A DNA-Based Algorithm for Minimizing Decision Rules: A Rough Sets Approach

Ikno Kim*, *Member, IEEE*, Yu-Yi Chu, Junzo Watada, *Member, IEEE*, Jui-Yu Wu, and Witold Pedrycz, *Fellow, IEEE*

***Abstract*—Rough sets are often exploited for data reduction and classification. While they are conceptually appealing, the techniques used with rough sets can be computationally demanding. To address this obstacle, the objective of this study is to investigate the use of DNA molecules and associated techniques as an optimization vehicle to support algorithms of rough sets. In particular, we develop a DNA-based algorithm to derive decision rules of minimal length. This new approach can be of value when dealing with a large number of objects and their attributes, in which case the complexity of rough-sets-based methods is NP-hard. The proposed algorithm shows how the essential components involved in the minimization of decision rules in data processing can be realized.**

***Index Terms*—Data processing, decision rules, DNA-based algorithm, knowledge support system, rough sets.**

I. Introduction

ROUGH SETS offer new approaches to machine learning, knowledge discovery in data, and knowledge support systems. Rough set theory has thus become an important basis for reasoning, inductive learning, and knowledge reduction. The rough set method (the rough set theory-based method) is a new discovery method for a data classification system that is used when there are different types of object data (any applicable knowledge-based data) whose attributes can be reduced and classified in order to provide comprehensive information.

In rough sets, a decision table is often used to represent objects, attributes, and decision attributes. In general, decision tables can be equivalently interpreted in terms of a collection of *if-then* decision rules. To effectively deal with decision rules, one has to consider the possibility that they can be effectively reduced. In this regard, [1] reports a number of reduction algorithms and presents applications, in which rough sets were used to minimize sets of decision rules when making decisions under conditions of uncertainty. The computation of all minimal length decision rules in order to manage the uncertainty in expert systems was proposed by Grzymala-Busse [2]. A variable precision model of rough sets was generalized and introduced by Ziarko [3]. The incremental identification of decision rules was proposed by Skowron *et al.* [4], whose study also elaborated the discernibility matrices and functions. Shan *et al.* [5] proposed an incremental algorithm for identifying all minimal length decision rules.

Issues of computational complexity have become of profound relevance when dealing with problems that involve a large number of data and attributes. This becomes particularly apparent, because the minimization of decision rules and the computation of all minimal length decision rules are NP-hard [6]. Recognizing this challenge, in this study, we propose a DNA-based algorithm for the computation of all minimal length decision rules. The DNA-based algorithm exhibits a significant potential when dealing with a large number of objects and their attributes. Given this capacity, it helps establish a new frontier in the enhancement of collaborative data processing that exploits both computer-based and DNA-based technologies.

Since Adleman [7] first proposed a new computing paradigm, there have been several different types of meaningful bio-inspired algorithms and related applications, cf. [8]–[13]. In this study, we exploit the DNA paradigm in the setting of rule minimization and rough sets. In this paper, a novel DNA-based algorithm is proposed as a powerful alternative means of finding a truly optimal solution. This means that it has a different ultimate purpose than that of common heuristic and meta-heuristic algorithms.

II. Rough Set Method

In this section, the rough set method is generally defined with a model decision table. This definition will prove helpful in presenting our DNA-based algorithm for minimizing decision rules.

A. Rough Set Theory

The rough set method is built on the basis of a mathematical concept underlying a set theory, which is known as rough set theory. The theory of rough sets was first proposed by Pawlak [14]. Several studies of rough set theory [15], [16] also identified many advantages it offered in data processing in the evaluation of significant data, analysis of hidden data, minimization of sets of decision rules, and the discovery of new knowledge, particularly in areas related to knowledge support systems.

Manuscript received April 20, 2010; revised July 22, 2011; accepted August 01, 2011. Date of current version October 26, 2011. This work was supported by Waseda University Global COE Program International Research and Education Center for Ambient SOC sponsored by MEXT, Japan. *Asterisk indicates corresponding author.*

*I. Kim is with the Graduate School of Information, Production and Systems, Waseda University, Kitakyushu 808-0135, Japan (e-mail: octoberkim@akane.waseda.jp).

Y.-Y. Chu and J. Watada are with the Graduate School of Information, Production and Systems, Waseda University, Kitakyushu 808-0135, Japan (e-mail: bolero168@ruri.waseda.jp; watada@waseda.jp).

J.-Y. Wu is with the Department of Biochemistry, School of Medicine, Taipei Medical University, Taipei, Taiwan (e-mail: jwu@tmu.edu.tw).

W. Pedrycz is with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton T6R 2G7, Canada (e-mail: pedrycz@ee.ualberta.ca)..

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNB.2011.2168535

TABLE I
MODEL DECISION TABLE (8 OBJECTS INCLUDE CONDITION ATTRIBUTES, CONDITION ATTRIBUTE VALUES, AND DECISION CLASSES).

Object	Colour	Density	Feature	Pattern	Decision Class
Object-1	(C, 4)	(D, 3)	(F, 2)	(P, 1)	Target-2
Object-2	(C, 3)	(D, 4)	(F, 4)	(P, 2)	Target-2
Object-3	(C, 1)	(D, 1)	(F, 1)	(P, 3)	Target-1
Object-4	(C, 4)	(D, 3)	(F, 3)	(P, 2)	Target-1
Object-5	(C, 2)	(D, 2)	(F, 1)	(P, 1)	Target-2
Object-6	(C, 3)	(D, 4)	(F, 4)	(P, 2)	Target-1
Object-7	(C, 4)	(D, 1)	(F, 2)	(P, 3)	Target-2
Object-8	(C, 2)	(D, 2)	(F, 1)	(P, 1)	Target-1

Let us consider an equivalence relation, denoted by Ξ, in a universe U. In rough set theory, the form of a pair (U, Ξ) is given by elementary knowledge, and if any x is an element of U, representing $x \in U$, then the equivalence class of x is defined as follows:

$$[x]_{\Xi} = \{y \in U | (x,\ y) \in \Xi\}, \tag{1}$$

where all the objects in $[x]_{\Xi}$ are collected to become a part of the same category for the initial partition, and the element of the restored category is x. The categories of this form $[x]_{\Xi}$ are elementary granules of knowledge and are also known as equivalence classes of the equivalence Ξ.

Let us denote a subset of U as X. The three disjoint sets for any subset $X \subseteq U$ represent the objects that are (1) exactly in X (class 1); (2) exactly not in X (class 2); and (3) possibly in X (class 3). In more detail, the lower approximation of X (class 1), denoted as Ξ_{A}, is defined by the union of all the elementary sets as follows:

$$\Xi_{\mathrm{A}}(X) = \{x_i \in U | [x_i]_{\Xi} \subseteq X\}, \tag{2}$$

where the union of all the elementary sets is $x_i, i = 1, 2, \ldots, n$, and each of them is classified as exactly belonging to the subset X. Second, the upper approximation of X (classes 1 and 3), denoted as Ξ^{A}, is also defined by the union of all the elementary sets as follows:

$$\Xi^{\mathrm{A}}(X) = \{x_i \in U | [x_i]_{\Xi} \cap X \neq \phi\}, \tag{3}$$

where the union of all the elementary sets is also $x_i, i = 1, 2, \ldots, n$, and each of them is classified as possibly belonging to the subset X. Finally, the boundary region of X (class 3), denoted as Ξ^{B}, is also defined by the union of all the elementary sets as follows:

$$\Xi^{\mathrm{B}}(X) = \Xi^{\mathrm{A}}(X) - \Xi_{\mathrm{A}}(X), \tag{4}$$

where none of the elementary sets were found to be classified for certain [6], [17].

In this study of the derivation of decision rules of minimal length, we initially focus on subsets of the lower approximation, which should be determined first for each of all the subsets of the decision table in the creation of decision matrices. In the present computational algorithm, a discernibility matrix is used to determine subsets of the lower approximation for each decision value of the decision attribute. Our DNA-based algorithm will also first accurately determine all the subsets of the lower approximation before deriving decision rules of minimal length.

B. Model Decision Table

In rough sets, there are two different systems (an information system and a decision system). The information system only includes a finite set of objects and attributes. In this paper, we discuss a *complete decision system*, which deals with condition attribute values.

A decision system, denoted as DS, is defined as $DS = (U, \Gamma, \varepsilon, \omega)$, where Γ is a finite set of condition attributes, ε is a decision attribute, and ω is a value assignment. U is composed of objects $x_1, x_2, \ldots, x_n$, which correspond to $U = \{x_1, x_2, \ldots, x_n\}$. First, let us denote n condition attributes as $\zeta_1, \zeta_2, \ldots, \zeta_n$, while Γ is composed of n condition attributes that correspond to $\Gamma = \{\zeta_1, \zeta_2, \ldots, \zeta_n\}$. Second, when dealing with condition attribute values for each of n condition attributes $\zeta_1, \zeta_2, \ldots, \zeta_n$, let us denote m condition attribute values as $\psi_1^{\zeta 1}, \psi_2^{\zeta 1}, \ldots, \psi_m^{\zeta 1}, \psi_1^{\zeta 2}, \psi_2^{\zeta 2}, \ldots, \psi_m^{\zeta 2}, \ldots$, and $\psi_1^{\zeta n}, \psi_2^{\zeta n}, \ldots, \psi_m^{\zeta n}$. Finally, let us denote n pair sets of both the above n condition attributes and m condition attribute values, as $I_{\zeta 1}^{c}, I_{\zeta 2}^{c}, \ldots, I_{\zeta n}^{c}$.

A case of this mapping takes the form ε: $U \rightarrow D^v$, where D^v is a set of decision values in the decision system, where the value assignment ω encompasses the decision attribute ε, expressing ω: $U \times (\Gamma \cup \{\varepsilon\}) \rightarrow I^v \cup D^v$, and where $\{\varepsilon\}$ is a set of decision attributes, denoted as E, representing $E = \{\varepsilon\}$, which is treated as a single decision attribute. Hence, a pair (ε, u) belongs to D^v, which is meant to satisfy all the value assignments. In fact, a set of all the attributes is represented by $\Gamma_t = \Gamma \cup \{\varepsilon\}$ in the case, in which the decision part is added. Assuming that n decision values of the decision attribute are denoted as $\tau_1, \tau_2, \ldots, \tau_n$ and that a set of decision values corresponds to $D^v = \{\tau_1, \tau_2, \ldots, \tau_n\}$, n decision classes in the given n-decision value can be denoted as $D_{\tau 1}^v, D_{\tau 2}^v, \ldots, D_{\tau n}^v$.

Table I shows a model decision table that was used in this study to minimize the number of decision rules in rough sets. This example of a decision table represents eight different proposed objects, four different types of condition attributes with attribute values, and one decision attribute with decision values. We have to minimize (simplify) these given decision rules provided by the proposers. By minimizing these decision rules, we

can easily grasp the characteristics of this decision table. Each set of elements in Table I is represented as follows:

(1) $U = \{\text{Object-1}, \text{Object-2}, \ldots, \text{Object-8}\}$;

(2) $\Gamma = \{\text{Color(C)}, \text{Density(D)}, \text{Feature(F)}, \text{Pattern(P)}\}$;

(3)

$$I^c_{\text{Color}} = \{(\text{C}, (\text{blue} = 1)), (\text{C}, (\text{purple} = 2)), (\text{C}, (\text{red} = 3)), (\text{C}, (\text{yellow} = 4))\};$$

(4)

$$I^c_{\text{Density}} = \{(\text{D}, (\text{very low} = 1)), (\text{D}, (\text{low} = 2)), (\text{D}, (\text{high} = 3)), (\text{D}, (\text{very high} = 4))\};$$

(5)

$$I^c_{\text{Feature}} = \{(\text{F}, (\text{simple} = 1)), (\text{F}, (\text{a bit simpler} = 2)), (\text{F}, (\text{a bit more complex} = 3)), (\text{F}, (\text{complex} = 4))\};$$

(6) $I^c_{\text{Pattern}} = \{(\text{P}, (\text{solid} = 1)), (\text{P}, (\text{liquid} = 2)), (\text{P}, (\text{gas} = 3))\}$; and

(7) $D^v = \{\text{Target-1}, \text{Target-2}\}$ (in this case, Target-1 corresponds to Decision Class-1 and Target-2 corresponds to Decision Class-2).

III. DNA-Based Algorithm

In designing the DNA-based algorithm, the process of encoding the DNA sequences is the main part of the process that is achieved by using a binary adjacency matrix that can be transformed from its own digraph. There are subsequently several sorts of molecular manipulation techniques that use the encoded DNA sequences and are implemented to design the DNA-based algorithm. The DNA-based algorithm is used to detect the subsets of the lower approximations in each of the decision classes, and to determine all the minimal length decision rules on the basis of the results of these lower approximation detections. In this section, we describe the DNA-based algorithm and how it is structured in order to minimize the decision rules in rough sets.

A. DNA-Digraph

In a digraph with n nodes, there are two distinct categories of element nodes: (1) an object element node; and (2) a node of both a condition attribute and a condition attribute value (we abbreviate this node as the term "a pair element node"). In a process similar to that used for the decision table, for the digraph, again assuming that (1) n given object element nodes are $x_1, x_2, \ldots, x_n$, and a set of these element nodes is U, represented as $U = \{x_1, x_2, \ldots, x_n\}$ for the object element node; and (2) n given pair element nodes composed of both n condition attributes $\zeta_1, \zeta_2, \ldots, \zeta_n$ and m condition attribute values $\psi_1^{\zeta 1}, \psi_2^{\zeta 1}, \ldots, \psi_m^{\zeta 1}, \psi_1^{\zeta 2}, \psi_2^{\zeta 2}, \ldots, \psi_m^{\zeta 2}, \ldots$, and $\psi_1^{\zeta n}, \psi_2^{\zeta n}, \ldots, \psi_m^{\zeta n}$, and a set of these pair element nodes is denoted as P_s, which is represented as a pair matrix, denoted by P, which expresses

$$P = \begin{bmatrix} P(\zeta_1, \psi^{\zeta_1}{}_1) & P(\zeta_1, \psi^{\zeta_1}{}_2) & \cdots & P(\zeta_1, \psi^{\zeta_1}{}_m) \\ P(\zeta_2, \psi^{\zeta_2}{}_1) & P(\zeta_2, \psi^{\zeta_2}{}_2) & \cdots & P(\zeta_2, \psi^{\zeta_2}{}_m) \\ \vdots & \vdots & \ddots & \vdots \\ P(\zeta_n, \psi^{\zeta_n}{}_1) & P(\zeta_n, \psi^{\zeta_n}{}_2) & \cdots & P(\zeta_n, \psi^{\zeta_n}{}_m) \end{bmatrix}. \tag{5}$$

A digraph structure consists of both n object element nodes and n pair element nodes, which are used for generating the minimal decision rules. If there are some condition attributes that include less condition attribute values than other condition attributes, then the specific symbol "ϕ" is used for any of the empty entries. We refer to this new style of digraph as a *DNA-digraph*, as shown in Fig. 1.

For the DNA calculation, an example in Fig. 5(a) illustrates a model of a DNA-digraph that is constructed on the basis of the model decision table. This model DNA-digraph is composed of the eight object element nodes, $U = \{x_1, x_2, \ldots, x_8\}$, and the four condition attributes, $\zeta_1, \zeta_2, \ldots, \zeta_4$. Here, each of the condition attributes (ζ_1, ζ_2, and ζ_3) includes four condition attribute values, and the remaining condition attribute (ζ_4) includes three condition attribute values, representing a pair matrix P that can be expressed as

$$P = \begin{bmatrix} P(\zeta_1, \psi^{\zeta_1}{}_1) & P(\zeta_1, \psi^{\zeta_1}{}_2) & P(\zeta_1, \psi^{\zeta_1}{}_3) & P(\zeta_1, \psi^{\zeta_1}{}_4) \\ P(\zeta_2, \psi^{\zeta_2}{}_1) & P(\zeta_2, \psi^{\zeta_2}{}_2) & P(\zeta_2, \psi^{\zeta_2}{}_3) & P(\zeta_2, \psi^{\zeta_2}{}_4) \\ P(\zeta_3, \psi^{\zeta_3}{}_1) & P(\zeta_3, \psi^{\zeta_3}{}_2) & P(\zeta_3, \psi^{\zeta_3}{}_3) & P(\zeta_3, \psi^{\zeta_3}{}_4) \\ P(\zeta_4, \psi^{\zeta_4}{}_1) & P(\zeta_4, \psi^{\zeta_4}{}_2) & P(\zeta_4, \psi^{\zeta_4}{}_3) & \phi \end{bmatrix}. \tag{6}$$

In a DNA encoding part, the above pair matrix is employed mainly when encoding double-encoded substrings for type 3 and their complementary substrings for type 6.

In the DNA-digraph, the two distinctly different types of the preceding element nodes are the same type of node, both of which can be simply expressed as element node types. The current existing relationship between any two element nodes in the DNA-digraph is denoted as e, where e is the existing relationship of a direct line that connects those two nodes. The three types of relational directions between two specific element nodes: (1) if an object element node x_i includes a direct relation with a pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$, which represents $x_i \bar{e} (\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$, and no direct relation represents $x_i \bar{e} (\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$; (2) if a pair element node $(\zeta_\beta, \psi_\beta^{\zeta\beta})$ includes a direct relation with an object element node x_j, which represents $(\zeta_\beta, \psi_\beta^{\zeta\beta}) \bar{e} x_j$, and no direct relation represents $(\zeta_\beta, \psi_\beta^{\zeta\beta}) \bar{e} x_j$; and (3) if a pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$ includes a direct relation with a pair element node $(\zeta_\beta, \psi_\beta^{\zeta\beta})$, which represents $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha}) e (\zeta_\beta, \psi_\beta^{\zeta\beta})$, and no direct relation represents $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha}) \bar{e} (\zeta_\beta, \psi_\beta^{\zeta\beta})$.

To represent the arcs among element nodes in the DNA-digraph, three types of relational arcs are sought: (1) when an object element node x_i includes a directional arrow that reaches a pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$, which represents

$$b^a{}_l = \overrightarrow{(x_i, (\zeta_\alpha, \psi^{\zeta_\alpha}{}_\alpha))}, \; i, \; \alpha, \text{ and } l = 1, 2, \ldots, n; \tag{7}$$

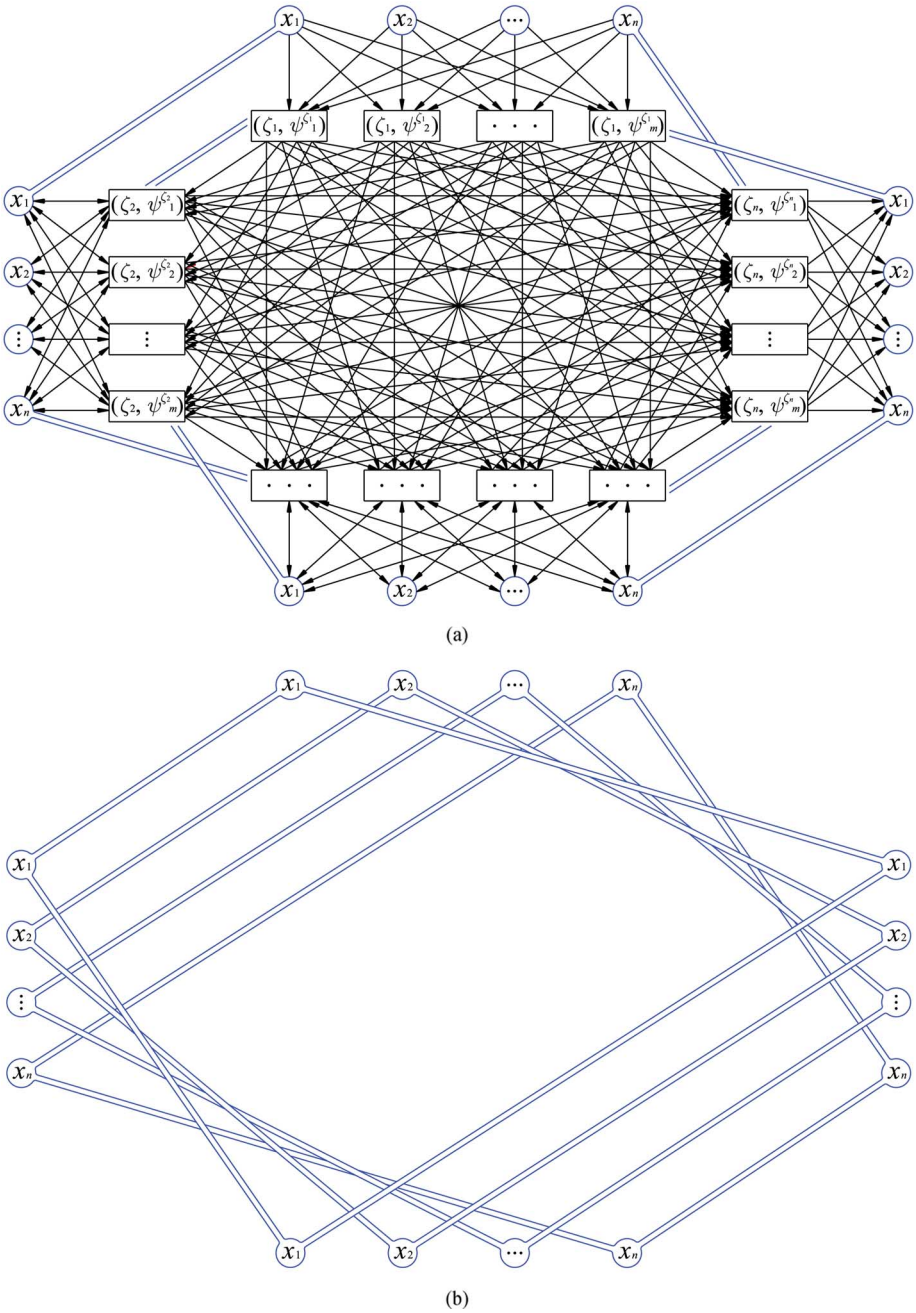

Fig. 1. DNA-digraph for the minimization of decision rules: (a) direct relations among n object element nodes x_n and n pair element nodes $(\zeta_n, \psi_m^{\zeta n})$; (b) the connection representation of objects in each n object element node, one by one.

(2) when a pair element node $(\zeta_\beta, \psi_\beta^{\zeta\beta})$ includes a directional arrow that reaches an object element node x_j, which represents

$$b^b{}_l = \overrightarrow{((\zeta_\beta, \psi^{\zeta_\beta}{}_\beta), x_j)},\ \beta,\ j,\ \text{and}\ l = 1,\ 2, \ldots,\ n; \quad (8)$$

and (3) when a pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$ includes a directional arrow that reaches a pair element node $(\zeta_\beta, \psi_\beta^{\zeta\beta})$, which represents

$$b^c{}_l = \overrightarrow{((\zeta_\alpha, \psi^{\zeta_\alpha}{}_\alpha), (\zeta_\beta, \psi^{\zeta_\beta}{}_\beta))},\ \alpha,\ \beta,\ \text{and}\ l = 1,\ 2, \ldots,\ n \quad \text{for}\ \alpha \neq \beta. \quad (9)$$

In addition, (1) an arc subset from the object element node x_i to the pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$ is denoted as B^a, where B^a is composed of all the possible arcs that represent $B^a = \{b_1^a, b_2^a, \ldots, b_n^a\}$; (2) an arc subset from the pair element node $(\zeta_\beta, \psi_\beta^{\zeta\beta})$ to the object element node x_j is denoted as B^b, where B^b is composed of all the possible arcs that represent $B^b = \{b_1^b, b_2^b, \ldots, b_n^b\}$; and (3) an arc subset from the pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$ to the pair element node $(\zeta_\beta, \psi_\beta^{\zeta\beta})$ is denoted as B^c, where B^c is composed of all the possible arcs that represent $B^c = \{b_1^c, b_2^c, \ldots, b_n^c\}$.

The two sets of object element nodes U and the pair element nodes P_s and the three different types of the arc subsets B^a, B^b, and B^c can be properly expressed in a binary adjacency

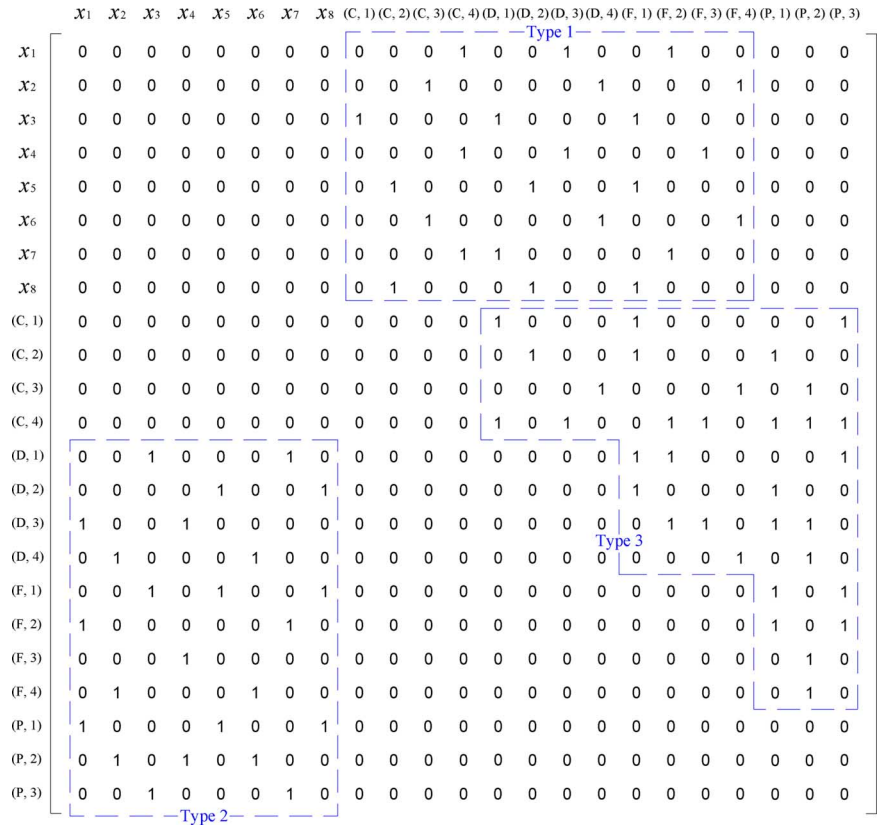

Fig. 2. Binary adjacency matrix of the given object and pair element nodes for the model DNA-digraph.

matrix. Fig. 2 shows the binary adjacency matrix of the model DNA-digraph. The model binary adjacency matrix comes with rows and columns that are labeled as follows:

$$r_{i,\,j}, i \text{ and } j = 1,\ 2, \ldots,\ 23 \text{ for all } (i,j) \in L, \tag{10}$$

where L is the set of all possible row and column labels (i is the row label and j is the column label) in the model binary adjacency matrix. Basically, this binary adjacency matrix is constructed by setting $r_{i,j} = 1$ wherever there is any one of three different type arcs in the example DNA-digraph (1) from an object element node x_i to a pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$; (2) from a pair element node $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$ to an object element node x_j; and (3) from a pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$ to another pair element node $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$, where (1) implies $x_i e(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$; (2) implies $(\zeta_\beta, \psi_\beta^{\zeta_\beta}) e x_j$; and (3) implies $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha}) e(\zeta_\beta, \psi_\beta^{\zeta_\beta})$, while the binary adjacency matrix is also constructed by setting $r_{i,j} = 0$ elsewhere, where (1) implies $x_i \bar{e}(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$; (2) implies $(\zeta_\alpha, \psi_\beta^{\zeta_\beta}) \bar{e} x_j$; and (3) implies $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha}) \bar{e}(\zeta_\beta, \psi_\beta^{\zeta_\beta})$.

B. Process for Encoding DNA

The binary adjacency matrix is employed to encode DNA sequences. The ordered rows and columns of the binary adjacency matrix, including the object and pair element nodes, can be transformed into specific desired DNA sequences in a systematic way.

As shown in Fig. 2, the binary adjacency matrix of the model DNA-digraph (8 object element nodes and 15 pair element nodes) is configured to be 23×23 (23 rows and 23 columns). In the model DNA-digraph, each directional order of two element nodes includes its own row and column, and the rows and columns in the model binary adjacency matrix are labeled as $r_{i,j}$, i and $j = 1, 2, \ldots, 23$. For the process of the DNA-based algorithm, seven different types are created in order to generate an initial library of DNA fragments, as shown in Fig. 3. Each of the seven types includes its own row and column labels, which are mainly defined in order to encode object and pair element nodes in single-stranded DNA (ssDNA) form. All the given DNA substrings correspond to types 1, 2, and 3, and their complementary substrings of two different given DNA substrings correspond to types 4, 5, 6, and 7. All the different types of both double-encoded and complementary substrings of DNA sequences are basically formed in ssDNA.

First, type 1 in the DNA-digraph represents a double-encoded substring that is each of the two different single element nodes x_i and $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$. For type 1, the two different types of element nodes, in which there is a direction of the arrow that indicates the direction from the object element node x_i to the pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$, are denoted as a double-encoded substring $x_i - 3'^{upper} \rightarrow 5'^{upper} - (\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$. In the binary adjacency matrix of the model DNA-digraph, as shown in Fig. 2, all of the row and column labels are denoted as i and j for type 1, and the entries are defined as

$$r_{i,j} = 1 \text{ for } i = 1, 2, \ldots, 8,\ j = 9, 10, \ldots, 20, \text{and all } (i,j) \in L. \tag{11}$$

As shown above, the two different element nodes can be encoded as a single oligonucleotide (two different unique sites are

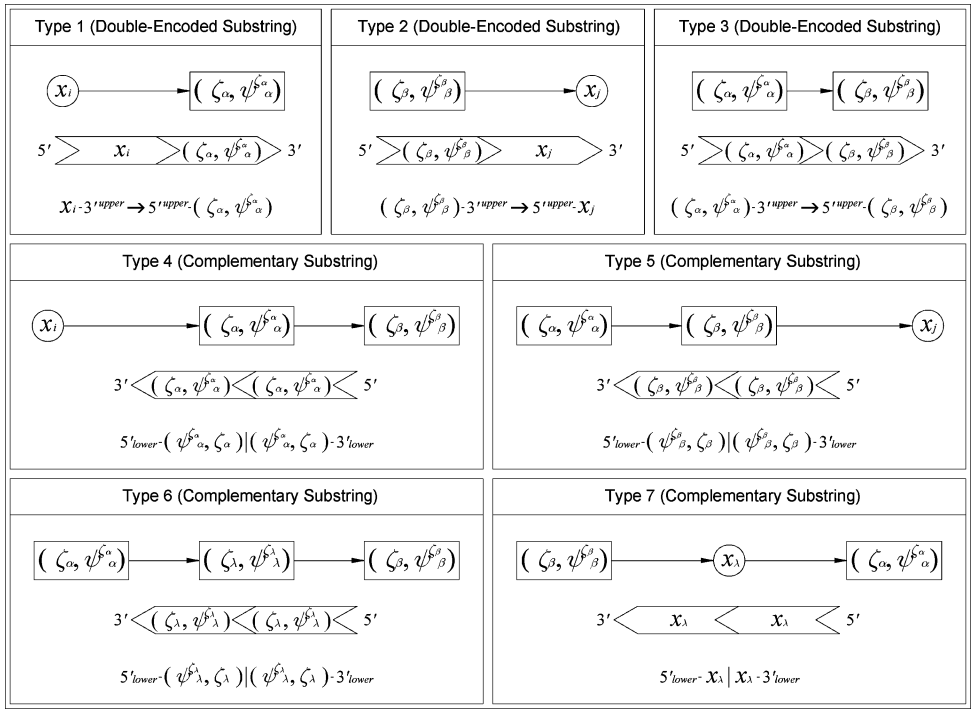

Fig. 3. Seven different types of substrings used to encode object and pair element nodes in ssDNA.

unified to become a single site), and each of all the single encoded oligonucleotides includes its own length for DNA base pairs (bp: after hybridization and end-filling DNA). Thus, (1) each object element node x_i, $i = 1, 2, \ldots, 8$ was set having the same length of 18 bp; and (2) each pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$ corresponds to $5'^{upper}$-$(\mathrm{C} \vee \mathrm{D} \vee \mathrm{F}, 1 \vee 2 \vee 3 \vee 4)$ with a different length bp in each different pair element node.

Second, type 2 in the DNA-digraph also represents a double-encoded substring that is each of the two different single element nodes $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$ and x_j. The two different types of element nodes for type 2, in which there is also a direction of the arrow that indicates the direction from the pair element node $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$ to the object element node x_j, are denoted as a double-encoded substring $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$-$3'^{upper} \rightarrow 5'^{upper}$-$x_j$. As shown in Fig. 2, in type 2, all the row and column labels in the binary adjacency matrix of the model DNA-digraph are denoted as i and j, and the entries are defined as

$$r_{i,j} = 1 \text{ for } i = 13, 14, \ldots, 23,\ j = 1, 2, \ldots, 8, \quad \text{and all } (i,j) \in L. \quad (12)$$

The two different element nodes shown above can also be encoded as a single oligonucleotide, including its own length of DNA bp as type 1; thus, (1) each pair element node $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$ corresponds to $(\mathrm{D} \vee \mathrm{F} \vee \mathrm{P}, 1 \vee 2 \vee 3 \vee 4) - 3'^{upper}$ with a different length bp in each different pair element node; and (2) each object element node x_j, $j = 1, 2, \ldots, 8$ was set with all nodes having the same length of 18 bp.

Third, type 3 in the DNA-digraph also represents a double-encoded substring that is each of the two different single pair element nodes $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$ and $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$. For type 3, the two different types of pair element nodes, in which there is a direction of the arrow that indicates the direction from the pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$ to the other pair element node $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$, are denoted as a double-encoded substring $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha}) - 3'^{upper} \rightarrow 5'^{upper} - (\zeta_\beta, \psi_\beta^{\zeta_\beta})$. In the binary adjacency matrix of the model DNA-digraph, as shown in Fig. 2, all the row and column labels are denoted as i and j for type 3, and the entries are defined as

$$r_{i,j} = 1 \text{ for } i = 9, 10, \ldots, 20, j = 13, 14, \ldots, 23, \quad \text{and all } (i,j) \in L. \quad (13)$$

As shown above, the two different pair element nodes can be encoded as a single oligonucleotide, including its own length of DNA bp as types 1 and 2; thus, (1) each pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$ corresponds to $(\mathrm{C} \vee \mathrm{D} \vee \mathrm{F}, 1 \vee 2 \vee 3 \vee 4) - 3'^{upper}$ with a different length bp in each different pair element node; and (2) each pair element node $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$ corresponds to $5'^{upper} - (\mathrm{D} \vee \mathrm{F} \vee \mathrm{P}, 1 \vee 2 \vee 3 \vee 4)$ with the different length bp in each different pair element node.

Complementary substrings correspond to types 4, 5, 6, and 7, which are used for connecting two of the double-encoded substrings (types 1 and 3, types 3 and 2, types 3 and 3, and types 2 and 1) since (1) the left double-encoded substring in a $5'$ to $3'$ direction includes the ending DNA sequence, which indicates the element node; and (2) the other right substring in a $5'$ to $3'$ direction includes the starting DNA sequence, and also indicates the same element node that the ending DNA sequences of the left substring indicates. The force of the hybridizations and DNA ligations [18] in the DNA-digraph induce all three different DNA substrings (three different element nodes linked by directions) to connect to one another so that they then line up sequentially and assume the form of double-stranded DNA (dsDNA). Meanwhile, types 4, 5, and 6 have the same purposes for their complementary substrings, which are particularly for attaching three different element nodes, which should contain

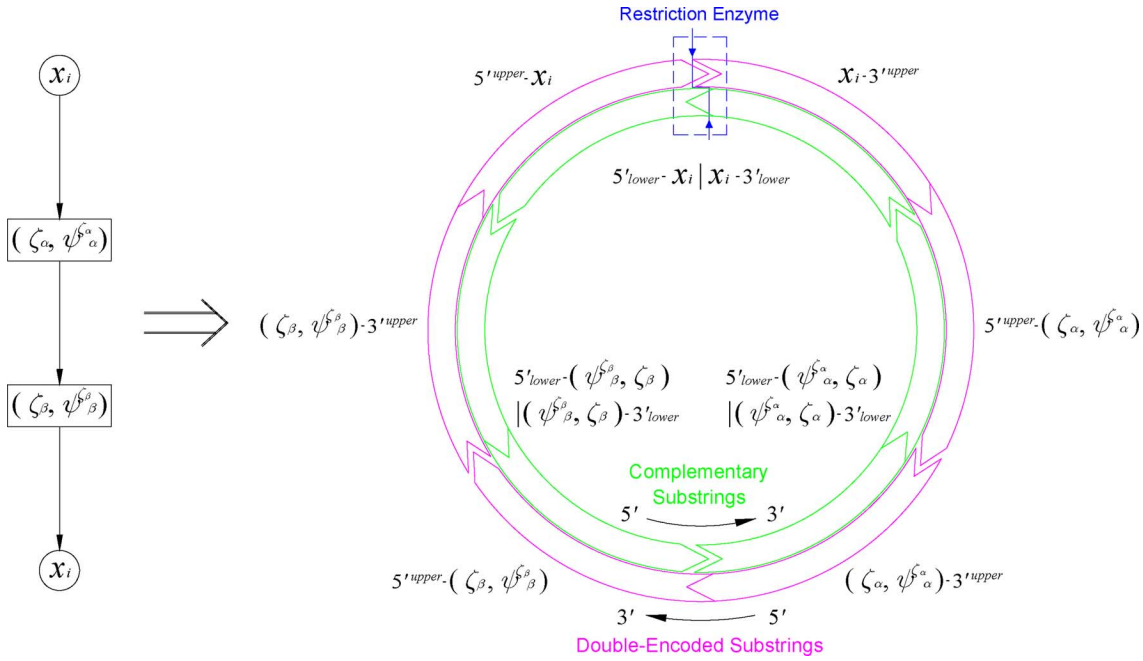

Fig. 4. Example representation of one object element node with two pair element nodes hybridized to become a circular DNA fragment.

a pair element node that is located in the middle of the three nodes. In particular, type 7 has its own purpose for the complementary substring for attaching three different element nodes, which should contain an object element node that is located in the middle of the three nodes.

Fourth, in the DNA-digraph, we created type 4 to attach three different linked element nodes, which are an object element node (starting element node) x_i, and two different pair element nodes (middle and ending element nodes) $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$ and $(\zeta_\beta, \psi_\beta^{\zeta\beta})$. For type 4, a complementary substring of the middle pair element node should be created to link together these three element nodes, and it is then denoted as a complementary encoding $5'_{lower} - (\psi_\alpha^{\zeta\alpha}, \zeta_\alpha)|(\psi_\alpha^{\zeta\alpha}, \zeta_\alpha) - 3'_{lower}$. For type 4, the complementary substring of the middle pair element node is denoted as $5'_{lower} - (1 \vee 2 \vee 3 \vee 4, \mathrm{C} \vee \mathrm{D} \vee \mathrm{F})|(1 \vee 2 \vee 3 \vee 4, \mathrm{C} \vee \mathrm{D} \vee \mathrm{F}) - 3'_{lower}$.

Fifth, type 5 in the DNA-digraph was created for the attachment of three different linked element nodes, which are two different pair element nodes (starting and middle element nodes) $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$ and $(\zeta_\beta, \zeta_\beta^{\zeta\beta})$, and an object element node (ending element node) x_j. A complementary substring of the middle pair element node for type 5 should also be created to link together these three element nodes, and it is denoted as a complementary encoding $5'_{lower} - (\psi_\beta^{\zeta\beta}, \zeta_\beta)|(\psi_\beta^{\zeta\beta}, \zeta_\beta) - 3'_{lower}$. For type 5, the complementary substring of the middle pair element node is denoted as $5'_{lower} - (1 \vee 2 \vee 3 \vee 4, \mathrm{D} \vee \mathrm{F} \vee \mathrm{P})|(1 \vee 2 \vee 3 \vee 4, \mathrm{D} \vee \mathrm{F} \vee \mathrm{P}) - 3'_{lower}$.

Sixth, in the DNA-digraph, we created type 6 to attach three different linked pair element nodes (starting, middle, and ending nodes), which are denoted as $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$, $(\zeta_\lambda, \psi_\lambda^{\zeta\lambda})$, and $(\zeta_\beta, \psi_\beta^{\zeta\beta})$, respectively. For type 6, a complementary substring of the middle pair element node should also be created to link together these three pair element nodes, and it is denoted as a complementary encoding $5'_{lower} - (\psi_\lambda^{\zeta\lambda}, \zeta_\lambda)|(\psi_\lambda^{\zeta\lambda}, \zeta_\lambda) - 3'_{lower}$. For type 6, the complementary substring of the middle pair element node is denoted as $5'_{lower} - (1 \vee 2 \vee 3 \vee 4, \mathrm{D} \vee \mathrm{F})|(1 \vee 2 \vee 3 \vee 4, \mathrm{D} \vee \mathrm{F}) - 3'_{lower}$.

Finally, type 7 in the DNA-digraph has a different form from the three types 4, 5, and 6 described above. Type 7 was created for the attachment of three different linked element nodes, which are two different pair element nodes (starting and ending element nodes) $(\zeta_\beta, \psi_\beta^{\zeta\beta})$ and $(\zeta_\alpha, \psi_\alpha^{\zeta\alpha})$, and an object element node (middle element node) that is denoted as x_λ. A complementary substring of the middle object element node for type 7 should also be created to link together these three element nodes, and it is denoted as a complementary encoding $5'_{lower} - x_\lambda | x_\lambda - 3'_{lower}$. For type 7, the complementary substring of the middle object element node is denoted as $5'_{lower} - 1 \vee 2 \vee \cdots \vee 8|1 \vee 2 \vee \cdots \vee 8 - 3'_{lower}$.

C. Studies for a Simulation Experiment

In this study, a splicing operation method [19] is applied to the DNA-based algorithm for the formation of minimal decision rules based on the DNA encoding process of the seven types described above. This splicing operation model concatenates the crosswise DNA fragments in order for them to be linked together through a DNA ligation process when there are two different kinds of encoded DNA sequences that must be properly spliced. The simulation molecular control program Vector NTI was used to represent the length of the DNA strands of all the minimized decision rules.

For the model DNA-digraph, each DNA substring pattern is described by the subsets of each object element node and pair element node, which correspond to the DNA sequences. Each complementary substring pattern is described by each concatenation of DNA fragments, in correspondence with the complementary substrings of both the object and the pair element

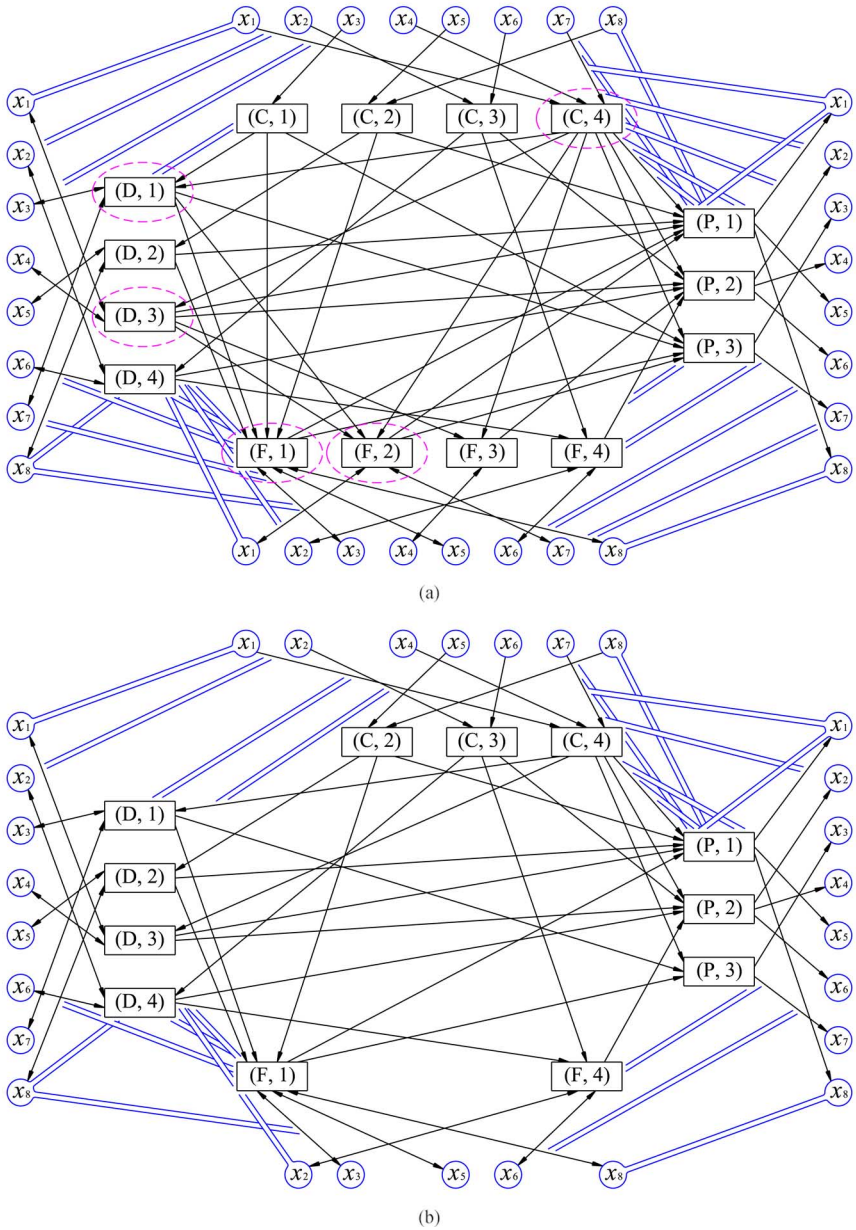

Fig. 5. Representations of two different model DNA-digraphs: (a) a model DNA-digraph built from the model decision table. In more detail, the dashed circle lines are shown in Fig. 6; (b) a newly built DNA-digraph after removing the first condition group of decision rules.

nodes. All the DNA substrings that are used for encoding the double-encoded substrings (types 1, 2, and 3) and their complementary substrings (types 4, 5, 6, and 7) should be generated. At the same time, in this encoding process, the fitting restriction enzymes have been added to each object element node of the DNA substrings and their complementary substrings. The main function of the restriction of enzymes is to cleave the DNA strands in a loop, and the length of the DNA strands (circular types) can be measured in another way. All of the detected circular DNA fragments should be cleaved and denatured, so they then become linear DNA fragments rather than circular DNA fragments for the length measurement process. The generation of DNA substrings is associated with large quantities of strands, i.e., at least twenty eight DNA strands in each DNA substring and complementary substring are generated for the model DNA-digraph, using an amplification technique of the polymerase chain reaction.

All of the possible double-encoded substrings among the element nodes in the model DNA-digraph and each complementary substring are generated to first detect subsets of the lower approximations in each decision class, and second, to detect all minimal length decision rules. The first detection process is mainly executed by measuring the length of the DNA strands among the given object element nodes, using the encoded types 3 and 6. The second detection process is mainly executed by detecting all the circular DNA fragments, by using the encoded types 1, 2, 4, 5, and 7 and reusing types 3 and 6. For example, Fig. 4 shows an example circular DNA fragment of an object element node x_i and two pair element nodes $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$ and $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$. The pair element nodes are connected to the object

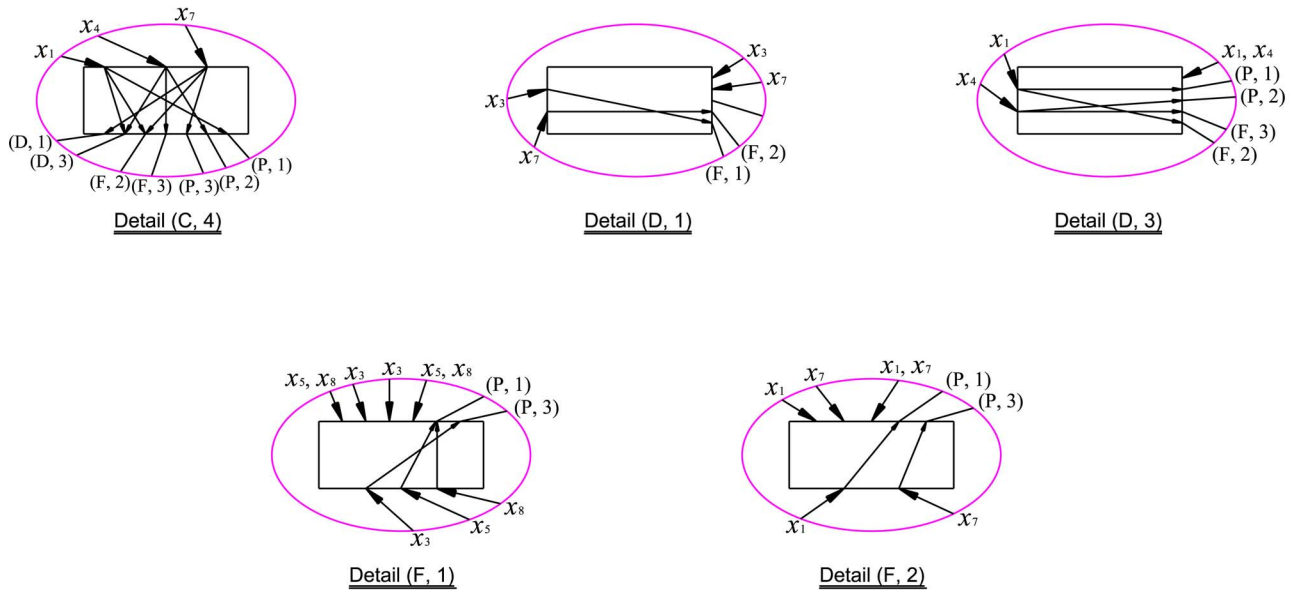

Fig. 6. Detail representations of the specific pair element nodes (C, 4), (D, 1), (D, 3), (F, 1), and (F, 2), in each of which the arcs are directed from the different object element nodes to the groups of the two or more pair element nodes at the same condition attribute.

element node, thus becoming the circular DNA fragment in hybridization. The procedure of the DNA-based algorithm described for the minimal decision rules is presented in more detail in the following steps:

1) Step 1 (Digraph-1): A DNA-digraph can be generated by the relations of each given object, including two or more pairs (condition attributes and condition attribute values) in its decision table, based on the structure of the DNA-digraph with n object element nodes and n pair element nodes, as shown in Fig. 1. Fig. 5(a) shows the model DNA-digraph, in which each DNA substring of the DNA sequence is composed of the arcs of specific pair element nodes from the different types of object element nodes to the same condition attributes, as shown in Fig. 6.

2) Step 2 (Encoding-1): For the DNA-digraph, the double-encoded substrings (from the given DNA substrings) and their complementary substrings can be encoded based upon the method of encoding the directed object and pair element nodes in DNA. In the model decision table, the DNA sequences of the existing arcs of the pairs (types 3 and 6) are generated first and encoded in ssDNA to identify subsets of the lower approximations in each of the decision classes.

3) Step 3 (Hybridization-1): All the encoded pair element node substrings (type 3) and their complementary substrings (type 6) are artificially synthesized and placed in a test tube. For this hybridization, the encoded pair element node substrings and their complementary substrings are heated to approximately $94°\mathrm{C}$ and then cooled to approximately $20°\mathrm{C}$ at intervals of $1°\mathrm{C/min}$. These are all hybridized based on the Watson-Crick complementary rules.

4) Step 4 (Simulated Gel Electrophoresis-1 and Removal-1): The DNA strands are parted according to their size by employing a simulated gel electrophoresis apparatus. The size of the parted DNA strands is then measured by comparing them to DNA strands of known length. The lengths of all the hybridized DNA strands should be measured by the simulated gel electrophoresis and should then be classified into each of two different decision classes for the model of the decision table. If there are two or more hybridized DNA strands whose measurements indicate the same length, this means that these hybridized DNA strands must correspond to objects that completely include the same condition attribute values, irrespective of decision values. The reason for this is that each condition attribute value was encoded using its own DNA strand length. For example, if two DNA strands that were measured as having the same length are not included in the same decision class, but rather, are separately included in each different decision class despite having the same length, then those two strands should be removed from all of the decision classes. After this removal process, each decision class of the remaining hybridized DNA strands can be classified into each subset of the lower approximation in each decision class. The DNA strands among the two given decision classes are removable groups of the same length, which cannot be included in either the lower approximation in Decision Class-1 or Class-2.

5) Step 5 (Denaturing): After determining each lower approximation in each decision class, the hybridized DNA strands of types 3 and 6 are heated to approximately $94°\mathrm{C}$ to become ssDNA in the status of dsDNA. All of the denatured ssDNA will be reused for the hybridization-2 and DNA ligation process in Step 8.

6) Step 6 (Marking-1 and Digraph-2): To reduce the number of DNA encoding sequences, it is necessary to find some object element node or nodes that only belong to one of the lower approximation subsets and are included in the same decision class, while only being connected to a specific pair element node together. This means that any of the other object element nodes, except the object element nodes, meet the above requirements

and are not connected to this specific pair element node. In this case, the arc subsets between object and pair element nodes are defined as

$$B_1 = \left\{ \overrightarrow{(x_i, (\zeta_\alpha, \psi^{\zeta_\alpha}{}_\alpha))} \in B^a | x_i \in \Xi_{\mathrm{A}}(D^v{}_\tau) \right\}, \quad (14)$$

which is denoted as B_1 and represents an arc subset from an object element node x_i to a pair element node $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$, and

$$B_2 = \left\{ \overrightarrow{((\zeta_\beta, \psi^{\zeta_\beta}{}_\beta), x_j)} \in B^b | x_j \in \Xi_{\mathrm{A}}(D^v{}_\tau) \right\}, \quad (15)$$

which is denoted as B_2 and represents an arc subset from a pair element node $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$ to an object element node x_j, in which each subset of the object element nodes, either x_i or x_j (or both of them), is included in one of the lower approximation subsets. At this point, each subset of the pair element nodes, either $(\zeta_\alpha, \psi_\alpha^{\zeta_\alpha})$ or $(\zeta_\beta, \psi_\beta^{\zeta_\beta})$ (or both of them), should first be clearly marked as a first condition group (a group of single pair element nodes) of decision rules in one of the decision classes. A new DNA-digraph can be constructed after all the subsets of these arcs (B_1 and B_2) are removed. This process can reduce the number of DNA encoding sequences used to obtain accurate minimal decision rules. Fig. 5(b) shows a new model DNA-digraph that has been constructed following the removal process.

7) Step 7 (Encoding-2): A new binary adjacency matrix that consists of the reduced number of object and pair element entries can be created by transforming the new model DNA-digraph that was constructed in Step 6, as shown in Fig. 5(b). All the double-encoded substrings and all of their complementary substrings can be encoded based on the method of encoding the object and pair element nodes in DNA. The DNA substrings of the existing arcs of the object and pair element nodes in the model DNA-digraph correspond to types 1, 2, 4, 5, and 7, which should be generated second and encoded in ssDNA. At the same time, the encoded DNA sequences of types 3 and 6 are reused in order to compute all the minimal length decision rules in each decision class.

8) Step 8 (Hybridization-2 and DNA Ligation): All the DNA sequence substrings of the encoded object and pair element nodes (types 1 and 2) and their complementary substrings (types 4, 5, and 7), as well as the added DNA sequence substrings of the pair element nodes (type 3) and their complementary substrings (type 6) are also artificially synthesized and placed in a test tube. For this hybridization, the DNA sequence substrings of the element nodes and their complementary substrings were hybridized in the process of Step 3. In this step, however, the DNA ligases should be added for the bonding of the different encoded DNA sequences to ensure their DNA ligations among ssDNA.

9) Step 9 (Cleavage and Affinity Separation-1): All the circular DNA fragments (one or more) should be discovered and distinguished from all of the hybridized and ligated DNA strands. Before these circular DNA fragments are classified into each of the object groups, the circular DNA fragments should be (1) cleaved at any one point by using restriction enzymes to create linear DNA fragments; and (2) heated to approximately 94°C to become ssDNA. A technique of affinity separation is used to make a classification of each object group, using the complementary substrings of all of the object element nodes with magnetic beads.

10) Step 10 (Simulated Gel Electrophoresis-2 and Removal-2): All the lengths of each object group of the cleaved and denatured DNA strands can be measured via the simulated gel electrophoresis. Each object group of the DNA strands should be classified in each single test tube and loaded in each lane. After the loading process, if there are two or more DNA strands that indicate the same length (the same bp), then these strands should be removed. However, if two or more DNA strands that correspond to two or more object element nodes, which are included in one of lower approximation subsets (as the results in the next section show), then one lower approximation subset is composed of Object-3 and 4, and the other lower approximation subset is composed of Object-1 and 7), then these distinguishable DNA strands should not be removed. The DNA strands across all the lanes are the removable groups of the same lengths. They are not included in the same lower approximation subset.

11) Step 11 (Affinity Separation-2 and Marking-2): After the removal process, in each decision class, each object of the remaining DNA fragments should contain two or more pair element nodes that have been clearly distinguished on the basis of the affinity separation method. For this process, the complementary substrings of all the pair element nodes with magnetic beads are also prepared. Each object in each lower approximation subset includes its own subset of pair element nodes that should be marked second, in order to become a second condition group composed of multiple pair element nodes of the decision rules in each decision class.

D. Study Results of the Simulated Experimentation

For the DNA sequence designs and sizes, as shown in Fig. 3, we created seven different types of encoding designs. Based on these encoding design types, we transformed the given decision table into a digraph that was composed of object and pair (condition attribute and condition attribute value) nodes, each of which had different DNA encoded with different lengths. All encoded DNA was manipulated according to the above experimental study steps.

The two processes of the simulated gel electrophoresis-1 and -2 were used to find the specific DNA strands, which represented subsets of the lower approximations and condition groups of each object in each decision class. Each lane included only its own group of DNA fragments in different lengths of DNA strands. We will now present more details of the experimental study results.

First, the length representation results of the simulated gel electrophoresis-1 for the first detection process showed the remaining DNA fragments. These were transformed into the two different subsets of the lower approximations in two different decision classes (1 and 2). As a result of the first detection process, the two removable groups that had the same length were (1) Object-2 (Decision Class-2) and Object-6 (Decision

Class-1); and (2) Object-5 (Decision Class-2) and Object-8 (Decision Class-1). Thus, all of the given objects in the model decision table were classified into subsets of the lower approximations in each decision class (the decision value of the decision attributes), which were represented in the directions of DNA sequencing as follows:

(1) Decision Class-1:
$\Xi_{\mathrm{A}}(D^{v}{}_{\tau 1}) = \{\text{Object-3}, \text{Object-4}\}$ derived from (Object-3)

$$\begin{aligned}(\mathrm{C},1)-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{D},1)|(\mathrm{D},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{F},1)|(\mathrm{F},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},3)\end{aligned}$$

at 762 bp and (Object-4)

$$\begin{aligned}(\mathrm{C},4)-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{D},3)|(\mathrm{D},3)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{F},3)|(\mathrm{F},3)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},2)\end{aligned}$$

at 1168 bp; and

(2) Decision Class-2:
$\Xi_{\mathrm{A}}(D^{v}{}_{\tau 2}) = \{\text{Object-1}, \text{Object-7}\}$ derived from (Object-1)

$$\begin{aligned}(\mathrm{C},4)-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{D},3)|(\mathrm{D},3)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{F},2)|(\mathrm{F},2)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},1)\end{aligned}$$

at 1128 bp and (Object-7)

$$\begin{aligned}(\mathrm{C},4)-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{D},1)|(\mathrm{D},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{F},2)|(\mathrm{F},2)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},3)\end{aligned}$$

at 1032 bp.

These two subsets (lower approximations) were retrieved in order to encode the minimal number of DNA fragments and to determine a first condition group (a group of single pair element nodes) of decision rules in each decision class, which was represented in DNA sequencing directions, based on the marking-1 process, as follows:

(1) Decision Class-1:
(Object-3) $x_3 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{C},1)$ and (Object-4) Both $x_4 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{F},3)$ and $(\mathrm{F},3) - 3'^{upper} \rightarrow 5'^{upper} - x_4$; and

(2) Decision Class-2:
(Object-1) Both $x_1 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{F},2)$ and $(\mathrm{F},2) - 3'^{upper} \rightarrow 5'^{upper} - x_1$ and (Object-7) Both $x_7 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{F},2)$ and $(\mathrm{F},2) - 3'^{upper} \rightarrow 5'^{upper} - x_7$.

Second, the length representation results of the simulated gel electrophoresis-2 for the second detection process showed the remaining DNA fragments. These were transformed into four different subsets of the pair element nodes in four different objects (Object-1, -3, -4, and -7). Here, (1) Object-3 and -4 were a subset of the lower approximation in Decision Class-1; and (2) the Object-1 and -7 were the other subset of the lower approximation in Decision Class-2. The four subsets of the pair element nodes in the four different objects, which belonged to two different decision classes, were determined as a second condition group (a group of multiple pair element nodes) of decision rules in each decision class, which was represented in the DNA sequencing directions, based on the marking-2 process, as follows:

(1) Decision Class-1:
(Object-3)

$$\begin{aligned}x_3-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{D},1)|(\mathrm{D},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{F},1)|(\mathrm{F},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},3)|(\mathrm{P},3)-3'^{upper}\rightarrow 5'^{upper}-x_3\end{aligned}$$

at 568 bp, (Object-3)

$$\begin{aligned}x_3-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{D},1)|(\mathrm{D},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{F},1)|(\mathrm{F},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-x_3\end{aligned}$$

at 472 bp, (Object-3)

$$\begin{aligned}x_3-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{F},1)|(\mathrm{F},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},3)|(\mathrm{P},3)-3'^{upper}\\ &\rightarrow 5'^{upper}-x_3\end{aligned}$$

at 280 bp, (Object-4)

$$\begin{aligned}x_4-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{C},4)|(\mathrm{C},4)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{D},3)|(\mathrm{D},3)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},2)|(\mathrm{P},2)-3'^{upper}\\ &\rightarrow 5'^{upper}-x_4\end{aligned}$$

at 1556 bp, (Object-4)

$$\begin{aligned}x_4-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{C},4)|(\mathrm{C},4)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},2)|(\mathrm{P},2)-3'^{upper}\\ &\rightarrow 5'^{upper}-x_4\end{aligned}$$

at 1148 bp, and (Object-4)

$$\begin{aligned}x_4-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{D},3)|(\mathrm{D},3)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},2)|(\mathrm{P},2)-3'^{upper}\rightarrow 5'^{upper}-x_4\end{aligned}$$

at 516 bp; and

(2) Decision Class-2:
(Object-1)

$$\begin{aligned}x_1-3'^{upper} &\rightarrow 5'^{upper}-(\mathrm{C},4)|(\mathrm{C},4)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{D},3)|(\mathrm{D},3)-3'^{upper}\\ &\rightarrow 5'^{upper}-(\mathrm{P},1)|(\mathrm{P},1)-3'^{upper}\\ &\rightarrow 5'^{upper}-x_1\end{aligned}$$

at 1532 bp, (Object-1)

$$x_1 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{C},4)|(\mathrm{C},4) - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{P},1)|(\mathrm{P},1) - 3'^{upper} \rightarrow 5'^{upper} - x_1$$

at 1124 bp, (Object-1)

$$x_1 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{D},3)|(\mathrm{D},3) - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{P},1)|(\mathrm{P},1) - 3'^{upper} \rightarrow 5'^{upper} - x_1$$

at 492 bp, (Object-7)

$$x_7 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{C},4)|(\mathrm{C},4) - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{D},1)|(\mathrm{D},1) - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{P},3)|(\mathrm{P},3) - 3'^{upper} \rightarrow 5'^{upper} - x_7$$

at 1460 bp, (Object-7)

$$x_7 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{C},4)|(\mathrm{C},4) - 3'^{upper} \rightarrow 5^{upper} - (\mathrm{D},1)|(\mathrm{D},1) - 3'^{upper} \rightarrow 5'^{upper} - x_7$$

at 1364 bp, and (Object-7)

$$x_7 - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{C},4)|(\mathrm{C},4) - 3'^{upper} \rightarrow 5'^{upper} - (\mathrm{P},3)|(\mathrm{P},3) - 3'^{upper} 5'^{upper} - x_7$$

at 1172 bp.

Each of the above resolved groups of decision rules had their own exact lengths, as each of the two condition groups of decision rules in each of the two decision classes.

All things considered, the results of the DNA-based algorithm indicated the two different condition groups of decision rules in two different classes, which were transformed into two subsets of *if-then* rules as follows:

(1) Decision Class-1:
if color is blue, *then* Target-1 is selected; *if* feature is a bit more complex, *then* Target-1 is selected; *if* density is very low and feature is simple, *then* Target-1 is selected; *if* feature is simple and pattern is gas, *then* Target-1 is selected; *if* color is yellow and pattern is liquid, *then* Target-1 is selected; and *if* density is high and pattern is liquid, *then* Target-1 is selected; and

(2) Decision Class-2:
if feature is a bit simpler, *then* Target-2 is selected; *if* color is yellow and pattern is solid, *then* Target-2 is selected; *if* density is high and pattern is solid, *then* Target-2 is selected; *if* color is yellow and density is very low, *then* Target-2 is selected; and *if* color is yellow and pattern is gas, *then* Target-2 is selected.

The above *if-then* decision rules in each decision class were the minimized decision rules and the minimal lengths of decision rules, which were finally determined by the DNA-based algorithm.

At the same time, as demonstrated by the obtained results, the DNA-based algorithm could be extended to deal with a large number of object sets and their condition attribute sets by taking advantage of the main characteristics of the DNA encoding, though this would require the installation of a large amount of instrumentation at much expense. The proposed DNA-based algorithm could be used to support super-parallel processing that is used for the minimization of decision rules in rough sets, or eligible sets with large numbers of objects and their condition attributes.

IV. Conclusions

Based on the obtained results, we are able to arrive at several general conclusions.

First, we have developed a novel way of handling the information present in decision tables via a DNA-based algorithm, which was used for the efficient discovery of subsets of the lower approximations, which were classified into decision classes, and for the minimization of decision rules.

Second, in this DNA-based algorithm, a novel encoding method encoded two completely different characteristics of both the object and pair element nodes in DNA. This has been proposed and adapted in order to create a new splicing operation.

Finally, the proposed encoding method based on our DNA-based algorithm made the best use of the finding of subsets of the lower approximations and minimal decision rules. It may be possible for the proposed DNA-based methods to be employed along with computer-based technologies to enhance cooperative data processing, and thus lead to the construction of a new knowledge support system.

References

[1] R. Slowinski, *Intelligent Decision Support: Handbook of Application and Advances of the Rough Sets Theory*. Norwell, MA: Kluwer, 1992, pp. 363–372.

[2] J. W. Grzymala-Busse, "A comparison of tree strategies to rule induction from data with numerical attributes," *Electron. Notes Theor. Comput. Sci.*, vol. 82, no. 4, pp. 132–140, 2003.

[3] W. Ziarko, "Variable precision rough set model," *J. Comput. Syst. Sci.*, vol. 46, no. 1, pp. 39–59, 1993.

[4] A. Skowron and C. Rauszer, "The discernibility matrices and functions in information systems," in *Intelligent Decision Support: Handbook of Application and Advances of the Rough Sets Theory*. Norwell, MA: Kluwer, 1992, pp. 331–362.

[5] N. Shan and W. Ziarko, "Data-based acquisition and incremental modification of classification rules," *Comput. Intell.*, vol. 11, no. 2, pp. 357–370, 1995.

[6] L. Polkowski, *Advances in Soft Computing: Rough Sets*. Heidelberg, Germany: Physica-Verlag, 2002, pp. 18–45.

[7] L. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, no. 11, pp. 1021–1024, 1994.

[8] I. Kim and J. Watada, "Decision making with an interpretive structural modeling method using a DNA-based algorithm," *IEEE Trans. NanoBiosci.*, vol. 8, no. 2, pp. 181–191, 2009.

[9] I. Kim and J. Watada, "A fuzzy density analysis of subgroups by means of DNA oligonucleotides," in *Intelligent Systems and Technologies, Methods and Applications, Studies in Computational Intelligence*. Berlin, Germany: Springer-Verlag, 2009, vol. 217, pp. 31–45.

[10] I. Kim, J. Watada, and W. Pedrycz, "A DNA-based algorithm for arranging weighted cliques," *Simul. Model. Practice Theory*, vol. 16, no. 10, pp. 1561–1570, 2008.

[11] J. A. Rose, R. J. Deaton, M. Hagiya, and A. Suyama, "Coupled equilibrium model of hybridization error for the DNA microarray and tag-antitag systems," *IEEE Trans. NanoBiosci.*, vol. 6, no. 1, pp. 18–27, 2007.

[12] S. A. Tsaftaris and A. K. Katsaggelos, "Retrieval efficiency of DNA-based databases of digital signals," *IEEE Trans. NanoBiosci.*, vol. 8, no. 3, pp. 259–270, 2009.

[13] J. Watada, "DNA computing and its application," in *Computational Intelligence: A Compendium*, J. Fulcher and L. C. Jain, Eds. Berlin, Germay: Springer-Verlag, 2008, pp. 1065–1086.
[14] Z. Pawlak, "Rough sets," *Int. J. Comput. Inf. Sci.*, vol. 11, no. 5, pp. 341–356, 1982.
[15] Z. Pawlak, "Rough classification," *Int. J. Man-Mach. Studies*, vol. 20, no. 5, pp. 469–483, 1984.
[16] Z. Pawlak, S. K. M. Wong, and W. Ziarko, "Rough sets: Probabilistic versus deterministic approach," *Int. J. Man-Mach. Studies*, vol. 29, no. 1, pp. 81–95, 1988.
[17] J. F. Peters and A. Skowron, *Transactions on Rough Sets VI*. Berlin, Germay: Springer-Verlag, 2007, pp. 351–396.
[18] P. J. Smith and C. J. Jones, *DNA Recombination and Repair*. New York: Oxford Univ. Press, 1999, pp. 112–129.
[19] I. Antoniou, C. S. Calude, and M. J. Dinneen, *Unconventional Models of Computation: Discrete Mathematics and Theoretical Computer Science*. Berlin, Germay: Springer-Verlag, 2000, pp. 68–84.

Ikno Kim (S'08–M'11) received the B.S. degree in technology management from Osaka Institute of Technology, Japan, and the M.S. and Ph.D. degrees in information, production and systems engineering from Waseda University, Japan.

He was a Mechanical and Production Systems Designer at Dongjin Machinery Corporation, Ltd., where he was responsible for designing and developing many different kinds of automatic machines. He is currently a Visiting Researcher in the Graduate School of Information, Production and Systems, Waseda University, Japan. His current research interests include biological computation, industrial management, nanobioscience, operations research, and soft computing.

Dr. Kim received the Excellent Presentation Prize from the Czech-Japan Symposium in 2006, the Excellent Master's Thesis Prize from the Japan Industrial Management Association (JIMA), Kyushu Branch in 2007, and the Hibikino Award for the Excellent Master's Thesis from the Kitakyushu Foundation for the Advancement of Industry Science and Technology (FAIS), Japan in 2007.

Yu-Yi Chu received the medical science degree from Taipei Medical University, Taiwan. She is currently working toward the Ph.D. degreee in the Graduate School of Information, Production and Systems, Waseda University, Japan.

She is now joining the collaboration research with the Taipei Medical University on Biocomputing technology. She majors in medical technology and computing technology. Her research interest includes fuzzy analysis, biocomputing, and biosensor applications.

Junzo Watada (M'87) received the B.Sc. and M.Sc. degrees in electrical engineering from Osaka City University, and the Ph.D. degree from Osaka Prefecture University.

He is currently a Professor of management engineering, knowledge engineering, and soft computing in the Graduate School of Information, Production and Systems, Waseda University, Japan. He is the Principal Editor, a Co-Editor, and an Associate Editor of various international journals, including the *ICIC Express Letters*, the *International Journal of Systems and Control Engineering*, and *Fuzzy Optimization and Decision Making*. His current research interests include soft computing, tracking systems, knowledge engineering, and management engineering.

Dr. Watada received the Henri Coanda Medal Award from Inventico in Romania in 2002. He is a Life Fellow of the Japan Society for Fuzzy Theory.

Jui-Yu Wu received the Doctor of Chemistry (Biochemistry) from Wayne State University, Detroit, MI, in 2002.

He is currently an Assistant Professor of Department of Biochemistry, School of Medicine at Taipei Medical University, Taiwan. He is also an Adjunct Assistant Professor of Department of Material sciences and Engineering at National Taiwan University of Science and Technology, and an Associate Editor of the *International Journal of Biomedical Soft Computing and Human Sciences*. His research interests include chemical biology, biomolecular computing, nucleic acid chemistry, and biosensor applications.

Witold Pedrycz (M'88–SM'90–F'99) is Professor and Canada Research Chair (CRC—Computational Intelligence) in the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada. He is also with the Systems Research Institute of the Polish Academy of Sciences, Warsaw, Poland. He main research directions involve computational intelligence, fuzzy modeling and granular computing, knowledge discovery and data mining, fuzzy control, pattern recognition, knowledge-based neural networks, relational computing, and software engineering. He has published numerous papers in this area. He is also an author of 15 research monographs covering various aspects of computational intelligence and software engineering.

Dr. Pedrycz has been a member of numerous program committees of IEEE conferences in the area of fuzzy sets and neurocomputing. In 2009 he was elected a foreign member of the Polish Academy of Sciences. He is intensively involved in editorial activities. He is an Editor-in-Chief of *Information Sciences* and Editor-in-Chief of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A. He currently serves as an Associate Editor of the IEEE TRANSACTIONS ON FUZZY SYSTEMS and a number of other international journals. In 2007 he received a prestigious Norbert Wiener award from the IEEE Systems, Man, and Cybernetics Council. He is a recipient of the IEEE Canada Computer Engineering Medal. In 2009 he has received a Cajastur Prize for Soft Computing from the European Centre for Soft Computing for "pioneering and multifaceted contributions to Granular Computing."