Evolving robust gaits with AIBO

Conference Paper *in* Proceedings - IEEE International Conference on Robotics and Automation · February 2000

DOI: 10.1109/ROBOT.2000.846489 · Source: IEEE Xplore

Evolving Robust Gaits with AIBO

G. S. Hornby[1] S. Takamura[2] J. Yokono[3] O. Hanagata[3] T. Yamamoto[3] M. Fujita[3]

[1]Computer Science Dept.
Brandeis University
Waltham, MA 02454

[2]ER Business Incubation Dept
Sony Corporation
Tokyo, JAPAN 141-0001

[3]Digital Creatures Lab
Sony Corporation
Tokyo, JAPAN 141-0001

Abstract

An evolutionary algorithm is used to evolve gaits with the Sony entertainment robot, AIBO. All processing is handled by the robot's on-board computer with individuals evaluated using the robot's hardware. By sculpting the experimental environment, we increase robustness to different surface types and different AIBOs. Evolved gaits are faster than those created by hand. Using this technique we evolve a gait used in the consumer version of AIBO.

1 Introduction

Entertainment Robots must be fun to play with and interesting to watch. With AIBO, locomotion gaits are one of its most visible attributes. Previously [1], an evolutionary algorithm (EA) was implemented on AIBO for the automatic acquisition of gait parameters for dynamic gaits. One finding was that the evolved gaits tended to be fragile. They performed well on the robot and carpet with which they were evolved, but on a different surface type and also on different AIBOs they would often perform poorly. This paper presents work towards the automatic acquisition of robust gaits.

Three reasons for studying automatic acquisition of gaits are for re-developing gaits for new robots, transferring gaits from one version of a robot to a new version, and for finding gaits that would not have been discovered by people. When moving from the pre-AIBO, pet-type robot to AIBO, the hardware platform was very different and the previously developed gaits did not work. It was necessary to re-develop gaits for the new robot. During the development of AIBO, our lab created several prototypes. With each new hardware version of AIBO, some gaits did not perform well. Adapting each pre-defined gait to a new prototype can take several hours and is a boring task.

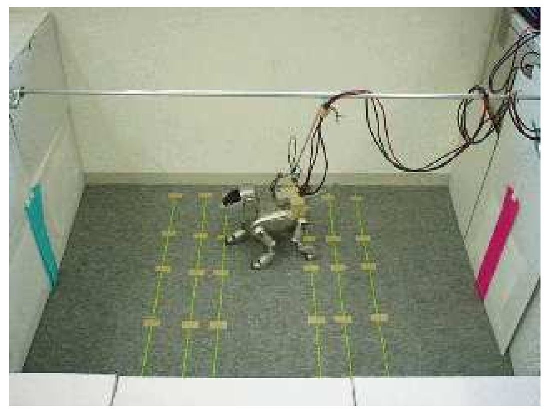

Figure 1: The experimental environment.

Finally, to make AIBO more entertaining we are interested in new behaviors for AIBO, such as new ways for it to walk or run. In all these cases a method of automatically creating, or adapting, gaits is a useful tool.

A gait created for AIBO, a consumer product, must be able to work well both on every AIBO built as well as on a variety of surface types – it must be *robust*. The gaits evolved in [1] tended to perform worse both on carpet types different from that which they were evolved on and on AIBOs different from which they were evolved with. On a different carpet (or with a different AIBO) the feet will often drag on the floor causing the robot to trip or turn. This is because the evolved gaits take shallow steps. In this paper we test whether sculpting the landscape will result in gaits that are more robust.

Gaits are evolved on two different surface types: a flat surface and one with ridges. The flat surface is made of carpet tiles similar to those used for Robocup and is the same surface as the one on which the experiments of [1] were performed. For the second surface type we lie plastic poles on this carpet to make ridges over which the robot must step (see figure 1). On this second surface the EA must evolve gaits with high steps to be able to move over the ridges and receive good fitness scores. We find that gaits evolved on the ridged surface have better overall performance on four different surface types than gaits evolved on the flat surface.

Another change for this work is we parameterize the gait type and evolve it. There are three advantages to this. It may be easier to evolve a trot or pace gait starting from a less difficult gait. By evolving the gait type we may find a gait with a better dynamic load. Finally, evolving the gait type allows us to find different types of gaits for AIBO.

For our problem of autonomous gait acquisition we chose an evolutionary approach as it has been applied to a variety of problems, achieving good results that are both novel and natural looking. Examples of this are Sims' work in evolutionary art and in evolving morphologies and controllers for simulated creatures [2] [3].

In addition to our work, evolutionary techniques have been used for gait acquisition of a legged robot in several other cases. Neural controllers were evolved for a 6-legged robot in [4] and [5] and an 8-legged robot in [6] and [7]. A binary string of on/off flags were evolved for a nitinol-actuated, 6-legged robot in [8]. In all cases the evolution produced static gaits for robots with more than 4 legs and required human intervention for evaluating the controllers.

Related to this work is the realization of dynamic gaits on a quadruped in [9]. Their robot was able to run on terrains with a small degree of irregularity. Unlike this work, their robot is constrained on the pitch plane by poles and the gaits are produced by a neural oscillator constructed by hand.

Although not used for acquisition of gaits, autonomous EAs have been used to evolve other behaviors with real robots. In these cases the behaviors evolved have been simple and the robots have had few degrees of freedom with few sensors. Examples of autonomously evolved behaviors are: forward, backward and stopping behaviors with a wheeled robot in [10]; homing navigation with a Khepera in [11]; and pursuer-evader behaviors with Kheperas in [12]. None of these behaviors would be particularly difficult to implement by hand nor would they be difficult to evolve in simulation (comparable behaviors have been successfully transferred from simulation to physical robot in [13]). In our case the gaits we evolve outperform those created by hand. Using this technique we have evolved a gait that moves at 900cm/min compared to 660cm/min for the fastest hand developed gait. Both perform well on different robots and surfaces.

The following section describes the robot platform and the locomotion module. Section 3 describes the evolutionary algorithm and method by which gait parameters are evolved. In sections 4 and 5 we present and discuss the results of our experiments. We conclude in section 6.

2 Locomotion Module

AIBO is a quadruped robot with 18 degrees of freedom and various sensors. A description of AIBO's hardware can be found in [14], which describes AIBO's precursor. In this section we describe the locomotion module that controls how AIBO moves and the parameters which are evolved.

The legs are controlled by a locomotion module that uses a vector of real-valued parameters to describe a gait. This reduces the problem of developing a gait to that of finding a set of parameters for the locomotion module. In total, there are 61 real-valued parameters used to define a gait for the locomotion module. The search space is reduced to 20 parameters by setting some parameters to fixed values (eg. setting body roll orientation to $0°$) and using the same value for multiple parameters (eg. setting the swing time for each leg to be the same). These 20 parameters are listed in table 1. They specify the position and orientation of the body, the swing path and rate of swinging of the legs, the amplitude of oscillation of the body's location and orientation, as well as specifying the point in the gait cycle when each leg swings. With a set of parameters, the locomotion module moves AIBO in any specified 2D translation and rotation – although for our experiments we test AIBO only on moving forward.

Two differences in the list of evolved parameters between [1] and this work are the removal of the gain variation parameters and the addition of parameters specifying relative swing times between legs. Gain variation was removed as the new locomotion module for AIBO did not support it. Previously, [1], the times when each leg would swing were fixed. In this way the type of gait (trot or pace) to be evolved was predetermined. Advantages of allowing the gait type to evolve are that it it might be easier to evolve a pace or trot gait starting from a half-trot, half-crawl gait. Second, a pure trot or pace gait may not be the best gait for the robot. Allowing the gait type to evolve may result in a better gait being found. Finally, con-

Table 1: Parameter List For A Gait

parameter	unit	initial range
body center x	mm.	105 - 125
body center z	mm.	-10 - 10
body pitch	degrees	-10 - 10
posture center x	mm.	0 - 20
all legs y	mm.	-5 - 15
front legs z	mm.	10 - 30
rear legs z	mm.	-5 - 15
step length	n.a.	60 - 100
swing height front	mm.	25 - 45
swing height rear	mm.	25 - 45
swing time	ms.	460 - 540
swing mult.	n.a.	3 - 5
ampl body x	mm.	-10 - 10
ampl body y	mm.	-25 - -5
ampl body z	mm.	-20 - 0
ampl yaw	degrees	-10 - 10
ampl pitch	degrees	-10 - 10
ampl roll	degrees	-5 - 15
L-R	n.a.	0.25 - 0.5
F-H	n.a.	0.5 - 0.75

verting the gait type to evolvable parameters allows for the discovery of new, and interesting gait types for AIBO.

Relative swing times for the different legs are specified as follows. Relative starting time for the swing of each leg is specified by assigning a value in the range [0-1) for each leg. This value specifies the point in the gait cycle when a leg is to start swinging. For example, a value of 0.0 indicates that the leg will start swing at the beginning of a gait cycle and a value of 0.5 indicates that the leg will start swinging halfway through a gait cycle.

Two offset parameters are used to specify the relative swing times of each leg. These parameters specify the offset between left and right legs, L-R, and fore and hind legs, F-H. The right foreleg is fixed to always start swinging at 0.0; the left foreleg starts swinging at L-R; the right hind-leg leg starts swinging at F-H; and the left hind-leg starts swinging at L-R + F-H (this value is adjusted to the range [0-1) by subtracting 1 if the sum is greater than, or equal to, 1).

Table 2 displays the swing starting time for each leg for different types of gaits. F-R and R-L values for these gaits are shown in table 3.

Table 2: Swing Starting Times for Different Gaits

Leg	Crawl	Trot	Pace	Skip
right foreleg	0.0	0.0	0.0	0.0
left foreleg	0.5	0.5	0.5	0.0
right hind-leg	0.75	0.5	0.0	0.5
left hind-leg	0.25	0.0	0.5	0.5

Table 3: Offset Values for Different Gaits

Parameter	Crawl	Trot	Pace	Skip
L-R	0.5	0.5	0.5	0.0
F-H	0.75	0.5	0.0	0.5

3 Evolutionary Method

Evolution takes place inside a pen, figure 1. At each end of the pen there is a strip of colored cloth to mark the center of that end. Using its on-board, digital camera the robot turns until it is centered on one colored strip of cloth. Once centered, the robot measures the distance to the color strip with its infrared sensor and proceeds to locomote for a fixed amount of time (7s for these experiments). The robot stops either at the end of this time or if it encounters a wall. Next the robot pans its head to find the color strip and measures its stopping distance. Using these two distances the robot scores the tested gait parameters by calculating its average speed during the trial. An individual's fitness is the average of three trials.

To simplify optimizing both velocity and straightness the score of a trial is the product of its velocity and straightness scores (averaged over three trials). Velocity, $v()$, is the average velocity of the robot during the trial. Straightness is a function of the angle between the robot's forward direction and the direction to the target color strip, θ, and the distance to the target strip, (see figure 2). Before calculating the straightness function, θ is converted to a 0-1 measure of offset by the function $f(\theta)$. The straightness function, $s()$, normalizes this value to account for the robot's distance from the color strip – with the robot at a fixed orientation θ will be larger when the robot is closer to the color strip. These functions are defined as:

$$score = v(d_{start}, d_{stop}, time) \times s(\theta, d_{stop}) \tag{1}$$

$$v(d_{start}, d_{stop}, time) = \frac{d_{start} - d_{stop}}{time} \tag{2}$$

$$s(\theta, d_{stop}) = \frac{d_{stop}(f(\theta) - 1) + 80 - 10f(\theta)}{70} \quad (3)$$

$$f(\theta) = 1 - \frac{|\theta|}{90°} \quad (4)$$

For the function $s()$, 80 and 10 are used as the constants because they are the distance sensor's maximum and minimum measurable distances. If the robot cannot find the color strip it is assumed that the robot's gait caused it to turn so sharply that it cannot pan its head far enough to face the color strip. In this case the individual receives a score of 0 for the trial, the same score it would receive if θ is 90°. An individual's fitness, used in the selection phase, is the average score over three trials.

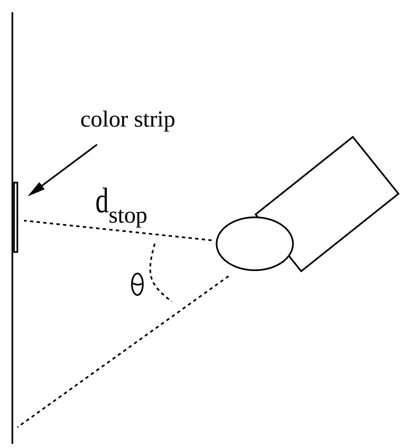

Figure 2: Measuring straightness.

A steady-state evolutionary algorithm (EA) running on-board the robot is used to evolve the parameters by optimizing fitness. The EA works by creating an initial population of different parameters to test. After evaluating the initial population new individuals are created by selecting better individuals from the population from which to create new individuals and inserting them into the population by replacing individuals with poor fitness.

The initial population is created with a uniform distribution over a given search range. Table 1 lists the twenty real-valued parameters used as genes and their initial search range. This initial range was determined from experience in hand developing gaits. Once individuals are created they are evaluated. With a dynamic gait many parameter configurations result in the robot falling over. To generate an initial population of non-falling individuals, sets of parameters in the initial population that cause the robot to fall are replaced with new, randomly generated individuals. When all individuals in the initial population are non-falling, evolution begins.

A tournament selection is used to select individuals for parents and the individuals to be replaced. First the algorithm decides whether to perform recombination or mutation. Then a number of individuals is randomly selected to be in the tournament. For recombination, 3 individuals are randomly selected and for mutation 2 individuals are randomly selected. The parent(s) is the individual(s) with higher fitness, and the individual with the lowest fitness is replaced by the offspring of the parent(s).

New individuals are created through recombination and mutation. Recombination takes two individuals as parents ($p1$ and $p2$) and creates one child individual (c). Each gene of the child is given a value according to the equation, $c_i = p1_i + \alpha_i(p1_i - p2_i)$. Here, c_i is the ith gene of the child individual; $p1_i$ and $p2_i$ are the ith gene of parents p1 and p2; and α_i is a random number in the range of -1 to 1. Mutation takes one parent individual and perturbs a few genes (1 to 8) by a small amount to generate a child individual. The genes to be mutated are selected randomly and the mutated value is, $c_i = p_i + \delta_i m_i$; where δ_i is a uniform random value in the range of -1 to 1. Values for m_i are set to 5% of a parameter's initial search range.

4 Results

Experiments were run on two types of surfaces, a flat surface and one with ridges. Three runs of evolution were performed on each surface type. The best individual evolved for each surface type was then run on different surfaces to see whether evolution on a surface with ridges produces better gaits.

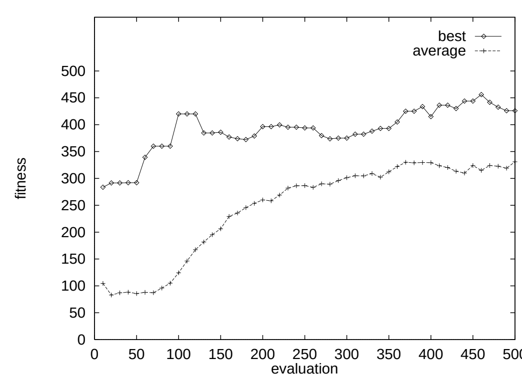

Figure 3: Evolution on a flat surface.

Each evolutionary run was for 500 evaluations, which

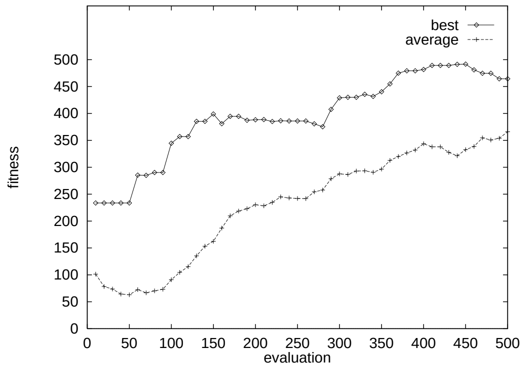

Figure 4: Evolution on a ridged surface.

took approximately 25 hours. Figure 3 contains a graph showing the results of evolution on a flat carpet and figure 4 contains a graph showing the results of evolution on a carpet with ridges. Each graph plots the highest fitness in the population and the average fitness of the population, averaged over three runs. Figure 5 shows an evolved gait.

(a) (b) (c)

Figure 5: Gait sequence for an evolved gait.

Table 4: Evolved Gaits Tested on Different Surfaces

	Flat Surface	Ridged Surface
office carpet	387 cm/min.	600 cm/min.
Robocup	369 cm/min.	598 cm/min.
tatami	369 cm/min.	576 cm/min.
wood	387 cm/min.	443 cm/min.

To compare robustness between individuals evolved on the two different carpet types we ran speed trials of the best individuals on a variety of surfaces using 3 different AIBOs. Table 4 lists the results of these trials. On all surfaces we tried, the individual evolved on the surface with ridges outperformed the individual evolved on the flat surface.

5 Discussion

Individuals evolved with the ridged surface were better than those evolved on the flat surface. Averaged over three different evolutionary runs individuals evolved on the ridged surface had higher fitness scores. Individuals evolved on the ridged surface also moved faster on the different surface types tested and with the 3 different AIBOs we tested with.

Evolved gaits were halfway between a crawl and a trot gait. Typical sets of values evolved for L-R and F-H were: (0.477, 0.686) and (0.452, 0.628). These would produce starting swing times for (right foreleg, left foreleg, right hind-leg, left hind-leg) of: (0.0, 0.477, 0.686, 0.163) and (0.0, 0.452, 0.628, 0.08).

Evolving with the ridged surface produced individuals with higher steps. In the early generations of these experiments, most individuals would drag their feet along the carpet. With the ridged surface, individuals with low steps would have a foot catch on a pole and fall over. These individuals would receive a low fitness score and be replaced by individuals with higher fitness. By the end of the evolutionary run individuals evolved steps high enough to move over the poles. In contrast, individuals evolved on the flat surface received no such pressure to use high steps. Comparisons between individuals from the two different types showed that the individuals evolved on the ridged surface generally had higher steps.

There were two problems we experienced with the poles. When moving over the ridged surface, AIBO would often step on top of a pole. Sometimes this would cause it to turn and change direction. This set of parameters would receive a poor score even though it may have been a good individual. Another problem was in pole placement. With too many poles AIBO would step on poles too frequently. With too few poles, or with poorly place poles, AIBO will have little interaction with them and the poles would not influence evolution. We settled on using six poles (as shown in figure 1). With this configuration both the front and rear legs need to move over at least two poles in a typical trial.

6 Conclusion

In this paper we presented our work in the autonomous evolution of dynamic gaits. We evolved vectors of 20 real-value parameters for our locomotion module. The evolutionary algorithm for this was run on-board the robot. AIBO evaluated the fitness for each individual without assistance by the experimenter. Using this technique we evolved gaits on both a smooth carpet and a carpet with ridges. We found that evolution on a carpet with ridges had better robustness to transference to other AIBOs and other carpets, see table 4.

These experiments produced slower results than previous trials. One reason may have been that we based our initial search space on a crawl gait – a typical crawl gait for AIBO is approximately 450cm/min. Compared to this our results from evolution with a ridged surface are much faster. Another reason for slower results may be the increased search range we use. For most parameters in table 1 the search range is more than twice as large as that used in [1]. Using a comparable number of trials to search a much large space we would not expect to achieve as good results.

By using the results of one trial to reduce the search space in a second trial better gaits can be evolved. In separate experiments using a ridged surface we evolved a trot-like gait for the consumer version of AIBO. Here we ran experiments first with a large initial search space. Using the best individual from the first run as the basis for a second evolutionary run we evolved a trot gait that moves at 900cm/min. This gait is more robust than the first trot gait evolved for AIBO and is faster than the fastest hand-developed gait of 660cm/min.

Acknowledgments

We wish to thank everyone in DCL Group 1 and in ER Business Incubation Department and our reviewers.

References

[1] G. S. Hornby, M. Fujita, S. Takamura, T. Yamamoto, and O. Hanagata. Autonomous evolution of gaits with the sony quadruped robot. In *Proceedings of the Genetic and Evolutionary Computation Conference*. Morgan Kaufmann, 1999.

[2] Karl Sims. Artificial evolution for computer graphics. *Computer Graphics*, pages 319–328, 1991.

[3] Karl Sims. Evolving virtual creatures. *Computer Graphics*, pages 15–22, 1994.

[4] M. Anthony Lewis, Andrew H. Fagg, and Alan Solidum. Genetic programming approach to the construction of a neural network for control of a walking robot. In *Proc. of Intrntl. Conf. on Robotics and Automation*, pages 2618–2623, May 1992.

[5] J. C. Gallagher, R. D. Beer, K. S. Espenschied, and R. D. Quinn. Application of evolved locomotion controllers to a hexapod robot. *Robotics and Autonomous Systems*, 19(1):95–103, 1996.

[6] N. Jakobi. Running across the reality gap: Octopod locomotion evolved in a minimal simulation. In P. Husbands and J-A Meyer, editors, *Proceedings of Evorob 98*, 1998.

[7] Frédéric Gruau and Kameel Quatramaran. Cellular encoding for interactive evolutionary robotics. Technical Report 425, University of Sussex, 1996.

[8] Gary B. Parker, David W. Braun, and Ingo Cyliax. Learning gaits for the stiquito. In P. Husbands and J-A Meyer, editors, *Proc. of 8th Intl. Conf. on Advanced Robotics*, pages 285–290, 1997.

[9] Hiroshi Kimura, Kazuaki Sakurama, and Seiichi Akiyama. Dynamic walking and running of the quadruped using neural oscillator. In *Proceedings of International Conference on Robotics and Automation*, pages 50–57. IEEE, 1998.

[10] Luc Steels. Emergent functionality in robotic agents through on-line evolution. In R. Brooks and P. Maes, editors, *Proc. of Fourth Workshop on Artificial Life*, pages 8–14, 1994.

[11] Dario Floreano and Francesco Mondada. Evolution of homing navigation in a real mobile robot. *IEEE Transaction on Systems, Man and Cybernetics–Part B: Cybernetics*, 26(3):396–407, 1996.

[12] Dario Floreano and Stefano Nolfi. Competitive co-evolutionary robotics: From theory to practice. In *From Animals to Animats V*. MIT Press, 1998.

[13] Nick Jakobi. *Minimal Simulations for Evolutionary Robotics*. PhD thesis, School of Cognitive and Computing Sciences, University of Sussex, May 1998.

[14] M. Fujita and K. Kageyama. An open architecture for robot entertainment. In Marina del Rey, editor, *Proc. of First Intl. Conf. on Autonomous Agents*, pages 234–239, 1997.