Spacial Representation for Artificial Chemistry Based on Small-World Networks

Hideaki Suzuki
ATR Network Informatics Laboratories, Kyoto 619-0288 Japan
hsuzuki@atr.jp

Abstract

A method to simulate molecular reactions with a graph whose nodes represent molecules or atomic modules and whose edges represent the bond/contact relation is proposed. The graph is updated by two actions: passive rewiring of contact edges that gives the graph a small-world property and active rewiring by active nodes' programs that rewire bond and contact edges. As examples, the replication of a chain molecule and the partitioning of a network are successfully simulated.

Introduction

One of the most important properties of artificial life models is spatial representation. The information space of a modern computer – the sequence of data words addressed with integer numbers – is far from the Euclidean liquid space wherein biological molecules move and interact. To compensate for this difference and realize molecular interactions in a computational medium, several artificial information space models have been proposed: one dimensional space of Tierra (Ray 1992; 1997) or SeMar (Suzuki 1999a; 1999b; 2000), two dimensional space of Avida (Adami and Brown 1994; Adami et al. 2000) or Amoeba (Pargellis 1996a; 1996b; 2001), lattice models by Ono, Madina *et al.* (Ono and Ikegami 1999; 2001; Madina et al. 2003), and Speroni's planar graph (Speroni di Fenizio et al. 2001). Despite having merits, these one, two, or three dimensional artificial spaces do not satisfy all of the conditions for the emergence of complex interaction between computational objects (Suzuki et al. 2003).

In a living cell, the movement of biological molecules is based on diffusion processes. After accomplishing a reaction, a molecule moves after being influenced by the forces of other molecules, collides with molecules, and finally meets its next molecule with which it will react. Such molecular movement can be modeled with mathematical expression that specifies the interval and order of molecular collisions; hence, if we were able to devise a mathematical model that has the same properties as molecular collisions, then, we would be able to simulate molecular interaction more abstractly.

Based upon this notion, we propose a new model of artificial chemistry called "Network-based Artificial Chemistry" (NAC) that represents molecular interaction by a graph whose nodes express molecules or atomic modules in macromolecules and whose edges express contact (collisions) or bonds between molecules. The graph is modeled using 'small-world' networks which include many more clusters than random networks (Watts and Strogatz 1998). While the contact edges are passively rewired according to a local rewiring rule that emulates molecular collisions in liquid, the contact and bond edges are actively rewired by the active nodes' functional programs. To test the ability of the NAC, we conducted two experiments: the replication of molecular chains by active molecules, *polymerase* and *helicase*, and the partitioning of a hydrophilic cluster by passive rewiring of hydrophilic and hydrophobic contact edges and active molecule *splitase*. The resultant graphs are visualized in figures.

After presenting the local rewiring rule for updating contact edges, the basic model for the NAC is given. Then, experimental results, closing remarks, and some conclusions are given.

Passive Rewiring Rule

Collisions between biological molecules are strongly influenced by the dimensionality and the structure of the space in which the molecules are moving. Generally speaking, the closer molecules are to each other, the more likely they are to collide with each other; hence, we can naturally assume that after colliding with molecule `B`, molecule `A` will make contact with molecules that have been in contact with `B` or other molecules that have been in contact with `A`.

The rewiring rule used in this paper is based on this conjecture. We describe the molecules' contact relation by undirected edges. At every time step, a starting node and its adjacent node, stopping node, is randomly

chosen from the network, and the edge between them is eliminated. Before eliminating it, a new stopping node is randomly chosen from nodes that have a distance of two to the starting node, creating a new undirected edge. When repeated, this rewiring process increases the cluster coefficient C of a graph while keeping the average path length L constant, creating a small-world network (Watts and Strogatz 1998). A similar rewiring procedure has already been proposed which focused on an acquaintance network in human society (Davidsen et al. 2002). A typical change of C and L by the present rule is shown in Fig. 1.

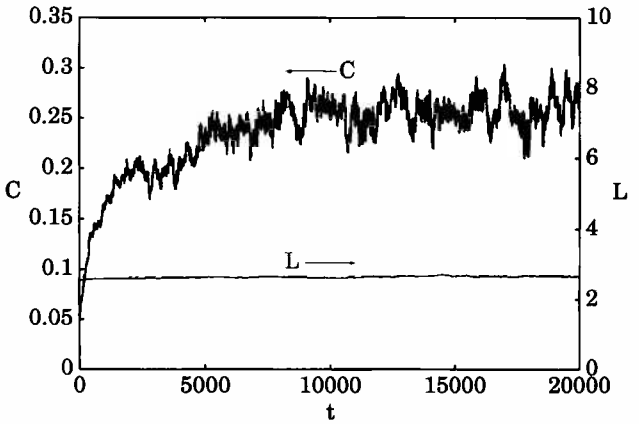

Figure 1: Cluster coefficient (C) and average path length (L) as a function of the rewiring number (t). The initial graph is random where the node number $N = 200$ and the average degree $\overline{K} = 10$. After t exceeds the edge number $N\overline{K}/2 = 1000$, C grows much larger than the random graph's theoretical value $\overline{K}/N = 0.05$, while L is kept equal to the random graph's value $\log N/\log \overline{K} = 2.3$.

Molecular Interaction in the NAC

In an actual biological cell, molecular activities are governed by four different kinds of forces: covalent, ionic, hydrogen, and van der Waals, listed in descending order of strength (Alberts et al. 1994). (Roughly speaking, a covalent bond is ten times stronger than an ionic bond, an ionic bond is ten times stronger than an hydrogen bond, and so on.) Specifically, the hydrogen bond is a key force for various interactions between molecules. The establishment of many hydrogen bonds between two macromolecules produces strong adhesion. Moreover, hydrogen bonds between water molecules and chemicals cause a hydrophilic/hydrophobic interaction and a partitioning of the solvent space.

On the analogy of these forces, we prepared four kinds of edges in the network: two kinds of directed edges for bonds, covalent (cv[ ]) and hydrogen (hy[ ]), and two kinds of undirected edges for contact, hydrophilic (il[ ]) and hydrophobic (ob[ ]). (il and ob were named after particular letters to distinguish from 'hydrophILic' and 'hydrophOBic'.) Here, cv[ ], etc. are node lists owned by each node. In the following, the rewiring of edges is always represented by the local modification of the content of the node lists. In this paper, the maximum sizes of cv[ ] and hy[ ] are always fixed at Cp = 2 and Hp = 3, respectively. (Cp is an abbreviation of 'cv pointer' whose first letter is capitalized to indicate that it stores the maximum value.) On the other hand, the maximum sizes of il[ ] and ob[ ], Ip and Op, are set to Ip = ∞ and Op = 0 for a hydrophilic node and Ip = 0 and Op = ∞ for a hydrophobic node, reflecting the 'hydro-property' of a node.

We prepared active and passive nodes in NAC. A passive node has the node lists plus a string representing its atomic constitution (sa) and an integer representing molecular status (ta). An active node has the node lists, sa and ta, plus several working registers for strings, integers, or nodes and its own program.

At every time step, the molecular graph is updated by the operation of active nodes and passive rewiring. In the active operation, every active node's program is decoded and executed. These operations not only modify its own working registers but also change the status (integer) or bond/contact edges of nearby nodes, which causes a molecular reaction or the transportation of nodes. In passive rewiring, we applied the local rewiring rule described in the previous section to the contact edges in il[ ] and ob[ ]. Although a deleted edge is randomly chosen regardless of the hydro-property of the node pair, if the hydro-properties of a created edge's node pair are in conflict with each other, the rewiring is canceled. (A hydrophilic node cannot have node information in ob[ ] and vice versa.) This forces the network to obey the rules for hydrophilic/hydrophobic interaction between molecules.

Through these operations, the bond relation in cv[ ] and hy[ ] is modified only by active operations, whereas the contact relation in il[ ] and ob[ ] is modified by both active and passive operations. This can be compared to biomolecular interaction in which molecular bonds cannot be rewired without enzymes, whereas molecular contact can be updated probabilistically by Brownian motions without catalysts.

Experiments

Replication of Molecular Chains

The elementary process of information replication in a living cell is done by complementary matching between nucleotides of DNA and RNA. In imitation, we prepared a network with a number of free hydrophilic nucleotide nodes, a polynucleotide (a chain of nucleotides)

bonded in covalent bonds cv[ ], and two kinds of hydrophilic enzyme nodes, named *polymerase* and *helicase*. Polymerase's function is to read a single strand of polynucleotide with hy[0] and polymerize nucleotides with hy[1], making a double strand composed of identical nucleotide pairs. Helicase splits a double strand of polynucleotide into single ones. See Appendices A and B for the detailed programs for polymerase and helicase. Every nucleotide node has one out of four different 'base strings' in sa. The head and tail nucleotide nodes have special status numbers in ta by which polymerase and helicase identify the head/tail of a chain.

Starting from a random network which includes these molecules, we simulated the NAC reactions using the update rules presented in the previous section. The replication process of the chain is basically the same as that of Squirm3 in (Hutton 2002); however, NAC is unique because it propels replication not by rules prepared as physical laws outside but by the programs stored in active enzyme nodes. See Fig. 2 for a typical run of the replication; after 7360 time steps, the replication of a nucleotide chain is successfully carried out.

Partitioning of the Network

A lipid is an amphipathic molecule composed of hydrophilic and hydrophobic parts, and when it agglomerates, it constitutes a sheet so that hydrophobic parts will not be in contact with water (solvent). According to recent discoveries in molecular biology (Alberts et al. 1994), the inside and outside sheets of a lipid bilayer have different phospholipids (hydrophilic parts); hence, the biological membrane is asymmetrical.

In this experiment, we described a lipid with a pair of nodes, hydrophilic and hydrophobic, bonded by cv[ ]. According to strings stored in the sa, the hydrophilic nodes of the pairs are classified as inner or outer, by which a particular hydrophobic enzyme named *splitase* actively cuts hydrophilic contact bonds between a hydrophilic node that might belong to the inside of the membrane and a hydrophilic node that might belong to the outside of the membrane. See Appendix C for the detailed algorithm of the splitase program.

To test the possibility that lipid pairs might constitute a membrane in NAC, we conducted a number of experiments with a network including free nucleotide, lipid pairs, and splitase. Starting from a random network, we updated the network with the rules described in the previous section while visualizing the structure of the network. From these experiments, we observed that the complete partitioning of a hydrophilic cluster cannot take place unless passive rewiring works on the network, the splitase program is properly designed, and the numbers of lipid pairs and splitase are appropriately adjusted. Figure 3 shows a typical successful run of these experiments where the initially-connected hydrophilic subgraph is partitioned into two isolated subgraphs after about ten thousand time steps. We can conclude from these results that under appropriate conditions, NAC can represent the partitioning of a hydrophilic region by lipid molecules through the passive rewiring that facilitates the occurrence of clusters and the active rewiring by the splitase.

Discussion

A new method of spatial representation for artificial chemistry was proposed. Using a local rewiring rule that increases clusters, NAC can describe the passive transportation of molecules by diffusion as well as molecular reactions or active transportation by enzymic molecules. The ability of NAC was demonstrated using two experiments: template replication and domain partitioning.

We discuss several conclusions and possible future extensions of NAC in the following section.

Flexibility to represent molecular movement: A graph is a purely mathematical framework that can represent a variety of molecular interactions. Being obviated from the limitation of the rigidness of previous physical space models, NAC expresses both strong and weak interactions between artificial molecules or atomic modules in a unifying manner. This might enable NAC to represent the following molecular movement or activities in a cell: partitioning a reaction domain by semi-permeable membrane, freely changing the number of symbols/molecules in a cell, mingling between the partitioned compartments, transportation of signal molecules in or between compartments, random rearrangement of symbols, and selective transportation of symbols by active molecules. These activities are equivalent to the latter half of the ten necessary conditions for an artificial environment presented in (Suzuki et al. 2003), suggesting that NAC might be a promising approach for constructing an artificial environment for the evolution of complex forms of life.

High-dimensional space representation: In 2000, the author argued that "the dimension of a computational medium is of vital importance for the richness of the functions possibly emerging in the computational system. ... The dimension determines the number of operator objects that can cooperate simultaneously. ... We cannot expect the emergence of higher functions in a computational system when only a small number of operator objects can cooperate for a single task" (Suzuki 2000). Using the graph representation, NAC allows a

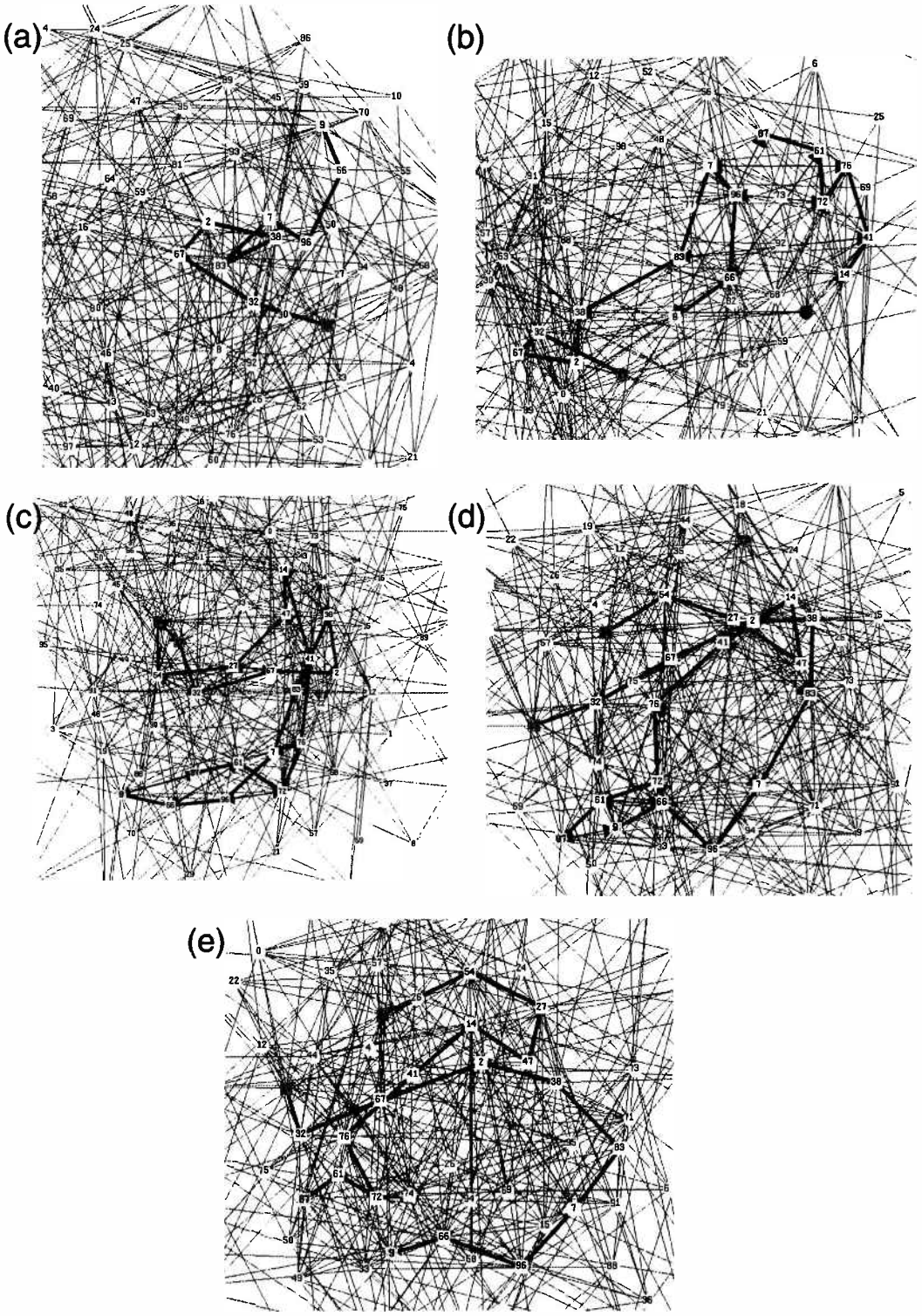

Figure 2: A typical run of the replication of a polynucleotide in a NAC cell with node number 100 and average degree 10 for hydrophilic edges. The network includes one polymerase and one helicase. (a) part of a snapshot of the initial random network at t=0, (b) during polymerization at t=3350, (c) one double strand at t=5000, (d) during the unzipping at t=7350, and (e) two unzipped strands at t=7360. Here and in the next figure, bold black edges represent covalent bond, thin black edges represent hydrogen bond, and thin grey edges represent hydrophilic contact. A white node is a nucleotide, a dark grey node in (b) is polymerase, and a node #28 in (d) is helicase. The head/tail of a polynucleotide is colored grey. The graphs are visualized with (aiSee, software).

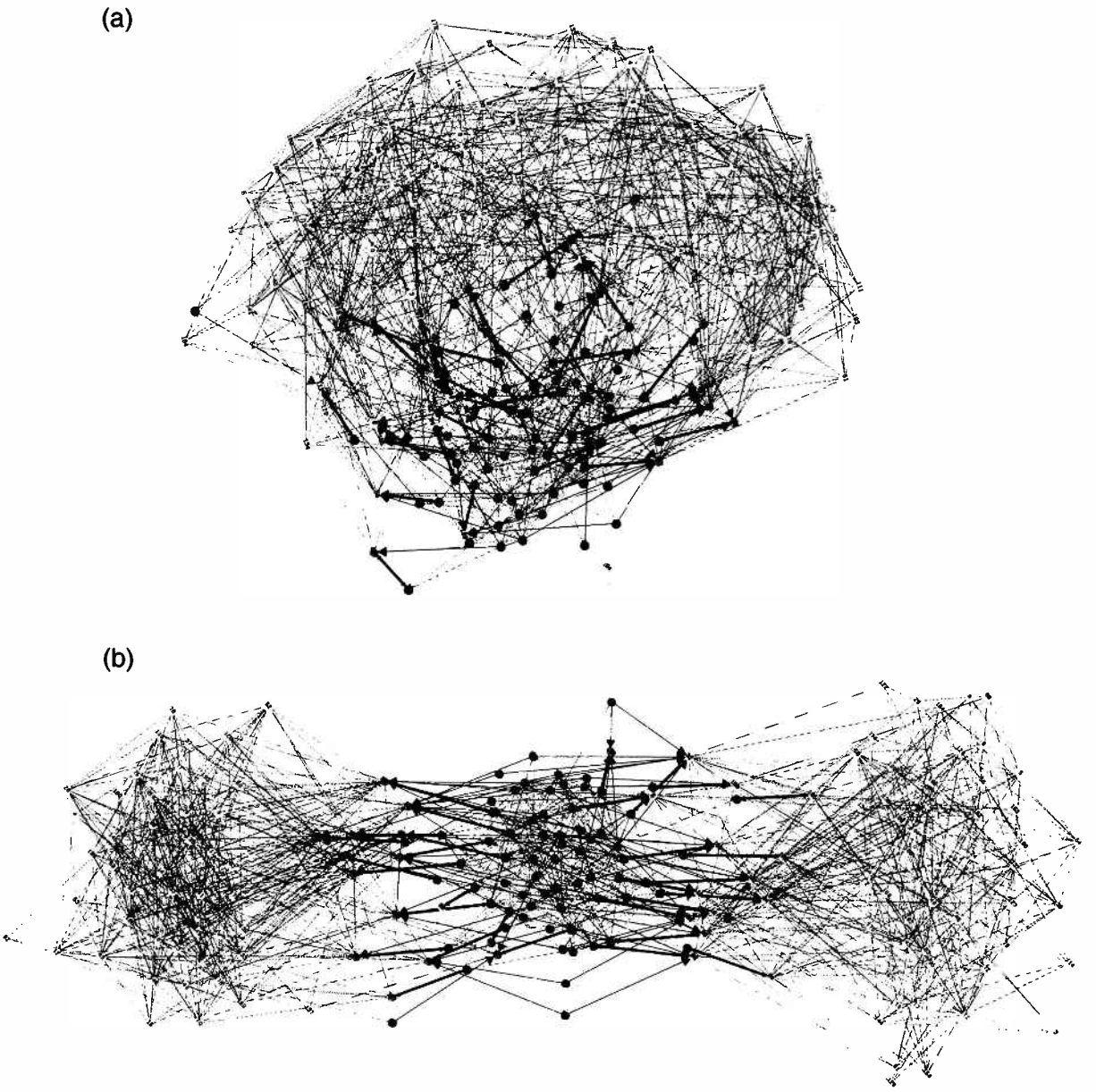

Figure 3: A typical run of the partitioning of the hydrophilic cluster in a NAC cell that has a total of 200 nodes and an average hydrophilic/hydrophobic degree of 10/20, respectively. The network includes 30 lipid pairs (15 inside and 15 outside), 40 nodes of splitase, and 100 free nucleotides. (a) snapshot at t=3000 and (b) t=10000. Thin black edges represent hydrophilic contact, and thin red edges represent hydrophobic contact. White nodes are hydrophilic nucleotides, gray nodes are hydrophilic parts of lipid, and black nodes are hydrophobic parts of lipid. Dark gray nodes which are located in the inner cluster in (b) are splitase.

node to be adjacent to any number of nodes in the network. Though particular size limitations are present for such adjacent node lists as cv[ ] or hy[ ], if several active nodes are properly connected, they can simultaneously work on a single task at the same time, suggesting that NAC can emulate such complex molecular cooperation as protein-protein interaction. The author believes that the replication in Fig. 2 demonstrates this ability of NAC, even as simple as it is. Combined with the flexibility mentioned above, this striking property of NAC might provide us with a method to simulate the organization and movement of a 'protein complex,' a highly-functional cluster of active nodes in a cell.

Future extensions: The following research topics are left for future studies: implementing a 'folding' mechanism which changes a polynucleotide into a single active molecule and makes an active node program from the string sequence of a polynucleotide; enabling a NAC cell to self-reproduce by putting the replicating and partitioning molecules into a single cellular network and synchronizing their activities; mathematically analyzing molecular diffusion in a liquid space, and examining the plausibility of the passive rewiring rule of NAC.

Dr. Ono of ATR labs provided useful comments for this work. Dr. K. Shimohara of ATR labs actively encouraged the study. This study was supported by the National Institute of Information and Communications Technology (NICT) and Doshisha University's Research Promotion Funds, Japan.

Appendix A: Pseudocode of Polymerase

The Network Cell is implemented in Java (Cadenhead and Lemay 2004). A node is prepared as an *object* in Java, and cv[ ], hy[ ], il[ ], ob[ ], sa, ta, and other node variables (na, nb, and nc) are prepared as *instance variables*. The program is implemented as a *method* of the object for an active node. Here and in the subsequent appendices, '.' is a specific Java operator representing a class member: for example, na.ta means ta of na.

```
if(hy[0] is null) // Template head is not yet found
{
  na <- (random adjacent node);
  if(na is a nucleotide AND na.ta=3)
  {
    hy[0] <- na; // hy[0] = template head
    na.ta <- 5;
  }
}
if(hy[0] is not null AND hy[1] is null)
// The first nucleotide is not yet found.
{
  na <- (random adjacent node);
  if(na is a nucleotide AND na.ta=-1 AND
     hy[0].sa=na.sa) // checking nucleo. identity
  {
    hy[1] <- na; // hy[1] = first nucleotide
    na.ta <- 5;
  }
}
if(hy[0] is not null AND hy[1] is not null AND
   hy[0].ta is not 4) // Under polymerization
{
  na <- (random adjacent node);
  if(na is a nucleotide AND na.ta=-1 AND
     hy[0].cv[0].sa=na.sa)
                    // checking nucleo. identity
  {
    hy[0].hy[0] <- hy[1]; // creates a new bond
    hy[1].hy[0] <- hy[0]; // creates a new bond
    hy[1].cv[0] <- na;    // creates a new bond
    na.cv[1] <- hy[1];    // creates a new bond
    hy[0] <- hy[0].cv[0]; // walks along chain
    hy[1] <- na;          // walks along chain
    hy[1].ta <- 2;
  }
}
if(hy[0] is not null AND hy[1] is not null AND
   hy[0].ta=4) // Polymerization is finished.
{
  hy[0].hy[0] <- hy[1]; // creates the final bond
  hy[1].hy[0] <- hy[0]; // creates the final bond
  hy[0].ta <- 6;  hy[1].ta <- 6;
  hy[0] <- null;        // releases the chain
  hy[1] <- null;        // releases the chain
}
```

Appendix B: Pseudocode of Helicase

```
if(hy[0] is null)
               // Double strand is not yet found
{
  na <- (random adjacent node);
  if(na is a nucleotide AND na.ta=6)
  {
    hy[0] <- na;       // hy[0] = strand tail
    hy[1] <- na.hy[0]; // hy[1] = strand tail
    hy[0].ta <- 4;  hy[1].ta <- 4;
  }
}
if(hy[0] is not null AND hy[0].ta is not 5)
                              // Under unzipping
{
  hy[0].hy[0] <- null;   // cuts the bond
  hy[1].hy[0] <- null;   // cuts the bond
  hy[0] <- hy[0].cv[1];  // walks along chain
  hy[1] <- hy[1].cv[1];  // walks along chain
}
if(hy[0] is not null AND hy[0].ta=5)
                           // Unzipping is finished.
{
  hy[0].hy[0] <- null;   // cuts final bond
  hy[1].hy[0] <- null;   // cuts final bond
  hy[0].ta <- 3;  hy[1].ta <- 3;
  hy[0] <- null;         // releases the chain
```

```
  hy[1] <- null;          // releases the chain
}
```

Appendix C: Pseudocode of Splitase

```
na <- ob[random];
na <- na.cv[1];
if(na is a hydrophilic part of inner lipid)
{
  hy[0] <- na; // hy[0] = inner lipid surface
}else{
  hy[1] <- na; // hy[1] = outer lipid surface
}
if(hy[0] is not null AND hy[1] is not null)
                                  // lipid is found
{
  nb <- hy[0].il[random];
  nc <- hy[1].il[random];
  cuts il[]-edge between hy[0] and hy[1];
  cuts il[]-edge between hy[1] and nb;
  cuts il[]-edge between hy[0] and nc;
  cuts il[]-edge between nb and nc;
}
```

References

Adami, C., Brown, C.T.: Evolutionary learning in the 2D artificial life system "Avida." In: Brooks, R., Maes, P. (eds.): Artificial Life IV: Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems, MIT Press, Cambridge (1994) 377–381

Adami, C., Ofria, C., Collier, T.C.: Evolution of biological complexity. Proc. Natl. Acad. Sci. USA **97** (2000) 4463-4468

aiSee is a commercial software for visualizing graphs with various algorithms such as the spring model. http://www.aisee.com/

Alberts, B., Bray, D., Lewis, J., Raff, M., Roberts, K., Watson, J.D.: Molecular Biology of the Cell, The Third Edition. Garland Publishing, New York (1994)

Cadenhead, R., Lemay, L.: Sams teach yourself Java 2 in 21 days. Macmillan Computer Pub. (2004)

Davidsen, J., Ebel, H., Bornholdt, S.: Emergence of a small world from local interactions - modeling acquaintance networks. Physical Review Letters **88**(12) (2002) 128701

Hutton, T.J.: Evolvable self-replicating molecules in an artificial chemistry. Artificial Life **8** (2002) 341-356

Madina, D., Ono, N., Ikegami, T.: Cellular evolution in a 3D lattice artificial chemistry. In: Banzhaf, W., Christaller, T., Dittrich, P., Kim, J.T., Ziegler, J. (eds.): Advances in Artificial Life (7th European Conference on Artificial Life Proceedings), Springer-Verlag, Berlin (2003) 59–68

Ono, N., Ikegami, T.: Model of Self-Replicating Cell Capable of Self-Maintenance. In: Floreano, D. et al. (eds.): Advances in Artificial Life (5th European Conference on Artificial Life Proceedings), Springer-Verlag, Berlin (1999) 399-406

Ono, N., Ikegami, T.: Artificial chemistry: computational studies on the emergence of self-reproducing units. In: Kelemen, J., Sosik, P. (eds.): Advances in Artificial Life (6th European Conference on Artificial Life Proceedings), Springer-Verlag, Berlin (2001) 186-195

Pargellis, A.N.: The spontaneous generation of digital "Life". Physica D **91** (1996) 86-96

Pargellis, A.N.: The evolution of self-replicating computer organisms. Physica D **98** (1996) 111-127

Pargellis, A.N.: Digital life behavior in the Amoeba world. Artificial Life **7** (2001) 63-75

Ray, T.S.: An approach to the synthesis of life. In: Langton, C.G. et al. (eds.): Artificial Life II: Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems (Santa Fe Institute Studies in the Sciences of Complexity, Vol. 10). Addison-Wesley (1992) 371–408

Ray, T.S.: Selecting Naturally for Differentiation. In: Koza, J.R. et al. (eds.): Genetic Programming 1997: Proceedings of the Second Annual Conference. Morgan Kaufmann, San Francisco (1997) 414–419

Speroni di Fenizio, P., Dittrich, P., Banzhaf, W.: Spontaneous formation of proto-cells in an universal artificial chemistry on a planar graph. In: Kelemen, J., Sosik, P. (eds.): Advances in Artificial Life (6th European Conference on Artificial Life Proceedings), Springer-Verlag, Berlin (2001) 206-215

Suzuki, H.: An approach to biological computation: unicellular core-memory creatures evolved using genetic algorithms. Artificial Life **5** N.4 (2000) 367–386

Suzuki, H.: Evolution of self-reproducing programs in a core propelled by parallel protein execution. Artificial Life **6** N.2 (2000) 103–108

Suzuki, H.: Core memory objects with address registers representing high-dimensional interaction. In: Calude, C.S. et al. (eds.): Pre-Proceedings of the Workshop on Multiset Processing (WMP-CdeA 2000), Centre for Discrete Mathematics and Theoretical Computer Science Research Report 140 (2000) 249-264

Suzuki, H., Ono, N., Yuta, K.: Several necessary conditions for the evolution of complex forms of life in an artificial environment. Artificial Life **9**(2) (2003) 537-558

Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature **393** (1998) 440-442