数理モデル化と問題解決　39－7
（２００２．５．１０）

Rewriting a 1D Artificial Chemistry System by Kinase Computing

Jian-Qin LIU and Katsunori SHIMOHARA

ATR Human Information Science Laboratories,
Hikaridai 2-2-2, Seika-cho, Soraku-gun, Kyoto, 619-0288, JAPAN

Abstract : This paper presents a method for rewriting a one-dimensional artificial chemistry system by kinase computing.

Keywords: artificial chemistry, artificial life, molecular computing, kinase computing, LS-systems.

キナーゼコンピューティングによる人工化学システムの書き換え

劉　健勤　下原勝憲
ATR 人間情報科学研究所
京都府相楽郡精華町光台２－２－２、６１９－０２８８

あらまし：本報告では、キナーゼコンピューティングによる人工化学システムの書き換え方法について議論する。
キーワード:人工化学、人工生命、分子計算、キナーゼコンピューティング、LS システム。

1. Introduction

In the early stages of our studies on kinase computing[1], we have been exploring its high degree of parallelism for molecular computing. This effort has led us to focus on the features of kinase computing as a new and unconventional computing paradigm with potential biological implementation. With kinase computing, we can achieve a coupling mechanism for different processing elements (PEs) that corresponds to the complexity within the process of problem solving itself. This cuts the cost of time and space into a linear order, and increases the speed of molecular computers. For example, we have solved the 3-SAT benchmark problem both in linear time and space[1].

A kinase is a special kind of enzyme. There are two main other kinds of enzymes related to our work: (a) simple enzymes in chemical engineering where the existing techniques for making them are mature, but information coding by molecules is difficult, and (b) simple enzymes within DNA computing, which offer efficient self-assembly, but present other difficulties, such as vulnerability to errors and the fact that a huge number of DNA molecules are necessary when they are separated for computing. Based on these considerations, we selected kinases because that they are natural and exist in living cells, and because it is easy to regulate them in cells *in situ*. This eliminates the heavy cost of manipulating the molecules (DNA, RNA or other bio-molecules), just to use the cells for computing tasks (although we must control or regulate them). Technically speaking, selection processes using kinases can achieve the linear cost of 3-SAT problem solving for NP problems.

We have determined that kinase computing can be extended and generalized into a rigorous formal model (i.e., a formal system) that conceptualizes or formalizes the signal pathways of *in situ* cells regulated by kinases in order to carry out parallel computing by a hyper-graph rewriting mechanism. From this it can be inferred that the central idea is to use interactions among molecules in the cell *in situ* for computing. In a more general sense (i.e., studying algorithms), the main goal of our study of kinase computing is to construct a new and unconventional computing paradigm that includes a methodology with the targets of achieving a high degree of parallelism and

creating a coupled mechanism for different PEs in order to increase the complexity within the algorithm itself and thus reduce the computing cost. These efforts are aimed at solving the kind of problems that consume considerable computing resources for conventional computing paradigms where silicon devices are used and limitations exist in conventional methods. With respect to the underlying architecture of parallel computing by molecules, we can use this method to avoid the problem of lowered efficiency in the neighborhood of PEs in parallel computing. Since we do not need to allocate the connection of PEs, the communication tasks between the PEs are much smaller than those in conventional computing methods.

The motivation for our research on the computational theory of kinase computing is the possibility of making a breakthrough in the current situation by introducing biologically-inspired ideas into computation. As such, we are trying to make "computers" that are equipped with interactions among "molecules." From the viewpoint of algorithm theory, fine-grained computing is carried out under a condition of partial constraints in the level of coarse-grained representation. Where the fine-grained scheme is used for high speed, coarse-grained representation is used to describe programs according to the designing schemes of our algorithms. Consequently, a simulation of the computing mechanism for molecular computing by kinase-guided signal pathways clearly shows that we can theoretically use cells *in situ* to make faster computers.

As for the application of kinase computing to artificial life (artificial chemistry), we envision the use of coupled parallel units for computing, rather than those at the single PE level. An important point here is that we propose using a set of coupled computing units and treating it as a whole object in order to carry out computing processes by transduction between the computing units for high parallelism. Each unit -- pathway -- in our method is formalized or regarded as a bigraph, in which the input-to-output can be formalized as a topograph and the internal structure of the pathway can be formalized as a monograph. This is a rigorous, formal, model-based parallel computing method. We can achieve complexity by enlarging the units' sizes to reduce the computing cost. In other words, we increase the complexity of the computing units in order to reduce the cost of the computing processes. Our method can thus generate a hypergraph-based GP or rewrite artificial chemistry systems by means of high-dimensional interactions, under certain conditions. Briefly speaking, in this paper, kinase computing is regarded as a software program that can adaptively program the granular modules among the computing systems. Briefly put, kinase computing is a new biologically-inspired information processing paradigm that is rigorously supported by theoretical computer science, with faithfulness of Rho family GTPases [2] by software simulation[1].

In this paper, we will discuss the application of our method to the field of artificial chemistry.

2. Kinase Computing for Rewriting Processes in Artificial Chemistry Systems

2.1 Formalization of Kinase Computing

2.1.1 The Preliminaries

The notations[3,4,5] used here are listed as:

A -- an alphabet set.
$\tau(a)$ -- the rank associated to a symbol ($a \in A$).
V_H -- a vertex set.
E_H -- a hyperedge set.
H -- a hypergraph, i.e., $H = < V_H, E_H >$, where $V_H \cap E_H = \varnothing$.
$lab_H(e)$ -- the label assigned to a hyperedge e in A.
$\tau(lab_H(e))$ -- the length of the sequence of vertices with $lab_H(e)$.
$R_p(A) = \{pah_a \mid a \in A\}$, where pah_a is $(\tau(a)+2)$-ary. This is the relation defined for the object set of pathways.
$|H|_3 := < V_H \cup E_H \cup K_H, (pah_{aH})_{a \in A} > \in STR(R_p(A))$, This is the corresponding structure defined for pathways.
Tog_H -- the category of topographs.
Mog_H -- the category of monographs.
Big_H -- the category of bigraphs.
$G_H = G(U_H, ctrl_H, G^T_H, G^M_H): \iota_m \to \iota_n$,

this is the representation for hypergraphs by bigraph forms where $U_H \in V_H \cup E_H$, $ctrl_H \in K_H$, G^T_H in Tog_H, G^M_H in Mog_H, G_H in Big_H, m, n $\in$ N.

K_H -- the control set corresponding to the G_H.

Cag_H -- the category defined based on the set of K_H.

pah (x, y_1, ..., y_{l1}, z_1, ..., z_{l2}) -- the predicate defined for pathways, where $l_1, l_2, l_3, l_4, l_5 \in$ N, x $\in E_H$, $y_1, \ldots, y_{l1} \in V_H$, $lab_H(x) = a$, $n = \tau(a)$, $z_1, \ldots\ldots, z_{l2}$ are the controls in the set of K_H.

$PATs_H$ -- the equivalent class of pathways.

$\equiv_H$ -- the equivalent relationship of $PATs_H$.

TRANS(G^T_H) -- the predicate set for the transductions exerted on the topographs: $G^T_H = (V, ctrl_H, prt)$: ι_m -> ι_n, where *prt* is the parent of the vertexes in rewriting.

TRANS(G^M_H) -- the predicate set for the transductions are exerted on the monographs: $G^M_H = (V, ctrl_H, \equiv_H)$.

BRS_H -- the category of bigraphical reactive systems.

$react_H$ -- the operators of the interactions of bigraphs based on the rule set of Q.

cag_H -- the category of those hypergraphs for the mapping: ι_m -> ι_n in the G_H, the input and output of pathways pah(.) in $R_p(A)$.

2.1.2 Formalization

Let cGRiT($|H|_3$, Big_H (K_H, $react_H$)) be the rewriting system on the hypergraphs with certain constraints (conditions), constructed by transduction on Big_H (K_H, $react_H$), where $react_H$ is used in BRS_H. Notice that the formal system we discuss here is functionally equivalent to the construct representation in[1].

Based on category cag_H, we can get the operation by predicates on the GRT($|H|_3$, Big_H (K_H, $react_H$), $\Im_H$) is:

INTERACTION (pah (x, y_1, ..., y_{l1}, z_1, ..., z_{l2}), pah (x', y_1, ..., y_{l3}, z_1, ..., z_{l4}), GRT($|H|_3$, Big_H (K_H, $react_H$), $\Im_H$))

which refers to the fact that x and x' share certain common parts of the hyperedges, y_1, ..., y_{l1} in pathway one, and y_1, ..., y_{l3} in another pathway. These two sets also share certain common parts of vertexes, and the neighborhood for the operation exerted is δ_H for the V_H and E_H in GRT($|H|_3$, Big_H (K_H, $react_H$), $\Im_H$). We have that:

Proposition 1: There exists $mono_H$, the set of monoid operators that is inferred from the interactions on $(pah_{aH})_{a \in A}$ of

$$GRiT(|H|_3, Big_H(K_H, react_H)),$$

which satisfies the condition of McNaughton languages for

$$G_H = G(U_H, ctrl_H, G^T_H, G^M_H): I^* \rightarrow J^*.$$

Proposition 2: The operations of $mono_H$ can keep the congruence of the (hyper)graph rewriting on

$$G_H = G(U_H, ctrl_H, G^T_H, G^M_H): I^* \rightarrow J^*,$$

if they satisfy the condition of McNaughton languages for I* -> J*, and the interactions of $mono_H$ can be inferred by

TRANDN (pah (x, y_1, ..., y_m, z_1, ..., z_p ...z_q, ..., z_m)) $\wedge$ VALPATH (pah (x, y_1, ..., y_m, z_1, ..., z_p ...z_q, ..., z_k)),

where p, q, m, n, k $\in$N for GRiT($|H|_3$, Big_H (K_H, $react_H$)).

2.2 An Example

The generalization of the model of "genomic dynamics" that we proposed previously[6] (which was once used to model bio-molecular computing) can be extended into a kind of methodology for studies on artificial life. The main parts of this model consist of information generation and the mutual relationships between them (conditional simulation/approximation by symbolic dynamics) where a kind of abstract form can be inferred. The artificial genomic sequences in the meaning of artificial life, and the equivalent string rewriting rules in the view of computation theory are helpful for studying artificial life by molecular computing. The most important parts that we consider here are the two circle structures for information generation[6] (which, of course, can be

generalized into high-dimensionally topological forms), that produce Xij, which denotes the location j in the circle i, and Xi'j', which denotes the location j' in the circle i', where i,i',j,j' ∈N. The sequences {i1,i2,...,iL} and {i1,i2,..., iL'} (i1, i2, ..., iL, i1, i2, ..., iL', L, L' ∈ N) are not identical in all processes. However, the coupling relations can be exerted onto the generation structure. In addition to factors such as random setting and dynamical updating within the generation structure, the locations and symbols are the main objects operated for the computational performance[6].

For a 1D artificial chemistry system obtained from a "genomic dynamics" model, we take the following two string rewriting rules as examples:

"A -> aA" and "B -> Bt",

where A and B refer to strings. By the combinatorial and repeated use of the above rules, a pattern such as "aacccacccaaa" can be produced that includes the building block "ac". This coupling of the rewriting rules can be represented as the following graph:

```
a -> G'
     |
     G
     |
     G" <- t
```

where G, G', G" refer to the hypergraphs. The minimum set of rewriting operations on hypergraphs can be {1 VR for "a", 1 VR for "c", 2 HR for "a", 2 HR for "c", 1 HR for the whole graph}. The rewriting operation of kinase computing for the 1D artificial chemistry system mentioned above consists of the hypergraph rewriting rules that can be equivalently inferred based on the combinatorial forms of the graph rewriting rule: "Gu * Gv -> Gw (Gu, Gv, Gw refer to hypergraphs)". This method of interaction is equivalent to the generation rule of Gz -> Gq ∪ Φg (Gz, Gq, Φg refer to the hypergraphs, with the condition that the complexity of Φg < the complexity of Gq) where the related predicates are formally defined in sections 2.1.1 and 2.1.2. The subsets of V_H and E_H related to the hypergraphs obtained from the rewriting rules mentioned above can cover the existence of the confluent subsets of McNaughton language, which are a prerequisite for the operations for rewriting an artificial chemistry system under the context of our discussion in this paper.

3. Conclusion

Based on the work reported here, it is expected that kinase computing will improve certain aspects of the performance of artificial life systems (e.g., speed). Future work will be focussed on "upgrading" the current version of our method by the "quantization" of kinase computing.

Acknowledgement:

The authors sincerely thank Prof. Kozo Kaibuchi and his laboratory for their invaluable help. J.-Q. Liu sincerely thanks Prof. Kozo Kaibuchi, Prof. Kazuki Joe, Dr. Shinya Kuroda, Dr. Mutsuki Amano, Prof. Wolfgang Banzhaf, Dr. Peter Dittrich, Dr. Pietro Speronni di Fenizio, Dr. Hideaki Suzuki and Dr. Yasuhiro Suzuki for their advice and suggestions.

This research was conducted as part of "Research on Human Communication" with funding from the Telecommunications Advancement Organization of Japan.

Main References

[1] Jian-Qin Liu and Katsunori Shimohara, Kinase Computing in Rho-GTPases: analysis and simulation, IPSJ SIG Notes, vol.2001, No.91, Sept.17, 2001, pp.29-32.

[2] K. Kaibuchi, S. Kuroda and M. Amano, Regulation of the cytoskeleton and cell adhesion by the Rho family GTPases in mammalian cells, Annu. Rev. Biochem., 68 (1999) 459-486.

[3] B. Courcelle, The expression of graph properties and graph transformations in monadic second-order logic, Chapter 5 of the "Handbook of graph grammars and computing by graph transformations, Vol. 1: Foundations", G. Rozenberg (ed.), World Scientific (New-Jersey, London), 1997, pp. 313-400.

[4] Robin Milner, Bigraphical reactive systems: basic theory, September 2001 (http://www.cl.cam.ac.uk/users/rm135 /bigraphs.ps.gz).

[5] Jian-Qin Liu and Katsunori Shimohara, Graph rewriting in topology II: computability, Proceedings of the 5th symposium on algebra, language and computation (edited by T. Imaoka and K. Shoji, Shimane Univ., Nov.17-18, 2001), pp. 63-68.

[6] Jian-Qin Liu and Katsunori Shimohara, DNA computing by genomic dynamics II: a simulation wetware prototype dynamical DNA computation, Proc. of AROB 5th'00 (Oita, Japan, January 2000) pp. 785-788.