Situated and Embodied Evolution in Collective Evolutionary Robotics

Yukiya USUI Takaya ARITA
Graduate School of Human Informatics, Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-8601, Japan
E-mail :{yukiya@create, ari@info}.human.nagoya-u.ac.jp

Abstract

Evolutionary robotics is a challenging technique for creation of autonomous robots based on the mechanism of Darwinian evolution. In the conventional evolutionary robotics, the "simulate-and-transfer" method has been adopted. We believe that the most likely candidate methodology in evolutionary robotics for near future is "Situated and Embodied Evolution", in which real robots in real world evolve based on the interactions with actual environment and real robots. It becomes important when realizing Situated and Embodied Evolution to decentralize the algorithm for evolution computation, because it could make implementation of efficient systems easier and could accelerate diversification in robot behavior. This paper proposes a distributed and asynchronous genetic algorithm for flexible and efficient robotic systems that realize Situated and Embodied Evolution. This paper also reports on the performance of Situated and Embodied Evolution based on the results of the preliminary experiments on the robotic system we have implemented.

Keywords: Situated and embodied evolution, Evolutionary robotics, Genetic algorithm.

1 Introduction

Evolutionary robotics is a challenging technique for creation of autonomous robots based on the mechanism of Darwinian evolution [1]. In the conventional evolutionary robotics, the "simulate-and-transfer" method has been adopted (Figure 1(a)). However, several issues are increasingly problematic for the method.
1) It is very difficult or takes long time to simulate complex behavior of robots and complex environment.
2) It is necessary to model the environment every time when a new task is given.
3) Scalability to the number of the robots is poor in case of the systems with a population of robots having

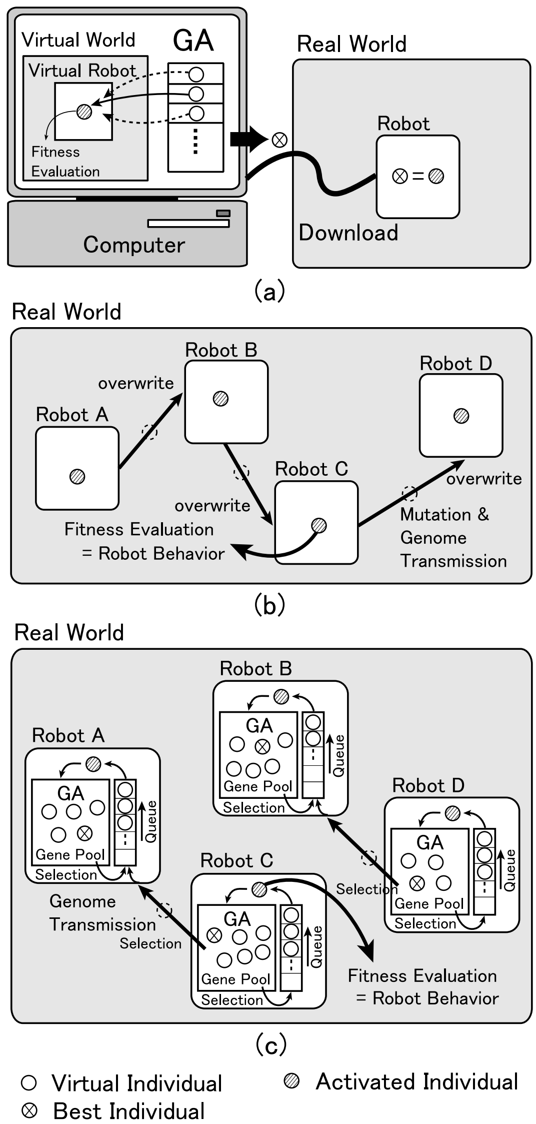

Figure 1: Schematic diagram for evolution of robots: (a) "Simulate-and-transfer" method, (b) "1 genome per robot" method, (c) Proposed method.

complex interaction among them.

We believe that the most likely candidate methodology in evolutionary robotics for near future is “Situated and Embodied Evolution”, in which real robots in real world evolve based on the interactions with actual environment and real robots. It becomes important when realizing Situated and Embodied Evolution to decentralize the algorithm of evolution (genetic algorithm), because decentralization of evolutionary computation could make implementation of efficient systems easier and could accelerate diversification in robot behavior.

Very few studies regarding Situated and Embodied Evolution have been conducted. Among them, Watson et al. proposed a method to realize Situated and Embodied Evolution based on a motivation that is similar to ours described above [2]. They adopted a straightforward method for evolutionary computation, in which each robot represents one individual and population share their genetic information by transmitting information among them when they encounter (Figure 1(b)). However, the progress of evolution in this method directly depends on the number of the robots and the frequency of encounter with other robots.

This paper proposes a distributed and asynchronous genetic algorithm for flexible and efficient robotic systems with adequate scalability that realize Situated and Embodied Evolution. There are two levels of optimization in this method (Figure 1(c)). There is transmission of good genes among robots when they encounter. Also, each robot executes a genetic algorithm within itself by emulating many “virtual individuals” based on time-sharing. This method thus reduces dependence of the number of the robots and of the frequency of encounter with other robots on the speed of evolution, which can realize flexible and efficient robotic systems with adequate scalability. This paper also reports on the performance of Situated and Embodied Evolution based on the results of the preliminary experiments on the robotic system we have implemented.

2 A Model for Situated and Embodied Evolution

Parallelization of genetic algorithms (GA) has been discussed in the field of evolutionary computation, motivated mainly by the desire to reduce the overall computation time. Most of the proposed parallel GAs fall into a class which has come to be called “island model” parallel GA. Island model parallel GA divides a population into subpopulations and assigns them to processing elements on a parallel or distributed computer. Then each subpopulation searches the optimal solution independently, and exchanges individuals periodically.

Our distributed genetic algorithm for Situated and Embodied Evolution can be called island model parallel GA in that each robot has a subpopulation, searches the optimal solution, and exchange good individuals. However, there is a significant difference in our model from the conventional island model parallel GA as follows.

1) Communication topology and frequency are dynamic, which depends on the robot behavior, especially encounters of robots.

2) Fitness evaluation is conducted as robot behavior in real word, which needs quite long time compared with other evolutionary operations done in the robots.

3) Optimal solution varies depending on the behavior range and physical characteristics of each robot, besides the dynamic property of the environments.

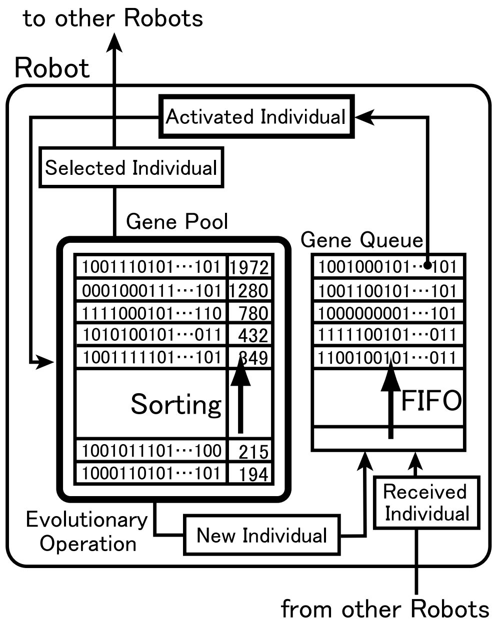

Figure 2: Evolutionary processing in each robot.

Typical implementation of evolutionary computation in each robot is shown in Figure 2. Each robot has a “gene pool” and a “gene queue”. The gene pool has an evolved subpopulation, whose individuals are expressed as genomes and sorted by their fitness values. A new individual is generated by selecting (copying) 2 individuals from the gene pool based on roulette wheel selection, and operating one point crossover and mutation. The new individual is then put into the gene queue, and waits for being evaluated. New individuals migrated from other robots are also put into the gene queue, and reevaluated in this robot, because there can be difference in their environments and physical characteristics among robots. A dequeued individual is loaded to specify the robot behavior, and after a given length of time, it is attached with the fitness value, and stored into the gene pool. The gene pool has a limited capacity, and therefore the evaluated individuals will be discarded if their fitness values are lower than the one of the worst individual in the gene pool. This mechanism realizes time sharing among many virtual individuals in each robot.

Migration procedure runs independently of the above-described GA process in each robot. An individual to be transmitted is selected (copied) from the gene pool based also on roulette wheel selection asynchronously. Each robot has a chance to send its selected individual every predefined time interval, the timing of which is randomly decided every event. The robot sends a selected individual with following probability which depends on its fitness value.

$$If(A <= C)\{ \quad P = 50(\frac{C-A}{M-A}) + 50 \quad \} \tag{1}$$

$$else \;\; if(A > C)\{ \quad P = 50(\frac{C-S}{A-S}) \quad \} \tag{2}$$

(P: Probability of transmitting, A: Average fitness, M: Maximum fitness, S: Minimum fitness, C: Fitness of the selected individual)

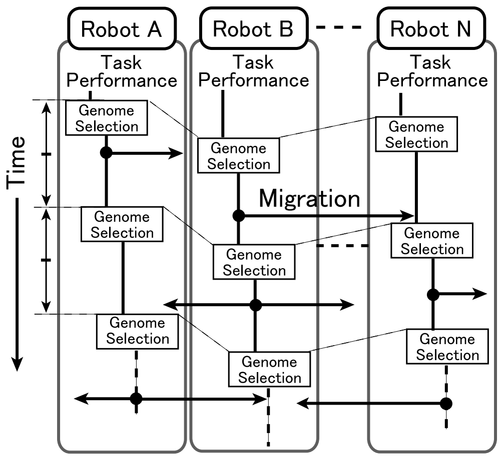

Figure 3: Asynchronous migration of individuals.

3 Preliminary Experiments

We have implemented a minimum experimental robotic system for the purpose of evaluating the proposed scheme described in the previous section. We used six Khepera miniature mobile robots (Figure 4). The issue of power supply also becomes important when realizing the Situated and Embodied Evolution paradigm. Our solution in the preliminary experiments is to adopt a power supply mechanism by which each robot moves around in a floor-and-ceiling structure and receives power continuously from a pantograph located on top of it (Figure 5). Also, a charging battery built in each robot backs up the mechanism. Infra-red communication is used for transmission of individuals between robots.

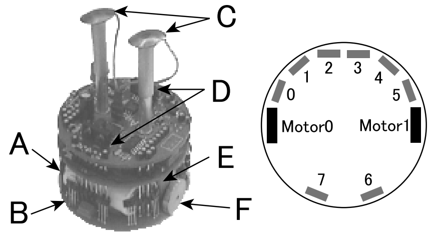

Figure 4: Khepera robot (Left - Khepera+IR communication turret A: Battery, B: Infra-red sensor, C: Pantograph, D: Infra-red emitter/receiver, E: Incremental DC motor, F: Wheel, Right - Layout of 8 infra-red sensors).

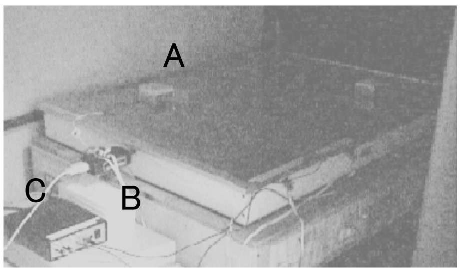

Figure 5: Experimental setup (A: Continuous power supply, B: Infra-red emitter/receiver unit, C: Power supply).

We adopted a simple two-layer neural network to control the behavior of each robot (Figure 6). The structure of the neural network (connection weights and the thresholds) was evolved by distributed genetic algorithm described in the previous section. In the neural network, 7 input nodes corresponded to 6 sensor inputs and a threshold, each of which was expressed by 5 bit genome information. There were 2 output nodes corresponding to right and left motor outputs. So, the length of the genome was 70 bits.

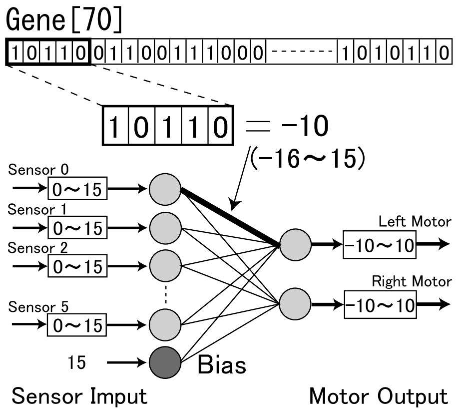

Figure 6: Relation between individual information and the structure of neural network.

Robot control programs (neural networks) for an avoidance task were evolved in the preliminary experiments for the purpose of confirming the effectiveness of the model. Each individual is evaluated by the length of movement without hitting the walls or other robots as follows:

$$fitness \mathrel{+}= SensorOff()(R\_Motor + L\_Motor), \quad (3)$$

where *SensorOff*() returns 0 when two or more infra-red sensors are activated, and otherwise returns 1, and *R_Motor* and *L_Motor* correspond to the rotation speeds of the right and left motors. Fitness value of each robot is increased at predefined time intervals when there is no input from eight infra-red sensors (which means that there are no obstacles in the neighborhood of the robot) and at least one motor is activated.

The result is shown in Figure 7, where the horizontal axis represents the total number of evaluated individuals, and the vertical axis represents the fitness. This graph shows the typical results of following 4 cases: the cases when the size of the gene pool is 1 ("1 genome per robot method"), 5 and 10, and the case in which there is no migration among robots and the size of the gene pool is 5. Average fitness of every 20 individuals is plotted in each case. We can see from this figure that the case with gene pool size of 5 with migration shows the best performance though the optimal size depends at least on the number of robots and the given task. It is also shown that migration has a large role in this scheme. In general, there can be an unexpected difference in behavior among robots in real world. Migration of good individuals can improve the performance of the robots with unevolved gene pool.

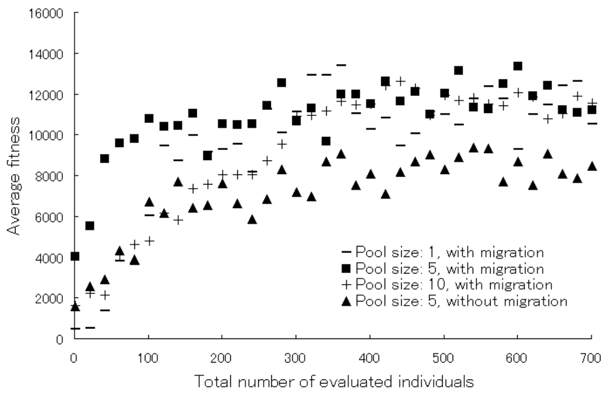

Figure 7: Evolution of robot behavior (Number of robots: 3, Mutation rate: 2%).

4 Conclusion

We have proposed a distributed and asynchronous genetic algorithm for flexible and efficient robotic systems with adequate scalability that realize Situated and Embodied Evolution. We have also reported on the performance of Situated and Embodied Evolution based on the results of the preliminary experiments on the robotic system we have implemented. Further experiments will include investigation of performance evaluation of the scheme targeting at more practical tasks.

References

[1] Stefano Nolfi and Dario Floreano, *Evolutionary Robotics*, MIT Press, 2000.

[2] Richard A. Watson, Sevan G. Ficici, and Jordan B. Pollack, "Embodied Evolution: Embodying an Evolutionary Algorithm in a Population of Robots," *1999 Congress on Evolutionary Computation*, IEEE Press, 335-342, 1999.