Second Asian Symposium on Industrial Automation and Robotics
BITEC, Bangkok, Thailand
May 17-18, 2001

AUTOMATIC SYNTHESIS OF ROBOT PROGRAMS FOR A BIPED STATIC WALKER BY EVOLUTIONARY COMPUTATION

Chaisukkosol CHANCHAI, Chongstitvatana PRABHAS
Department of Computer Engineering
Chulalongkorn University, Bangkok 10330. Thailand
Tel: (662) 218-6983 Fax: (662) 218-6955
42702899@student.netserv.chula.ac.th, prabhas@chula.ac.th

Abstract: *This work proposes a method to synthesis programs by Evolutionary Computation (EC). The complicated problem of biped robot walking is chosen to test the proposed method. Walking motion is divided into six stages and the evolution is carried out stage-by-stage. The locomotion is restricted to forward walking on the flat and smooth surface with static balance. The synthesis process composed of both simulation and the experiment with a real robot. The result of the experiment shows that various types of gaits are achievable and the biped walking is stable.*
Keywords: *Evolutionary computation, biped robot walking, static balance walking.*

1. INTRODUCTION

Automatic programming for a robot to achieve a task has been a long term goal of robotic research community. Programming a robot by human is difficult and error prone. Modern robots are very complex, some robot has sophisticated mechanisms that enable it to perform human tasks such as the humanoid robot P2 by Honda (Hirai, 1998). The limitation of using these robots is the difficulty in programming them to achieve a desired task.

Evolutionary Computation is a family of algorithms some of which can produce solutions in the form of "programs". It is applicable to robot problems. Many works have been demonstrated, for example, (Koza, 1992; Davidor, 1990; Polvichai, 1996). Evolutionary Computation can be regarded as a weak search method. It is effective for a wide range of problems such as symbolic regression, job scheduling, robot control and so on.

Evolutionary Computation is a search method based on population. A number of candidate solutions are evolved generation by generation to converge to a final solution. The search is guided by the measure of goodness of candidate solutions, called "fitness function" which is defined for a particular problem to be solved.

Many robot program synthesis problems that have been attempted using evolutionary computation are the problems that have low number of degree of freedom and mostly are the work in simulation. This is because of the high cost of computation and the huge number of candidate solutions to be evaluated. It is well known that transferring the solution from simulation to the real world is not very successful (Brooks, 1991a). Many aspects of the real world can not be sufficiently simulated. To improve the success rate, the real world should be engineered such that the simulation can predict the effect in the real world with high degree of accuracy. This is difficult if not impossible in many tasks which robots are intended to be used.

This work proposes to synthesis programs for a biped static walker. This task is chosen because it contains high degree of freedom. A walking robot is interesting because it can travel in many terrain that are not accessible to a typical wheel-based mobile robot. A biped robot is also more appropriate in the area that is constructed for human, such as in a car, in a tunnel, on an elevator etc. A biped is deemed to be more difficult to control than a multilegged robot as it has to perform balancing with minimum degree of redundancy. Genetic Algorithm is used to synthesis programs. The walking task is divided into stages and the program is synthesized stage-by-stage. In each stage, the solutions from simulation are validated using the experiment in the real world. These validated solutions become the initial state of the next stage of the synthesis. This is the key to improve the transferring of solutions form simulation to the real world. The subsequent sections explain the proposed method in more details.

2. RELATED WORKS

There are many researches on generating robot programs. Genetic Algorithms (GA) and Genetic Programming (GP) two of the most popular methods in Evolutionary Computation are widely used. Hirai (1998) developed a humanoid robot that has full body, head arms, and legs. It could walk perfectly like human, it could walk up and down the staircase, turn left and right, and walk on any surface. This robot, however, used manual programming. Polvichai (1996) demonstrated the automatic generating of robot programs by using Genetic Programming (GP). The robot is 3-joint arm moving in two dimensions. Experiments were performed in simulation, and the results were validated in a real robot.

There are many works on generating robot program to control biped locomotion. Zheng, et al, (1988) developed biped walking from a level surface to sloping surfaces with positive gradients. Inaba, et al, (1995) constructed an ape-like biped that can walk with static balance. Kun and Miller (1997) applied neural network to perform adaptive static balance of biped walking.

Regarding works that apply EC to solve the biped walking problem. Cheng (1995) developed a walking robot with dynamic balance. In this work, GA is applied to search for control gains and nominal trajectory for a 5-link biped locomotion. The aim is to walk in different constraints, such as, walking on an incline surface, walking at a high speed, and walking with a specified step size. The biped is experimented in simulation. Rodrigue (1996) used GA to find the minimum torque that is necessary for walking. The fitness function is defined as robot posture similarity between the ideal posture and the actual posture. The experiment is also performed in simulation. Arakawa (1996, 1997) focused on using GA to produce a natural motion trajectory and optimize walking energy. The learning system was performed in simulation and the result was confirmed by the real robot.

Most researches experimented only in the simulation. In this work, there are both simulation and real world learning included in the experiment.

3. GENETIC ALGORITHMS

Genetic Algorithms (GAs) (Goldberg, 1989) is a branch subfield of EC. A candidate solution, called ‘individual’, is a string. The process of GA is shown in Fig 1.

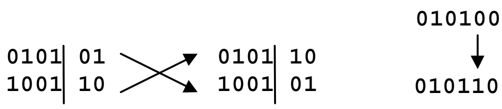
```
Initial Population
repeat
    Fitness Evaluation
    Selection
    Crossover
    Mutation
until (Terminated condition)
```

Figure. 1. Process of GAs.

First, a number of individuals, called ‘population’, are initialized as the first generation. All individuals are evaluated to find their fitness. The genetic operators—cross over, and mutation—are applied to produce the next generation. The population in the next generation is evaluated again. Selection, crossover, and mutation are executed repeatedly until the terminated condition is achieved.

Regarding selection process, there are many methods to select individuals. Generally, individuals are selected according to fitness value. That means the higher fitness individual is more likely to be selected than the lower one. Nevertheless, this method can cause all individuals to become similar to each other. In this work, we use a selection, called “Combined Rank Selection” (Winston, 1992). Using this selection, we can maintain the diversity of the population.

Crossover is an operation that exchanges some parts of an individual with another one (see Fig. 2.a). For mutation, some bits of a string are randomly changed (Fig. 2.b). The mutation occurs by a small probability, 0.001.



(a) (b)

Figure. 2. (a) Crossover
(b) Mutation

4. EXPERIMENT

The objective is to synthesis the biped robot control program automatically. This work restricts the walking task to the biped that can walk forward on the flat and smooth surface with a static balance.

The experimental biped is 25 centimeter high, and the area of sole is 4.5 * 5.0 cm^2. It has two hips, two knees, and two ankles, rotated in sagital plane (Fig. 3). The biped does not have a torso, but it has a tail moving in frontal plane. The reason of using tail instead of torso is tail will cause the biped ’s center of gravity (C.G.) positioning lower than torso, and the biped can keep its balance easier.

An individual is composed of two parts: a length, and a sequence of walking commands.

length	a sequence of walking commands

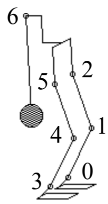


Figure. 3. biped construction

The sequence of walking commands. The form of walking command is:

m: r

where ‘m’ is motor command {0+, 0-, 1+, 1-, …, 6+, 6-}. The biped has 7 motors numbered 0-6. The signs ‘+’ and ‘-’ mean increasing or decreasing angle of the motor by ‘r’.

In the fitness evaluation, the number of walking commands before falling and rolling is stored in ‘length’ part.

Walking motion of one step is divided in to six stages (Fig. 4). GA is used to synthesize control program for each stage step-by-step, called “stage evolution” (Brooks, 1991b). With this approach, the fitness function can be set appropriately with the subgoal of each stage. Thus, the final solution can be achieved more rapidly.

The initial biped posture is standing on two feet. In the first stage, robot stands on a right foot. In the second stage, lifting the left leg and laying down in the third stage. In the forth stage, robot stands on the left foot. The fifth and sixth stage is lifting the right leg and laying down. After the final stage, the posture is adjusted to the initial posture. Therefore, the control program can be repeated to create a continuous walking.

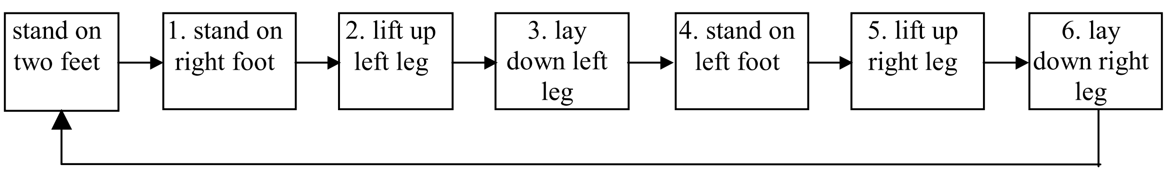


Fig. 4. six stages of walking motion.

There are two types of fitness function: general fitness function, and particular fitness function. The general fitness function consists of three variables:

$$Fit = \frac{k_1F + k_2R}{k_3\, t}$$

where F = 1 when the robot falls otherwise 0, R = 1 when the robot turns otherwise 0, T is the duration that the robot can achieve stable walk, k_1, k_2, k_3 are appropriate constants. The general fitness function promotes the behavior that is stable and walk straight forward without turning.

The particular fitness function for each stage is shown in Table. 1. Both fitness functions are minimized function.

Stage	Fitness function
1	distance between C.G. and center of right foot.
2	distance of left foot forward from right foot.
3	height of left foot from ground.
4	distance between C.G. and center of left foot.
5	distance of right foot forward to left foot.
6	height of right foot from ground.

Table. 1. particular fitness function for each stage

The experiment with an actual robot takes a very long time. During the evolution, the robot must perform a very large number of trial and error movements. To avoid this limitation, we use a simulation. The experiment with the real robot is performed to validate the solutions at the end of each stage. Human observation is used to validate the behavior of the robot.

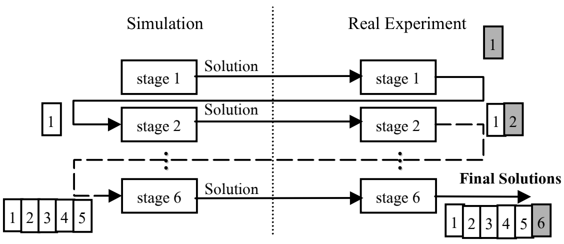


Fig. 5. Learning system.

A number of solutions of each stage from simulation are selected to evaluate in the behavior of the real robot of that stage (Fig. 5). The validated result becomes an initial state of the next walking stage. After six stages, the total solutions will emerge.

GAs parameters are shown in Table 2.

population size	500
generation	200
crossover probability	1.0
mutation probability	0.001

Table. 2. GAs parameters

5. RESULT

The robot can walk continuously more than 15 steps, with the speed 40 second per step. Figure 6. shows an example of a full step. Figure 7 shows the movement of C.G. during a one-step walk. The solid line shows the foot that is on the ground. The dotted line shows the foot that is lifted. This figure is drawn from the simulation result. It can be seen that the stability of biped locomotion is marginal, especially in stage 2 – 4.

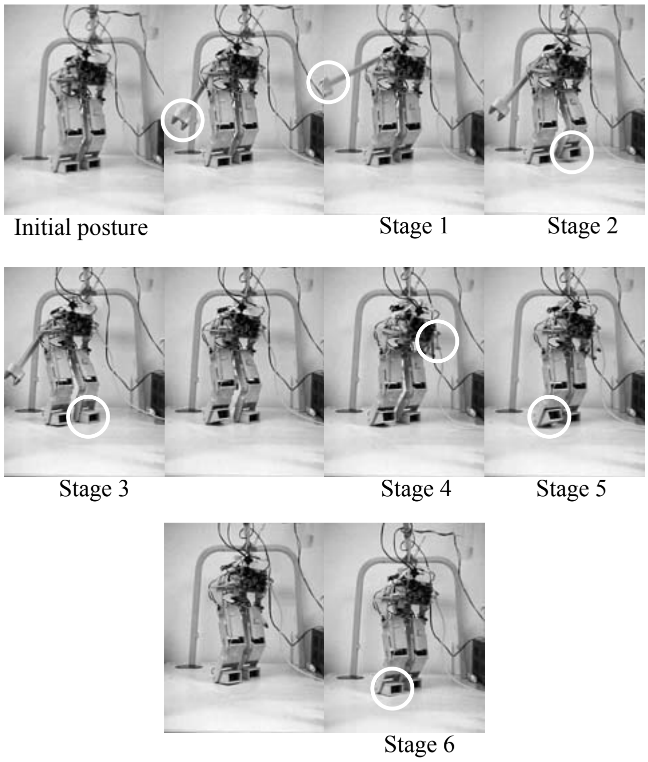


Fig. 6. Result.

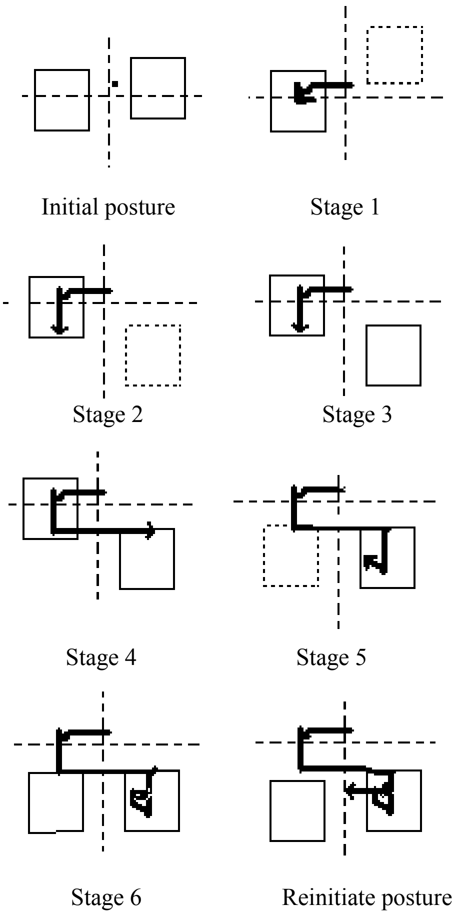


Figure. 7. Movement of C.G.

At the end of each stage in simulation, 20 individuals are selected to be validated in the real experiment. An average number of successful individual in the real experiment of each stage is 7. We found that even without general fitness function, the final solution could still be achieved. Moreover, the stable duration term in general fitness function causes the unnecessary movement.

The forth stage is the most difficult stage to evolve. As the robot must transfer its weight to another foot, especially when the length of stride is large. The length of stride is determined by the fitness function in the second stage. Sometimes, the unexpected behavior emerges in the forth stage such as moving the leg backward before weight transfer.

6. CONCLUSION

In this work, we investigate a method to generate control program for a walking biped. Walking motion is divided into six stages. GA is used to synthesize the robot control program stage-by-stage. The fitness function can be set differently and appropriately in each stage. This work uses simulation combined with the experiment in a real robot. The results show that the real robot can achieve a stable and continuous walk.

The experiment in the real world is used to select and validate the result from the simulation. The cooperation between simulation and real world experiment is the key to achieve a solution that works in the real world.

7. REFERENCES

Arakawa, T., Fukuda, T., (1996), “Natural motion trajectory generation of biped locomotion robot using genetic algorithm through energy optimization”. Proc. of IEEE Int. Conf. on Systems, Man and Cybernetics, Vol. 2, Page(s): 1495 -1500

Arakawa, T., Fukuda, T., (1997), “Natural motion generation of biped locomotion robot using hierarchical trajectory generation method consisting of GA, EP layers”. Proc. of IEEE Int. Conf. on Robotics and Automation, Vol. 1, Page (s): 211 -216

Brooks, R. A., (1991a), “Artificial Life and Real Robots”, Towards a Practice of Autonomous Systems: European Conference on Artifcial Life, Paris, France, MIT Press, December, Page(s): 3-10.

Brooks, R. A., (1991b), “Intelligence without representation”, Artificial Intelligence, 47, Page(s): 139-160.

Cheng, M.-Y., Lin, C.-S., (1995), “Genetic algorithm for control design of biped locomotion”. Proc. of Int. Conf. on Intelligent Systems for the 21st Century, Vol. 2, Page(s): 1315 -1320

Davidor, Y., (1990), “Robot programming with a genetic algorithm”, Proc. of IEEE Int. Conf. on Computer Systems and Software Engineering, Page(s): 186 -191

Goldberg, D. E., (1989), “Genetic Algorithm in Search, Optimization, and Machine Learning”, Addison Wesley.

Hirai, K., Hirose, M., Haikawa, Y., Takenaka, T,. (1998), “The development of Honda humanoid robot”, Proc. of IEEE Int. Conf. on Robotics and Automation, Vol. 2, Page(s): 1321 -1326

Inaba, M., Kanehiro, F., Kagami, S., Inoue, H., (1995), “Two-Armed Bipedal Robot that can Walk, Roll Over and Stand up”, Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. 'Human Robot Interaction and Cooperative Robots', Vol. 3, Page(s): 297 – 302.

Koza, J. R., and J. P. Rice., (1992)., “Automatic programming of Robots using Genetic Programming”, in AAAI-92 Proc. 10th National Conf. on Artificial Intelligence, Page(s): 194-201.

Kun, A. L., Miller, W. T., (1997), "Adaptive Static Balance of a Biped Robot Using Neural Networks," Proc. of the IASTED Int. Conf. on Robotics and Manufacturing, Cancun, Mexico, May 29-31, pp. 245-248,

Polvichai, J. (1996)., “Learning of a robot arm by genetic programming”, Master Thesis, Chulalongkorn University.

Rodrigues, L., Prado, M., Tavares, P., Da Silva, K., Rosa, A., (1996), “Simulation and control of biped locomotion-GA optimization”, Proc. of IEEE Int. Conf. on Evolutionary Computation, Page(s): 390 – 395

Winston, P., (1992)., “Artificial intelligence”, Reading MA: Addison-Wesley, Page(s): 505-528.

Yuan F. Zheng., Jie Shen., Fred R. Sias, Jr., (1988), “A motion control scheme for a biped robot to climb sloping surfaces”, Proc. IEEE Int. Conf. on Robotic and Automation, Vol. 2, Page(s): 814-816.