# Division Charts as Granules and Their Merging Algorithm for Rule Generation in Nondeterministic Data

| | |
|---|---|
| 著者 | Sakai Hiroshi, Wu Mao, Nakata Michinori |
| journal or publication title | International Journal of Intelligent Systems |
| volume | 28 |
| number | 9 |
| page range | 865-882 |
| year | 2013-06-03 |
| URL | http://hdl.handle.net/10228/00006137 |

doi: info:doi/10.1002/int.21608

# Division Charts as Granules and Their Merging Algorithm for Rule Generation in Non-deterministic Data

Hiroshi Sakai [a], Mao Wu [b], and Michinori Nakata [c]

[a] *Department of Basic Sciences, Faculty of Engineering, Kyushu Institute of Technology, Tobata, Kitakyushu 804-8550, Japan*
*email: sakai@mns.kyutech.ac.jp*

[b] *Department of Integrated System Engineering, Graduate School of Engineering, Kyushu Institute of Technology, Tobata, Kitakyushu, 804-8550, Japan*
*email: wumogaku@yahoo.co.jp*

[c] *Faculty of Management and Information Science, Josai International University, Gumyo, Togane, Chiba 283, Japan*
*email: nakatam@ieee.org*

Abstract:
We have been proposing a framework *Rough Non-deterministic Information Analysis (RNIA)*, which considers granular computing concepts in tables with incomplete and non-deterministic information, as well as rule generation. We have recently defined an expression named *division chart* with respect to an implication and a subset of objects. Each division chart takes the role of the minimum granule for rule generation, and it takes the role of contingency table in statistics. In this paper, we at first define a division chart in *Deterministic Information Systems (DISs)*, and clarify the relation between a division chart and a corresponding implication. We also consider a merging algorithm for two division charts, and extend the relation in *DISs* to *Non-deterministic Information Systems (NISs)*. The relation gives us the foundations of rule generation in tables with non-deterministic information.

Keywords: Granular computing; Rough sets; Division charts; Contingency table; Non-deterministic information; Rule generation

## 1 Introduction

Rough set theory offers a mathematical approach to vagueness and uncertainty, and the rough sets based concepts have been recognized to be very useful [3,10,12]. This theory usually handles tables with deterministic information, which we call *Deterministic Information Systems (DISs)*. Many applications of this theory to information analysis, data mining, rule generation, machine learning and knowledge discovery have been investigated [14,15].

Granular computing [5,13], which covers several computing frameworks, is now investigated about its possibility and applicability. Rough sets seem a special case of granular computing, and it is very important research to extend several useful concepts in rough sets to concepts in granular computing.

*Non-deterministic Information Systems (NISs)* and *Incomplete Information Systems (IISs)* were proposed for handling information incompleteness in *DISs* [2,4,6–8,10]. *NISs* and *IISs* are known as the important framework for handling information incompleteness in tables, and a lot of theoretical work has been reported. We followed this robust framework, and we have been developing algorithms and software tools. We are simply calling this work *Rough Non-deterministic Information Analysis (RNIA)* [18–20].

In this paper, we newly define an expression, which we name *Division Chart*. For any descriptors $[A, val_A]$ and $[B, val_B]$, we define a division chart $DC([A, val_A], [B, val_B])$, and this division chart shows us most of information for an implication $[A, val_A] \Rightarrow [B, val_B]$. Namely, this division chart takes the role of the contingency table in statistics, and we may see a division chart is a contingency table in rule generation. We also define a division chart $DC(\wedge_i [A_i, val_i], [B, val_B])$ recursively.

For this recursive definition, we propose a merging algorithm to obtain a division chart $DC([A, val_A] \wedge [C, val_C], [B, val_B])$ from $DC([A, val_A], [B, val_B])$ and $DC([C, val_C], [B, val_B])$. This will be useful to know information for rule generation, namely we can handle each attribute independently. When we need to consider a set of attributes, we employ this merging algorithm.

The paper is organized as follows: Section 2 considers division charts in $DISs$, and clarifies the concept of consistency by division charts. Furthermore, we newly introduce a merge of two division charts. Section 3 extends each issue in $DISs$ to $NISs$, and consider rule generation by division charts instead of previously defined $inf$ and *sup* blocks. Finally, Section 4 concludes this paper. In the appendix, we show examples by the current software tool for division charts.

## 2 Preliminary and Division Charts in DISs

This section reviews $DISs$ and some definitions, then we consider division charts in $DISs$.

### 2.1 Definitions in DISs

A *Deterministic Information System* (*DIS*) [9,12,11] $\psi$ is a quadruplet:

$$\psi = (OB, AT, \{VAL_A |\ A \in AT\}, f),$$

where $OB$ is a finite set whose elements are called *objects*, $AT$ is a finite set whose elements are called *attributes*, $VAL_A$ is a finite set whose elements are called *attribute values* and $f$ is such a mapping:

$$f : OB \times AT \to \cup_{A \in AT} VAL_A.$$

We usually consider tabular representation of $\psi$ like Table 1.

A pair $[A, v]$ ($A \in AT$, $v \in VAL_A$) is called a *descriptor*, and each candidate of rule is defined by these descriptors. For a descriptor $[A, v]$, let $[x]_{[A,v]}$ denote an equivalence class below:

Table 1
An Exemplary Deterministic Information System $\psi_1$.

| $OB$ | *temperature* | *headache* | *nausea* | *flu* |
|---|---|---|---|---|
| 1 | *very_high* | *yes* | *yes* | *yes* |
| 2 | *high* | *yes* | *yes* | *yes* |
| 3 | *normal* | *yes* | *yes* | *no* |
| 4 | *very_high* | *yes* | *no* | *yes* |
| 5 | *very_high* | *yes* | *yes* | *yes* |
| 6 | *high* | *no* | *no* | *no* |
| 7 | *normal* | *no* | *yes* | *no* |
| 8 | *high* | *no* | *no* | *no* |

$$[x]_{[A,v]} = \{y \in OB \mid f(y, A) = f(x, A) = v\}.$$

An equivalence class $[x]_{\wedge_i [A_i, v_i]}$ for a conjunction of descriptors is defined by $\cap_i [x]_{[A_i, v_i]}$.

We often consider two disjoint sets: $CON \subseteq AT$ which we call *condition attributes* and $DEC \subseteq AT$ which we call *decision attributes*. Usually, $DEC$ is a singleton set $\{Dec\}$. An implication for attributes $CON$ and $DEC = \{Dec\}$ is generally a formula $\tau$ in the following form:

$$\tau : \wedge_{A \in CON}[A, val_A] \Rightarrow [Dec, val],$$
$$(val_A \in VAL_A, DEC = \{Dec\} \subseteq AT, val \in VAL_{Dec}).$$

In most of work on rule generation, we try to obtain a set of appropriate implications defined above. For simplifying the notation, let $[CON, val_{CON}]$ denote $\wedge_{A \in CON}[A, val_A]$, and we handle an implication $\tau : [CON, val_{CON}] \Rightarrow [Dec, val]$.

An object $x \in OB$ is *consistent* with any $y \in OB$,

if $f(x, A) = f(y, A)$ for every $A \in CON$
implies $f(x, Dec) = f(y, Dec)$.

If object $x$ is consistent, we also say an implication $\tau$ defined by object $x$ is *consistent*. In order to specify an object $x$ defining $\tau$, we may employ a notation $\tau^x$.

In [9,12,11], a rule is defined by a consistent implication, and the problem of rule generation is converted to the problem on reduction of attributes and attribute values. However, the definition by consistency is slightly strong, therefore the following criteria were also introduced into each implication $\tau^x$.

$$\begin{aligned}&support(\tau^x) = |[x]_{[CON,val_{CON}]} \cap [x]_{[Dec,val]}|/|OB|,\\ &accuracy(\tau^x) = |[x]_{[CON,val_{CON}]} \cap [x]_{[Dec,val]}|/|[x]_{[CON,val_{CON}]}|.\end{aligned}$$

For threshold values $\alpha$ and $\beta$ ($0 < \alpha, \beta \leq 1.0$), if $support(\tau^x) \geq \alpha$ and $accuracy(\tau^x) \geq \beta$ hold, we define that this $\tau^x$ is a rule. The case $\beta = 1.0$ corresponds to the rule definition by consistency, namely the rule definition by two criteria is more general than that by consistency. This paper follows these definitions of rules.

**Proposition 1** *In a DIS, let $\tau^x$ and $\eta^y$ be implications for $x, y \in [x]_{CON\cup\{Dec\}}$. Then, $\eta$ is equal to $\tau$. Furthermore, the following holds.*

$$support(\eta^y) = support(\tau^x),\ accuracy(\eta^y) = accuracy(\tau^x).$$

**(Proof)**
*$y \in [x]_{CON\cup\{Dec\}}$ means that $f(y, A) = f(x, A)$ for each $A \in CON \cup \{Dec\}$. Therefore,*

$$\begin{aligned}&\eta^y = (\wedge_{A\in CON}[A, f(y, A)] \Rightarrow [Dec, f(y, Dec)])\\ &= (\wedge_{A\in CON}[A, f(x, A)] \Rightarrow [Dec, f(x, Dec)]) = \tau^x.\end{aligned}$$

*According to the above equation, two equations on support and accuracy clearly hold.* □

Proposition 1 seems trivial in *DISs*. Since $x$ and $y$ belong to $[x]_{CON\cup\{Dec\}}$, we may consider any $\tau^y$ for calculating *support* and *accuracy*. However this property may not hold in *NISs*.

Now, we consider to divide each equivalence class $[x]_{[CON,val_{CON}]}$ by $[Dec, val]$. Here, we can consider the next two types of implications for each $y \in [x]_{[CON,val_{CON}]}$ in Table 2.

Table 2
A table for obtainable implications from $[x]_{[CON,val_{CON}]}$ with respect to $[Dec, val]$. Here, $val_{CON} \Rightarrow val$ means $[CON, val_{CON}] \Rightarrow [Dec, val]$.

| *Case* | *CON* | $\{Dec\}$ | *Implications* |
|---|---|---|---|
| 1 | $val_{CON}$ | $val$ | $\tau : val_{CON} \Rightarrow val$ |
| 2 | $val_{CON}$ | $val'$ $(val' \neq val)$ | $\eta : val_{CON} \Rightarrow val'$ |

For two cases 1 and 2 in Table 2, we define ① and ② below, and we name them *components*.

$$① = \{y \in [x]_{[CON,val_{CON}]} |\ y\ defines\ \tau\},$$
$$② = \{y \in [x]_{[CON,val_{CON}]} |\ y\ defines\ \eta\}.$$

Clearly, $[x]_{[CON,val_{CON}]} = ① \cup ②$ holds.

Fig. 1 shows this division, and we name this figure *Division Chart* $DC([CON, val_{CON}], val)$ of $[x]_{[CON,val_{CON}]}$ by $[Dec, val]$.

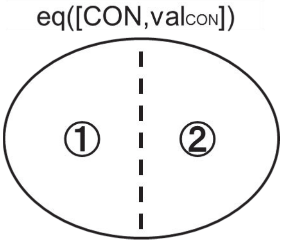


Fig. 1. A division chart $DC([CON, val_{CON}], val)$ of $[x]_{[CON,val_{CON}]}$ by $[Dec, val]$ in a *DIS*.

As for criteria $support(\tau^x)$ and $accuracy(\tau^x)$ $(x \in ①)$, we easily obtain the following by using components.

**Proposition 2** *We have the following.*
$support(\tau^x) = |①|/|OB|$, $(|①| :$ *cardinality*$)$,
$accuracy(\tau^x) = |①|/(|①| + |②|)$.

Furthermore, we can characterize the following property of the consistency in $DISs$ by using the condition of division charts, which show us visual information.

**Proposition 3** *[20,21] For descriptors $[CON, val_{CON}]$ and $[Dec, val]$ in a $DIS$, (1), (2), (3) and (4) in the following are equivalent.*
*(1) Each object $y \in [x]_{[CON, val_{CON}]}$ is consistent.*
*(2) $[x]_{[CON, val_{CON}]} \subseteq [x]_{[Dec, val]}$.*
*(3) The component ② $= \emptyset$ in $DC([CON, val_{CON}], val)$.*
*(4) $accuracy(val_{CON} \Rightarrow val)$=1.0.*

**Remark 1** *In rough sets and granular computing, we often employ equivalence classes, and division charts also take the role of equivalence classes according to Proposition 3. We try to employ division charts for handling rough sets-based concepts instead of equivalence classes.*

### *2.3 A Merged Division Chart by Two Division Charts in DISs*

Now, we consider merging two division charts. Let us consider descriptors $[CON1, val_{CON1}]$, $[CON2, val_{CON2}]$ ($CON1 \cap CON2 = \emptyset$) and $[Dec, val]$. For two division charts with the same decision attribute values $DC([CON1, val_{CON1}], val)$, $DC([CON2, val_{CON2}], val)$, we generate $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$. This merged division chart means a chart with respect to an implication below:

$$\tau : [CON1, val_{CON1}] \wedge [CON2, val_{CON2}] \Rightarrow [Dec, val].$$

In this case, we have Table 3 related to obtainable implications.

Table 3
A table for obtainable implications from $[x]_{[CON1, val_{CON1}] \wedge [CON2, val_{CON2}]}$ with respect to $[Dec, val]$.

| *Case* | *CON* | *DEC* | *Implications* |
|---|---|---|---|
| 1 | $(val_{CON1}, val_{CON2})$ | $val$ | $\tau : (val_{CON1}, val_{CON2}) \Rightarrow val$ |
| 2 | $(val_{CON1}, val_{CON2})$ | $val'$ $(val' \neq val)$ | $\eta : (val_{CON1}, val_{CON2}) \Rightarrow val'$ |

In Table 3, we also define ①, ② and $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$ in Fig. 2.

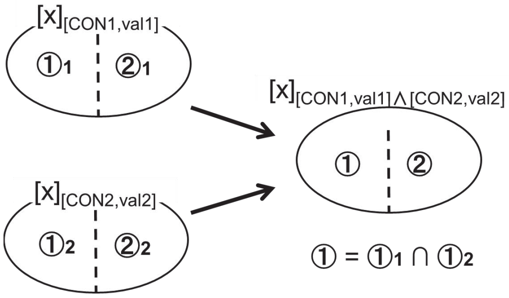


Fig. 2. A merged division chart of $[x]_{val_{CON1} \wedge val_{CON2}}$ by $[Dec, val]$ in a $DIS$.

For this merged division chart, we have the following proposition.

**Proposition 4** *For $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$ in Fig.2, the following holds.*

$$\textcircled{1} = \textcircled{1}_1 \cap \textcircled{1}_2,$$
$$\textcircled{2} = ([x]_{[CON1, val_{CON1}]} \cap [x]_{[CON2, val_{CON2}]}) \setminus \textcircled{1}.$$

Since Proposition 2 and 3 also holds in this merged division chart in Fig. 2, we can recursively apply this merging process for obtaining rules.

**Example 1** *Let us consider $\psi_1$ in Table 1, and we fix $\alpha = 0.0$ and $\beta = 0.9$.*
*(CASE 1) For $DC([temperature, high], yes)$ corresponding to an implication $\tau_1 : [temperature, high] \Rightarrow [flu, yes]$, $\textcircled{1}_1 = \{2\}$ and $\textcircled{2}_1 = \{6, 8\}$ holds. Here, $support(\tau_1) = |\textcircled{1}_1|/8 = 1/8 > \alpha$ and $accuracy(\tau_1) = |\textcircled{1}_1|/(|\textcircled{1}_1| + |\textcircled{2}_1|) = 1/3 < \beta$. Therefore, this $\tau_1$ is not recognized as a rule.*
*(CASE 2) Similarly, for $DC([headache, yes], yes)$ corresponding to an implication $\tau_2 : [headache, yes] \Rightarrow [flu, yes]$, $\textcircled{1}_2 = \{1, 2, 4, 5\}$ and $\textcircled{2}_2 = \{3\}$ holds. Here, $accuracy(\tau_2) = |\textcircled{1}_2|/(|\textcircled{1}_2| + |\textcircled{2}_2|) = 4/5 < \beta$. Thus, $\tau_2$ is not recognized as a rule, either.*
*(CASE 3) Then, we consider a merged division chart corresponding to an implication $\tau_{1,2} : [temperature, high] \wedge [headache, yes] \Rightarrow [flu, yes]$. Here, $[2]_{[temperature, high] \wedge [headache, yes]} = \{2\}$, $\textcircled{1} = \textcircled{1}_1 \cap \textcircled{1}_2 = \{2\} \cap \{1, 2, 4, 5\} = \{2\}$, $\textcircled{2} = \{2\} \setminus \textcircled{1} = \emptyset$, $support(\tau_{1,2}) =$*

$|①|/8 = 1/8 > \alpha$, *accuracy*$(\tau_{1,2}) = |①|/(|①| + |②|) = 1/1 > \beta$. *Like this, we examine the criterion values of rules by using each merged division chart.*

**Remark 2** *In a DIS $\psi$, we consider the following set $GR(\psi)$:*

$$GR(\psi) = \{DC([A, val_A], val) \mid A \in AT, val_A \in VAL_A,\\ val \in VAL_{Dec}\}.$$

*We may see each division chart a granule for rule generation in $\psi$, and we can pick up any rule by using the merging process of these granules. This process is similar to Apriori algorithm [1] defined in the transaction data.*

In the following sections, we extend the property of division charts in $DISs$ to the property in $NISs$.

## 3 Division Charts in NISs and Rule Generation

This section reviews $NISs$ and some definitions, then we consider division charts in $NISs$ and rule generation.

### 3.1 Definitions in NISs

A *Non-deterministic Information System* ($NIS$) [8,10] $\Phi$ is also a quadruplet:

$$\Phi = (OB, AT, \{VAL_A | A \in AT\}, g),$$

where $g$ is such a mapping:

$$g : OB \times AT \to P(\cup_{A \in AT} VAL_A)$$
(a power set of $\cup_{A \in AT} VAL_A$).

Every set $g(x, A)$ is interpreted as that there is an actual value in it but it is not known. We usually consider tabular representation of $\Phi$ like Table 4.

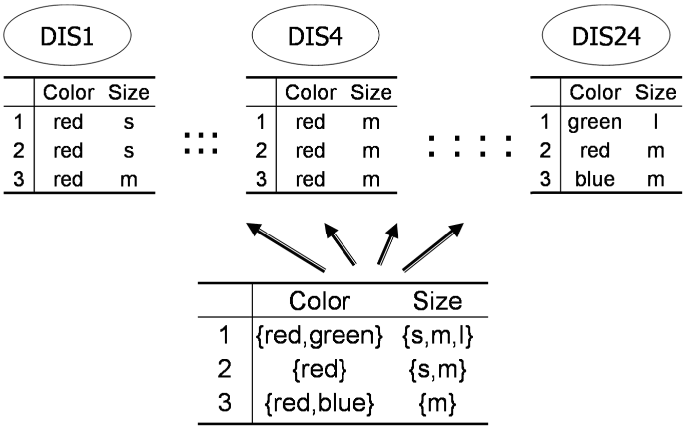
Table 4
An Exemplary Non-deterministic Information System $\Phi_1$.

| $OB$ | $temperature$ | $headache$ | $nausea$ | $flu$ |
|---|---|---|---|---|
| 1 | $\{very\_high\}$ | $\{yes, no\}$ | $\{yes\}$ | $\{yes\}$ |
| 2 | $\{high, very\_high\}$ | $\{yes\}$ | $\{yes\}$ | $\{yes\}$ |
| 3 | $\{normal, high\}$ | $\{yes\}$ | $\{yes\}$ | $\{yes, no\}$ |
| 4 | $\{very\_high\}$ | $\{yes\}$ | $\{yes, no\}$ | $\{yes\}$ |
| 5 | $\{very\_high\}$ | $\{yes, no\}$ | $\{yes\}$ | $\{yes\}$ |
| 6 | $\{high\}$ | $\{no\}$ | $\{yes, no\}$ | $\{yes, no\}$ |
| 7 | $\{normal\}$ | $\{no\}$ | $\{yes\}$ | $\{no\}$ |
| 8 | $\{normal, high\}$ | $\{no\}$ | $\{yes, no\}$ | $\{no\}$ |

For $\Phi$=$(OB, AT, \{VAL_A | A \in AT\}, g)$ and a set $ATR \subseteq AT$, we name the following $DIS$ a *derived* $DIS$ (for $ATR$ from a $NIS$ $\Phi$).

$$\psi = (OB, ATR, \{VAL_A | A \in ATR\}, h), \; (h(x, A) \in g(x, A)).$$

In Table 4, there are 1024 (=$2^{10}$) derived $DISs$. Fig. 3 is another example of a $NIS$.



DIS1

| | Color | Size |
|---|---|---|
| 1 | red | s |
| 2 | red | s |
| 3 | red | m |

DIS4

| | Color | Size |
|---|---|---|
| 1 | red | m |
| 2 | red | m |
| 3 | red | m |

DIS24

| | Color | Size |
|---|---|---|
| 1 | green | l |
| 2 | red | m |
| 3 | blue | m |

| | Color | Size |
|---|---|---|
| 1 | {red,green} | {s,m,l} |
| 2 | {red} | {s,m} |
| 3 | {red,blue} | {m} |

Fig. 3. An example of $\Phi_2$ and a set of derived DISs $DD(\Phi_2)$.

For a $NIS$ $\Phi$, let $DD(\Phi)$ denote a set $\{\psi \mid \psi$ is a derived $DIS$ from $\Phi\}$, and let $\psi^{actual}$ denote a derived $DIS$ with actual attribute values. For $NIS$ $\Phi_2$, a set $DD(\Phi_2)$ consists of 24 derived $DISs$. Then, we have the following modal concepts.

(Certainty) If a formula $\alpha$ holds in each $\psi \in DD(\Phi)$, $\alpha$ also holds in $\psi^{actual}$. In this case, we say $\alpha$ *certainly holds* in $\psi^{actual}$.
(Possibility) If a formula $\alpha$ holds in some $\psi \in DD(\Phi)$, there exists such a possibility that $\alpha$ holds in $\psi^{actual}$. In this case, we say $\alpha$ *possibly holds* in $\psi^{actual}$.

In order to handle two modalities, we defined two *blocks* $inf$ and $sup$ below.

**Definition 1** *[18,19] In* $\Phi = (OB, AT, \{VAL_A | A \in AT\}, g)$, *we define the following two sets of objects, i.e., inf and sup blocks, for each descriptor* $[A, val_A]$ *(*$A \in ATR \subseteq AT$, $val_A \in VAL_A$*).*

*(1)* $inf([A, val_A]) = \{x \in OB |\ g(x, A) = \{val_A\}\}$,
*(2)* $inf(\wedge_{A \in ATR}[A, val_A]) = \cap_{A \in ATR}\ inf([A, val_A])$,
*(3)* $sup([A, val_A]) = \{x \in OB |\ val_A \in g(x, A)\}$,
*(4)* $sup(\wedge_{A \in ATR}[A, val_A]) = \cap_{A \in ATR}\ sup([A, val_A])$.

Clearly, an equivalence class $[x]_{[A,val_A]}$ defined in $\psi \in DD(\Phi)$ satisfies the following:

$$inf([A, val_A]) \subseteq [x]_{[A,val_A]} \subseteq sup([A, val_A]).$$

Intuitively, $inf$ and $sup$ blocks define the minimum set and the maximum set with respect to a descriptor $[A, val_A]$, respectively. By using $inf$ and $sup$ blocks, we considered how to compute two modalities depending upon $DD(\Phi)$. The number of all derived $DISs$ increases in exponentially, therefore an *explicit method*, such that every definition is sequentially computed in each $\psi \in DD(\Phi)$, is not suitable. We have proposed some algorithms which do not depend upon $|DD(\Phi)|$, especially *NIS-Apriori* rule generation algorithm [19].

However, the calculation depending upon $inf$ and $sup$ is very complicated. Furthermore, we also noticed Proposition 3 and Remark 1, namely division charts will take the role of equivalence classes.

Therefore in the subsequent sections, we move the role of Definition 1 to division charts, because division charts give us visual and more comprehensive information.

### *3.2 A Division Chart in NISs*

In $NISs$, a block $sup([CON, val_{CON}])$ is the maximum set for $[x]_{[CON,val_{CON}]}$, so we consider a division of $sup([CON, val_{CON}])(\neq \emptyset)$ by a descriptor $[Dec, val]$. In this case, we have obtainable implications in Table 5, and we have the following division chart $DC([CON, val_{CON}], val)$ in Figure 4.

Table 5
A table for obtainable implications from $sup([CON, val_{CON}])$ with respect to $[Dec, val]$. Here, $val_{CON} \Rightarrow val$ means $[CON, val_{CON}] \Rightarrow [Dec, val]$.

| | $CON$ | $DEC$ | $Implications$ |
|---|---|---|---|
| 1 | $val_{CON} \in inf$ | $val \in inf$ | $val_{CON} \Rightarrow val$ |
| 2 | $val_{CON} \in inf$ | $val \in sup \setminus inf$ | $val_{CON} \Rightarrow val, val_{CON} \Rightarrow val'$<br>$(val \neq val')$ |
| 3 | $val_{CON} \in inf$ | $val \notin sup$ | $val_{CON} \Rightarrow val'$ |
| 4 | $val_{CON} \in sup \setminus inf$ | $val \in inf$ | $val_{CON} \Rightarrow val, val'_{CON} \Rightarrow val$<br>$(val_{CON} \neq val'_{CON})$ |
| 5 | $val_{CON} \in sup \setminus inf$ | $val \in sup \setminus inf$ | $val_{CON} \Rightarrow val, val_{CON} \Rightarrow val',$<br>$val'_{CON} \Rightarrow val, val'_{CON} \Rightarrow val'$ |
| 6 | $val_{CON} \in sup \setminus inf$ | $val \notin sup$ | $val_{CON} \Rightarrow val', val'_{CON} \Rightarrow val'$ |

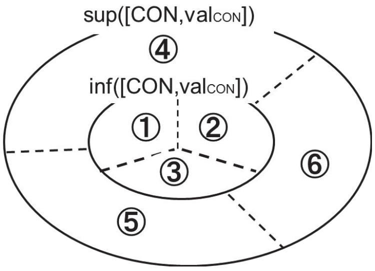


Fig. 4. A division chart $DC([CON, val_{CON}], val)$ of $sup([CON, val_{CON}])$ by $[Dec, val]$ in a $NIS$. Clearly, ⓙ ∩ ⓚ $= \emptyset$ $(j \neq k)$.

For each case in Table 5, let ⓚ denote the component defined by $k$th case ($k = 1, 2, \cdots, 6$). Here, the implication $val_{CON} \Rightarrow val$ is obtainable in ①, ②, ④, and ⑤. In ③ and ⑥, this implication is not obtainable.

**Remark 3** *In Table 5, we have an implication $\tau : val_{CON} \Rightarrow val$ from four components ①, ②, ④, and ⑤. Therefore in $NISs$, each $\tau^x$ ($x \in$ ⓙ$, j = 1, 2, 4, 5$) has the different character. An implication $\tau^x$ ($x \in$ ①) appears in each $\psi \in DD(\Phi)$, but $\tau^x$ ($x \notin$ ①) appears in a subset of $DD(\Phi)$. In $NISs$, there may be $\tau^x$ satisfying the condition of a rule and there may be $\tau^y$ ($y \neq x$) not satisfying the condition of a rule. We define that an implication $\tau$ is a rule, if there is an implication $\tau^x$ (for an object $x$) satisfying the condition of a rule. We see this $\tau^x$ is a piece of evidence of a rule $\tau$. Like this, Proposition 1 in $DISs$ may not hold in $NISs$.*

In Table 3, the obtainable implication is unique. However in Table 5, the obtainable implication may not be unique, the concept of consistency and *support*, *accuracy* values are variable according to the choice of an implication. Intuitively, this choice of an implication causes to reduce a set $DD(\Phi)$. We have the following proposition.

**Proposition 5** *[20] For every $NIS$, (1), (2) and (3) in the following are equivalent.*
*(1) An object $x \in sup([CON, val_{CON}]) \cap sup([Dec, val])$ ($\neq \emptyset$) is consistent in each $\psi \in DD(\Phi)$.*
*(2) $g(x, CON) = \{val_{CON}\}$, $g(x, \{Dec\}) = \{val\}$, and*
*$sup([CON, val_{CON}]) \subseteq inf([DEC, val])$ hold.*
*(3) Components ②, ③, ⑤ and ⑥ are all empty sets.*

Proposition 5 is an extension from Proposition 3 in $DISs$. We have previously employed (2) in Proposition 5 for proving theorems, however (3) in Proposition 5 seems more comprehensive.

In $NISs$, $support(\tau^x)$ and $accuracy(\tau^x)$ ($\tau$ defined by object $x$) are variable according to derived $DISs$. Therefore, we consider the following criterion values. The actual value exists between the minimum and the maximum values [19].

$$\begin{array}{l}
minsupp(\tau^x) = Min_{\psi \in DD(\Phi)}\{support(\tau^x)\ in\ \psi\},\\
minacc(\tau^x) = Min_{\psi \in DD(\Phi)}\{accuracy(\tau^x)\ in\ \psi\},\\
maxsupp(\tau^x) = Max_{\psi \in DD(\Phi)}\{support(\tau^x)\ in\ \psi\},\\
maxacc(\tau^x) = Max_{\psi \in DD(\Phi)}\{accuracy(\tau^x)\ in\ \psi\}.
\end{array}$$

In the above definition, each value depends upon $|DD(\Phi)|$, but we can easily calculate them by using a division chart corresponding to $\tau^x$.

**Example 2** *Let us consider* $\Phi_1$ *in Table 4.*
*For* $DC([headache, yes], yes)$ *corresponding to an implication* $\tau_1$ : $[headache, yes] \Rightarrow [flu, yes]$, $sup([headache, yes]) = \{1, 2, 3, 4, 5\}$, ① $= \{2, 4\}$, ② $= \{3\}$, ③ $= \emptyset$, ④ $= \{1, 5\}$, ⑤ $= \emptyset$ *and* ⑥ $= \emptyset$ *holds. In components* ②, ④, ⑤ *and* ⑥, *we may choose one of implications, and this choice causes the variation of support and accuracy.*
*(CASE 1:* $minsupp(\tau^2)$ *(*$2 \in$ ①*))*
*In order to reduce support, we should not choose* $val_{CON} \Rightarrow val$. *We choose* $val_{CON} \Rightarrow val'$ *from* $3 \in$ ②, $val'_{CON} \Rightarrow val$ *from* $4, 5 \in$ ④. *Then,* $\tau^x$ *occurs twice, and we have* $minsupp(\tau^1) = 2/8$.
*(CASE 2:* $minacc(\tau^2)$ *(*$2 \in$ ①*))*
*Since* $M/N \leq (M + 1)/(N + 1)$ *for natural numbers* $N$ *and* $M$ *(*$M \leq N$*), we also should not choose* $val_{CON} \Rightarrow val$. *Furthermore, we should choose* $val_{CON} \Rightarrow val'$ *as much as possible. If we have the same choice as CASE 1, accuracy is the minimum and we have* $minacc(\tau^2) = 2/3$.
*(CASE 3:* $maxsupp(\tau^2)$ *(*$2 \in$ ①*))*
*In order to increase support, we should choose* $val_{CON} \Rightarrow val$. *We choose* $val_{CON} \Rightarrow val$ *from* $3 \in$ ②, $val_{CON} \Rightarrow val$ *from* $4, 5 \in$ ④.

*Then, $\tau^x$ occurs 5 times, and we have $maxsupp(\tau^1) = 5/8$.*
*(CASE 4: $maxacc(\tau^2)$ $(2 \in \textcircled{1})$)*
*Since $M/N \leq (M+1)/(N+1)$ for natural numbers $N$ and $M$ $(M \leq N)$, we also should choose $val_{CON} \Rightarrow val$. If we have the same choice as CASE 3, accuracy is the maximum and we have $maxacc(\tau^2) = 5/5$.*

Like Example 2, we examine the criterion values of rules by using a division chart. In each proposition in the following, we consider a division chart $DC([CON, val_{CON}], val)$ corresponding to $\tau : [CON, val_{CON}] \Rightarrow [Dec, val]$.

**Proposition 6** *For $\tau^x$ $(x \in \textcircled{1})$, the following holds.*

$$minsupp(\tau^x) = |\textcircled{1}|/|OB|,$$

$$minacc(\tau^x) = |\textcircled{1}|/(|\textcircled{1}| + |\textcircled{2}| + |\textcircled{3}| + |\textcircled{5}| + |\textcircled{6}|),$$

$$maxsupp(\tau^x) = (|\textcircled{1}| + |\textcircled{2}| + |\textcircled{4}| + |\textcircled{5}|)/|OB|,$$

$$maxacc(\tau^x) = \frac{(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{4}|+|\textcircled{5}|)}{(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{3}|+|\textcircled{4}|+|\textcircled{5}|)}.$$

**(Proof)**
*We choose $val_{CON} \Rightarrow val'$ in components $\textcircled{2}$, $\textcircled{5}$, $\textcircled{6}$ and choose $val'_{CON} \Rightarrow val$ in $\textcircled{4}$. In this selection, $support(\tau^x)$ is the minimum. Since $M/N \leq (M+1)/(N+1)$ for natural numbers $N$ and $M$ $(M \leq N)$, the accuracy value is reduced by choosing $val_{CON} \Rightarrow val'$. Namely, $accuracy(\tau^x)$ is also the minimum in the above selection of implications.*
*On the other hand, We choose $val_{CON} \Rightarrow val$ in components $\textcircled{2}$, $\textcircled{4}$ $\textcircled{5}$ and select $val'_{CON} \Rightarrow val$ in $\textcircled{6}$. In this selection, $support(\tau^x)$ is the maximum, and $accuracy(\tau^x)$ is also the maximum.* □

**Proposition 7** *For $\tau^x$ $(x \in \textcircled{2} \cup \textcircled{5})$, the following holds.*

$$minsupp(\tau^x) = (|\textcircled{1}| + 1)/|OB|,$$

$$minacc(\tau^x) = (|\textcircled{1}| + 1)/(|\textcircled{1}| + |\textcircled{2}| + |\textcircled{3}| + |\textcircled{5}| + |\textcircled{6}|),$$

$$maxsupp(\tau^x)=(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{4}|+|\textcircled{5}|)/|OB|,$$

$$maxacc(\tau^x)=\frac{(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{4}|+|\textcircled{5}|)}{(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{3}|+|\textcircled{4}|+|\textcircled{5}|)}.$$

**(Proof)**
*Since $x \in \textcircled{2} \cup \textcircled{5}$, this object $x$ is counted in the numerator. As for the maximum values, we choose the same as Proposition 6, and we have the same formulas.* □

**Proposition 8** *For $\tau^x$ ($x \in \textcircled{4}$), the following holds.*

$$minsupp(\tau^x)=(|\textcircled{1}|+1)/|OB|,$$

$$minacc(\tau^x)=(|\textcircled{1}|+1)/(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{3}|+|\textcircled{5}|+|\textcircled{6}|+1),$$

$$maxsupp(\tau^x)=(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{4}|+|\textcircled{5}|)/|OB|,$$

$$maxacc(\tau^x)=\frac{(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{4}|+|\textcircled{5}|)}{(|\textcircled{1}|+|\textcircled{2}|+|\textcircled{3}|+|\textcircled{4}|+|\textcircled{5}|)}.$$

**(Proof)**
*In the previous propositions for the minimum, we chose $val'_{CON} \Rightarrow val$ in $\textcircled{4}$. However, $x \in \textcircled{4}$ holds, therefore this object $x$ is counted in the numerator and denominator. As for the maximum values, we choose the same as Proposition 6, and we have the same formulas.* □

Proposition 6, 7, 8 in $NISs$ are extensions from Proposition 2 in $DISs$. Proposition 6, 7, 8 show us how to calculate criterion values by using division charts, and this calculation does not depend upon $|DD(\Phi)|$.

### *3.4 A Merged Division Chart by Two Division Charts in NISs*

Now, we extend a merged division chart in $DISs$ (Fig. 2) to that in $NISs$. In $NISs$, let us consider two descriptors $[CON1, val_{CON1}]$, $[CON2, val_{CON2}]$ ($CON1 \cap CON2 = \emptyset$) and $[Dec, val]$, again. Furthermore, let us consider Fig. 5.

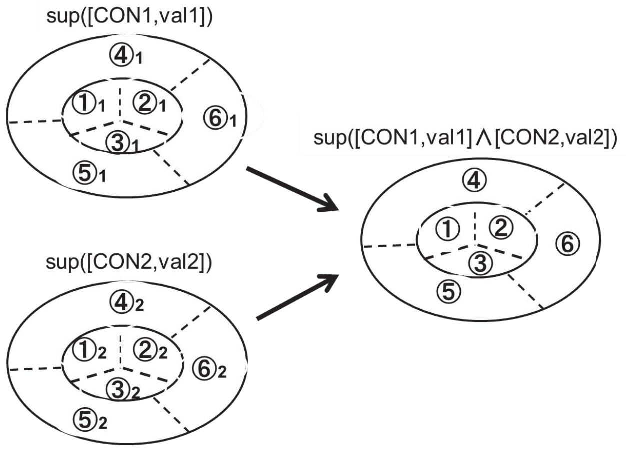


Fig. 5. A division chart of $sup([CON1, val_{CON1}] \wedge [CON2, val_{CON2}])$ by $[Dec, val]$ in a $NIS$.

By using each component, we newly generate ①, ②, $\cdots$, ⑥ for $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$. We have the following equation.

$$
\begin{aligned}
& sup([CON1, val_{CON1}] \wedge [CON2, val_{CON2}]) \\
& = sup([CON1, val_{CON1}]) \cap sup([CON2, val_{CON2}]) \\
& = (\textcircled{1}_1 \cup \textcircled{2}_1 \cup \cdots \cup \textcircled{6}_1) \cap (\textcircled{1}_2 \cup \textcircled{2}_2 \cup \cdots \cup \textcircled{6}_2) \\
& = \Sigma_{1 \leq s,t \leq 6}\ \textcircled{s}_1 \cap \textcircled{t}_2 = \Sigma_{1 \leq s,t \leq 6}\ C_{st}, \\
& \quad (\textcircled{s}_1 \cap \textcircled{t}_1 = \emptyset,\ \textcircled{s}_2 \cap \textcircled{t}_2 = \emptyset \text{ for } s \neq t).
\end{aligned}
$$

According to the above equation, we sequentially consider each $C_{st}$ in Table 6. Since $[Dec, val]$ is unique in two division charts, we can reduce the number of combinations.

Table 6
A combination of components $C_{st} = \textcircled{s}_1 \cap \textcircled{t}_2$ $(1 \leq s, t \leq 6)$.

| | $\textcircled{1}_2$ | $\textcircled{2}_2$ | $\textcircled{3}_2$ | $\textcircled{4}_2$ | $\textcircled{5}_2$ | $\textcircled{6}_2$ |
|---|---|---|---|---|---|---|
| $\textcircled{1}_1$ | $C_{11}$ | $C_{12}$ | $C_{13}$ | $C_{14}$ | $C_{15}$ | $C_{16}$ |
| $\textcircled{2}_1$ | $C_{21}$ | $C_{22}$ | $C_{23}$ | $C_{24}$ | $C_{25}$ | $C_{26}$ |
| $\textcircled{3}_1$ | $C_{31}$ | $C_{32}$ | $C_{33}$ | $C_{34}$ | $C_{35}$ | $C_{36}$ |
| $\textcircled{4}_1$ | $C_{41}$ | $C_{42}$ | $C_{43}$ | $C_{44}$ | $C_{45}$ | $C_{46}$ |
| $\textcircled{5}_1$ | $C_{51}$ | $C_{52}$ | $C_{53}$ | $C_{54}$ | $C_{55}$ | $C_{56}$ |
| $\textcircled{6}_1$ | $C_{61}$ | $C_{62}$ | $C_{63}$ | $C_{64}$ | $C_{65}$ | $C_{66}$ |

**Proposition 9** $C_{12}{=}C_{13}{=}C_{15}{=}C_{16}{=}\emptyset$.

**(Proof)** *For any* $y \in \textcircled{1}_1$,

$g(y, CON1){=}\{val_{CON1}\}$ *and* $g(y, DEC){=}\{val\}$

*hold by the definition of components. Similarly, in each* $y \in \textcircled{t}_2$ $(t{=}2, 3, 5, 6)$, $g(y, DEC) \neq \{val\}$. *Therefore, there is no* $y$ *satisfying* $y \in \textcircled{1}_1$ *and* $y \in \textcircled{t}_2$ $(t{=}2, 3, 5, 6)$. *Therefore,* $C_{12}{=}C_{13}{=}C_{15}{=}C_{16}{=}\emptyset$. □

According to Proposition 9, we similarly have the following.

**Proposition 10** *The following holds.*
*(1)* $C_{21}{=}C_{23}{=}C_{24}{=}C_{26}{=}\emptyset$,
*(2)* $C_{31}{=}C_{32}{=}C_{34}{=}C_{35}{=}\emptyset$,
*(3)* $C_{42}{=}C_{43}{=}C_{45}{=}C_{46}{=}\emptyset$,
*(4)* $C_{51}{=}C_{53}{=}C_{54}{=}C_{56}{=}\emptyset$,
*(5)* $C_{61}{=}C_{62}{=}C_{64}{=}C_{65}{=}\emptyset$.

**Proposition 11** $C_{11}$ $({=}\textcircled{1}_1 \cap \textcircled{1}_2)$ *belongs to component* $\textcircled{1}$ *in* $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$.

**(Proof)** *For any* $y \in \textcircled{1}_1 \cap \textcircled{2}_1$,
$g(y, CON1){=}\{val_{CON1}\}$, $g(y, DEC){=}\{val\}$,
$g(y, CON2){=}\{val_{CON2}\}$, $g(y, DEC){=}\{val\}$ *hold.*
*Therefore, this* $y$ *satisfies*
$g(y, CON1{\cup}CON2) = \{(val_{CON1}, val_{CON2})\}$, $g(y, DEC){=}\{val\}$.
*Namely,*
$y \in \textcircled{1}$ *in* $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$. □

**Proposition 12** $C_{14}$ $({=}\textcircled{1}_1 \cap \textcircled{4}_2)$ *belongs to component* $\textcircled{4}$ *in* $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$.

**(Proof)** *For any* $y \in \textcircled{1}_1 \cap \textcircled{4}_2$,
$g(y, CON1){=}\{val_{CON1}\}$, $g(y, DEC){=}\{val\}$,
$val_{CON2} \in g(y, CON2)$ $(|g(y, CON2)| \neq 1|)$ *hold.*
*Therefore, this* $y$ *satisfies*
$(val_{CON1}, val_{CON2}) \in g(y, CON1 \cup CON2)$, $g(y, DEC){=}\{val\}$.
*Namely,*

*$y \in \textcircled{4}$ in $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$.* □

**Proposition 13** *$C_{22}$ ($=\textcircled{2}_1 \cap \textcircled{2}_2$) belongs to component $\textcircled{2}$ in $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$.*

**(Proof)** *For any $y \in \textcircled{2}_1 \cap \textcircled{2}_2$,*
*$g(y, CON1) = \{val_{CON1}\}$, $g(y, CON2) = \{val_{CON2}\}$, $val \in g(y, DEC)$ hold. Therefore, this $y$ satisfies*
*$g(y, CON1 \cup CON2) = \{(val_{CON1}, val_{CON2})\}$, $val \in g(y, DEC)$.*
*Namely,*
*$y \in \textcircled{2}$ in $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$.* □

Similarly, we have the next proposition.

**Proposition 14** *The following holds for each component $\textcircled{3}$, $\textcircled{4}$, $\textcircled{5}$ and $\textcircled{6}$ in $DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$.*
*(1) $C_{33}$ belongs to component $\textcircled{3}$.*
*(2) $C_{41}$ and $C_{44}$ belong to component $\textcircled{4}$.*
*(3) $C_{25}$, $C_{52}$ and $C_{55}$ belong to component $\textcircled{5}$.*
*(4) $C_{36}$, $C_{63}$ and $C_{66}$ belong to component $\textcircled{6}$.*

As a result, we have the next theorem.

**Theorem 1** *We can calculate each component in*
*$DC([CON1, val_{CON1}] \wedge [CON2, val_{CON2}], val)$*
*from $DC([CON1, val_{CON1}], val)$ and $DC([CON2, val_{CON2}], val)$.*
*Namely,*

$$\begin{aligned}
&\textcircled{1} = \textcircled{1}_1 \cap \textcircled{1}_2,\ \textcircled{2} = \textcircled{2}_1 \cap \textcircled{2}_2,\ \textcircled{3} = \textcircled{3}_1 \cap \textcircled{3}_2,\\
&\textcircled{4} = (\textcircled{1}_1 \cap \textcircled{4}_2) \cup (\textcircled{4}_1 \cap \textcircled{1}_2) \cup (\textcircled{4}_1 \cap \textcircled{4}_2),\\
&\textcircled{5} = (\textcircled{2}_1 \cap \textcircled{5}_2) \cup (\textcircled{5}_1 \cap \textcircled{2}_2) \cup (\textcircled{5}_1 \cap \textcircled{5}_2),\\
&\textcircled{6} = (\textcircled{3}_1 \cap \textcircled{6}_2) \cup (\textcircled{6}_1 \cap \textcircled{3}_2) \cup (\textcircled{6}_1 \cap \textcircled{6}_2).
\end{aligned}$$

Theorem 1 is an extension from Proposition 4. Proposition 6, 7 and 8 are also applicable to this merged division chart. Since each merged division chart is corresponding to an implication $\tau$, we can calculate *support*($\tau$) and *accuracy*($\tau$) easily. We have the next Remark 4, which is extended from Remark 2.

**Remark 4** *In a NIS $\Phi$, we consider the following set $GR(\Phi)$:*

$$GR(\Phi) = \{DC([A, val_A], val) \mid A \in AT, val_A \in VAL_A,$$
$$val \in VAL_{Dec}\}.$$

*We may see each division chart a granule for rule generation in $\Phi$, and we can pick up any rule by using the merging process of these granules. We think this process is useful for improving previously proposed NIS-Apriori algorithm [16,19]. Formerly, we employed inf and sup blocks for handling rules, however we can generate rules by using $GR(\Phi)$ and merging algorithm with Theorem 1.*

### *3.5 Computational Complexity on Division Charts*

We briefly consider the computational complexity.

(1) As for generating $GR(\Phi)$, we at first prepare each array for a division chart $DC([A, val_A], val)$, then we sequentially examine the tuple of each object in $OB$. Namely, the order depends upon the size in the following:

$$|OB| \times (\Sigma_{A \in AT \setminus \{Dec\}} |VAL_A| \times |VAL_{Dec}|).$$

The procedure for generating $GR(\Phi)$ seems not time-consuming.

(2) As for merging two division charts, we obtain ①$_1 \cdots$ ⑥$_1$ and ①$_2 \cdots$ ⑥$_2$ from $GR(\Phi)$, and apply Theorem 1 to them. This procedure is not time-consuming, either. However, the number of the combination of two division charts may become large. The number is the following:

$$(\Sigma_{A,B \in AT,\ A \neq B} |VAL_A| \times |VAL_B|) \times |VAL_{Dec}|.$$

In rule generation, this process of handling each combination is the most time-consuming. We are adding constraint $support(\tau) \geq \alpha$ to each $DC([A, val_A], val)$ ($\tau : [A, val_A] \Rightarrow [Dec, val]$), and we are reducing the number of the combination. It is also the same situation as the *Apriori* algorithm [1].

## 4 Concluding Remarks

We proposed division charts over a $DIS$ and a $NIS$, and considered how to merge two division charts in a $DIS$ and a $NIS$. Especially, we clarified the property for merging two division charts. Due to this property, we can easily generate $DC((val_{CON1}, val_{CON2}), val)$ from $DC(val_{CON1}, val)$ and $DC(val_{CON2}, val)$. Previously, we have proved the calculation of $minsupp(\tau^x)$, $minacc(\tau^x)$, $maxsupp(\tau^x)$, $maxacc(\tau^x)$ by using $inf$ and $sup$ blocks in Definition 1. However, the proofs by $inf$ and $sup$ were complicated and not comprehensive. On the other hand, division charts give us visual and comprehensive information. Furthermore, $GR(\Phi)$ in Remark 4 and the merging algorithm with Theorem 1 can be applicable to improve our previously implemented *NIS-Apriori* [16,19]. Thus, we conclude that division charts afford new granular computing-based framework for rule generation. We are now implementing a software tool depending upon division charts.

## Acknowledgment

The authors would like to thank anonymous referees for their useful comments. This paper is an extended version of 'Division Charts and their Merging Algorithm in Rough Non-deterministic Information Analysis' presented at IEEE Granular Computing 2012.

This work is supported by the Grant-in-Aid for Scientific Research (C) (No.22500204), Japan Society for the Promotion of Science.

## References

[1] Agrawal R, Srikant R. Fast algorithms for mining association rules in large databases. In: Bocca JB, Jarke M, Zaniolo C, editors. Proc. VLDB'94. Morgan Kaufmann; 1994. pp 487–499.

[2] Grzymała-Busse JW. Data with missing attribute values: Generalization of indiscernibility relation and rule induction. Transactions on Rough Sets 2004;1:78–95.

[3] Komorowski J, Pawlak Z, Polkowski L and Skowron A. Rough Sets: a tutorial.

In: Pal S and Skowron A, editors. Rough Fuzzy Hybridization. Springer; 1999. pp 3-98.

[4] Kryszkiewicz M. Rules in incomplete information systems. Information Sciences. 1999;113:271–292.

[5] Lin TY. Uncertainty and Knowledge Theories: New Era in Granular Computing. In: Lin TY et al., editors. Proc. IEEE Granular Computing. 2012. pp 8–11.

[6] Lipski W. On semantic issues connected with incomplete information databases. ACM Transactions on Database Systems. 1979;4:262–296.

[7] Lipski W. On Databases with Incomplete Information. Journal of the ACM. 1981;28:41–70.

[8] Orłowska E, Pawlak Z. Representation of nondeterministic information. Theoretical Computer Science. 1984;29:27–39.

[9] Pawlak, Z. Information systems theoretical foundations. Information Systems. 1981;6:205–218.

[10] Pawlak Z. Systemy Informacyjne: Podstawy Teoretyczne (in Polish). Wydawnictwo Naukowo-Techniczne; 1983.

[11] Pawlak Z. Some issues on rough sets. Transactions on Rough Sets. 2004;1:1–58.

[12] Pawlak Z. Rough Sets. Kluwer Academic Publishers; 1991.

[13] Pedrycz W, Skowron A, Kreinovich V, editors. Handbook of Granular Computing. Wiley; 2008.

[14] Polkowski L, Skowron A, editors. Rough Sets in Knowledge Discovery 1: Methodology and Applications, Studies in Fuzziness and Soft Computing. Springer; 1998;18.

[15] Polkowski L, Skowron A, editors. Rough Sets in Knowledge Discovery 2: Applications, Case Studies and Software Systems, Studies in Fuzziness and Soft Computing. Springer; 1998;19.

[16] RNIA software logs:
http://www.mns.kyutech.ac.jp/∼sakai/RNIA
http://getrnia.appspot.com

[17] Sakai H. Effective procedures for handling possible equivalence relations in non-deterministic information systems. Fundamenta Informaticae. 2001:48;343–362.

[18] Sakai H, Okuma A. Basic algorithms and tools for rough non-deterministic information analysis. Transactions on Rough Sets. 2004;1:209–231.

[19] Sakai H, Ishibashi R, Koba K, Nakata M. Rules and apriori algorithm in non-deterministic information systems. Transactions on Rough Sets. 2008;9:328–350.

[20] Sakai H, Okuma H, Wu M, Nakata M. A descriptor-based division chart table in rough non-deterministic information analysis. Smart Innovation, Systems and Technologies 15. Springer; 2012. pp 25–34.

[21] Sakai H, Wu M, Yamaguchi N, Nakata M. Division Charts and their Merging Algorithm in Rough Non-deterministic Information Analysis. In: Lin TY et al., editors. Proc. IEEE Granular Computing. 2012. pp 491–496.

[22] UCI Machine Learning Repository:
`http://mlearn.ics.uci.edu/MLRepository.html`

## Appendix 1

The following is the logging data for obtaining a set of all division charts $DC(\{2,3\},\{4\})$ *(2:headache, 3:nausea, 4:flu)* in Table 4.

```
?-init.
File Name for Read Open:flu.pl.
[DIS:1 or NIS:2]:2.
EXEC_TIME=0.0(sec)
yes
?- dct([2,3],4).
dc(1,[yes,yes,yes],[2],[3],[],[1,4,5],[],[]).
dc(2,[yes,yes,no],[],[3],[2],[],[],[1,4,5]).
dc(3,[yes,no,yes],[],[],[],[4],[],[]).
dc(4,[yes,no,no],[],[],[],[],[],[4]).
dc(5,[no,yes,yes],[],[],[7],[1,5],[6],[8]).
dc(6,[no,yes,no],[7],[],[],[8],[6],[1,5]).
dc(7,[no,no,yes],[],[],[],[],[6],[8]).
dc(8,[no,no,no],[],[],[],[8],[6],[]).
EXEC_TIME=0.0(sec)
yes
```

The first $dc(1,[yes,yes,yes],[2],[3],[],[1,4,5],[],[])$ is corresponding to an implication $[headache, yes] \wedge [nausea, yes] \Rightarrow [flu, yes]$.
(CASE 1) $dc([headache, yes],[flu, yes]) : \textcircled{1}_1 = \{2,4\}, \textcircled{2}_1 = \{3\},$
$\textcircled{3}_1 = \{\}, \textcircled{4}_1 = \{1,5\}, \textcircled{5}_1 = \{\}, \textcircled{6}_1 = \{\}$.
(CASE 2) $dc([nausea, yes],[flu, yes]) : \textcircled{1}_2 = \{1,2,5\}, \textcircled{2}_2 = \{3\},$
$\textcircled{3}_2 = \{7\}, \textcircled{4}_2 = \{4\}, \textcircled{5}_2 = \{6\}, \textcircled{6}_2 = \{8\}$.
(CASE 3) Due to Theorem 1, we can obtain

$dc([headache, yes] \wedge [nausea, yes], [flu, yes])$ below:
$① = ①_1 \cap ①_2 = \{2\}$,     $② = ②_1 \cap ②_2 = \{3\}$,
$③ = ③_1 \cap ③_2 = \{\}$,
$④ = (①_1 \cap ④_2) \cup (④_1 \cap ①_2) \cup (④_1 \cap ④_2) = \{1, 4, 5\}$,
$⑤ = (②_1 \cap ⑤_2) \cup (⑤_1 \cap ②_2) \cup (⑤_1 \cap ⑤_2) = \{\}$,
$⑥ = (③_1 \cap ⑥_2) \cup (⑥_1 \cap ③_2) \cup (⑥_1 \cap ⑥_2) = \{\}$.

## Appendix 2

The following is a $DC([shape, 1], [severity, 0])$ in *Mammographic* data (960 objects, 6 attributes, 180 missing values and the number of derived $DISs$ is more than 10 power 90) in UCI machine learning repository [22]. This division chart (in a list expression) divides $sup([shape, 1])$ by $[severity, 0]$, and any object appears in the following obtained division chart is related to an implication $[shape, 1] \Rightarrow [severity, 0]$.

```
dc(1,[1,0],
[3,5,7,12,15,19,22,26,30,33,34,41,42,47,52,66,75,77,85,87,88,
92,94,96,104,105,108,115,121,123,127,138,142,143,149,152,167,
171,174,182,183,187,190,194,196,199,210,215,217,227,229,242,243,
249,250,252,261,273,281,298,301,303,305,308,309,318,322,324,325,
327,331,342,343,344,349,353,364,365,371,372,373,380,383,384,400,
416,421,422,423,430,442,446,447,454,460,461,463,465,468,470,472,
473,474,475,477,479,481,483,484,497,503,511,515,516,518,524,545,
572,573,577,580,582,585,586,604,607,608,615,617,631,634,636,644,
658,667,680,682,702,704,706,711,719,720,732,735,739,743,750,754,
770,775,777,779,780,781,784,794,809,816,817,822,825,828,840,841,
844,845,852,855,867,868,869,882,886,894,902,905,909,911,916,919,
927,929,934,938,943],··· ①
[],··· ②
[1,4,8,10,18,76,82,89,106,166,193,212,247,278,280,283,341,420,426,
444,445,492,521,564,598,603,616,670,673,678,689,701,785,791,823,
891,935,944],··· ③
[6,48,83,128,157,163,236,255,387,388,389,394,476,519,531,561,581,
661,778],··· ④
[],··· ⑤
```

[9,20,54,74,496,537,554,614,660,662,752,824]). $\cdots$ ⑥