Using Genetic Algorithms To Establish Efficient Walking Gaits For An Eight-Legged Robot

B.L. Luk[1], *S. Galt*[2] *and S. Chen*[3*]

[1] Center for Intelligent Design, Automation and Manufacturing
City University of Hong Kong
Kowloon, Hong Kong, P.R. China

[2] PORTECH Ltd
Unit 15 The Tanneries, Brockhampton Lane, Havant
Hampshire PO9 1JB, U.K.

[3] Department of Electronics and Computer Science
University of Southampton, Highfield
Southampton SO17 1BJ, U.K.

Abstract

In the design and development of a legged robot, many factors need to be considered. As a consequence, creating a legged robot that can efficiently and autonomously negotiate a wide range of terrain is a challenging task. Many researchers working in the area of legged robotics have traditionally looked towards the natural world for inspiration and solutions, reasoning that these evolutionary solutions are appropriate and effective because they have passed the hard tests for survival over time and generations. This paper reports the use of genetically inspired learning strategies, commonly referred to as genetic algorithms, as an evolutionary design tool for improving the design and performance of an algorithm for controlling the leg stepping sequences of a walking robot. The paper presents a specific case of finding optimal walking gaits for an 8-legged robot called Robug IV and simulated results are provided.

Keywords: Mobile robots, legged robots, walking gaits, genetic algorithms.

*Contact author. Tel./Fax: +44 (0)23 8059 6660/4508; E-mail: sqc@ecs.soton.ac.uk

1 Introduction

Legged robots present significant advantages over wheeled or tracked mechanisms due to their ability to move in very rough and unstructured terrains and to step over obstacles. However, without efficient walking strategies these advantages cannot be realised. Robots that stumble and bump into obstacles are classic examples of robots with inefficient walking strategies; with each terrain contact there is a loss of energy and accelerated wear and tear on the vehicle, resulting in a reduction of the expected lifespan. Designing a legged robot that is capable of walking over a variety of terrains efficiently and autonomously is a challenging task that involves expertise from a wide range of disciplines. When designing a legged robot it is often useful to consider biological walking systems, which tend to be much more versatile and seem to be more effective and elegant, in order to emulate these or similar mechanisms in the design. Indeed, researchers working in the area of legged robotics have traditionally looked toward the natural world for inspiration and solutions (Evoy and Fourtner 1973, Lee and Orin 1988, Song and Waldron 1989,Cruse 1990, Cruse *et al.* 1993, Galt 1996).

Previous research into biological walking machines has identified certain key areas in the evolution process that has resulted in natural walking mechanisms being so versatile. For example, the terrestrial crab is well known for its agility (Burrows and Hoyle 1973), the reasons for which are largely due to the reduction, through natural evolution, of the abdomen which made lateral walking feasible (Herreid 1981). This naturally invoked improvement to the crab skeleton has thus opened up the potential for extremely rapid movement because it has been made possible for adjacent limbs to move freely without fear of interfering with each other. It is thus clear that tuning and optimisation techniques that can emulate this process of “improvement by natural evolution” could possibly be beneficial in the design and development of a walking robot by arriving at solutions that may not have been possible to derive manually (or were simply never contemplated).

This paper will discuss such a method of “natural evolution emulation” with which to improve the mechanisms for controlling the stepping sequence of a legged robot. This is achieved by applying Genetic Algorithms (GAs) (Goldberg 1989) – probabilistic search and optimisation procedures based on natural evolution principles. Within this paper it is shown that by employing GA optimisation within the walking system of a legged robot, favourable gait behaviour can be achieved. By analysing these GA-derived solutions it is then possible to learn ideal behaviour patterns and gain valuable knowledge with which to update the control mechanisms, or alternatively with which to improve future robot designs. This paper explores the use of GAs for this purpose, and in particular focuses on the case of the application of GAs in order to produce optimally stable walking motions under normal conditions, and also for the case when one leg becomes inoperative. The work was conducted on an 8-legged robot called Robug IV.

2 Robug IV

The research described within this paper is specific to the development of Robug IV (Figure 1). Many of the design features of Robug IV were inspired by "natural solutions". The genesis of the robot structure is based on the emulation of arthropod walkers and climbers; in particular the entomological and crustacean groups. Robug IV has eight pneumatically powered legs, each consisting of two physical links with 4 joints – a knee joint, an abductor joint, a hip joint and an ankle joint (see Figure 2). Each leg module has 4 dedicated microprocessors, one for each joint in the leg, with the ankle joint processor serving as the leg "master processor". The joints are all driven by a pneumatic drive system at approximately 1300kPa to achieve a high power-to-weight ratio and inherent compliance; these qualities are important in walking robots because they allow for the development of lightweight machines without compromising the payload capabilities, whilst minimising the possibility of damage when operating in unstructured environments. The pneumatic drive system conveniently allows for the attachment of vacuum gripper feet at the end of each leg for climbing.

Each dedicated leg microprocessor is responsible for the low level control and the leg dynamics computations. The Siemens C167 RISC processor was chosen for this purpose - this processor operates at 20MHz and each one has been equipped with 1MB of RAM and 1MB ROM. The remaining high level control is conducted by means of an external PC (Pentium II 300) connected to the robot by means of a high-speed data link and CAN bus system. The system hardware topology is shown in Figure 3. A detailed description of Robug IV can be found in (Galt 1998).

The mechanical and electronic systems of Robug IV have been designed very much with a high degree of modularity in mind so that the existing design can be modified easily and functionality can be added without any major restructuring of the existing design. For example, the actuators, robot body structure and leg mechanism can be modified easily to suit new applications. The system has an open bus architecture to allow various tool packages and navigation systems to be added without any major rework. It is anticipated that the sophistication and flexibility of Robug IV will consequently open up the accessibility of such robotic systems to various industrial applications such as building inspection, reactor pressure vessel maintenance and landmine clearing operations.

3 Walking gaits

The walking gait algorithm is the decision based process regarding which leg should be lifted or placed to provide time-space co-ordination of the motion of the various legs of the robot. It is the most crucial process in the control of legged robot motion. In the extreme case this decision is made with regards to factors such as the condition of the terrain, stability requirements, ease of control, smoothness of body motion, speed requirements, mobility requirements and power requirements. This presents a highly complicated problem which is most commonly reduced in gait algorithms by concentrating on performing smooth walking motions over variable terrain while maintaining vehicle stability and velocity. Theoretically, the more legs that a walking mechanism has at its disposal, the greater the possibility of successfully negotiating any terrain without losing balance or reaching deadlock (the point at which none of the legs can be moved without exceeding defined minimum vehicle stability and force limits). This rationale is only relevant if the gait strategy is adequately capable of generating an appropriate sequence of leg stepping motions that will result in efficient and stable walking behaviour, regardless of terrain properties.

The walking gait, or *corporate motion of the legs*, can be defined as the time and location of the placing and lifting of each foot, co-ordinated with the motion of the body in order to move the body from A to B (Song and Waldron 1989). Gaits describe and determine the speed, direction of motion and the mobility of a walking machine. The number of legs, leg geometry and performance are very much related to the selected gait and thus an efficient gait for one robot may not necessarily be good for another (McGhee and Iswandhi 1979). Due to the complexity of gait analysis, many aspects of multi-leg walking motions are still poorly understood. However, several researchers in the past two decades have worked on this subject, in particular McGhee and Iswandhi (1979), and Song and Waldron (1989). Their work has resulted in the formation of a mathematical foundation for gait analysis. Based on this foundation, many useful definitions and theorems have been devised. The following subsections give a description of those terms and definitions used within this paper for those new to gait theory.

3.1 Definitions for gait analysis

The following basic definitions, adapted from the original terminology first established in (McGhee and Iswandhi 1979, Song and Waldron 1989), are a collection of new definitions specific to the leg numbering scheme for Robug IV as shown in Figure 2.

Definition 1 The *support phase*, S_i, is the period in which leg i is in contact with the ground and is thus seen to be “supporting” the robot body. During this phase, the leg is referred to as being in state 0.

Definition 2 The *transfer phase*, T_i, is the period for which leg i is no longer in the support phase and is moving toward its target position. During this phase, the leg is referred to as being in state 1.

Definition 3 The *leg cycle time*, t_i, is the time for leg i to complete a single periodic cycle of the support and transfer phases.

Definition 4 The *leg duty factor*, β_i, is the fraction of the cycle time for which leg i is in the support phase, thus:

$$\beta_i = \frac{S_i}{t_i} \tag{1}$$

Typical value of the leg duty factor is $0.5 \leq \beta \leq 0.75$.

Definition 5 The *leg phase*, ϕ_i, is the fraction of a cycle period by which the contact of leg i on the ground lags behind a phase reference point. The phase reference point is the start of the cycle for moving all the robot legs.

Definition 6 The *gait phase formula*, g, consists of the respective leg phases for describing a gait of an n-legged vehicle:

$$g = (\phi_0, \phi_1, \cdots, \phi_{n-1}) \tag{2}$$

Definition 7 A *gait event* is a change of state of a leg during locomotion. There are two possible events to signify the changes between support and transfer phases, that is lifting and placing events.

Definition 8 A *gait state* is an n-element array whose elements have values in the range 0 to 1, corresponding to the n leg phases of an n-legged vehicle at that time.

Definition 9 A *regular gait* is a gait with the same duty factor for all legs.

3.2 Gait diagrams

In the analysis of walking gaits it is useful to represent the walking behaviour graphically. There are many standard methods for this purpose (Song and Waldron 1989). The *gait diagram* is amongst the simplest and most useful graphical representations (Hildebrand 1967). In the diagram, each horizontal line is assigned to a leg. Black line segments denote the leg being in the support phase and blank areas represent the transfer phases. Hence, these alternating black line segments and blank areas represent the gait events of a walking sequence. The beginning and end of the black line segments denote the placing and lifting of the feet respectively. This method provides a simple yet informative display of the leg activity during walking. From the gait diagram we can easily judge the duration of the leg support phases and the ratio of support to transfer phases.

To help the understanding, gait diagrams are used throughout this paper to provide a graphical representation of the gait characteristics over time (i.e. the gait state). Figure 4 shows the gait diagram for a very stable walking gait known as the "metachronal" walking gait with a duty factor of 0.75. The metachronal gait is characteristic in that the placing of each foot on one side of the robot body runs from either front to back or back to front as a wave and further, each pair of legs on either side of the robot body is 180^{o} out of phase. The legs are numbered so as all even-numbered legs are positioned on one side of the body whilst odd-numbered legs are on the other side (as shown in Figure 2).

3.3 Stability in locomotion

The more legs that a walking machine has the greater the potential for that vehicle to walk in a state of continuous static stability. A biped robot will probably reveal utmost manoeuvrability and land mobility whilst having the least stability and, as a result of the latter, require the most complex control system for balance maintenance. An eight-legged robot is at an advantage over bipeds, quadrupeds and hexapods in its potential for negotiating a wide range of rough terrains whilst maintaining a good static stability, thus eliminating the requirement for dynamic balancing within the control structure. Until now, most efforts to build dynamically balanced robots have been confined to bipeds (Raibert 1986) and hopping machines (Raibert and Sutherland 1983, Raibert *et al.* 1983, 1986). One reason for this is that the present state of the art in digital computers has allowed the implementation of only simplified dynamic models for walking machines. Also, the complexities of the locomotion systems in biological creatures have prevented proper understanding of the involved mechanics and controls.

For a legged robot that is not designed for dynamic balancing, such as Robug IV, it is imperative that the vehicle is prevented from becoming unbalanced for prolonged periods. Thus it is vitally important to have ability to evaluate the stability of the robot and predict if and when the robot becomes unstable. A leg partakes in the maintenance of the vehicle's stability during the support phase, whereas throughout the transfer phase the leg loses contact with the supporting surface and fails to serve any useful purpose (other than to return to a suitable supporting position as soon as possible). The most commonly used measure of legged vehicle stability is known as the *stability margin* and a description of the terms and methods used to evaluate this quantity for a particular gait is listed in definitions 10–13:

Definition 10 The *support plane* is defined as a best-fit plane obtained by doing a linear regression on the points of support at any given state. The support plane on level terrain is the surface on which the contact points of the feet lie, but it is more difficult to define a support plane on uneven terrain.

Definition 11 The *support polygon* of the robot is the two-dimensional point set in a horizontal plane obtained from convex hull of the vertical projection of all the contact points of the legs in the support phase. The contact between foot and ground is idealised to a point contact. In a real, distributed foot contact, the contact point can be interpreted as the centre of pressure.

Definition 12 The *stability margin*, SM, is the shortest distance of the vertical projection of the centre of gravity of the robot to the boundaries of the support polygon in the horizontal plane (see Figure 5). The stability margin is one of the commonly used methods for analysing gait algorithms, as maintaining static stability is generally the most significant aim when deriving leg stepping sequences for a statically stable robot. Within this research, the overall stability of the vehicle at a given time is defined as the current value for the stability margin.

Definition 13 The *gait stability margin* is the minimum stability margin for a gait over all the support patterns encountered in the entire cycle. The gait stability margin is a measurement of the performance of a gait algorithm over a period time.

The gait diagrams and stability traces are combined in one single graph for convenience as shown in Figure 6. The stability trace represents the ratio of the current vehicle stability to the maximum possible stability for the robot as a function of time. A negative value of the stability means that the vehicle has lost its stability at that time.

4 Evolving gait strategies

In comparison to walking robots, the walk of an insect appears to be much more versatile and seems to be more effective and elegant, which is hardly surprising given the millions of years that nature has had with which to refine the biological design of such creatures.

Under certain conditions of motion, such as walk, run, crawl or climb, a certain gait is optimum when applied to a legged robot, the reasons for which are related to stability, leg structure, speed or other factors. From inspection of data accumulated on the walking habits of the ghost crab (Barnes 1975), which is seen to display lateral walking habits, it can be seen that the crab often walks with the “metachronal” gait. Most laterally walking crabs use this gait under normal walking conditions, and it can be shown that when the gait is implemented on Robug IV it provides a high degree of stability.

In general, it is of great interest to be able to determine the optimum gaits for various conditions and the reasons for their optimality. One way in which these natural solutions can be derived artificially is by incorporating GA learning mechanisms into the gait models. GAs are based on the biological evolutionary process and can thus provide a means of emulating natural evolution. The GAs used for this research have three basic operators called *reproduction*, *crossover* and *mutation*. The reproduction operator creates a new population by randomly selecting individuals as parents from the existing population, according to a weighted-random analysis of their fitnesses. The selected individuals are “mated” using the crossover operator. Infrequent random changes are made to each chromosome using the mutation operator, the invocation probability of which is kept low to avoid losing a large number of good chromosomes. For a detailed description of this so-called *simple* GA, see Goldberg (1989).

In order to test and develop gait algorithms for Robug IV, a software package, named ROBSIM, has been specifically designed and developed to model the kinematics and static forces of Robug IV. The need for ROBSIM software becomes immediately apparent when one considers the prospect of testing gait algorithms on the actual robot. Even if there were enough resources available to provide a reasonably sized test terrain (i.e. approximately 100m^2 or more), the prospect of making any mistakes with such a high powered machine cannot be attractive to many. Also, safety is another main concern of carrying the test on the actual robot. Since GA is essentially a “trial-and-error” based method, it is clear that such research is best conducted by means of simulation.

The robot model employed within ROBSIM was developed from a detailed analytical study of the geometry of the motion of the robot, with respect to a fixed reference coordinate system. A sequence of mathematical equations were developed using Denavit-Hartenberg representation (Fu *et al.* 1987) which provide an analytical description of the spatial displacement of the robot as a function of time, in particular the relations between the joint-variable space and the position and orientation of the end-effectors (feet) of the robot. The resulting kinematic model allows us to identify the position of each leg with respect to the robot body. In addition, the force distribution of the robot is modelled based on the kinematic position of the legs that are in support at that time. The ROBSIM software was written using the C++ language. Extensive use of the object-oriented programming features of C++ was made and the software was structured so as to constitute a library of routines which can be called individually as needed in the main control software for the robot hardware. Employing this design methodology means that there is a much closer link between simulated control performance and actual control performance as the same routines are used to simulate the robot as to control it.

In the next two sections it is shown that walking gaits with optimal or near-optimal stability margins can be obtained by using a GA to facilitate the derivation of the optimal gait parameters and that these resulting gaits are comparable to those found in nature.

5 A GA-based gait tuning algorithm

In order for the GA to be able to derive gaits with optimal or near optimal stability, there must exist a procedure in which the GA can analyse and optimise a series of walking gaits based on the gait performance. Figure 7 highlights the basic procedure behind the GA-based gait tuning algorithm that has been developed for this purpose. For the initial population of chromosomes, the bit pattern contained within each chromosome is initialised entirely at random. In applying the GA to carry out the optimisation procedure described above, consideration must be given to both the encoding of the walking gait parameters within the GA and the methods by which the GA can control the gait. The process of creating a new population, which is achieved using the standard GA functions of selection, crossover and reproduction, is executed based on a random-weighted principle so that the chromosomes which had bit patterns that produced the gaits of the highest stability within the old population have the best chance of contributing to the next population. The following subsections describe the encoding scheme and fitness analysis procedure employed in our GA-based gait tuning algorithm.

5.1 Chromosome encoding of the gait parameters

A walking gait is represented by a series of parameters that describe the individual leg phases and duty factors (see definitions 4 and 5). For a non-changing periodic gate, these parameters define the precise timing of the leg lifting and placing events, thus controlling body stability for a fixed velocity. For Robug IV, eight parameters are used to represent the phases for which legs 0 to 7 lag behind the phase reference point. Each phase parameter is represented as 7 binary digits in the chromosome, as shown in Figure 8, thus each individual chromosome contains a 56-bit binary string describing the phase values that (together with duty factors) define the walking gait. The phase of each leg within the walking gait must be a number between 0 and 1. A phase of 0 means that particular leg is in phase with the phase reference point whilst a phase of 0.5 means that leg is 180° out of phase with respect to the phase reference point (as described in definition 5). Note that signed phases are not required as a lagging phase of 45° (-0.125) is equivalent to a leading phase of 315° (+0.875) for a sustained gait cycle. Given that each phase is represented by 7-bits, individual leg phase values between 0.0 and 0.9921875 (0° to 360° in 128 steps of size 7.8125×10^{-3}) can be encoded within the chromosome.

Using this technique, it is also possible to encode the duty factor within the chromosome, although this idea was rejected as there is no real benefit in doing so. If the duty factors for each leg were encoded then the GA would try to optimise the gait stability by increasing all the duty factors, thus increasing the amount of time each leg is in the support phase compared to the time the leg is in the transfer phase (this will obviously improve stability). This increase in the duty factors would continue until the maximum duty-factor limit was reached (the point at which the transfer leg velocity exceeds the maximum that is theoretically possible using the physical robot). Consequently, in this work, the duty-factor for each leg was fixed at 0.66; therefore each leg is in the support phase twice as long as it is in the transfer phase. This has previously been found to be an acceptable value in practical applications. By fixing the duty factor, the walking gait is described alone by its leg phases. Thus, the work described in this paper is aimed specifically towards the optimisation of regular gaits.

5.2 Analysing fitness

The chromosome fitnesses are evaluated as a function of the stability of the robot. It was previously shown in subsection 3.3 how the stability of the robot can be analysed for a particular leg configuration. This technique is used throughout the entire cycle of simulating the 30 second actual walking tests conducted on each of the gaits decoded from the chromosomes in the current population. The length of the walking test was set at 30 seconds as this was found to be sufficient to enable several complete gait cycles to be completed at a vehicle speed of 0.1ms^{-1} regardless of the direction of travel. Note that the simulation is run "off-line" so that the 30 second actual walking test can be completed in less than 0.4 seconds (on a Pentium 166), thus the running time of the GA algorithm is about 100-generations being processed per hour (for a population size of 100).

The goal of the GA-based gait tuning algorithm is to optimise the stability of the robot by maximising the stability margin, SM, hence high stability results in a high fitness value for that chromosome. The fitness of chromosome n is evaluated as follows:

$$f_n = SM_{\text{min}}^2 \tag{3}$$

where SM_{min} represents the minimum stability margin value during the simulated 30 second walking test. Note that the minimum stability margin is squared in order to bias the GA towards evolving only gaits with high stability. Thus, squaring the stability margin acts like a filter to reduce the probability of reproduction from chromosomes with low and medium stability gaits.

6 Experimental results

Two tests were conducted using a GA to find gaits which offered maximum stability. The first test was for the robot walking over flat terrain in normal operating conditions, that is, walking with a laterally (sideways) directed body velocity. The second test was identical to the first test except that one leg was made inoperative to simulate a mechanical breakage, thus rendering the leg useless. The fitness function of the GA was based on the stability of the robot evaluated over a set walking period. The individual chromosomes of the GA were encoded to represent the co-ordinating parameters for each leg, namely the phase, that describe the time relationships between the legs which thus define the basic walking motions of the robot.

6.1 Improving regular gait stability

The GA parameters used for the first test are summarised in Table 1. These parameters were found to be adequate through experience of using the system. The termination condition for the GA occurs when the maximum generation number (100 in this case) is reached. The resulting GA-generated gait is given in Figure 9 and the averaged fitness, smoothed by an exponential weighting filter, is given in Figure 10. Figure 9 shows the derived walking gait for the fully operational robot, which can be seen to be approximately tetrapodal. It is believed to be no coincidence that this type of gait has been shown to exist in nature and is characteristic of the walking behaviour of the ghost crab over flat terrain (Barnes 1975). The gait phase formula for this GA-generated gait is:

$$g = (0.371,\ 0.996,\ 0.941,\ 0.445,\ 0.016,\ 0.379,\ 0.395,\ 0.969)$$

It should be noted that the above gait is tuned for stability, not force loading. It is obvious that lifting 4 legs at a time will not optimise the maximum leg loading. Other criteria such as leg loading and body velocity can be included into the fitness function of the GA and this would doubtless give different results.

6.2 Compensating for leg redundancy

We have seen the use of GAs to emulate natural evolution to arrive at a similar solution. A more practical application of the GA-based design tool is to a situation in which there is no simple way of copying nature. For the second test we will assume the robot to have an inoperative limb, which may have been caused by damage or a system failure. Research data concerning the walking behaviour of crabs with a leg removed is very limited and thus in this situation there is no comprehensive data of a “natural solution” as such.

The GA was again applied to the walking gait, but this time leg 0 was inoperative and did not contribute to stability at all. The parameters for the GA were retained from the previous test. The results given in Figure 11 represent the gait diagram and vehicle stability for the gait decoded from the chromosome with the greatest fitness after the GA gait tuning algorithm had completed 100 generations. Figure 12 depicts the smoothed averaged fitness. Close inspection of the resulting gait diagram, in comparison to the previous results given in Figure 9, shows that legs 1 and 2, which were approximately in phase in the previous results, are now co-ordinated in such a way that there is always one of the two legs in contact with the walking surface thus contributing to the maintenance of the vehicle’s stability. Experience supports this action as it has been previously found that lifting all three legs of a “corner tripod” of Robug IV (in this case the tripod formed by legs 0, 1 and 2), is a very hazardous state and often results in unstable or critical stability situations. In this case, as leg 0 cannot provide any contribution to the stability, the only possible course of action is to ensure that legs 1 and 2 are never lifted simultaneously, as in the solution provided by the GA-based gait tuning algorithm.

7 Discussion and conclusions

The work presented in this paper has shown that GAs can be used to emulate natural evolution and the gait generated by GA for Robug IV on flat terrain is very similar to that evolved by ghost crab. It has also been shown that solutions can be found for the problem when a leg is inoperative, a case for which natural evolutionary solutions are not well understood. The criterion used in this research was mainly stability. However, for practical industrial applications, other criteria are needed in addition to the stability criterion. For example, if the robot is required to carry a heavy robot manipulator, it will be essential to optimise the criterion of payload or leg loading. On the other hand, if the robot is needed to perform plasma cutting, the criteria for generating the gait will be different. It will be more important for the robot to achieve a constant body velocity and body altitude for the cutting operation. In order to incorporate multiple criteria into the gait generation process, it will require the use of the multi-objective GA (Fonseca and

Fleming 1993). In fact this is one of the research areas that this work can be extended.

Besides finding the appropriate walking gait solutions through the use of GA for an existing robot, the research results have highlighted another issue which the GA can be very useful for designing walking robots in the future. As we compare Figure 10 and Figure 12, the stability fitness of 8-legged walking gait is better than 7 legged walking gait under the same test environment. This shows that the GA can provide useful information for evaluating and comparing various robot mechanism designs and possibly finding the optimal design solution (or near optimal solution) for the robot. After all, the nature has evolved not just the walking gaits, but also the walking mechanism for legged creatures. Again, this needs further investigations on how the gait generation and the robot mechanism design can be combined within GA for obtaining a system solution for a particular application.

Acknowledgements

The Robug IV project was supported by the EPSRC grant (GR/M16894) and the EC ESPRIT grant (WIRED). The authors would also like to thank PORTECH Ltd for their industrial support and PAC Ltd for their help in setting up the communication network for the robot.

References

Barnes, W.J.P., 1975, Leg coordination during walking in the crab, Uca pugnax. *J. Comp. Pysiol.*, **96**, 237–256.

Burrows, M., and Hoyle, G., 1973, The mechanism of rapid running in the ghost crab, ocypode cerathophthalma. *J. Experimental Biology*, **58**, 327–349.

Cruse, H., 1990, What mechanisms coordinate leg movement in walking arthropods? *Trends in Neurosciences*, **13**, 15–21.

Cruse, H., Muller, U., and Dean, J., 1993, Artificial neural nets for a 6-legged walking system. in *From Animals to Animats 2* (MIT Press).

Evoy, W.H., and Fourtner, C.R., 1973, Nervous control of walking in the crab. *J. Comp. Physiol.*, **83**, 303–318.

Fonseca, C.M., and Fleming, P.J., 1993, Genetic algorithms for multi-objective optimization: formulation, discussion and generalization. in *Proc. 5th Int. Conf. Genetic Algorithms* (San Mateo, CA), 416–423.

Fu, K.S., Gonzalez, R.C., and Lee, G.S.G., 1987, *Robotics – Control, Sensing, Vision and Intelligence* (McGraw-Hill International Editions).

Galt, G., 1996, *A Survey of the Existing Gait Generation Strategies for Legged Robots* (Internal Research Report No. 96/0016, University of Portsmouth, UK).

Galt, G., 1998, *Soft Computing Based Motion Control for an Eight-legged Robot* (PhD Thesis, University of Portsmouth, UK).

Goldberg, D.E., 1989, *Genetic Algorithms in Search, Optimisation and Machine Learning* (Addison Welsey).

Herreid, C.F., 1981, Locomotion in arthropods: an evolutionary phatasmagoria. in *Locomotion and Energetics in Arthropods* (Plenum Press).

Hildebrand, M., 1967, Symmetrical gaits of horses. *Science*, **150**, 701–708.

Lee, W.J., and Orin, D.E., 1988, Omnidirectional supervisory control of a multilegged vehicle using periodic gaits. *IEEE J. Robotics and Automation*, **4**, 635–641.

McGhee, R.B., and Iswandhi, G.I., 1979, Adaptive locomotion of a multilegged robot over rough terrain. *IEEE Trans. Systems, Man and Cybernetics*, **9**, 176–182.

Raibert, M.H., 1986, *Legged Robots that Balance* (Cambridge, Massachusetts: MIT Press).

Raibert, M.H., Brown, H.B., and Chepponis, M., 1986, Experiments in balance with a 3-D one-legged hopping machine. *Int. J. Robotics Research*, **3**, 75–92.

Raibert, M.H., Brown, H.B., Chepponis, M., Hastings, E., Murthy, S.S., and Wimberly, F.C., 1983, *Dynamically Stable Legged Locomotion* (Progress Report, Leg Laboratory, Carnegie-Mellon University).

Raibert, M.H., and Sutherland, I.E., 1983, Machine that walk. *Scientific American*, **248**, 44–53.

Song, S.M., and Waldron, K.J., 1989, *Machines that Walk: the Adaptive Suspension Vehicle* (MIT Press).

Parameter	**Value**
Population Size	50
Generation Number	100
Chromosome Size (bits)	56
Crossover Probability	0.9
Mutation Probability	0.025

Table 1: Parameter settings for the GA.

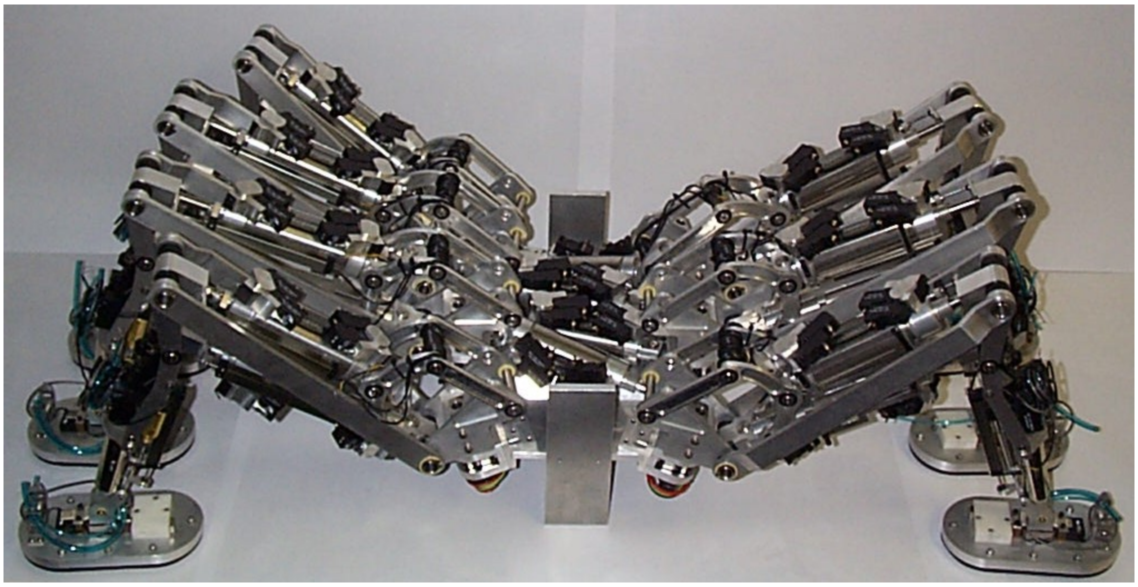
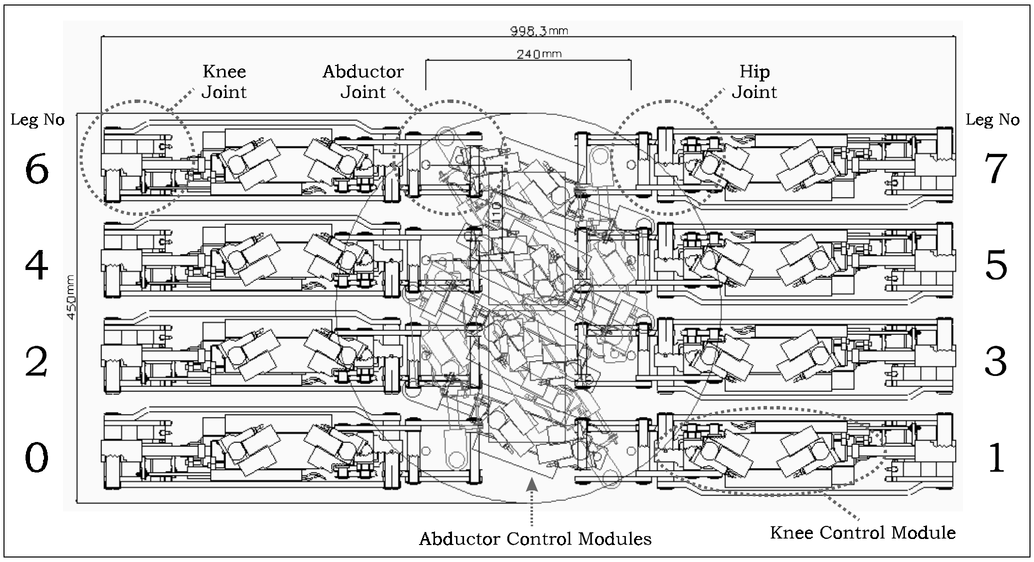
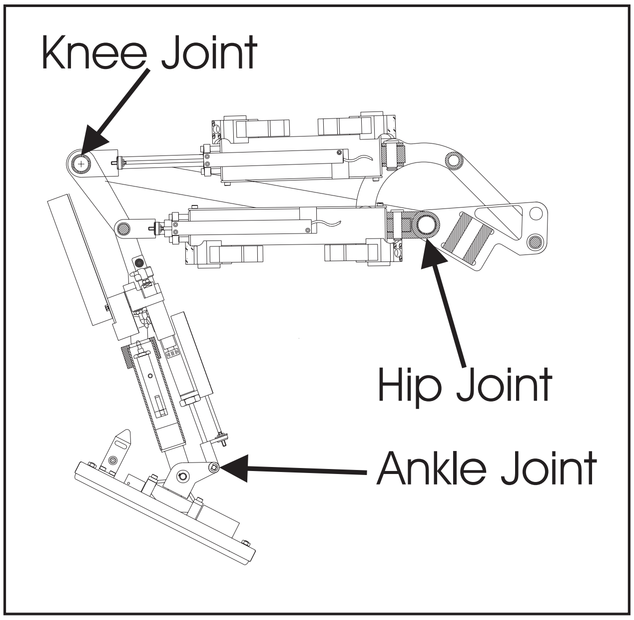


Figure 1: Robug IV. A fully extended robot leg has an approximate length of 1 m.



(a) Plan view of the robot schematic and leg numbering scheme



(b) Side view of a Robug IV leg

Figure 2: Robot schematic and leg numbering scheme.

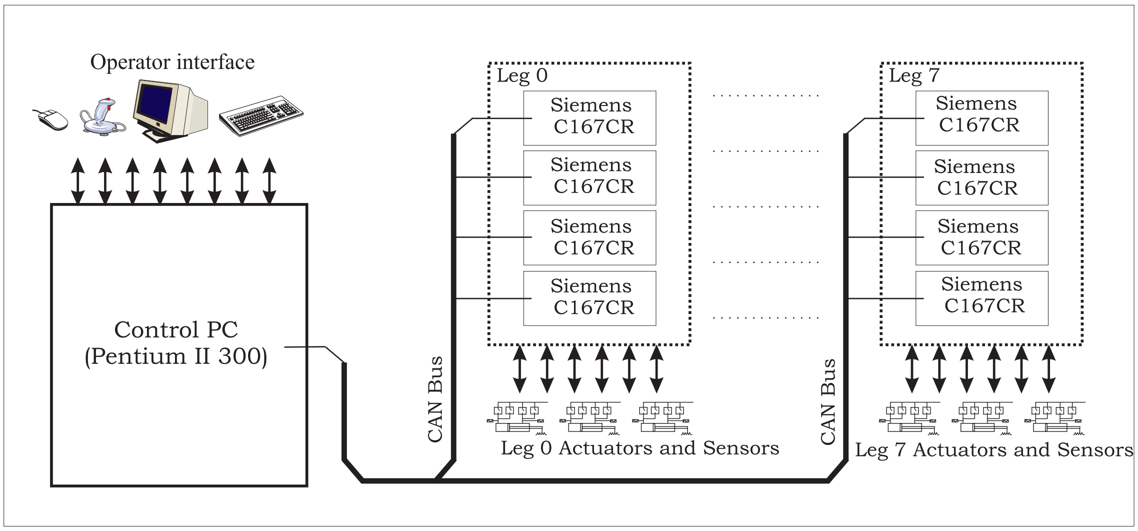


Figure 3: Hardware topology.

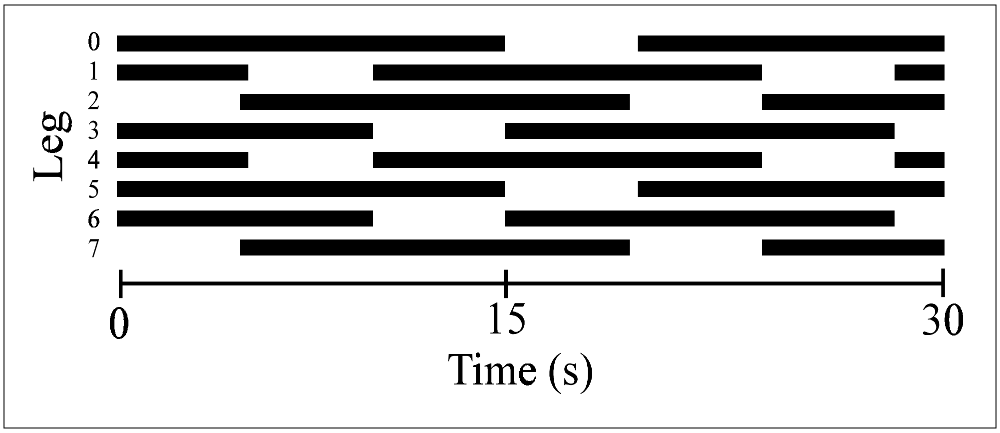


Figure 4: Gait diagram of a metachronal gait.

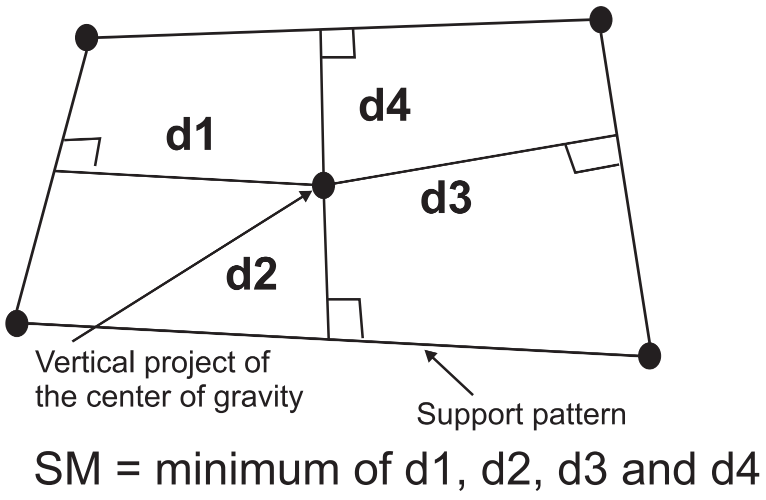


Figure 5: Stability margin SM of a support pattern.

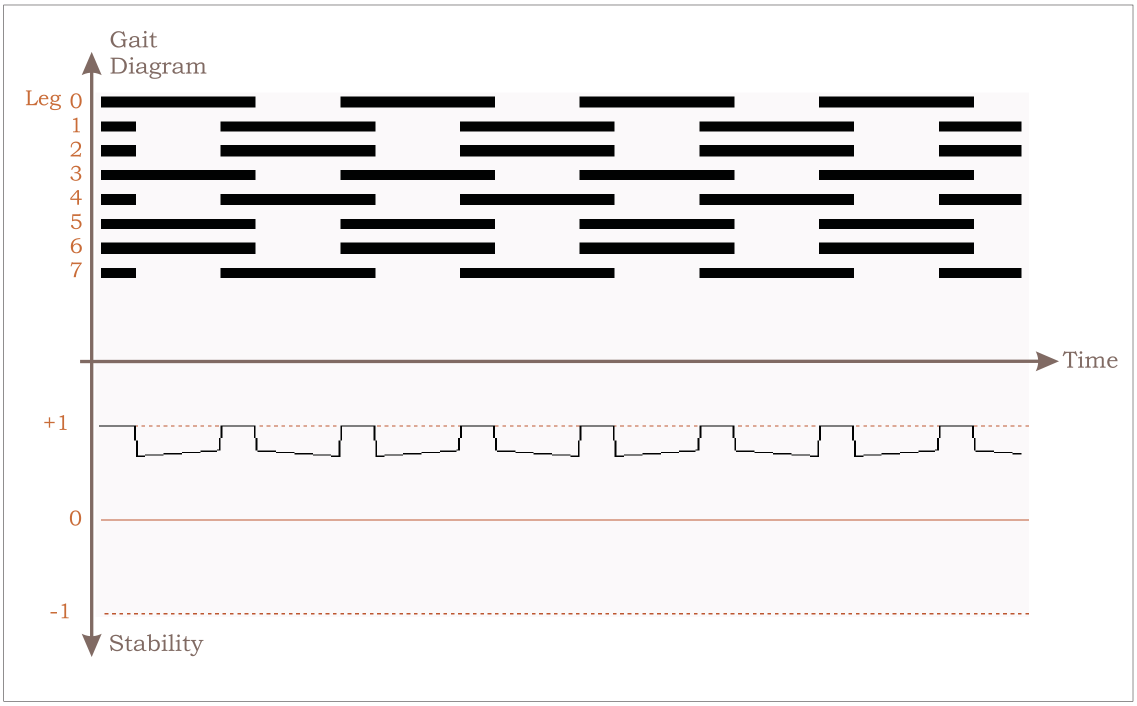


Figure 6: Combined gait diagram and stability trace for a tetrapod walking gait.

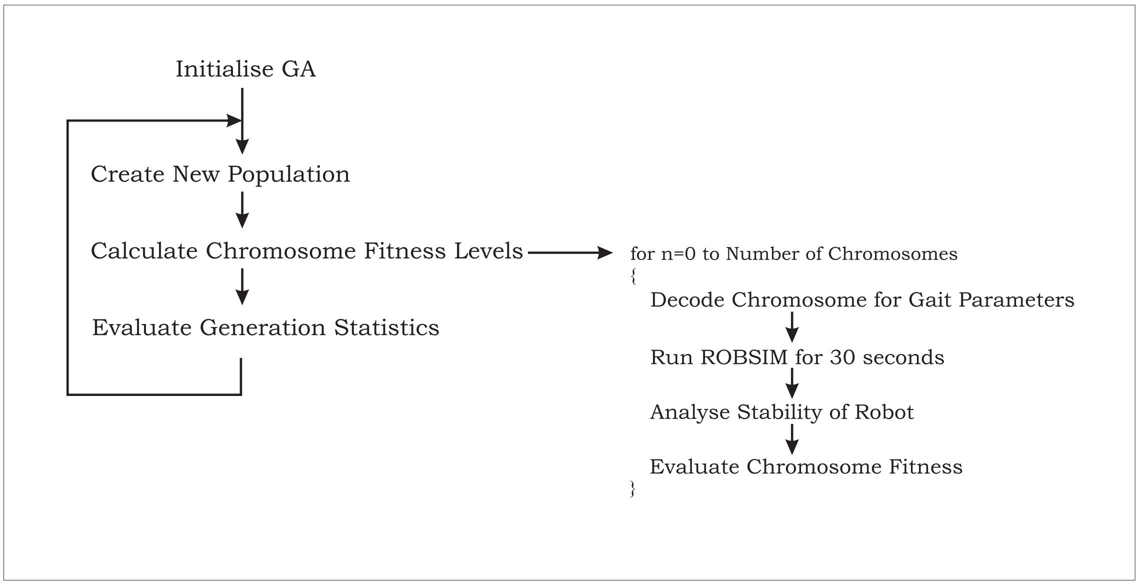


Figure 7: The GA-based gait tuning algorithm.

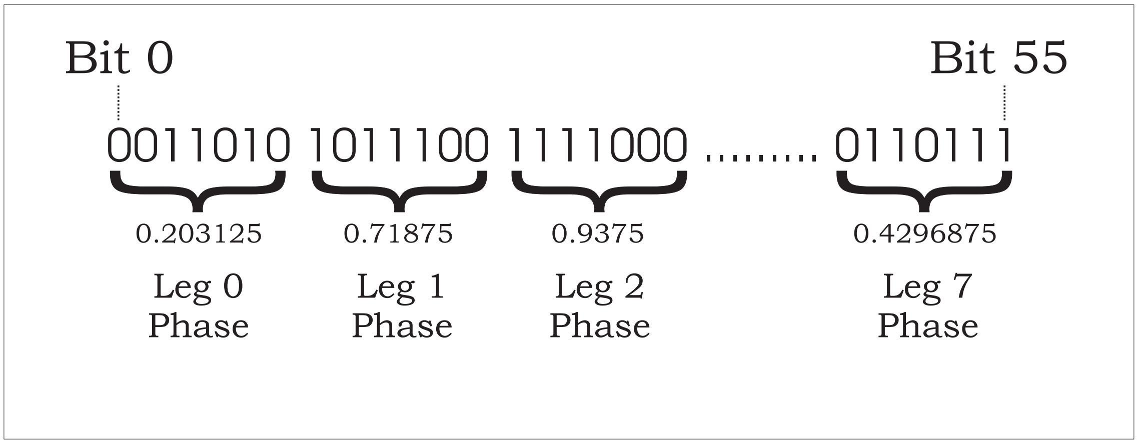


Figure 8: Gait phase encoding within the GA chromosome.

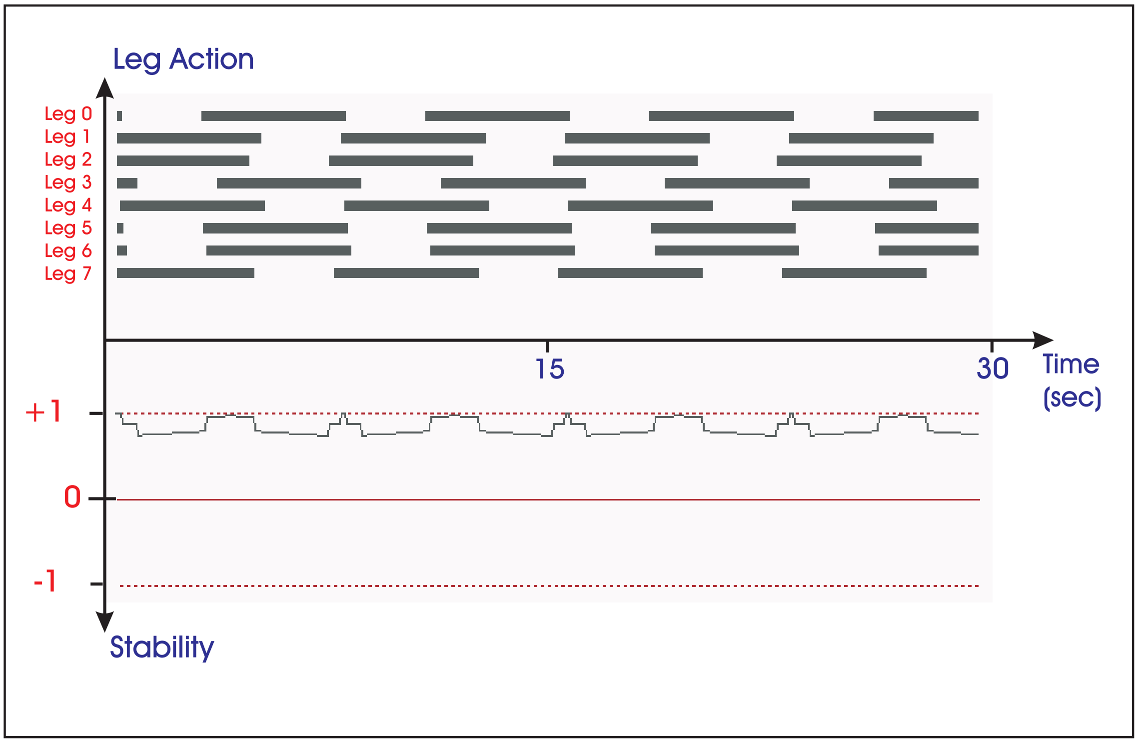


Figure 9: GA-generated walking gait for normal walking on a flat surface.

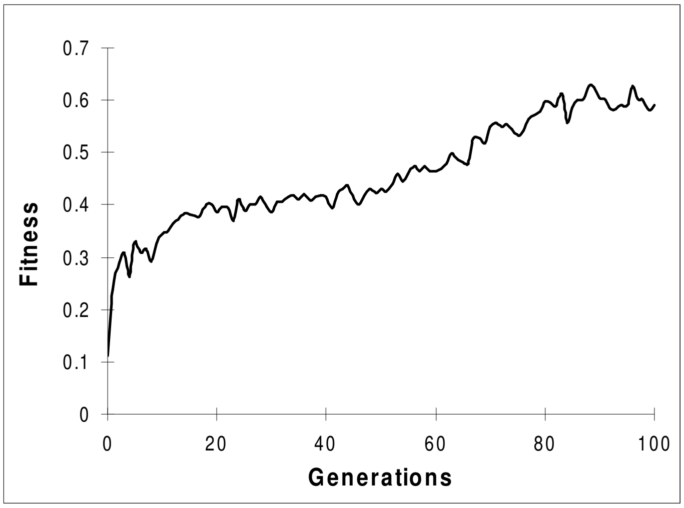


Figure 10: Evolution statistics for level walking.

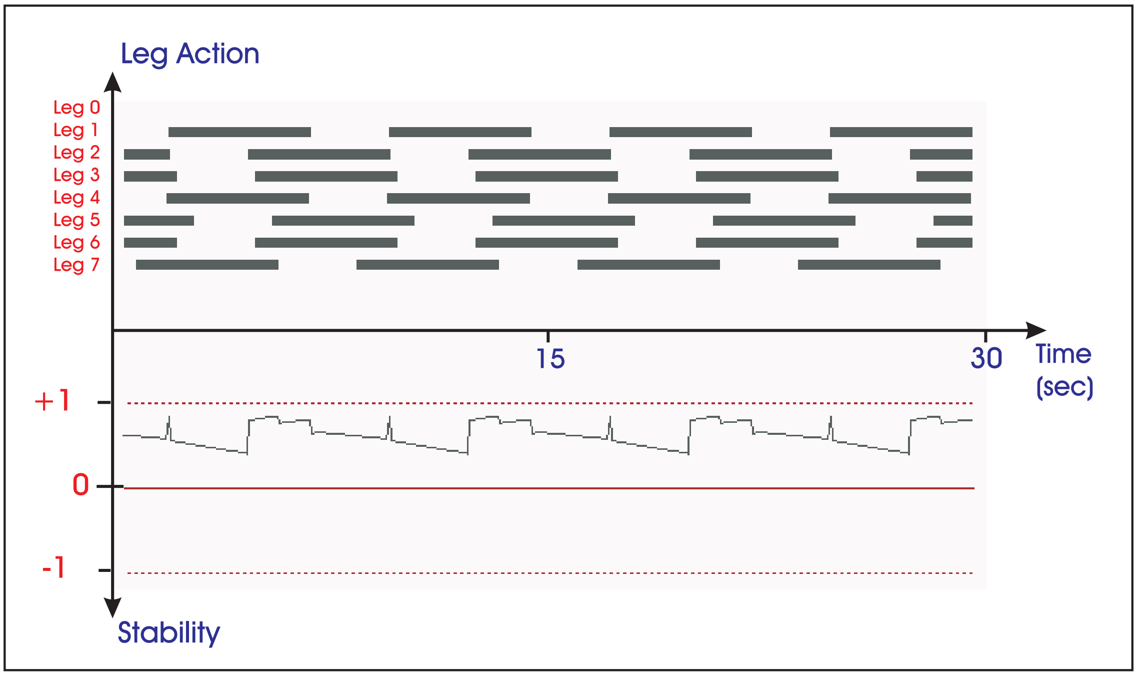


Figure 11: GA-generated walking gait for when one leg is made inoperative.

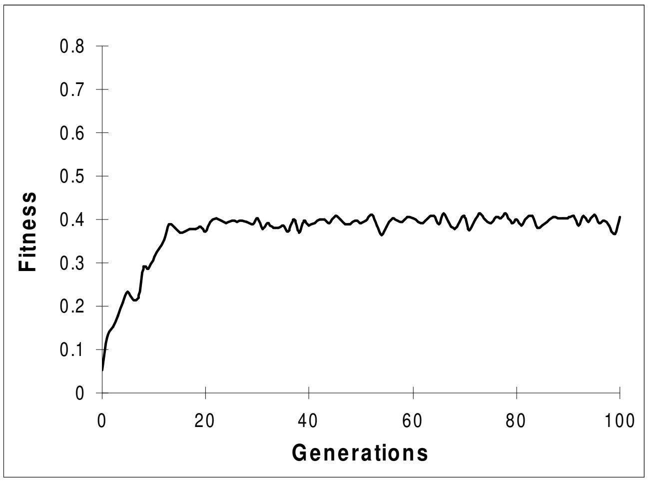


Figure 12: Evolution statistics for when one leg is made inoperative.