# AN EVOLVED NEURAL NETWORK FOR FAST QUADRUPEDAL LOCOMOTION

IRENE MARKELIC

*Bernstein Center for Computational Neuroscience, Bunsenstrasse 10*
*Goettingen,,37073, Germany*

KEYAN ZAHEDI

*Fraunhofer Institute for Autonomous Intelligent Systems (AIS), Schloss Sankt Augustin*
*Sankt Augustin, 53757,Germany*

This paper presents a modular neural network controller for fast locomotion of a quadruped robot. It was generated by artificial evolution techniques using a physical simulation of the Sony Aibo ERS-7. Co-evolution was used to develop neuromodules controlling the single legs as well as the coordination between the four legs. The final neurocontroller utilizes a central pattern generator and does not make use of available sensory inputs. In experiments with the physical robot a top walking speed of 47.34 cm/s was measured, where lateral leg movement contributed considerably to the achieved high velocity.

## 1. Introduction

The control of legged locomotion in robots is still a challenging problem, especially for fast locomotion. Apart from some exceptions like Raibert's robots [1] most legged machines move rather slowly [2]. It has been found that locomotion in many organisms is mostly driven by central pattern generators (CPGs) [3]. These are neural networks producing a rhythmic pattern without the need of sensory feedback [4]. For walking machines a CPG can be realized by an oscillatory artificial neural network. Such oscillators were used for instance by Billard and Ijispeert to realize a continuous passage between a walking, scratching and lying down behavior of an Aibo robot [5]. Kimura used neural oscillators to realize dynamic walking and running in a quadruped [6]. To find appropriate controllers for a hexapod, Beer used genetic algorithms and a simulation of the robot [7]. Finally he transferred the generated controllers to a physical machine [8].

Cruse et al. developed neural networks by means of a simulation and implemented them in a hexapod [9]. In this paper a recurrent neural network for fast quadrupedal walking of a Sony Aibo ERS-7 robot is presented. It was derived by artificial evolution techniques and is for use in the international competition of robot soccer. Since velocity is a key feature, many teams are concerned with fast locomotion. Published velocities of Aibo walking using the ERS-7 model are between 39.7 - 43.0 cm/s [10,11,12,13]. The paper is organized as follows: In section 2 (The Experimental Setup) the neuron model is introduced, and the tools employed for evolution and composition of the network are explained. In the third section (Results) the resulting controller is presented and analyzed. In the final section (Discussion) the results are discussed and related to other works.

## 2. The Experimental Setup

All experiments were conducted using the time discrete dynamics of networks with standard additive neurons, using the hyperbolic tangent as transfer function ; i.e.,

$$a_i(t+1) = \Theta_i + \sum_{j=1}^{N} \omega_{ij}\sigma(a_i(t)), \quad i=1,..N \tag{1}$$

where $a_i$ denotes the activation, $\Theta i$ the bias term of the $i$th neuron, $t$ denotes a discrete time step, $\omega_{ij}$ refers to the weight of synapse rom neuron $j$ to $i$, and $N$ is the total number of neurons in a given network. Input neurons were treated as buffers for sensor signals. For evolution the program package ISEE (Integrated Structure Evolution Environment ) was used, which was designed to realize Artificial Evolution experiments. It implements the $ENS^3$ algorithm [14] and offers the physical simulator YARS based on ODE (Open Dynamics Engine). ISEE permits to influence the evolutionary process and the network by various parameters that can be set during runtime, for example, to foster small networks by assigning cost terms to neurons and synapses. It also supports co-evolution, i.e. the simultaneous evolution of several populations. Furthermore, the generation of network structure as well as parameter optimization is possible.

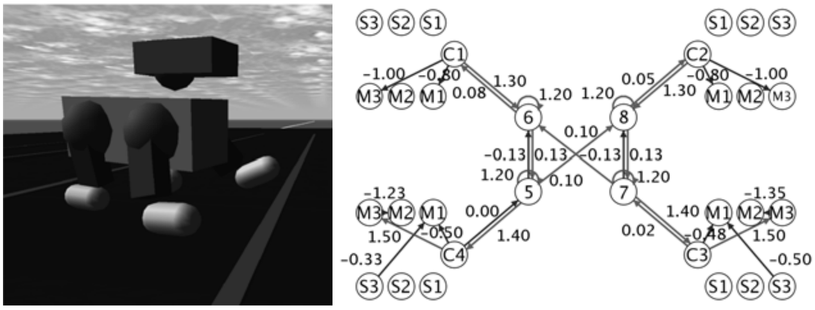


Figure 1.Left: The simulated robot. Right: The neurocontroller generating the fastest walking behaviour. Circles denote neurons and are connected via synapses. Neurons C1 to C4 are connector neurons which form the junction between leg modules and coordination module.

A dog-like robot, the Aibo ERS-7 was used as physical platform for the evolved controller. Each of its four legs has three degrees of freedom realized by the three motors, M1, M2 and M3. The first motor, M1, moves the leg back and forth, whereas M2 moves it sideways. M3 moves only the lower limb of the leg back and forth. For each of these motors a sensor (S1, S2, and S3), is detecting its deflection angle.

The physical simulation of the robot, as shown in Fig. 1 left, was realized with YARS. As relevant features to be modelled for a walking behavior were considered: body parts (head, trunk and limbs) defined by their dimensions and weights as well as the 12 leg motors which were each defined by a deflection angle, strength, and velocity, and finally the according 12 deflection angle sensors. Since hinge torque and velocity of the motors used for the ERS-7 are not published they had to be measured experimentally. For maximal force 0.43 Ncm were used, and 286 °/s for maximal angular velocity. To instantiate the controller in the robot it was rewritten as C++ program and incorporated into the freely available German Team Software which provides access to motor control and sensory information.

Based on the hypothesis that control of insect walking can be considered hierarchical and modular [16], and assuming that both fore legs and both hind legs each carry out the same step cycle, a modular approach, and thus co-evolution, was chosen as an appropriate method. It was decided to use one network per leg,

where the networks for both fore- and both hind legs were identical. All leg modules were then connected to a coordination module. A minimal leg controller contained seven neurons, one input neuron per sensor and one output neuron per motor. The seventh neuron was a hidden neuron by which the connection to the coordination module was realized and which is referred to as connector neuron. The resulting network formed the neurocontroller, comp. Fig. 1, right.
When a reasonable walking behavior was observed during simulation the corresponding controller was transferred to the hardware, and the speed of the resulting gait measured. Therefore a test course was set up with two labels, indicating one meter and the robot had to cover this distance from a flying start and with fully charged batteries while being filmed. Each neurocontroller was tested 10 times, and its speed was set to be the average value of these runs.
Various experiments were conducted, and the best results were achieved when using a CPG as coordination module. The employed CPG, depicted in Fig. 2A, was adopted from [17]**.** Its output are four sinusoidal signals with equal phase shift (see Fig. 2, bottom left), of which each leg module received one in the order: right hind leg, right foreleg, left hind leg and left foreleg. This, together with the corresponding phase shift, generated a reasonable walking gait. An evolution run was started by providing initial network structures for the leg modules in order to avoid the bootstrap-problem. They caused a pendulum like movement of the legs which was triggered by sensory input. Parameter settings of the evolutionary $ENS^3$ program enabled an alteration of structure, bias and synaptic weights of these networks. Structure and synaptic weights of the CPG were chosen to be fixed, because already small changes often had undesirable effects for the output signals, e.g. the loss of periodicity. The fitness function $F$ was given by

$$F = \sum_{t=0}^{T} x(t)^2 - c(t) * \alpha \tag{2}$$

where $t$ denotes discrete time, and $x$ the position along the x-axis of the simulated robot in world coordinates, which was equal to the covered forward distance. It was squared to increase selection pressure. To punish individuals with high frequency oscillating legs, a problem that frequently occurred, the change of direction in leg movement was counted, denoted by $c$. It was then multiplied by a factor, $\alpha$, which was adjustable during runtime. In addition, low obstacles were built into the simulated environment which could not be overcome by individuals with fast oscillating leg movements, so that they could not gain high fitness values. After 4320 generations a neurocontroller leading to a speed of 39.56 cm/s was generated.

## 3. Results

The resulting network structure is shown in Fig.2, right. It is observable that no further hidden neurons, apart from the connector neuron, were part of the leg networks and that the only driving force of the controller was the CPG. Although sensory input from 12 sensors was available during evolution, only one sensor in the hind leg modules was effciently used. It was found that this sensor input was important for the initial behavior of the simulated robot. It modulated the trajectory of the signals that controlled M1 of both hind legs in such a way that the simulated robot could get in a position from which it could start walking. Because the physical robot was not started from such a position ( it was put on the floor when moving) it was not affected by this mechanism. Thus, a controller for walking without any incorporation of sensory feedback had been evolved.

Signals were fed back into the CPG coming from the connector neurons. Although the according synapse weights were small, between [-0.08, 0.08] they had impact on the CPG's output signals, as can be observed when comparing Fig. 2, bottom middle and bottom right. Changes in amplitude (C1, C4), period (C1, C2, C3, C4), phase shift (C4, C3) and signal shape (C3) are observable.

With this controller the robot did not walk in a straight line, which was due to the different signals the motors received. Therefore synaptic weights were carefully adjusted so that all legs received a similar signal (see Fig. 2 top, right).

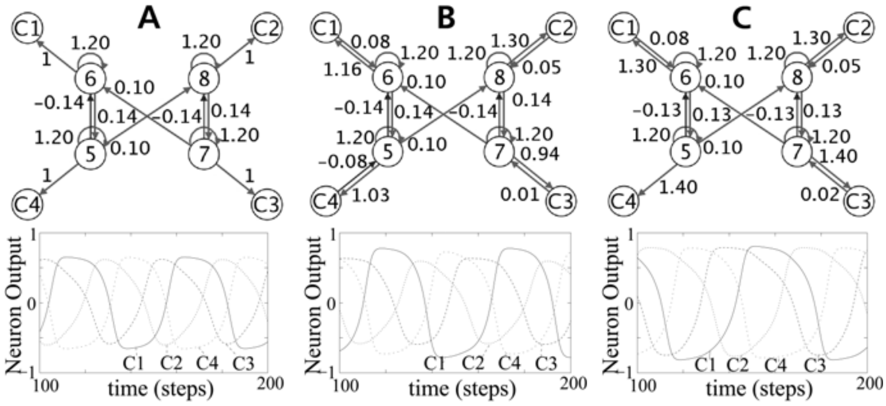


Figure 2. Top row: The CPG with connector neuron of each leg (C1-C4). The originally used CPG, left, the CPG with evolutionary altered parameters, middle, and the manually adjusted CPG, right. The corresponding bias values are listed in table 1. Bottom row: The plot below each CPG shows the output of the four connector neurons. Each curve has is labelled with the corresponding neuron.

Furthermore parameters were altered to find out which stride frequency and length were most appropriate. This was done manually, but could have been done using evolution. As was already indicated by the longer period time of CPG signals after evolution, the best results were attained with a lower CPG signal frequency of 1.8Hz. The initial frequency in comparison was 2.2Hz. In Fig. 3 the signal of M1 of the right foreleg is plotted along with the according sensor signal. It is observable that the motor lags behind the given signal and never reaches the target amplitude. Thus, with a larger period the motor can follow the given signal better, which finally results in a larger step length. In contrast, a higher frequency automatically led to smaller steps and due to the inertia of the motors to a lower speed of the walking behavior.

Leg posture was determined by the bias values of the according output neurons. Changing these values enabled a straighter posture of the legs which caused larger steps. The final neurocontroller (see Fig 2, right) generated a walking behavior with a speed of 45.95 cm/s, with a top speed of 47.34 cm/s. Videos of the behaviour are available under www.fraunhofer.de/~zahedi/aibo.html. The obtained gait is a walk with duty factor $\beta = 0.58 \pm 0.2$ for all legs. Where duty factor is the ratio between the duration that a leg has ground contact and the duration when it is lifted off the ground during one stride. One major difference to other Aibo walking behaviors was the incorporation of M2 in the hind leg modules, which resulted in a sideways movement of these legs. It caused the robot to rock from one side to the other while walking (compare video). Lesion experiments showed that this resulted in a gain of speed of 11.8%.

Table 1. A comparison of bias values of the different CPGs. The enumeration of neurons is according Fig. 2, nn = neuron number, bo = bias of original CPG, ba = bias of evolutionary altered CPG, and bop = bias of optimized CPG.

| nn | bo | ba | bop | nn | bo | ba | bop |
|---|---|---|---|---|---|---|---|
| **C1** | 0 | -0.003 | 0 | 5 | 0.01 | 0.017 | 0.0175 |
| **C2** | 0 | 0.0016 | 0 | 6 | 0 | 0.0053 | 0.0053 |
| **C3** | 0 | -0.002 | 0 | 7 | 0 | 0.0037 | 0.0037 |
| **C4** | 0 | 0.0065 | 0 | 8 | 0 | -0.0044 | -0.0044 |

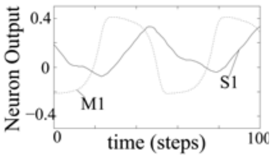

Fig. 3. Output of M1 and S1. S1 shows the actual deflection of the motor. Whereas M1 can be interpreted as a setpoint signal.

## 4. Discussion

Modular neurocontrollers were evolved to control a physical Aibo-ERS-7 robot. Using a physical simulator and additional adjustments, the final controller was able to generate a fast straight walking behavior for this robot of about 47,3 cm/s. It was found that a sideway movement of hind legs resulted in a higher walking velocity. This can be explained by the shift of the robot's center of gravity from side to side, caused by the lateral hind leg movement. When one bodyhalf was tilted towards the ground, carrying most of the weight, the other side was moved upwards, leaving more time and force for legs on this side to be moved.Thus, it is concluded that a pace gait in addition to the horizontal body tilt can lead to a higher walking speed for this particular robot, than a trot or a walk  Experiments by Beer suggest that sensory input is always incorporated when available [8]. That this does not happen in this case may result from the fact that sensory input was simply not needed for the task at hand, because of the driving CPG. Evolution took place in an environment that indeed contained obstacles, but obstacle height was relatively low. Once it was provided that legs were lifted up high enough there was no reason for a more sophisticated behavior. That no sensory input was fed into the network supports an hypothesis by Full and Koditschek [19] that rapid locomotion relies rather on feedforward control than on continuous sensorimotor feedback. This is thought to be due to the little time available to process proprioceptive input [19]. For future work, it is desirable to steer the walking direction of the robot. For this, visual sensory input could be used as a tropism (e.g. using the robot's camera to follow a colored ball).

## References


1. M.H. Raibert & I.E. Sutherland, Sci. Am., **248,** 44 (1983).
2. R.M. Alexander, *Phys. Rev.*, **69,** 1199 (1989).
3. S. Grillner, *Curr. Opin. Neurobiol*, **1(4)**, 231, (1991).
4. F. Delcomyn, *Science*, **210(4469)**, 492 (1980).
5. A. Billard & A.J. Ijspeert, *Proc. IJCNN*, **VI**, 637 (2000)
6. H. Kimura et al, *Aut. Rob.*, **7(3)**, 247 (1999).
7. R.D. Beer & J.C. Gallagher, *Adapt. Behav.,* **1***,* 91 (1992)
8. J.C. Gallagher et al., *Robot. Auton. Syst.*, **19**, 95 (1996).
9. H. Cruse et al., *Adap. Behav.*, **3(4)**, 385 (1995).
10. A. Shinohara & A. Ishino, www.jollypochie.org/papers/JollyPochie2004TR.pdf, (accessed 2006).
11. M. Quinlan *et al*., www.robots.newcastle.edu.au/publicationns/NUbotFinalReport2005.pdf,(accessed 2006).
12. D. Cohen et al., *www.cis.upenn.edu/robocup/UPenn04.pdf*, (accessed 2006).
13. T. Röfer, 8th International Workshop on RoboCup, (2004).
14. F. Pasemann *et al*., *Theory Biosc.,* **120**, 311 (2001).
15. M. Fujita & H. Kitano, *Aut. Rob.,* **5**, 7 (1998).
16. F. Delcomyn, *Aut. Rob.*, **7**, 259 (1999).
17. B. Klaassen *et al*., *VDI-Berichte*, **1841**, 633 (2004).
18. F. Pasemann et al., Proc. IWANN, **2686**,144 (2003).
19. R. Koditscheck, *J. Exp. Biol*, **22**, 3325 (1999).