, see Fig. 3).

B. Head Movement Generation

Head movement is generated similarly to locomotion, but a CPG for a given DOF is modelled as an Hopf oscillator, not coupled to any other oscillator. Each CPG, therefore, generates a rhythmic movement according to

$$\begin{bmatrix} \dot{x}_{\mathrm{i}} \\ \dot{z}_{\mathrm{i}} \end{bmatrix} = \begin{bmatrix} \alpha\left(\mu_{\mathrm{i}} - r_i^2\right) & -\omega_{\mathrm{i}} \\ \omega_{\mathrm{i}} & \alpha\left(\mu_{\mathrm{i}} - r_i^2\right) \end{bmatrix} \begin{bmatrix} x_{\mathrm{i}} - O_{\mathrm{i}} \\ z_{\mathrm{i}} \end{bmatrix}, \quad (2)$$

where i =tilt,pan,nod.

The control policy is the x_{i} variable, obtained by integrating the CPGs dynamical systems, and represents tilt, pan and nod joint angles in our experiments.

Note that the final movement for each of these joints is a rhythmic motion which amplitude of movement is specified by μ_{i}, offset by O_{i} and its frequency by ω_{i}.

The differential equations for locomotion and head movement are solved using Euler integration with a fixed time step of 1ms. The x_{i} trajectories represent angular positions and are directly sent to the PID controllers of the joint servomotors.

III. OPTIMIZATION SYSTEM

In this section, we explain how the head CPGs are optimized in order to reduce the camera (head) movement induced by locomotion itself. We will optimize the distance between the generated head movement for a set of head CPG control parameters and the one induced by locomotion.

In order to implement the head motion it is necessary one or several optimal combinations of amplitude, offset and frequency of each head oscillator. This is possible because we can easily modulate amplitude, offset and frequency of the generated trajectories according to changes in the μ_{i}, O_{i} and ω_{i} CPG parameters and these are represented in an explicit way by our CPG. Therefore, we have to tune these head CPG parameters. In order to optimize the combinations of the different head CPG control parameters the ES algorithm is used.

The multitude of parameter combinations is large, and it is difficult to derive an accurate model for the tested quadruped robot and for the environment. Besides, such a model based approach would also require some post-adaptation of results (because of backlash, friction, etc).

In this study, the search of parameters suitable for the implementation of the required head motion was carried out based on the data from a simulated quadruped robot. The (X,Y,Z) head coordinates, in a world coordinate system (Fig. 3), are recorded when a simulated robot walks during 30s and no head stabilization is performed. We are interested in the opposite of this movement around the (X,Y,Z) coordinates. This data was mathematically treated such as to keep only the oscillations in the movement and remove the drift that the robot has in the X coordinate and also the forward movement in the Z coordinate. From now on, this data is referred to as $(X,Y,Z)_{\mathrm{observed}}$.

In the simulation, we have set a cycle time of 30ms, that is, the time needed to perform sensory acquisitions, calculate the planned trajectories (integrating the differential equations) and send this data to the servomotors. The $(X,Y,Z)_{\mathrm{observed}}$

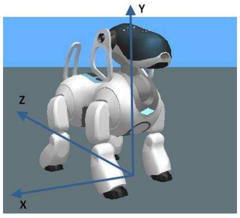

Fig. 3. World coordinate system.

data is sampled with a sample time of 30ms, meaning we have a total of 1000 samples. A simulated time of 30s corresponds to 10 strides of locomotion. This time is arbitrary and could have been chosen differently but seems well suited to find a model representative of the head movement induced by the locomotion controller.

The basic idea is to combine the CPG model for head movement generation with the optimization algorithm. Fig. 4 illustrates a schematics of the overall optimization system.

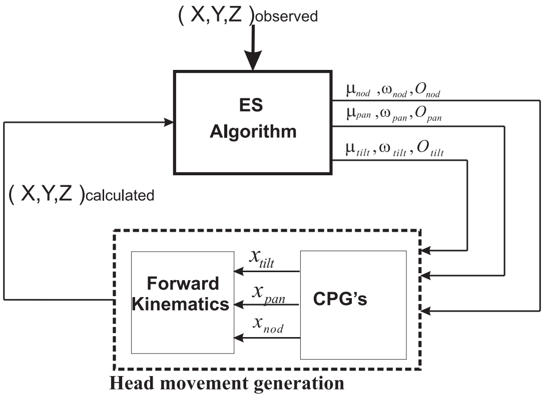

Fig. 4. Schematics of the optimization system.

Three head CPGs (2) generate during 30s rhythmic motions for the tilt, pan and nod joints. By applying forward kinematics, we calculate the resultant set of 1000 samples of $(X,Y,Z)_{\text{calculated}}$ head coordinates in the world coordinate system.

A. Problem Definition

The decision variables of the problem are the amplitude, offset and frequency of each tilt, pan and nod oscillators, that are necessary to generate the desired head movement. The search ranges of the head CPG control parameters were set beforehand as shown in Table I for the purpose of efficient learning and according to the limits of the tilt, pan and nod DOFs. Search for optimal parameters is carried out by performing the overall optimization system over a preset number of generations.

The combinations of amplitude, offset and frequency of each tilt, pan and nod oscillators, that are necessary to generate the desired head movement, form each individual of the population of ES. Thus, each individual consists on a vector of 9 CPG parameters, identified on the first column of Table I.

TABLE I

SEARCH RANGES OF CPG PARAMETERS

Parameter	Range	Unit
μ_{tilt}	$[0, 1406.3]$	
ω_{tilt}	$[1, 12]$	$(\text{rad}s^{-1})$
O_{tilt}	$\left[-75+\sqrt{\mu_{\text{tilt}}}, 0-\sqrt{\mu_{\text{tilt}}}\right]$	$(^{o})$
μ_{pan}	$[0, (1936+1936)]$	
ω_{pan}	$[1, 12]$	$(\text{rad}s^{-1})$
O_{pan}	$\left[-88+\sqrt{\mu_{\text{pan}}}, 88-\sqrt{\mu_{\text{pan}}}\right]$	$(^{o})$
μ_{nod}	$[0, (506.25+56.25)]$	
ω_{nod}	$[1, 12]$	$(\text{rad}s^{-1})$
O_{nod}	$\left[-15+\sqrt{\mu_{\text{nod}}}, 45-\sqrt{\mu_{\text{nod}}}\right]$	$(^{o})$

In order to compute the fitness function value of an individual, the CPG parameters will be the input of the head movement generation process (see Fig. 4) and by applying forward kinematics the resultant $(X,Y,Z)_{\text{calculated}}$ head coordinates are computed. The sum of the distances between each sample of the observed and calculated head coordinates is used as fitness function in order to evaluate the resulting head movement. Thus, the fitness of the ith individual is given by

$$f^{\text{i}} = \sum_{j=1}^{1000} \sqrt{\left(X_{\text{j}}-X'_{\text{j}}\right)^2+\left(Y_{\text{j}}-Y'_{\text{j}}\right)^2+\left(Z_{\text{j}}-Z'_{\text{j}}\right)^2} \quad (3)$$

where j is an head position sample (because the points are generated and acquired in a discrete manner); (X',Y',Z') represent the calculated head coordinates with the CPG parameters and (X,Y,Z) represent the offline observed head coordinates. Only head position errors are computed in the fitness function, because we only control three DOFs and as such cannot control head orientation.

B. Evolution Strategies

ESs work directly with the real representation of the parameter set, searching from an initial population (a set of individuals), requiring only data based on the objective function and constraints, and not derivatives or other auxiliary knowledge. The initial population of individuals is uniformly generated at random within the feasible region, i.e., between the corresponding upper and lower bounds. Note that in order to guarantee the feasibility of the initial individuals and all individuals generated during the search a repair mechanism was implemented. Thus, an infeasible solution is repaired exploring the relations among variables expressed by the box constraints. Then the fitness function value (3) for all the individuals is computed as described in previous section.

Traditionally, two distinct types of ESs differing basically on the selection procedure can be considered: the $(\mu/\rho+\lambda)$-ES and the $(\mu/\rho,\lambda)$-ES. In this nomenclature, μ and λ represent, respectively, the parent and offspring population sizes (for many problems, $\lambda/\mu \approx 7$ is suggested [18]); ρ represents the number of parents that are selected for recombination. In this work, we have adopted a $(\mu/\rho+\lambda)$-ES that uses an elitist selection scheme.

In this algorithm, each population member consists on a tuple of two vectors: a vector of real values representing the decision variables (the CPG parameters) and a vector of real standard deviations used to adapt step sizes during the search. Step sizes for mutation are themselves optimized during the search. Therefore, each decision variable i has an associated standard deviation σ_i. The search starts from an initial population which individuals are, in general, generated at random. The initial standard deviations σ_i were set according to equation:

$$\sigma_i^{(0)} = \frac{U_i - L_i}{\lambda\sqrt{n}} \qquad (4)$$

where U_i and L_i are, respectively, the upper and lower bounds of the decision variables (defined in Table I) and n is the dimension of the problem.

So, in a $(\mu/\rho+\lambda)$-ES, at a given generation, there are μ parents, and λ offspring are generated by recombination and mutation. Basically, the recombination operator consists on, before mutation, to recombine a set of chosen parents to find a new solution. In this work, we implemented a discrete recombination. So, a given number ρ $(1 \leq \rho \leq \mu)$ of parents are randomly chosen for recombination and each component of the offspring is chosen from one of the ρ parents at random. This procedure allows different combinations of the values of the decision variables from existing solutions in the population. Standard deviations are similarly recombined.

Next, the step sizes for mutation are adapted. Several self-adaptation schemes are possible. One possibility is to actualize the standard deviations σ_i (for each decision variable) according to the equation [18]:

$$\sigma_i^{(k+1)} = \sigma_i^{(k)} e^{z_i} e^{z} \qquad (5)$$

where $z_i \sim N(0,\Delta\sigma^2)$, $z \sim N(0,\Delta\sigma'^2)$ and $\Delta\sigma$ and $\Delta\sigma'$ are parameters of the algorithm. In the experiments conducted only this non-isotropic adaptation rule was considered.

On other hand, mutation creates new individuals by adding random normal distributed quantities with mean zero and variance σ_i^2 $(z_i^{(k)} \sim N(0,\sigma_i^2))$ to the vector of decision variables. Each generated individual should be evaluated in terms of fitness function value and projected into the feasible region, according to the range presented in Table I in order to maintain feasibility.

Next, the $\mu+\lambda$ individuals are sorted according to their fitness values. Finally, the best μ of all the $\mu+\lambda$ members become the parents of the next generation (i.e., the selection takes place between the $\mu+\lambda$ members). The process is repeated until a given stopping criteria is fulfilled.

C. Experimental Results

The optimization system was implemented in Matlab (Version 7.5) running in an Intel Pentium CPU 3.20 GHz (1024 MB of RAM) PC. The system of equations was integrated using the Euler method with 1ms fixed integration steps. The evaluation time for head movement generation is 30s.

In our implementation, the optimization system ends when the number of generations exceeds 150, or the population variability is inferior to a threshold set *a priori*. In this study,the number of parents was 10 and the number of offspring was 100, *i.e.* $(10+100)$-ES was used. When stochastic methods are used to solve problems, the impact of the random number seeds has to be taken into consideration and each optimization process should be run a certain number of times. In this experience we set it to 10.

Table II contains the Best, Mean and standard deviation (SD) values of the solutions found (in terms of fitness function and time) over the 10 runs.

TABLE II

PERFORMANCE OF ES ALGORITHM IN THE OPTIMIZATION SYSTEM

Fitness (mm)			Time (hours)		
Best	Mean	SD	Best	Mean	SD
3776.2	4183.1	572.6463	1.9837	2.4212	0.3327

Differences in these fitness values arise because we use the sum of 1000 samples of distances between and calculated head coordinates. So, small differences in the CPG parameters may lead to large differences in the fitness values. Fig. 5 shows the evolution of all the 10 runs (lighter lines), best (solid line) and mean (dashed line) fitness function value over 90 generations. The best individual has a fitness value of 3776.2 that was achieved at generation 90. The best run took 1h58min (CPU time) and each iteration took in average 79.3481 seconds.

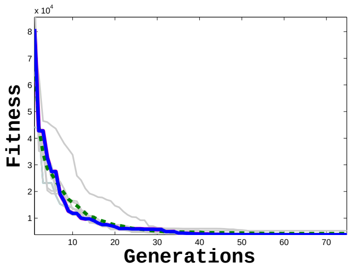

Fig. 5. 10 runs (lighter lines), best (solid) and mean (dashed) fitness evolution.

Table IV shows the tuned CPG parameters representing the best individual found. Table IV shows the tuned CPG

TABLE III

BEST POINT CPG PARAMETERS

Parameter	Value	Unit
μ_{tilt}	2.0456	
O_{tilt}	-0.065	$(^o)$
w_{tilt}	4.11	$(\text{rad}s^{-1})$
μ_{pan}	15.544	
O_{pan}	0.0918	$(^o)$
w_{pan}	2.12	$(\text{rad}s^{-1})$
μ_{nod}	0.1702	
O_{nod}	-1.92	$(^o)$
w_{nod}	3.98	$(\text{rad}s^{-1})$

parameters representing the best individual found.

TABLE IV

BEST POINT CPG PARAMETERS

Parameter	μ_{tilt}	O_{tilt}	w_{tilt}	μ_{pan}	O_{pan}	w_{pan}	μ_{nod}	O_{nod}	w_{nod}
Value	2.0456	-0.065	4.11	15.544	0.0918	2.12	0.1702	-1.92	3.98
Unit		$(^o)$	$(\text{rad}s^{-1})$		$O_{\text{pan}}(^o)$	$(\text{rad}s^{-1})$		$(^o)$	$(\text{rad}s^{-1})$

Fig. **??** depicts the distance between observed and calculated values of the head movement for the best individual of the initial population (dotted line) and the best individual after 90 generations (solid line). We can observe that peak-to-peak amplitude in distance decreases along the search.

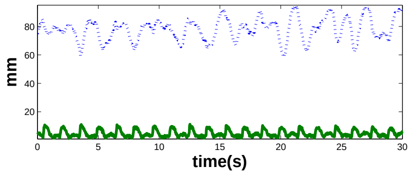

Fig. 6. Distance between observed and calculated values of the head movement for the best individual of the initial population (dotted line) and the best individual after 90 generations (solid line) of the optimization system.

Quantitatively, head movement is reduced about 29.3% for the best solution.

Fig. 7 depicts the time courses of the (X,Y,Z) calculated (solid line) head movement according to the head CPG control parameters of the best solution found. The observed (dotted line) head movement is also illustrated. Table V gives the maximal movement variation in the (X,Y,Z) coordinates for the calculated and observed movements. We conclude that the generated movements are quite similar in the X coordinate. The calculated movement is quite different in the Y and Z coordinate. This results from the fact that only the pan joint controls movement in the X coordinate, while both the tilt and nod joints control the Y and Z coordinates.

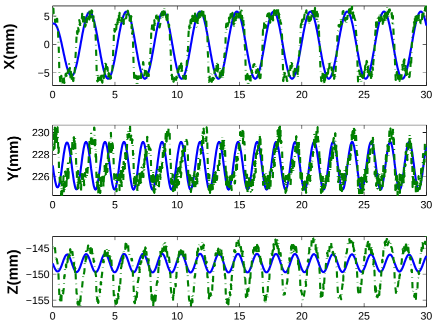

Fig. 7. (X,Y,Z) calculated (solid line) and observed (dotted line) head movement, during 30s, according to the head CPG control parameters from the best individual found.

TABLE V

MAXIMAL MOVEMENT VARIATION IN (X,Y,Z)

	Max ΔX (mm)	Max ΔY (mm)	Max ΔZ (mm)
Calculated Movement	11.802	4.3	3.6
Observed Movement	13.42	5.9	11.3

IV. SIMULATION RESULTS

In this section, we describe the experiment done in a simulated ers-7 AIBO robot using Webots [8]. 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 the joint angles of the hip and knee joints in the sagittal plane, that is 8 DOFs of the robot, 2 DOFs in each leg. Only walk gait is generated and tested.

The head controller generates the joint angles of the 3 DOFs: tilt, pan and nod. The other DOFs are not used for the moment, and remain fixed to an appropriately chosen value during the experiments.

The AIBO has a camera built into its head.

At each sensorial cycle (30ms), sensory information is acquired. The dynamics of the CPGs are numerically integrated using the Euler method with a fixed time step of 1ms thus specifying servo positions. Locomotion parameters were set as previously described. Head movement parameters were set to the tuned values by the ES algorithm (Table IV).

Because we are working in a simulated environment, we are able to build a GPS into the AIBO camera, that enable us to verify how the head effectively moves in an external coordinate system. Two simulations are performed: the robot walks during 30s with and without the feedforward solution and its GPS coordinates are recorded. Results are compared for these two simulations. Fig. 8 shows the GPS coordinates for the experiments with (solid line) and without the feedforward solution (dotted line). The overall experiment can be seen in the attached video.

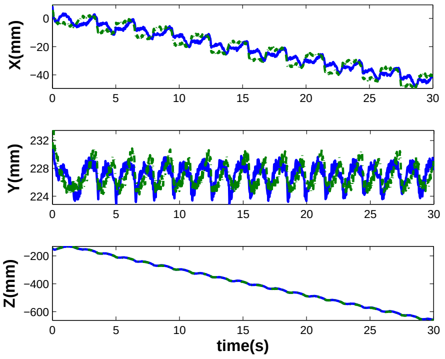

Fig. 8. (X,Y,Z) coordinates of the GPS positioned in the AIBO head when the robot walks during 30s. Solid and dotted lines indicate the experiment in which the feedforward solution is and is not implemented, respectively.

We expect that the proposed feedforward solution minimizes the variation of the GPS coordinates, meaning that the head remains near the same position during the experiment. We observe that the X coordinates of the marker position oscillate less. Note that there is some drift in the X coordinates, meaning the robot slightly deviates towards its side while walking. The observed peaks in the Y coordinate reflect the final stage of the swing phase and the begin of the stance phases of the fore legs, corresponding to an accentuated movement of the robot center of mass. This problem will be addressed in current work, by improving the locomotion controller and take into account balance control [3].

V. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In this article, we have proposed movement controllers to achieve head motion minimization of a quadruped robot that walks with a walking gait. Locomotion induces head shaking which must be minimized. Quadruped locomotion is generated by a network of CPGs modeled by dynamical systems. Head movement is generated by CPGs built-in in the tilt, pan and nod joints. These CPG parameters are tuned by an optimization system. This optimization system combines CPGs and the ES algorithm. As a result, set of parameters obtained by the ES allows to reduce the head movement induced by the locomotion.

This article extends previous work on combining dynamical systems and a Genetic Algorithm (GA) for head motion stabilization during quadruped robot locomotion [13]. Comparing results in terms of head motion, these are quite similar for the ES and the GA. However, ES outperforms GA in terms of CPU time.

B. Future Works

Currently, we are using other optimization methods, like the particle swarm optimization, and testing other fitness functions. 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. We further plan to extend our current work to online learning of the head movement similarly to [9].

Sensitivity analysis of parameters is currently under investigation.

REFERENCES

[1] S Degallier, C P. Santos, L Righetti and A Ijspeert,*Movement Generation using Dynamical Systems: a Drumming Humanoid Robot.* In Humanoids06 IEEE-RAS International Conference on Humanoids Robots, Genova, Italy, December 4-6 2006

[2] Goldberg, D.,*Genetic Algorithms in Search*, Optimization,and Machine Learning, Addison-Wesley, (1989).

[3] Castro, Luiz, Santos, C P; Oliveira, M and Ijspeert, A, *Postural Control on a Quadruped Robot Using Lateral Titl: a Dynamical System Approach*, European Robotics Symposium EUROS 2008, Prague, 2008.

[4] Takizawa, S, Ushida, S, Okatani, T, Deguchi, K.,*2DOF Motion Stabilization of Biped Robot by Gaze Control Strategy.* 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005). Aug. 2005:3809-3814

[5] Ravi Kaushik, Marek Marcinkiewicz, Jizhong Xiao, Simon Parsons, and Theodore Raphan*Implementation of Bio-Inspired Vestibulo-Ocular Reflex in a Quadrupedal Robot* 2007 IEEE International Conference on Robotics and Automation (ICRA 2007), Roma, Italy, 10-14 April 2007: 4861-4866

[6] Yamada, H.; Mori, M.; Hirose, S.,*Stabilization of the head of an undulating snake-like robot*, it 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007), San Diego, CA, USA, Oct 29 - Nov 2, 2007 : 3566-3571

[7] Yurak Son, Takuya Kamano, Takashi Yasuno, Takayuki Suzuki,and Hironobu Harada,*Generation of Adaptive Gait Patterns for Quadruped Robot with CPG Network Including Motor Dynamic Model*, Electrical Engineering in Japan, Vol. 155, No. 1, 2006

[8] O. Michel, “Webots: Professional mobile robot simulation,” *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 39–42, 2004.

[9] A. Sproewitz, R. Moeckel, J. Maye, M. Asadpour, and A.J. Ijspeert. *Adaptive locomotion control in modular robotics.* In Workshop on Self-Reconfigurable Robots/Systems and Applications IROS07, pages 81-84, November 2007.

[10] Ludovic Righetti and Auke Jan Ijspeert.*Pattern generators with sensory feedback for the control of quadruped locomotion.* IEEE International Conference on Robotics and Automation ICRA 2008, California, 2008.

[11] S. Grillner. *Neurobiological bases of rhythmic motor acts in vertebrates.* Science, Vol. 228, No. 4696, pp. 143-149, 1985

[12] J. Kennedy and R.C. Eberhart, *Particle swarm optimization*, in *IEEE International Conference on Neural Network*, 1995, pp. 1942–1948.

[13] Cristina Santos, Miguel Oliveira, Ana Maria A.C. Rocha and Lino 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.

[14] F. Patane, C. Laschi, H. Miwa, E. Guglielmelli, P. Dario, and A. Takanishi, *Design and development of a biologically-inspired artificial vestibular system for robot heads*, in IEEE Conference on Intelligent Robots and Systems, IROS'04, Sendai, Japan, September 28 - Ocotber 2, 1317- 1322, 2004.

[15] G. Asuni, G. Teti, C. Laschi, E. Guglielmelli, and P. Dario, *A robotic head neuro-controller based on biologically-inspired neural methods*, in IEEE Conference on Robotics and Automation, ICRA'05, Barcelona, Spain, April 18-22, 2005.

[16] R. Cromwell, J. Schurter, Scott Shelton, S. Vora, *Head stabilization strategies in the sagittal plane during locomotor tasks*, Physiotherapy Research International, Whurr Publishers Ltd, 9(1), pp.3342, 2004

[17] Rechenberg, I., *Evolutionsstrategie '94*, Frommann-Holzboog, Stuttgart, 1994.

[18] Schwefel, H.,-P, *Evolution and Optimum Seeking*, Wiley, New York, 1995.

[19] Costa, L. and Oliveira, P., Evolutionary algorithms approach to the solution of mixed integer non-linear programming problems, *Computers Chem. Engng.*, **25**, 2001, pp. 257–266.