การค้นพบความรู้ในข้อมูลวิยุตโดยการคำนวณแบบซอฟต์เชิงคณิตศาสตร์

Knowledge Discovery in Discrete Data by Using Mathematical Soft-Computing

โดย

ผศ.ดร.พรรณทิพย์ ภัทรอินทากร
สาขาวิชาคณิตศาสตร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

งานวิจัยนี้ได้รับทุนวิจัยจากทุนวิจัยคณะวิทยาศาสตร์
ประจำปีงบประมาณ พ.ศ. 2553

กิตติกรรมประกาศ

รายงานการวิจัยเรื่อง **"การค้นพบความรู้ในข้อมูลวิยุตโดยการคำนวณแบบซอฟต์เชิงคณิตศาสตร์" (Knowledge Discovery in Discrete Data by Using Mathematical Soft-Computing)** เป็นโครงการที่จัดทำขึ้นโดยได้รับการสนับสนุนจากทุนวิจัยคณะวิทยาศาสตร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อนำข้อมูลไปใช้ประกอบการจัดทำแนวทางการศึกษาวิจัย เพื่อสร้างขีดความสามารถในเรื่องยุทธศาสตร์การพัฒนาคุณภาพคน และสังคมไทยสู่สังคมแห่งภูมิปัญญาและการเรียนรู้ ประเภทการวิจัยประยุกต์ สาขาวิจัยคณิตศาสตร์ประยุกต์

ผู้วิจัยขอขอบคุณคณะกรรมการทุนวิจัยคณะวิทยาศาสตร์ซึ่งมี รศ.ดร.ดุษณี ธนะบริพัฒน์ เป็นประธาน และคณะทำงานอำนวยการโครงการวิจัยที่ได้ให้ข้อคิดเห็นและข้อเสนอแนะที่เป็นประโยชน์ยิ่งต่อการวิจัย ที่ให้ทุนสนับสนุนในการทำวิจัยและขอขอบคุณ อาจารย์ พรชัย ชัยสนิท ประธานสาขาคณิตศาสตร์ที่ช่วยติดตามดูแลงานวิจัยเป็นอย่างดี

ขอขอบคุณทุนวิจัยคณะวิทยาศาสตร์ที่อนุมัติงบประมาณอุดหนุนวิจัย เป็นค่าใช้จ่ายของโครงการวิจัยนี้และขอขอบคุณฝ่ายเลขานุการคณะทำงานที่ให้ความร่วมมือแก่นักวิจัยอย่างดียิ่ง

ผศ.ดร.พรรณทิพย์ ภัทรอินทากร
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2553

บทคัดย่อ

การวิจัยเกี่ยวกับข้อมูลวิยุตโดยใช้คณิตศาสตร์แขนงเก่า (hard computing) จะต้องมีการวิเคราะห์ความผิดพลาดด้วยเสมอว่า ในการสร้างตัวแบบนั้นจะให้ความผิดพลาดไม่เกินขอบเขตตามที่ได้พิสูจน์ไว้ แต่อย่างไรก็ดี สำหรับข้อมูลจริงที่มีขนาดใหญ่และมีค่าคลาดเคลื่อนในข้อมูลอยู่แล้วเป็นจำนวนมาก ไม่สามารถใช้คณิตศาสตร์แขนงเก่าวิเคราะห์ได้ถูกต้องเพียงพอ

ในปี 2523, Zdzislaw Pawlak ได้คิดค้นคณิตศาสตร์แขนงใหม่ขึ้นจากแนวคิดเรื่องเซต นั่นคือ กราฟสายงาน (flow graph) ซึ่งเป็นคณิตศาสตร์แขนงใหม่ชนิด คณนาแบบเปลี่ยนได้ (soft computing) โดยจะคำนวณค่าคลาดเคลื่อนโดยตรง จึงให้ความถูกต้องมากขึ้น อีกทั้งยังสามารถคำนวณด้วยวิธีการประมวลผลแบบขนานได้ จึงทำให้คำนวณได้เร็วขึ้นและใช้ความจุหน่วยเก็บน้อยลง ด้วยประสิทธิภาพดังกล่าว ทำให้วิธีนี้ได้ถูกนำไปใช้ในศาสตร์แขนงต่าง ๆ อย่างกว้างขวาง อาทิเช่น preference analysis, conflict analysis, rule analysis, propositional calculus, granular computing

จากข้อดีทั้งการใช้ความจุหน่วยเก็บน้อยลง ความเร็วในการคำนวณและความถูกต้องของวิธีนี้ในงานวิจัยเบื้องต้นดังกล่าว ทางผู้วิจัยจึงศึกษาหาความสัมพันธ์เชิงคณิตศาสตร์ระหว่างรัฟเซตและกราฟสายงานเชิงคณิตศาสตร์ที่สามารถนำไปวิเคราะห์ข้อมูลวิยุตได้ถูกต้องมากยิ่งขึ้นและค้นพบรูปแบบที่สำคัญในข้อมูลวิยุตได้

ABSTRACT

Mathematical rough set theory has attracted both practical and theoretical researchers. A significant extension of rough set theory is called *flow graphs*, invented by Zdzislaw Pawlak (1980). It is a knowledge representation in the form of information flow. Mathematical flow graphs are in soft computing area which provides less complexity, times and resources. Thus, flow graph is a promising approach to analyze data flow, decision trees, decision rules, probability learning, etc.

In this research, we propose a new extension of flow graphs in rough sets' context. We discuss some important properties of flow graphs with experimental on discrete data set.

รายงานวิจัยฉบับสมบูรณ์

โครงการ การค้นพบความรู้ในข้อมูลวิยุตโดยการคำนวณแบบซอฟต์เชิงคณิตศาสตร์

Knowledge Discovery in Discrete Data by Using Mathematical Soft-Computing

ทิศทางการวิจัย (Reseasrch Direction):

การศึกษาวิจัยเพื่อสร้างขีดความสามารถในเรื่อง การพัฒนาทรัพยากรมนุษย์ และการวิจัยในด้านการเรียนการสอนด้วยคอมพิวเตอร์ทั้งในส่วนของระบบซอฟต์แวร์และระบบฮาร์ดแวร์

แผนวิจัย (Research Plan):

เพื่อสร้างขีดความสามารถในเรื่องยุทธศาสตร์การพัฒนาคุณภาพคน และสังคมไทยสู่สังคมแห่งภูมิปัญญาและการเรียนรู้

ประเภทการวิจัย:

การวิจัยประยุกต์

สาขาการวิจัย:

คณิตศาสตร์ประยุกต์

ระยะเวลาโครงการ: 1 ปี

การดำเนินงาน ☒ ได้ดำเนินงานตามแผนที่วางไว้
☐ ได้ดำเนินงานล่าช้ากว่าแผนที่วางไว้

คำสำคัญ (Keyword) ของโครงการวิจัย: Flow graphs, Decision rule, Rough set theory, Data analysis

การเผยแพร่ผลงานวิจัย: ในช่วง 12 เดือนที่ผ่านมาได้มีการนำเสนอผลงานวิจัยแบบ Oral presentation ในงานประชุมวิชาการในระดับนานาชาติดังนี้

"An Extension of Rough Set Approximation to Flow Graph Based Data Analysis". The Seventh International Conference on Rough Sets and Current Trends in Computing (RSCTC 2010), pp. 418-427.

สารบัญเรื่อง

สารบัญรูป

รูป หน้า

I. INTRODUCTION

IN discrete data analysis, the overall modeling process typically consists of a number of errors, data imprecision and incompleteness that all require tuning and adjustments. In order to accommodate errorful, imprecise, and incomplete data, use of *rough set theory* is expedient. Rough set theory was introduced by Zdzislaw Pawlak in the early 1980's [1]. Rough set theory provides mathematicians with the ability to handle uncertainty with approximation. According to Pawlak [1], the power of rough set theory "...is that it does not need any preliminary or additional information about the data, such as probability distributions in statistics, basic probability assignment in the Dempster-Shafer theory, or grade of membership or the value of possibility in fuzzy set theory". A significant extension of rough set theory is called *flow graphs*. Rough set theory formed the basis of flow graphs in the early 2000s to support data visualization and enhance its efficiency.

Pawlak's flow graphs are different to *flow networks* introduced by Ford et al. that do not analyze optimal flow (c.f. [2]). Flow graphs represent the model of information flow from the given data set. The branches of a flow graph can be constructed as decision rules [3]. It is an efficient method for uncertainty management, partly because the branches of a flow graph are interpreted as decision rules. With every decision rules, there are three associated coefficients: *strength*, *certainty* and *coverage*. These coefficients satisfy Bayes' theorem. We can also discover dependency and correlation within a data set without reference to its probabilistic nature by using flow graphs [4]. Pawlak also proposed a new approach to data mining and knowledge discovery based on information flow distribution. In [5], Pawlak shown that a decision tree can be constructed as a flow graph. The author concluded that such a flow graph provides better structure of data compared to a decision tree. Pawlak revealed the relationship between flow graphs and probability. Pawlak revealed the relationship between flow graphs and probability. Rough sets also closely related with *modus ponen* (MP) and *modus tollens* (MT) inference rules [6], [7]. Preference analysis [8], rule analysis [9], propositional calculus [6], [7], data mining [5], [10] and granular computing [11] were analyzed by flow graphs efficiently. Essentially, Pawlak interpreted the union of all inputs x of y as the upper approximation of y and the union of all inputs x of y such that $cer(x,y) = 1$ as the lower approximation of y [8].

Pawlak's flow graph is a useful tool for knowledge discovery and has been successfully applied in many areas. Within the past decade, there has been much research about theoretical aspects of flow graphs which explored the complementary nature of their properties and other mathematical theories. At the same time, flow graphs applications became larger and more complex. Flow distribution in flow graphs can be exploited for approximation and reasoning. Based on flow graph contexts, we define fundamental definitions for rough sets: four categories of vagueness, accuracy of approximation, roughness of approximation and dependency degree. In addition, we state formulas to conveniently compute these measures for inverse flow graphs. To illustrate, a possible car dealer preference analysis is provided to support our propositions. New categories and measures assist and alleviate some limitations in flow graphs to discover new patterns and explanations.

II. Rough Set Theory

The theory of rough sets, proposed by Pawlak, provides a formal tool to deal with imprecise or incomplete information. The following rough sets theoretical backgroud is taken from [1]. This mathematical theory formed the basis of flow graphs later in the early 2000s [3], [5], [7], [8], [14].

In rough set based data analysis, the equivalence relation in an approximation space is commonly interpreted via the notion of information system. Rough sets are based on an *information system* (*data table*). An information system is a decision table, whose columns are labeled by attributes, rows are labeled by objects of interest and entries of the table are attribute values. More formally, it is a pair $S = (U, A)$, where U is a nonempty finite set of objects called the *universe* and A is a nonempty finite set of attributes such that a: $U \rightarrow V_a$ for every $a \in A$. The set V_a is called the *domain* of a.

Any subset B of A determines a binary relation $I(B)$ on U called an *indiscernibility relation*. It is defined as $(x, y) \in I(B)$ if and only if $a(x) = a(y)$ for every $a \in A$, where $a(x)$ denotes the attribute value of element x. If (x, y) belongs to $I(B)$, x and y are called *B-indiscernible*. Equivalence classes of the relation $I(B)$ are referred to as *B-elementary sets* (or equivalently *B-granules*, for more *granular computing* studies please refer to [8], [11]) denote by $B(X)$, i.e., $B(X)$ describes X in the terms of attribute values from B.

If we distinguish an information system to two disjoint classes of attributes, called *condition* and *decision attributes*, then the information system will be called a *decision system*, denoted by $S = (U, C, D)$, where $C \cap D = \emptyset$.

Suppose we are given an information system $S = (U, A)$, $X \subseteq U$, $B \subseteq A$ and $B(X)$ describes X in the terms of attribute values from B. We can approximate X using the information in B

by constructing the *B-lower* and *B-upper approximations* of X, denoted by $B_*(X)$ and $B^*(X)$ respectively. They are defined as

$B_*(X) = \bigcup_{x \in U} \{B(X) | \, B(X) \in X\}$,

$B^*(X) = \bigcup_{x \in U} \{B(X) | \, B(X) \cap X \neq \emptyset\}$,

respectively.

In addition, the set

$BN_B(X) = B^*(X) - B_*(X)$

will be referred to as *B-boundary region* of X.

As we can see from above definitions, they are expressed in terms of granules (small pieces) of knowledge. The lower approximation is the union of all granules which are entirely included in the set. The upper approximation is the union of all granules which have non-empty intersection with the set. The boundary region is the difference between upper approximation and lower approximation.

If the boundary region of X is an empty set

$BN_B(x) = \emptyset$,

then X is *crisp*.

In contrast, if

$BN_B(X) \neq \emptyset$,

then X is *rough*.

Pawlak also discussed two numerical characterizations of the imprecision of a subset X. More studies involve rough sets are in [1], [3], [5], [6], [7], [8], [10], [14].

Below, formally, we recall key feature definitions of approximations in rough set theory which is studied in depth later.

Definition 1: Let $S = (U, A)$ be an information system. For $X \subseteq U$, $B \subseteq A$. The B-lower approximations, B-upper approximations and B-boundary region of X are defined as $\underline{B}(X) = \bigcup_{x \in U} \{B(X) | B(X) \subseteq X\}$, $\overline{B}(X) = \bigcup_{x \in U} \{B(X) | \ B(X) \cap X \neq \emptyset\}$ and $BN_B(X) = \overline{B}(X) - \underline{B}(X)$, respectively.

In what follows we recall four basic classes of rough sets, i.e., four categories of vagueness.

Definition 2: Let $S = (U, A)$ be an information system. For $X \subseteq U$, $B \subseteq A$, the four categories of vagueness are defined as

- $\underline{B}(X) \neq \emptyset$ and $\overline{B}(X) \neq U$ iff X is roughly B-definable,
- $\underline{B}(X) = \emptyset$ and $\overline{B}(X) \neq U$ iff X is internally B-indefinable,
- $\underline{B}(X) \neq \emptyset$ and $\overline{B}(X) = U$ iff X is externally B-definable,
- $\underline{B}(X) = \emptyset$ and $\overline{B}(X) = U$ iff X is totally B-indefinable.

The intuitive meaning of this classification is the following. If X is roughly *B-definable*, we are able to employ B to decide for some elements of U whether they belong to X or $-X$[1].

If X is internally *B-indefinable*, this means that we can only decide whether some elements of U belong to $-X$ using B.

If X is externally *B-indefinable*, we can use B to justify for some elements of U whether they belong to X. Lastly,

if X is totally *B-indefinable*, we are unable to decide for any element of U whether it belongs to X or $-X$, using B.

Approximation of a rough set can be characterized numerically by some measurements as follows.

Definition 3: Let $S = (U, A)$ be an information system. For $X \subseteq U$, $B \subseteq A$, the accuracy of approximation, $\alpha_B(X)$, and roughness of approximation, $\gamma_B(X)$, are defined respectively as

[1] Where $-X = U - X$.

$\alpha_B(X) = \frac{card(\underline{B}(X))}{card(\overline{B}(X))}$

and

$\gamma_B(X) = 1 - \alpha_B(X) = 1 - \frac{card(\underline{B}(X))}{card(\overline{B}(X))}$,

where $card(X)$ denotes the cardinality of X.

Let us observe that, $0 \leq \alpha_B(X) \leq 1$. If $\alpha_B(X) = 1$, then X is *crisp* with respect to B and otherwise, if $\alpha_B(X) < 1$, then X is *rough* with respect to B.

Definition 4: Let $S = (U, A)$ be an information system and $F = \{X_1, X_2, \ldots, \ X_n\}$ be a partition of the universe U. For $B \subseteq A$, F depends on B to a degree

$k_B(F) = \frac{\sum_{i=1}^{n} card(\underline{B}(X_i))}{card(U)}$.

Definitions $2 - 4$ will be stated in the context of flow graphs elsewhere.

III. Flow Graphs

A flow graph was introduced by Pawlak. In this section we present basic definitions and some properties of flow graphs from [3], [5], [7], [8], [10], [14].

A *flow graph* is a *directed, acyclic, finite graph* $G = (\mathcal{N}, \mathcal{B}, \varphi)$, where $\mathcal{N}$ is a set of nodes, $\mathcal{B} \subseteq \mathcal{N} \times \mathcal{N}$ is a set of *directed branches*, φ: $\mathcal{B} \to R^+$ is a *flow function* and R^+ is the set of non-negative real numbers. If $(x, y) \in \mathcal{B}$ then x is an *input* of node y denoted by $I(y)$ and y is an *output* of node x denoted by $O(x)$.

The *input* and *output* of a flow graph G are defined by $I(G) = \{x \in \mathcal{N} \mid I(x) = \emptyset\}$ and $O(G) = \{x \in \mathcal{N} \mid O(x) = \emptyset\}$. These inputs and outputs of G are called *external nodes* of G whereas other nodes are called *internal nodes* of G.

If $(x, y) \in \mathcal{B}$ then we call (x, y) a *throughflow* from x to y. We will assume in what follows that $\varphi(x, y) \neq 0$ for every $(x, y) \in \mathcal{B}$.

With every node x of a flow graph G, we have its associated *inflow* and *outflow* respectively as: $\varphi_+(x) = \sum_{y \in I(x)} \varphi(y, x)$ and $\varphi_-(x) = \sum_{y \in O(x)} \varphi(x, y)$.

Similarly, an *inflow* and an *outflow* for the flow graph G are defined as: $\varphi_+(G) = \sum_{x \in I(G)} \varphi_-(x)$ and $\varphi_-(G) = \sum_{x \in O(G)} \varphi_+(x)$. We assume that for any internal node x, $\varphi_-(x) = \varphi_+(x) = \varphi(x)$, where $\varphi(x)$ is a *throughflow* of node x. Similarly then, $\varphi_-(G) = \varphi_+(G) = \varphi(G)$ is a *throughflow* of graph G.

As discussed by Pawlak [8], the above equations can be considered as *flow conservation equations* (or *pairwise consistent* [16]).

A. Normalized Flow Graphs

A flow graph is an alternative representation of knowledge, more specifically the information flow, in the given data. Several studies on traditional (rough set) flow graphs were in [3], [5], [7], [8], [10], [14] and for fuzzy set theory [13], [15]. In order to demonstrate interesting relationships between flow graphs and other disciplines, we consider normalized version of flow graphs (consider $\sigma \in [0,1]$ instead of R^+).

A *normalized flow graph* is a *directed, acyclic, finite graph* $G = (N, B, \sigma)$, where N is a set of *nodes*, $B \subseteq N \times N$ is a set of *directed branches* and φ: $B \rightarrow R^+$ is a flow function, $\varphi(G)$ is a *throughflow* of flow graph G and σ: $B \rightarrow [0,1]$ is a *normalized flow* of (x,y). The *strength* coefficient of (x,y) is given by

$\sigma(x,y) = \frac{\varphi(x,y)}{\varphi(G)}$,

where $0 \leq \sigma(x,y) \leq 1$.

The *strength_of_branch* $\times$ *100* can be understood as the percentage of a total flow through the branch.

With every node x of a flow graph G, *normalized inflow* and *outflow* are defined as:

$\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(y,x)$, and

$\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x,y)$.

Similarly, *normalized inflow* and *outflow* for the flow graph G are defined as:

$\sigma_+(G) = \frac{\varphi_+(G)}{\varphi(G)} = \sum_{x \in I(G)} \sigma_+(x)$, and

$\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(x)} \sigma_-(x)$.

For any internal node x,

$\sigma_+(x) = \sigma_-(x) = \sigma(x)$,

where $\sigma(x)$ is a *normalized throughflow* of x.

With every branch (x,y) of a flow graph G, the *certainty* and *coverage* coefficients of (x,y) are defined as:

$cer(x,y) = \frac{\sigma(x,y)}{\sigma(x)}$ and
$cov(x,y) = \frac{\sigma(x,y)}{\sigma(y)}$,
where $\sigma(x) \neq 0$ and $\sigma(y) \neq 0$.

Interestingly, properties:
$\sum_{y \in O(x)} cer(x,y) = 1$ and
$\sum_{y \in I(x)} cov(x,y) = 1$,
are the form of total probability theorem, and
$\sigma(x) = \sum_{y \in O(x)} cer(x,y)\sigma(x) = \sum_{y \in O(x)} \sigma(x,y)$ and
$\sigma(x) = \sum_{x \in I(x)} cov(x,y)\sigma(y) = \sum_{x \in I(x)} \sigma(x,y)$
are Bayes' rules [3], [5], [7], [8], [10], [14].

Moreover, $cer(x,y) = \frac{cov(x,y)\sigma(y)}{\sigma(x)}$, and $cov(x,y) = \frac{cer(x,y)\sigma(x)}{\sigma(y)}$ are holded.

Next, it is important to consider path and connection of flow graphs in some situations.

A (directed) *path* from x to y ($x \neq y$) in G, denoted by $[x \ldots y]$, is a sequence of nodes $x_1, \ldots, x_n$ such that $x_1 = x$ and $x_n = y$ and $(x_i, x_{i+1}) \in B$ for every i, $1 \leq i \leq n-1$.

The *certainty, coverage* and *strength of the path* $[x_1 \ldots x_n]$ are defined as:
$cer[x_1 \ldots x_n] = \prod_{i=1}^{n-1} cer(x_i, x_{i+1})$,
$cov\ [x_1 \ldots x_n] = \prod_{i=1}^{n-1} cov(x_i, x_{i+1})$ and
$\sigma[x \ldots y] = \sigma(x) cer[x \ldots y] = \sigma(y) cov[x \ldots y]$.

The set of all paths from x to y ($x \neq y$) in G, denoted by $\langle x, y \rangle$, is a *connection* of G determined by x and y.

Its associated *certainty, coverage* and *strength of the connection* $\langle x, y \rangle$ are defined as:
$cer\ \langle x,y \rangle = \sum_{[x \ldots y] \in \langle x,y \rangle} cer[x \ldots y]$,
$cov\ \langle x,y \rangle = \sum_{[x \ldots y] \in \langle x,y \rangle} cov[x \ldots y]$ and

$\sigma \langle x,y \rangle = \sum_{[x\ldots y]\in\langle x,y\rangle} \sigma[x\ldots y] = \sigma(x)cer\,\langle x,y\rangle = \sigma(y)cov\,\langle x,y\rangle$.

If $[x\ldots y]$ is a path such that x and y are input and output of the graph G, respectively, then $[x\ldots y]$ will be referred to as *complete*. The set of complete paths from x to y will be called a *complete connection* from x to y in G. For simplicity, from now we will consider complete paths and connections only.

If we substitute every complete connection $\langle x,y\rangle$ in G, where x and y are an input and an output of a graph G with a single branch (x,y) such that $\sigma(x,y) = \sigma\,\langle x,y\rangle$, $cer(x,y) = cer\,\langle x,y\rangle$ and $cov(x,y) = cov\,\langle x,y\rangle$ then we have a new flow graph with the property: $\sigma(G) = \sigma(G')$. This new flow graph is called a *combined flow graph*. It represents a relationship between its inputs and outputs more precisely.

IV. Rough Set Approximations and Flow Graphs

In this section, we provide a bridge between flow graphs and rough approximation. From standard definitions of approximations made by rough sets, we give these definitions in the context of flow graphs below.

Suppose we are given a normalized flow graph $G = (A, \mathcal{B}, \sigma)$, where $A = \{A_{l_1}, A_{l_2}\ , \ldots, A_{l_n}\}$ is a set of attributes[2], $\mathcal{B}$ is a set of directed branches and σ is a normalized flow function. A set of nodes in a flow graph G corresponding to A_{l_i} is referred to as a *layer* i. For $A = C \cup D$, we have that every layer corresponding to C will be called a *condition layer* whereas every layer corresponding to D will be called a *decision layer*. If an attribute A_{l_i} contains n_{l_i} values, we say that it contains n_{l_i} nodes.

We now consider how to approximate an attribute value $Y \in A_{l_{i+1}}$ from attribute values of A_{l_i} where $A_{l_i} = \left\{X_1, X_2, \ldots, X_{n_{l_i}}\right\}$, to indicate lower approximation, upper approximation and boundary region of Y.

In Definition 5, we recall Pawlak's definitions of lower approximation, upper approximation and boundary region for flow graphs.

Definition 5: [8] Let $G = (A, \mathcal{B}, \sigma)$ be a normalized flow graph, $A_{l_i} = \{X_1, X_2, \ \ldots, X_{n_{l_i}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j, Y), $j \in \{1, \ldots, n_{l_i}\}$, of the flow graph G, the union of all inputs X_j of Y is the upper approximation of Y (denoted $\overline{A_{l_i}}(Y)$), the union of all inputs X_j of Y, such that $cer(X_j, Y) = 1$, is the lower approximation of Y (denoted $\underline{A_{l_i}}(Y)$). Moreover, the union of all inputs X_j of Y, such that $cer(X_j, Y) < 1$, is the boundary region of Y (denoted $A_{l_i}N_{A_{l_i}}(Y)$).

In Definition 6, we state four categories of rough sets mentioned in Definition 2 in terms of

[2]In what follows, we regard $\mathcal{N}$ as A for simplicity.

flow graph.

Definition 6: Let $G = (A, \mathcal{B}, \sigma)$ be a flow graph, $A_{l_i} = \left\{X_1, X_2, \ldots, X_{n_{l_i}}\right\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j, Y), $j \in \{1, \ldots, n_{l_i}\}$, of G, we define four categories of vagueness as

- $\exists X_j\,[cer(X_j, Y) = 1]$ and $\exists X_j\,[X_j \notin I(Y)]$ iff Y is roughly A_{l_i}-definable,
- $\forall X_j\,[cer(X_j, Y) \neq 1]$ and $\exists X_j\,[X_j \notin I(Y)]$ iff Y is internally A_{l_i}-indefinable,
- $\exists X_j\,[cer(X_j, Y) = 1]$ and $\forall X_j\,[X_j \in I(Y)]$ iff Y is externally A_{l_i}-definable,
- $\forall X_j\,[cer(X_j, Y) \neq 1]$ and $\forall X_j\,[X_j \in I(Y)]$ iff Y is totally A_{l_i}-indefinable.

From the definition we obtain the following interpretation:

- if Y is roughly A_{l_i}*-definable*, this means that we are able to decide for some elements of U whether they belong to Y or $-Y$[3], using A_{l_i},
- if Y is internally A_{l_i}*-indefinable*, this means that we are able to decide whether some elements of U belong to $-Y$, but we are unable to decide for any element of U, whether it belongs to Y or not, using A_{l_i},
- if Y is externally A_{l_i}*-indefinable*, this means that we are able to decide for some elements of U whether they belong to Y, but we are unable to decide, for any element of U whether it belongs to $-Y$ or not, using A_{l_i},
- if Y is totally A_{l_i}*-indefinable*, we are unable to decide for any element of U whether it belongs to Y or $-Y$, using A_{l_i}.

Property 1: Let $G = (A, \mathcal{B}, \sigma)$ be a flow graph, $A_{l_i} = \left\{X_1, X_2, \ldots, X_{n_{l_i}}\right\}$, $2 \leq i \leq k$, be an attribute in layer i and W be a node in $A_{l_{i-1}}$. For any branch (X_j, W), $j \in \{1, \ldots, n_{l_i}\}$ in the inverse flow graph of G, the union of all output X_j of W in flow graph G is the

[3]Where $-Y = U - Y$.

upper approximation of W, the union of all outputs X_j of W in a flow graph G, such that $cov(W, X_j) = 1$, is the lower approximation of W. Moreover, the union of all outputs X_i of W, such that $cov(W, X_j) < 1$, is the boundary region of Y.

Proof: It can be proved in a straightforward way according to definition and property of inverse flow graph and Definition 5. ∎

The following example illustrates the four basic categories.

Example Suppose we are given the flow graph for the preference analysis problem demonstrated in Fig. 1. This flow graph describes four disjoint models of cars $X = \{X_1, X_2, X_3, X_4\}$. They are sold to four disjoint groups of customers $Z = \{Z_1, Z_2, Z_3, Z_4\}$ through three dealers $Y = \{Y_1, Y_2, Y_3\}$.

By Definition 5, when we consider customer Z_1: the lower approximation of Z_1 is an empty set, the upper approximation of Z_1 is $Y_1 \cup Y_2$ and the boundary region Z_1 is $Y_1 \cup Y_2$. Hence, by Definition 6, we conclude that Z_1 is internally Y-indefinable.

In Fig. 1 (only limited information is available), by using the set of dealers (Y) to approximate the customer group Z_1 together with the flow distribution visualized in layers two and three, our results can be summarized as the following.

- Since no branch connects Y_3 and Z_1, there is no customer Z_1 buys a car from dealer Y_3. As a result if dealer Y_3 plans to run new promotional campaigns, they do not need to pay attention to customer group Z_1 in these campaigns.
- If a customer buys a car through dealer Y_1 or Y_2, then we cannot conclude whether this is a customer in group Z_1 or not. Thus, if dealers Y_1 and Y_2 plan to run promotional campaigns, then they should, at least, target at customer group Z_1 in their campaigns.

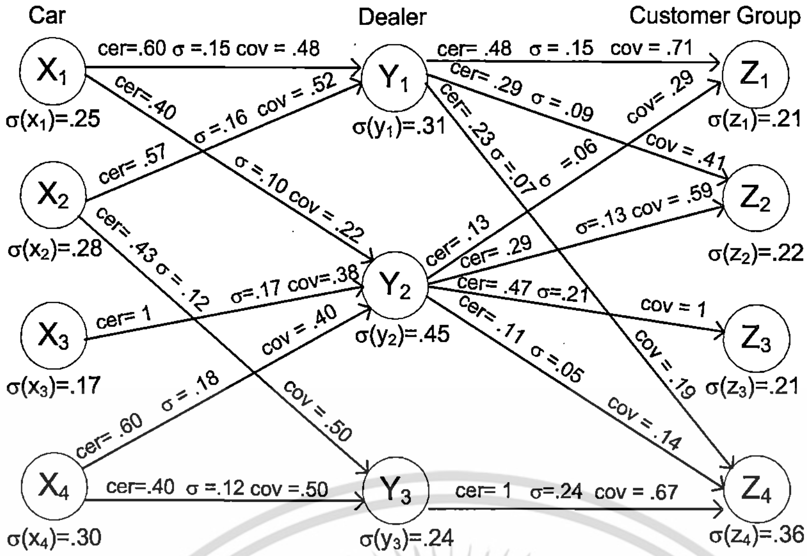

Fig. 1. A normalized flow graph.

Similarly, we can approximate all attribute values (node) in the inverse flow graph of G by using Property 1.

However, the flow graph perspective on rough sets' categories in Definition 6 do not provide approximations quantitively. Hence, in Definitions 7 and 8, we define two measures for flow graphs, the accuracy of approximation and the roughness of approximation.

Definition 7: Let $G=(A,\mathcal{B},\sigma)$ be a flow graph, $A_{l_i}=\left\{X_1, X_2, \ldots, X_{n_{l_i}}\right\}$, $1 \le i \le k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j, Y), $j \in \{1, \ldots, n_{l_i}\}$, of G, the accuracy of approximation, $\alpha_{A_{l_i}}(Y)$, is defined as:
$\alpha_{A_{l_i}}(Y) = \frac{card(\underline{A_{l_i}}(Y))}{card(\overline{A_{l_i}}(Y))}$.

We can use the accuracy of approximation to specify the quality of an approximation. Obviously, $0 \le \alpha_B(X) \le 1$. If $\alpha_{A_{l_i}}(Y) = 1$, then Y is crisp with respect to A_{l_i}, and otherwise, if $\alpha_{A_{l_i}}(Y) < 1$, then Y is rough with respect to A_{l_i}.

Definition 8: Let $G=(A,\mathcal{B},\sigma)$ be a flow graph, $A_{l_i}=\left\{X_1,X_2,\ldots,X_{n_{l_i}}\right\}$, $1\leq i\leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_i,Y), $i\in\{1,\ldots,n_{l_i}\}$, of G, the roughness of approximation, $\gamma_{A_{l_i}}(Y)$, is defined as:
$\gamma_{A_{l_i}}(Y)=1-\alpha_{A_{l_i}}(Y)=\frac{card(\overline{A_{l_i}}(Y))-card(\underline{A_{l_i}}(Y))}{card(\overline{A_{l_i}}(Y))}$.

We have that $0\leq\gamma_{A_{l_i}}(Y)\leq 1$. If $\gamma_{A_{l_i}}(Y)=0$, then Y is crisp with respect to A_{l_i}, and otherwise, if $\gamma_{A_{l_i}}(Y)<1$, then Y is rough with respect to A_{l_i}.

Property 2: Let $G=(A,\mathcal{B},\sigma)$ be a flow graph, $A_{l_i}=\left\{X_1,X_2,\ldots,X_{n_{l_i}}\right\}$, $1\leq i\leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j,Y), $j\in\{1,\ldots,n_{l_i}\}$, of G, we have
(1) $\alpha_{A_{l_i}}(Y)=\frac{\sum_{cer(X_j,Y)=1}\sigma(X_j)}{\sum_{X_j\in I(Y)}\sigma(X_j)}$ and
(2) $\gamma_{A_{l_i}}(Y)=\frac{\sum_{cer(X_j,Y)<1}\sigma(X_j)}{\sum_{X_j\in I(Y)}\sigma(X_i)}$.

Proof:
(1) From Definition 5, we have $card(\underline{A_{l_i}}(Y))=\sum_{cer(X_j,Y)=1}card(X_j)$ and $card(\overline{A_{l_i}}(Y))$ $=\sum_{X_j\in I(Y)}card(X_i)$. Since $card(X_j)=\varphi(X_j)=\sigma(X_j)\varphi(G)=\sigma(X_j)\varphi(U)$ and by Definition 7, then $\alpha_B(Y)=\frac{\sum_{cer(X_j,Y)=1}\sigma(X_j)}{\sum_{X_j\in I(Y)}\sigma(X_j)}$.
(2) It can be proved similarly to (1). ■

Let us briefly comment on Property 2(1) that the greater the boundary of Y, the lower is the accuracy. If $\alpha_{A_{l_i}}(Y)=1$, the boundary region of Y is empty.

Property 3: Let $G=(A,\mathcal{B},\sigma)$ be a flow graph, $A_{l_i}=\left\{X_1,X_2,\ldots,X_{n_{l_i}}\right\}$, $2\leq i\leq k$, be an attribute in layer i and W be a node in $A_{l_{i-1}}$. For any branch (X_j,W), $j\in\left\{1,\ldots,n_{l_j}\right\}$ in the inverse flow graph of G, we have
(1) $\alpha_{A_{l_j}}(W)=\frac{\sum_{cov(W,X_j)=1}\sigma(X_j)}{\sum_{X_j\in O(W)}\sigma(X_j)}$ and
(2) $\gamma_{A_{l_j}}(W)=\frac{\sum_{cov(W,X_j)<1}\sigma(X_j)}{\sum_{X_j\in O(W)}\sigma(X_j)}$.

Proof:

(1) From Property 1, we have $card(\underline{A_{l_j}}(W)) = \sum_{cov(X_j,W)=1} card(X_j)$ and $card(\overline{A_{l_j}}(Y))$ $=\sum_{X_j \in O(W)} card(X_j)$. Since $card(X_j) = \varphi(X_j) = \sigma(X_j)\varphi(G) = \sigma(X_j)\varphi(U)$ and by Definition 7, then $\alpha_{A_{l_j}}(W) = \frac{\sum_{cer(X_j,W)=1} \sigma(X_j)}{\sum_{X_j \in O(W)} \sigma(X_j)}$.

(2) It can be proved similarly to (1). ■

Example (Cont.) Consider the branches between dealer and customer group in Fig. 1. We can read from our flow graph that 24% of all customers buy cars through dealer Y_3 ($\sigma(Y_3) = 0.24$) and all of them are in customer group Z_3 ($cer(Y_3, Z_4) = 1$). There is only one branch (Y_3, Z_4) with $cer(Y_3, Z_4) = 1$. Thus, by Property 2(1), we have

$\alpha_Y(Z_1) = \alpha_Y(Z_2) = \alpha_Y(Z_3) = 0$ and

$\alpha_Y(Z_4) = \frac{\sum_{cer(Y_i,Z_4)=1} \sigma(Y_i)}{\sum_{Y_i \in I(Z_4)} \sigma(Y_i)} = \frac{\sigma(Y_3)}{\sigma(Y_1)+\sigma(Y_2)+\sigma(Y_3)} = 0.24.$

By employing the approach presented in our previous study, we can extract some interesting association rules. If the model of car X_2 (or X_4) is bought through dealer Y_3 then the customer group is Z_4 with support 0.12 and confidence 1.

These results imply that we should not make decisions involving customer groups Z_1, Z_2 and Z_3 solely by using dealers due to high imprecision. Nevertheless, we can partly check that it will be customer group Z_4 with low accuracy by using dealers.

Similarly, if we consider the roughness of approximation between dealer and customer group, then by Property 2(2), we have $\gamma_Y(Z_1) = \gamma_Y(Z_2) = \gamma_Y(Z_3) = 1$ and $\gamma_Y(Z_4) = 0.76$. We can draw a conclusion in a similar manner as we did for the roughness measure.

Please note that we can calculate the accuracy and the roughness of approximation between attributes in the inverse flow graph by using Property 3.

Another important topic in data analysis is dependency between attributes. We introduce dependency degree between any two attributes in Definition 9.

Definition 9: Let $G = (A, \mathcal{B}, \sigma)$ be a flow graph, $A_{l_i} = \left\{X_1, X_2, \ldots, X_{n_{l_i}}\right\}$ and $A_{l_{i+1}} = \left\{Y_1, Y_2, \ldots, Y_{n_{l_{i+1}}}\right\}$, $1 \leq i \leq k$, be any two adjacent layers. $A_{l_{i+1}}$ depends on A_{l_i} to a degree $k_{A_{l_i}}(A_{l_{i+1}}) = \frac{\sum_{l=1}^{n_{l_{i+1}}} card(\underline{A_{l_i}}(Y_l))}{card(U)}$.

If $k_{A_{l_i}}(A_{l_{i+1}}) = 1$, we say that $A_{l_{i+1}}$ depends totally on A_{l_i}, and if $k_{A_{l_i}}(A_{l_{i+1}}) < 1$, we say that $A_{l_{i+1}}$ depends partially in a degree $k_{A_{l_i}}(A_{l_{i+1}})$ on A_{l_i}. It is worth pointing out that our dependency measure is different to the one given by Pawlak [10]. The former gives dependency degree between two adjacent attributes (layers) while the latter gives dependency degree between two nodes connected by directed branch.

Property 4: Let $G = (A, \mathcal{B}, \sigma)$ be a flow graph, $A_{l_i} = \left\{X_1, X_2, \ldots, X_{n_{l_i}}\right\}$ and $A_{l_{i+1}} = \{X_1, X_2, \ldots, X_{n_{l_{i+1}}}\}$, $1 \leq i \leq k$, be any two adjacent layers. $A_{l_{i+1}}$ depends on A_{l_i} to a degree $k_{A_{l_i}}(A_{l_{i+1}}) = \sum_{cer(X_i,X_j)=1} \sigma(X_i)$.

Proof: From Definition 5, $\sum_{j=1}^{n_{l_{i+1}}} card(\underline{A_{l_i}}(X_j)) = \sum_{j=1}^{n_{l_{i+1}}} \sum_{cer(X_i,Y_j)=1} card(X_i)$.
Since $X_n \cap X_m = \emptyset$, $1 \leq n \neq m \leq n_{l_i}$, then $\underline{A_{l_i}}(X_n) \cap \underline{A_{l_i}}(X_m) = \emptyset$.
Thus $\sum_{j=1}^{n} card(\underline{A_{l_i}}(X_j)) = \sum_{cer(X_i,Y_j)=1} card(X_i)$.
Since $\varphi(X_i) = \sigma(X_i)\varphi(G) = \sigma(X_i)\varphi(U)$ and by Definition 9, we can write $\gamma_B(D) = \sum_{cer(X_i,X_j)=1} \sigma(X_i)$. ■

Property 5: Let $G = (A, \mathcal{B}, \sigma)$ be a flow graph, $A_{l_j} = \left\{X_1, X_2, \ldots, X_{n_{l_j}}\right\}$ and $A_{l_{j-1}} = \left\{X_1, X_2, \ldots, X_{n_{l_{j-1}}}\right\}$, $1 \leq j \leq k+1$, be any two adjacent layers in the inverse flow graph of G. $A_{l_{j-1}}$ depends on A_{l_j} to a degree $k_{A_{l_j}}(A_{l_{j-1}}) = \sum_{cov(X_i,X_j)=1} \sigma(X_i)$.

Proof: It can be proved similarly as Property 4 ■

Example (Cont.) Consider model of car and dealer in the flow graph G in Fig. 1. By Property 4, dealer depends on model of car to a degree

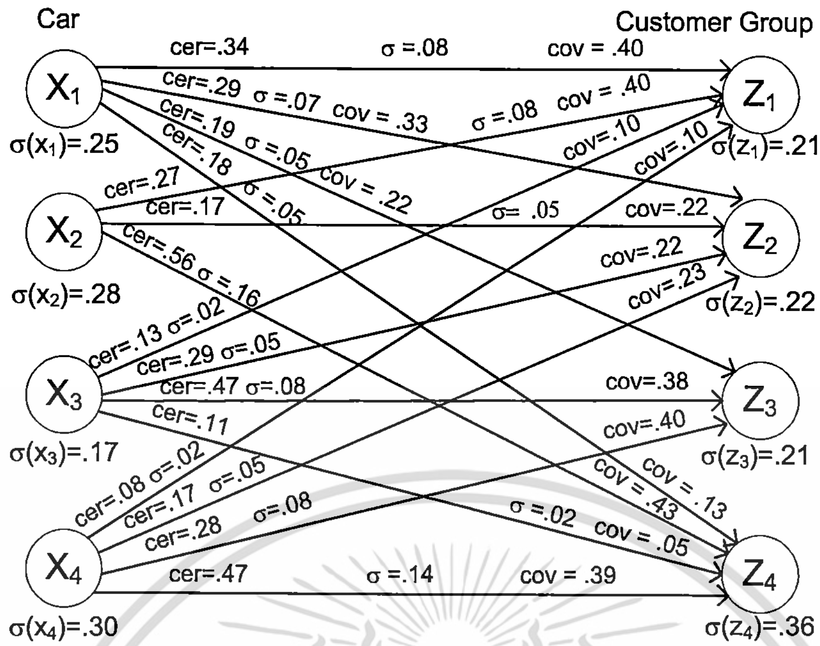

Fig. 2. A combined flow graph.

$\gamma_X(Y) = \sum_{cer(X_i,Y_j)=1} \sigma(X_i) = \sigma(X_3) = 0.17.$

On the other hand, if we consider customer and dealer in the inverse flow graph of G, then by Property 5, we obtain that dealer depends on customer group to a degree

$\gamma_Z(Y) = \sum_{cov(Y_i,Z_j)=1} \sigma(Z_i) = \sigma(Z_3) = 0.21.$

These results give a conclusion that dealers depend on customer groups more than models of cars.

In what follows, we aim to approximate a specific attribute value by some attribute values such that they are not in adjacent layers. We can use the concept of a *connection* to do this. More specifically, if we aim to approximate an attribute value in an output layer by attribute values in an input layer, then we will use the concept of *complete connection.*

Example (Cont.) For model of car and customer group in Fig. 1, we give a combined flow graph in Fig. 2. By Definition 5, for Z_4, the lower approximation of Z_4 is an empty set, the upper approximation and the boundary region of Z_4 are $X_1 \cup X_2 \cup X_3 \cup X_4$. Hence, by Definition 6, Z_4 is totally X-indefinable.

By Property 2, we have the accuracy and the roughness approximation of customer Z_4 by model of car as: $\alpha_X(Z_4) = 0$ and $\gamma_X(Z_4) = 1$.

Additionally, we can use Property 4 to compute the dependency between model of car and customer group, and the result is 0. From these results due to the imprecision and dependency, we should not make decisions involving customer group Z_4 by using only model of car. As before, we can approximate and measure them for the inverse flow graph in the same way. Comparing the obtained accuracy and roughness measures, we can draw a conclusion that from this population *dealer* is a better indicator for analyzing customer group Z_4 than *model of car*.

V. Conclusion and Suggestion

Pawlak's flow graph is a useful tool for knowledge discovery and has been successfully applied in many areas. Within the past decade, there has been much research about theoretical aspects of flow graphs which explored the complementary nature of their properties and other mathematical theories. At the same time, flow graphs applications became larger and more complex.

Thus, in this research, we introduce definitions and properties of rough set approximations, accuracy and roughness of approximation which are defined in terms of a flow graph. They can be useful when the initial data is in the form of flow graph and contains some limitations. We illustrate a car dealer preference analysis to support our propositions. They are successively applied to car data set efficiency. Our future work is to explore relationships between flow graphs and three-way decision rules. Additionally, we aim to invent new method to allow extensively computation for large data set.

REFERENCES

[1] Z.Pawlak, *Rough Sets. Theoretical Aspects of Reasoning about Data.* (Dordrecht: Kluwer Academic Publishers, 1991)

[2] L.R. Ford, D.R. Fulkerson, *Flows in Networks.* Princeton University Press, Princeton. New Jersey, 1962

[3] Z.Pawlak, Decision Algorithms and Flow Graphs: A Rough Set Approach. *J. of Telecom. and Informa Tech.* Vol. 3, 2003, 98–101.

[4] Z. Pawlak, "Rough Sets, Decision Algorithms and Bayes' Theorem." *Eur. J.of Oper. Res.* Vol. 136, 2002, pp 181–189.

[5] Z.Pawlak, Decision Trees and Flow Graphs, in Greco, S. (Eds.) *Rough Sets and Current Trends in Computing, LNAI 4259* (Berlin: Springer-Verlag, 2006) 1–11.

[6] S. Greco, Z.Pawlak, R. Slowinski, Generalized Decision Algorithms, Rough Inference Rules, and Flow Graphs, in Alpigini, J.J. et al. (Eds) *Rough Sets and Current Trends in Computing, LNAI, 2475* (Berlin: Springer-Verlag, 2002) 93–104.

[7] Z.Pawlak, Probability, Truth and Flow Graphs, in *Proc of Rough Sets in Knowledge Discovery and Soft Computing*, Warsaw, Poland, 2003, 1–9.

[8] Z. Pawlak, Rough Set and Flow Graphs, In Slezak, D. et al.(Eds.) *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, LNAI 3641* (Berlin: Springer-Verlag, 2005) 1–11.

[9] P. Pattaraintakorn, N. Cercone, K. Naruedomkul, Rule Learning: Ordinal Prediction based on Rough Set and Soft-computing. Appl. Math. Lett. 19. (12), 2006, 1300–1307.

[10] Z.Pawlak, Flow Graphs and Data Mining, in Peters, J.F., Skowron, A. (Eds.) *Transcations on Rough Sets III, LNCS 3400* (Berlin: Springer-Verlag, 2005) 1–36.

[11] J. Sun, H. Liu, C. Qi, H. Zhang, An Interpretation of Flow Graphs by Granular Computing in Greco, S. (Eds.) *Rough Sets and Current Trends in Computing, LNCS 4259* (Berlin: Springer-Verlag 2006) 448–457.

[12] L.A. Zadeh, Fuzzy set. Internation and Control 8, 1965, 338–353

[13] M. Alicja, M. Leszek, Flow Graph and Decision Tables with Fuzzy Attribdtes, *International Conference Artificial Intelligence and Soft Computing , LNAI 4029* (Berlin: Springer-Verlag, 2006) 268–277.

[14] Z.Pawlak, Flow Graphs and Decision Algorithms, in Wang, G. et al. (Eds.) *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, LNAI, 2639* (Berlin: Springer-Verlag 2003) 1–10.

[15] M. Alicja, M. Leszek, Fuzzy Implication Operators in Variable Precision Fuzzy Rough Sets Model, in L. Rutkowski et al. (eds.) *International Conference Artificial Intelligence and Soft Computing , LNAI 3037* (Berlin: Springer-Verlag, 2006) 498–503.

[16] C.J, Butz, W. Yan, B. Yang, The Computational Complexity of Inference using Rough Set Flow Graphs, in Slezak, D. et al. (Eds.) *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, LNAI, 364* (Berlin: Springer-Verlag, 2005), 335–344.