The Thirteenth International Symposium on Artificial Life and Robotics 2008(AROB 13th '08),
B-Con Plaza, Beppu, Oita, Japan, January 31-February 2, 2008

2D artificial life system using network-type assembly-like language: Influence of change in environment with costs of instructions

Yuhki SHIRAISHI and Fumiaki TAKEDA
Research and Development Center of Intelligent Recognition System
Research Institute of Kochi University of Technology
185 Miyanokuchi, Tosayamada-cho, Kami-shi, Kochi, 782-0034, JAPAN
{shiraishi.yuhki, takeda.fumiaki}@kochi-tech.ac.jp

Abstract

To unlock the full potential of evolution in digital media, a Tierra-like system using network-type assembly-like language has been constructed. In the system, like Avida, digital creatures, self-replicating programs, live in a discrete 2D torus space and an interaction between creatures is restricted locally. Bearing a genetic network in mind, network structure are introduced. In the previous works, it is shown that the possibility that the network-type model has more potential of evolution than a linear-type model like Avida in a simple environment.

In this paper, to study the potential of evolution more precisely, we model the effects of environment other than creatures. As one of the simplest models of such environment, the costs of execution, i.e. executing time, are introduced. The difficult environment to live in costs high to execute instructions, the easy environment does low. In computer experiments, we have investigated the influence of change in environment by analyzing the process of evolution and diversity of the system. Experimental results show that the network-type system keeps more stable diversity than the linear-type system does, even when the environment changes drastically. This indicates the possibility that the network-type system has more potential of evolution than the linear-type system does.

Keywords: Self-replication, Evolvability, Network structure, Diversity, Redundancy, Intron

1 Introduction

In the early work of Artificial Life, T. Ray has created Tierra motivated by a desire to observe the evolutionary process in media other than carbon chemistry. Evolutionary dynamics of digital creatures, self-replicating programs, were investigated in Tierra, and evolutionary processes and complex symbiotic relationships between creatures were observed [1, 2]. However, they have reached an evolutionary stable state, i.e. the number of species does not increase quickly and explosively as it does in the Cambrian period. This indicates that the system does not have the same potential for evolutionary innovation as physical systems. To unlock the full potential of evolution in digital media is considered as an open problem in Artificial Life [3].

To solve the problem, one of the authors have developed and investigated a system using network-type assembly-like language [4, 5, 6]. In the system, like Avida created by C. Adami [7],digital creatures, self-replicating programs, live in a discrete 2D torus space and an interaction between creatures is restricted locally. Bearing a genetic network in mind, network structure are introduced to the system. As shown in Figure 1, in the system, each grid in the space has a virtual CPU, a local memory, three registers, a flag, and a stack; and each grid has one creature at most. A memory cell, or *node*, has an instruction and two values, the addresses of the node executed next step, which are used to execute network-type assembly-like programs. In the previous works, it has been shown the possibility that the network-type model has more potential of evolution than a linear-type model like Avida in a simple environment, as the diversity of the network-type system is greater than that of the linear-type system in a computer experiment [6] and the network specific structure is utilized for evolution [4, 5].

In general, evolution can be considered as a process of adaptation for environment. Therefore, environment plays a key role as with creatures themselves in evolution. However, in the system, the world contains only creatures, and the creatures competed against each other only for space for living. Thus, to study the potential of evolution more precisely, we model the effects of environment other than creatures. As

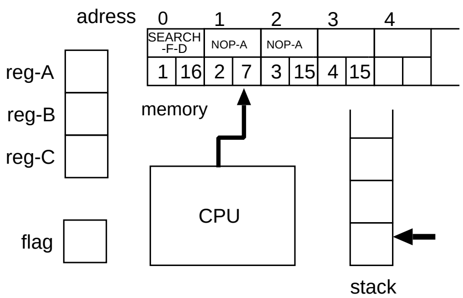

Figure 1: Structure of the virtual CPU in the network-type assembly model.

one of the simplest models of such environment, in this paper, we have introduced the costs of execution, i.e. executing time, each instruction into the system. The difficult environment to live in costs high to execute instructions, the easy environment does low. The variety of the distribution of costs among instructions means that of environment. In computer experiments, we have investigated the influence of change in environment by analyzing the process of evolution and diversity of the system comparing with the linear system.

2 Brief System Description

Figure 1 shows that a node has an instruction and other two values that are the addresses of the node to be executed next step. The method of an exact execution is as follows:

step 1. execute the instruction at memory address 0.

step 2. if `flag` equals 0,

step 2-1. move to the node whose address is written at lower left in the current node modulo the *size* (the number of nodes) of the creature.

step 2-2. otherwise, move to the node at lower right modulo the size of it.

step 3. execute the instruction of the current node.

step 4. jump to step 2.

The above process is repeated until a creature die out. When it copies itself to one of the 4-neighborhood cells (up, down, left, and right); (1) if there is no space, i.e., all four cells are already occupied by creatures, the oldest creature is killed and the new creature is copied to this cell, (2) otherwise, it is copied to one of the free cells selected randomly.

Table 1: Instruction sets.

Network	Linear
NOP-A NOP-B NOP-C	NOP-A NOP-B NOP-C
IF-EQU IF-EQU-MOD IF-NOT-EQU IF-NOT-EQU-MOD	IF-NOT-EQU
JUMP-F	JUMP-F JUMP-B
INC DEC	SHIFT-R SHIFT-L INC DEC
PUSH POP	PUSH POP
ADD SUD NAND	ADD SUB NAND
ALLOCATE DIVIDE	ALLOCATE DIVIDE
COPY-E COPY-N1 COPY-N2	COPY
SEARCH-F SEARCH-F-D	SEARCH-F SEARCH-B
FLIP-FLAG	

The instruction sets[1] and ancestor creatures are listed in Table 1 and Table 2.

As shown in Table 2, the network-type assembly creature have *introns*, which are unused next nodes in execution. Network-type system's creatures have introns naturally but ordinary linear-type system's don't have ones. This realizes the diversity of *species*[2] in the network system. It is already confirmed that the introns are utilized for evolution [4, 5].

3 Experiments

In this paper, to investigate the influence of change in environment, the distribution of the costs of instruction execution is changed randomly in a finite range every constant period. At the first *update*[3], a man-made ancestor (listed in Table 2) is set to a cell, where the world's size is 128×128, copy mutation rate is 0.005, divide mutation rate is 0.05, and *time-slice*[4] is 30.

After a few thousand updates, the size of dominant species decreased from 20 to 16, because, in general, the smaller the size of creature is, the faster the creatures self-replicate.

In this experiment, every 5,000 updates, the environment is changed drastically, the distribution of the costs is changed randomly between 1 and 30. Figure 2

[1] The details of the instructions are shown in [4, 5].

[2] In this paper, when two creatures have the same genes but introns, the two creatures belong to the same species; as for the network assembly, the values of next nodes are also same modulo the size of them.

[3] A artificial unite of time in the world.

[4] The number of instructions a creature execute every update.

Table 2: Ancestor creatures in network model and in linear model ("*": don't care).

Network				Linear
node no.	**Inst.**	next node flag=0	next node flag=1	**Inst.**
0	SEARCH-F	1	16*	SEARCH-F
1	NOP-A	2	7*	NOP-A
2	NOP-A	3	15*	NOP-A
3	ADD	4	15*	ADD
4	INC	5	18*	INC
5	ALLOCATE	6	3*	ALLOCATE
6	PUSH	7	6*	PUSH
7	NOP-B	8	15*	NOP-B
8	POP	9	5*	POP
9	NOP-C	10	11*	NOP-C
10	POP	11	9*	SUB
11	NOP-B	12	12*	NOP-B
12	COPY-E	13	13	COPY
13	COPY-N1	14	14	INC
14	COPY-N2	15	15	IF-NOT-EQU
15	INC	16	16	JUMP-B
16	IF-EQU	17	12*	NOP-A
17	DIVIDE	18	7*	DIVIDE
18	NOP-B	19	10*	NOP-B
19	NOP-B	5	19*	NOP-B

shows the time series of the number of creatures with dominant species. It is confirmed that while the fluctuations in the number of the linear system is very high, that of the network system is low. Figure 3 also shows the time series of Shannon's diversity:

$$H = -\sum_i p_i \log p_i,$$

where p_i is the proportional abundance of species i.Table 3 shows the mean and standard deviation of Shannon's diversity during 100,000 updates. These results show that the network-type assembly keeps more stable diversity than the linear-type assembly does.

Another experimental results are shown in Figure 4 and 5, and Table 4. In this case, the environment is changed not too drastically, the distribution of the costs is changed randomly between 1 and 20. These results also show that the network-type assembly keeps more stable diversity.

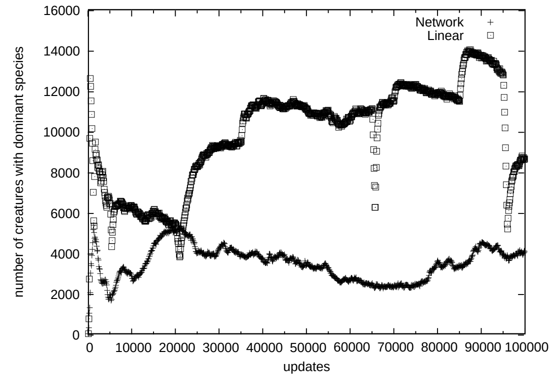

Figure 2: The number of creatures of dominant species v.s. updates: the distribution of the costs is changed randomly between 1 and 30 every 5,000 updates.

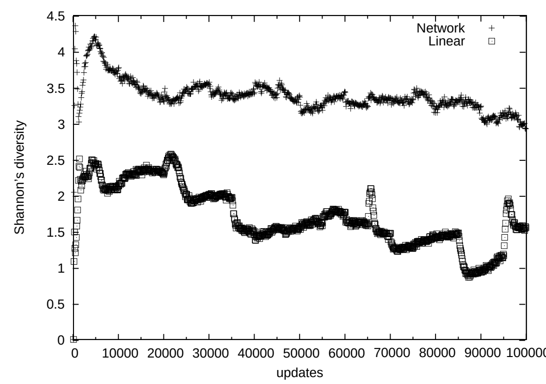

Figure 3: Shannon's diversity v.s. updates: the distribution of the costs is changed randomly between 1 and 30 every 5,000 updates.

Table 3: The mean and standard deviation of Shannon's diversity: the distribution of the costs is changed randomly between 1 and 30 every 5,000 updates.

	mean(m)	standard deviation(σ)
Network	3.38	0.24
Linear	1.71	0.42

4 Discussion

Experimental results shows that (1) the fluctuations in the number of the network system is lower than that of the linear system; and (2) the network-type system keeps more stable diversity than the linear-type system does, even when the environment have changed drastically. These mean that the creatures in the network system have high adaptability for the change of environmental conditions. Therefore, the creatures in the network system are supposed to continue to evolve without extinction. This indicates the possibility that

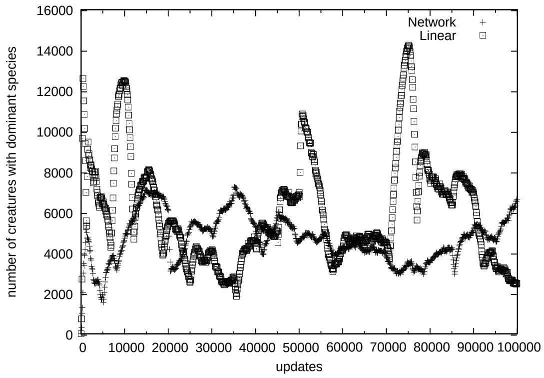

Figure 4: The number of creatures of dominant species v.s. updates: the distribution of the costs is changed randomly between 1 and 20 every 5,000 updates.

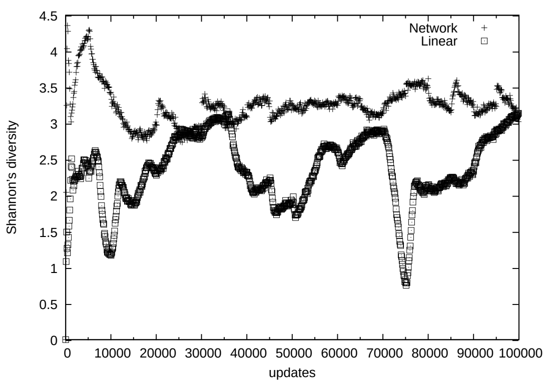

Figure 5: Shannon's diversity v.s. updates: the distribution of the costs is changed randomly between 1 and 20 every 5,000 updates.

Table 4: The mean and standard deviation of Shannon's diversity: the distribution of the costs is changed randomly between 1 and 20 every 5,000 updates.

	mean(m)	standard deviation(σ)
Network	3.25	0.28
Linear	2.36	0.49

the network-type system has more potential of evolution than the linear-type system does.

5 Conclusions

We have shown that, in computer experiments, the network-type system keeps more stable diversity than the linear-type system does, even when the environment have changed drastically. This indicates the possibility that the network-type system has more potential of evolution than the linear-type system does.

References

[1] T. S. Ray (1995), An Evolutionary Approach to Synthecic Biology: Zen and the Art of Creating Life, Artificial Life: An Overview, The MIT Press, pp.179-209

[2] T. S. Ray and J. F. Hart (2000), Evolution of Differentiation in Multithreaded Digital Organisms, Proceedings of the Seventh International Conference on Artificial Life (Artificial Life VII), The MIT Press, pp.132-140

[3] M. A. Bedau, J. S. McCaskill, N. H. Packard, S. Rasmussen et al. (2000), Open problems in Artificial Life, Artificial Life, vol.6, no.4, The MIT Press, pp.363-376

[4] Y. Shiraishi, K. Hirasawa, J. Hu, and J. Murata (2002), The Basic Study of Artificial Ecosystem Models Using Network-type Assembly-like Language, Proceedings of Genetic and Evolutionary Computation Conference (GECCO 2002), Late Breaking Papers, pp.412-418

[5] Y. Shiraishi, K. Hirasawa, J. Hu, and J. Murata (2004), The Basic Study of Artificial Ecosystem Models Using Network-type Assembly-like Language (in Japanese), IEEJ Trans. EIS, vol.124, no.2, pp.418-424

[6] Y. Shiraishi, K. Hirasawa, J. Hu, and J. Murata (2003), 2D Artificial Life System Using Network-type Assembly-like Language: A Comparative Study with Linear-type Assembly-like Language, Proceedings of the Eighth International Symposium on Artificial Life and Robotics (AROB 8th), pp.669-672

[7] C. Adami and C. T. Brown (1994), Evolutionary Learning in the 2D Artificial Life System "Avida", Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems (Artificial Life IV), The MIT Press,