

**MATHEMATICAL ASPECTS OF FLOW GRAPH APPROACHES TO
DATA ANALYSIS**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY PROGRAM IN APPLIED MATHEMATICS
FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2010

KMITL-2010-SC-D-001-047

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



COPYRIGHT 2010

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

หัวข้อวิทยานิพนธ์	กราฟสายงานทางคณิตศาสตร์ขั้นสูงเพื่อการวิเคราะห์ข้อมูล
นักศึกษา	นางสาว ควงรัตน์ จิตต์เจริญ
รหัสนักศึกษา	50067111
ปริญญา	ปรัชญาดุษฎีบัณฑิต
สาขาวิชา	คณิตศาสตร์ประยุกต์
พ.ศ.	2553
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ. ดร. พรรณทิพย์ ภัทรอินทการ

บทคัดย่อ

ทฤษฎีกราฟเซต เป็นทฤษฎีทางคณิตศาสตร์ที่มีความน่าสนใจทั้งในเชิงทฤษฎีและปฏิบัติ เนื่องจากเป็นทฤษฎีที่สามารถนำไปประยุกต์ใช้ในการวิเคราะห์ข้อมูลได้ดีและมีประสิทธิภาพ กราฟสายงานทางคณิตศาสตร์เป็นส่วนขยายส่วนหนึ่งของทฤษฎีกราฟเซต ในวิทยานิพนธ์ฉบับนี้ได้นำเสนอการวิเคราะห์ของกราฟสายงานทั้งในเชิงทฤษฎีและปฏิบัติ

จุดประสงค์หลักสำหรับการวิเคราะห์กราฟสายงานในวิทยานิพนธ์ฉบับนี้แบ่งเป็น 3 ส่วนหลัก ได้แก่ การศึกษาเกี่ยวกับเทคนิคการวิเคราะห์ โครงสร้าง และการวัด สำหรับเทคนิคการวิเคราะห์ ได้นำเสนอวิธีการคำนวณค่าสัมประสิทธิ์ certainty, covering และ strength ของกฎการตัดสินใจสำหรับข้อมูลในรูปเซตวิชันและสร้างกราฟสายงานที่สมนัยกับข้อมูลขึ้น ในส่วนของโครงสร้าง นำเสนอระเบียบวิธีใหม่ในการนำเสนอค่าสัมประสิทธิ์ของกราฟสายงานในรูปแบบเมทริกซ์และพิสูจน์สูตรในการคำนวณค่าสัมประสิทธิ์ของกราฟสายงาน และกราฟสายงานแบบรวมโดยใช้สมบัติของเมทริกซ์ นอกจากนี้ยังได้นำเสนอวิธีการรวมกราฟสายงานที่มาจากแหล่งข้อมูลที่แตกต่างกันพร้อมทั้งสูตรในการคำนวณค่าสัมประสิทธิ์ ท้ายที่สุดสำหรับการวัด ได้วิเคราะห์ความสัมพันธ์ระหว่างกราฟสายงานและกฎความสัมพันธ์ และในกรณีที่ต้องการใช้กราฟสายงานในรูปกราฟเซต วิทยานิพนธ์ฉบับนี้ได้นำเสนอความสัมพันธ์ต่าง ๆ ระหว่างทฤษฎีกราฟเซตกับกราฟสายงานและกราฟสายงานผกผัน

เพื่อสนับสนุนว่าทฤษฎีที่ได้นำเสนอขึ้นใหม่สามารถประยุกต์ใช้ได้ ผู้จัดทำได้เก็บรวบรวมข้อมูลจากมูลนิธิส่งเสริมโอลิมปิกวิชาการและพัฒนามาตรฐานวิทยาศาสตร์ศึกษา (สอวน.) เพื่อนำมาใช้ในการวิเคราะห์ ผลการทดลองแสดงให้เห็นว่าทฤษฎีที่นำเสนอสามารถใช้ได้จริงอย่างมีประสิทธิภาพ

Thesis Title	Mathematical aspects of flow graph approaches to data analysis
Student	Miss Doungnat Chitcharoen
Student ID.	50067111
Degree	Doctor of Philosophy
Program	Applied Mathematics
Year	2010
Thesis Advisor	Asst. Prof. Dr. Puntip Pattaraintakorn

ABSTRACT

Rough set theory has attracted both practical and theoretical researchers because of its efficient and effective applications to data analysis. A mathematical flow graph is a significant extension of rough set theory. In this thesis, both theoretical and practical discoveries related to flow graphs are established.

The three topics considered in this thesis are data driven technique, structure and measurement. Our data-driven technique is a mechanism to calculate the certainty, coverage and strength coefficients of decision rules from fuzzy attributes and then construct its corresponding flow graph. While studying the structure of flow graphs, we invent a new way to present all coefficients of a flow graph in a matrix form. Theorems and formulas to calculate coefficients of a flow graph and a combined flow graph both directly and by using properties of the matrix representation are given. We improve measures of flow graphs with new extensions of flow graphs to association rules in data mining. In addition, a flow graph view of rough sets' categories and measures leads to a new methodology of inductively reasoning from data. We present this perspective, as well as interesting relationships between, and properties of rough sets, flow graphs and inverse flow graphs.

To illustrate that these propositions are indeed feasible and applicable in real situations, we simulate and conduct experiments to the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN) data. The obtained mathematical properties and formulas can be employed to analyze the data successfully. To this end, a series of contributions are developed.

Acknowledgements

I am heartily thankful to my thesis Advisor, Asst. Prof. Dr. Puntip Pattaraintakorn, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the thesis. This thesis is supported by the Graduate school of Science King Mongkut's Institute of Technology Ladkrabang. I am also grateful to Asst. Prof. Dr. Chartchai Leenawong, Assoc. Prof. Dr. Kanlaya Naruedomkul, Asst. Prof. Dr. Puntani Pongsumpun, Dr. Sukuman Sarikavanij, Dr. Gregory M. Zaverucha and the Promotion of Academic Olympiads and Development of Science Education Foundation, Thailand respectively.

Finally, I thank my parents Udom and Pornpen Chitcharoen and my friends for giving me this opportunity to perceive PhD education. Without having their moral support, I wouldn't have accomplished.

Doungnat Chitcharoen

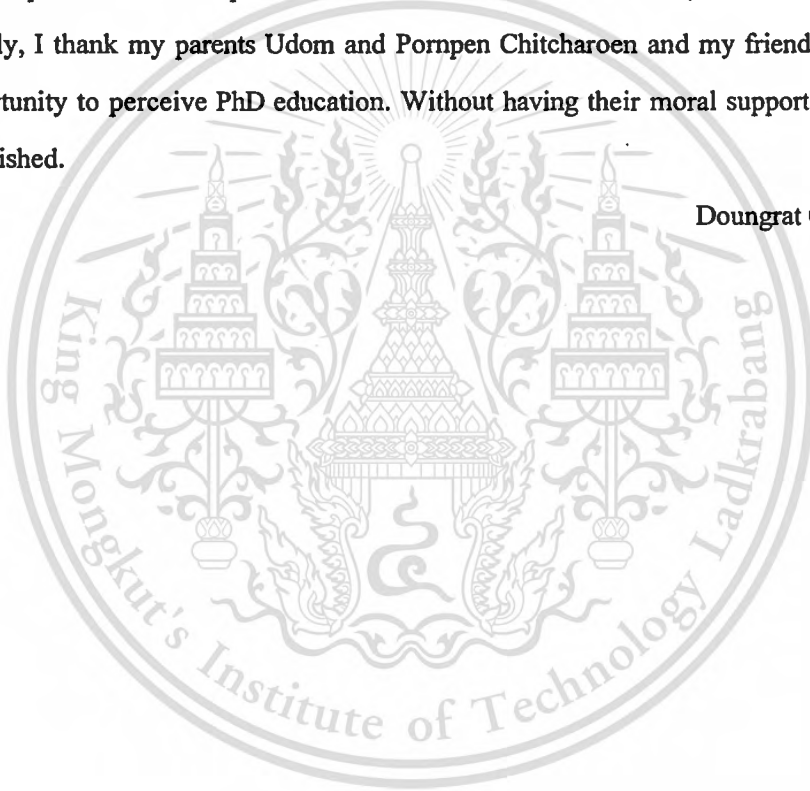


Table of Contents

	Page
Thai Abstract.....	I
English Abstract.....	II
Acknowledgements.....	III
Table of Contents.....	IV
List of Tables.....	VI
List of Figures.....	VII
CHAPTER 1 INTRODUCTION.....	1
1.1 Importance and Motivation.....	1
1.2 Objectives.....	2
1.3 Scope of the Study.....	3
1.4 Expected Results.....	3
1.5 Research Methodology.....	3
1.6 Schedule of Process of the Study.....	4
CHAPTER 2 LITERATURE REVIEW.....	5
CHAPTER 3 FLOW GRAPHS AND THEIR EXTENSION TO ASSOCIATION RULES.....	8
3.1 Flow Graphs and Normalized Flow Graphs.....	8
3.2 Paths and Connections.....	10
3.3 Dependencies.....	11
3.4 Illustrative Example for Flow Graph Construction from Descriptive Attributes....	12
3.5 Association Rules.....	14
3.6 Experimental Setting (POSN data set).....	16
3.7 Experimental Results.....	17

Table of Contents (Cont.)

	Page
CHAPTER 4 MATRIX FORMS OF FLOWGRAPHS AND THEIR HETEROGENEOUS DATA SOURCES APPLICATIONS.....	21
4.1 Flow Graphs' Representation in Matrix Forms.....	21
4.2 Multi-Datasets Flow Graphs.....	32
4.3 Experimental Results.....	35
CHAPTER 5 FLOW GRAPHS CONSTRUCTION FROM FUZZY SETS.....	40
5.1 Fuzzy Set Theory.....	40
5.2 Flow Graphs with Fuzzy Attributes.....	41
5.3 Illustrative Example for Flow Graph Construction from fuzzy Attributes.....	43
5.4 Experimental Results.....	48
CHAPTER 6 AN EXTENSION OF ROUGH SET APPROXIMATION TO FLOW GRAPH BASE ON DATA ANALYSIS.....	51
6.1 Rough Set Theory.....	51
6.2 Rough Set Approximation and Flow Graphs.....	53
CHAPTER 7 CONCLUSION AND RECOMMENDATION.....	59
7.1 Conclusion	59
7.2 Recommendation.....	60
REFERENCES.....	61
APPENDIX.....	65
APPENDIX A Data Set Used in Chapter 3.....	65
APPENDIX B Data Set Used in Chapter 5.....	68
APPENDIX C List of Publications	71
AUTHOR BIOGRAPHY.....	114

List of Tables

Tables	Page
3.1 Description of the POSN data.....	16
4.1 (a) POSN decision table of N^1	35
(b)A POSN decision table of N^2	35
4.2 Models for N^1	38
4.3 Models for N^2	38
4.4 Models for $N^1 \cup N^2$	39
5.1 Decision table with fuzzy attributes	41
5.2 An example decision table.....	44
5.3 Decision table with fuzzy attributes from Table 5.2.....	44
5.4 Strength between score of Number theory and Algebra ($\sigma(N_i, A_j)$).....	45
5.5 Strength between score of Algebra and total score ($\sigma(N_i, A_j)$).....	45
5.6 Strength between score of Number theory Algebra and total score ($\sigma(N_i, A_j, D_k)$)	46
5.7 Certainty and coverage coefficients between N_i and A_j	46
5.8 Certainty and coverage coefficients between A_j and D_k	47
5.9 Certainty and coverage coefficients between N_i and D_k	47
5.10 Certainty and coverage coefficients between N_i, A_j and D_k	48
A.1 A portion of POSN decision table used in chapter 3.....	65
B.1 A portion of POSN decision table used in chapter 5.....	66

List of Figures

Figures	Page
3.1 Flow graph initial data of voting problem	12
3.2 Social class, age and party votes normalized flow graph.....	13
3.3 Combined normalized flow graph of Fig. 3.2.....	13
3.4 Flow graph of initial Olympiad mathematics' students score from POSN.	18
3.5 Comparison of <i>sup</i> , <i>conf</i> and <i>accuracy/100</i> from four cases of POSN data set	19
4.1 A normalized flow graph with missing coefficients	31
4.2 Combined normalized flow graph of Fig. 4.1	32
4.3 A normalized flow graph of Fig. 4.1 with filled coefficients... ..	32
4.1 Flow graph of voting data of N^1 with 40 voters.	33
4.2 Flow graph of voting data of N^2 with 60 voters.	34
4.3 Flow graph of voting data of $N^1 \cup N^2$ with 40 voters.	34
4.4 Flow graph of initial data for N^1	35
4.5 Flow graph of initial data for N^2	36
5.1 Triangular membership functions.	48
5.2 Comparison of <i>cer</i> , <i>cov</i> of POSN data.	50
6.1 A normalized flow graph.....	54
6.2 Combined flow graph.	58

CHAPTER 1

INTRODUCTION

1.1 Importance and Motivation

Rough set theory was introduced by Zdzislaw Pawlak in 1982 [23]. The crux of rough set theory is the novel use of approximations to cope with uncertainty. Rough set theory plays an essential role in many applications such as decision analysis, machine learning, knowledge discovery, pattern recognition and etc. A *flow graph* is a mathematical model which is embedded in terms of rough sets. In 2002, Pawlak firstly introduced a mathematical flow graph¹ [25]. 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, while the entire flow graph describes a decision algorithm [28].

To use a mathematical flow graph to analyze real-world data, a number of tasks are involved. Since data may have several formats, the first task is to consider the type of data at hand. Afterwards, alternative structures and measurements are developed to achieve better analysis and more useful mathematical models and patterns. The three topics considered in this thesis are:

- (i) Data-driven technique: construction of flow graphs from fuzzy sets data type,
- (ii) Structure: invention of matrix forms of flow graphs and formulation of flow graphs from heterogeneous data sources,
- (iii) Measurement: mathematical connections of flow graphs to association rules and an extension of rough set approximation to flow graph.

Data driven technique This topic includes a mechanism to calculate the certainty, coverage and strength coefficients of decision rules from fuzzy attributes. Consequently, its corresponding flow graph is constructed.

Structure This topic involves a new way to present all coefficients of a flow graph in a matrix form. Theorems and formulas to calculate coefficients of a flow graph and a combined flow graph both directly and using properties of the matrix representation are given. While traditional algorithms for computing these three coefficients of the connection are exponential in l , an algorithm using proposed matrix representation is polynomial in l , where l is a number of layers of a flow graph.

Measurement New extensions of flow graphs to association rules in data mining and their coefficient are given. Furthermore a flow graph view on rough sets' categories and measures leads to a new methodology of inductively reasoning from data. In this study,

¹Pawlak's flow graphs are different to *flow networks* that do not analyze optimal flow [11].

this perspective, interesting relationships and properties among rough set, flow graphs and inverse flow graphs are offered.

To demonstrate the utility of flow graphs, real-world data collected from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN) and other benchmark data sets are experimented. The new mathematical properties and formulas can be employed to analyze this data successfully. The experimental results illustrate a desirable outcome as reported.

1.2 Objectives

The objectives of the research are as follows:

1. To devise an approach to construct flow graphs, decision rules and perform data analysis for decision table with fuzzy attributes.
2. To invent a new schema to represent throughflow of a flow graph and three coefficients of both normalized and combined normalized flow graphs in matrix form. Alternatively, starting from a flow graph with its throughflow matrix, Pawlak's formulas to calculate these three coefficients are reformed by matrix properties.
3. To improve algorithms for computing certainty, coverage and strength of the connection using our matrix representation. In some situation, if any one connected layer lacks coefficients. Hence, algorithms for computing these coefficients are provided.
4. In practical data analysis, data sets may be in the heterogeneous data sources form which require special treatment. Hence, this research offers a composition method for flow graphs and corresponding formulas to calculate their coefficients.
5. To illuminate a new extension of flow graphs to association rules in data mining. The relationship between the support and confidence of association rules and flow graphs is explored.
6. To define fundamental definitions for rough sets: four categories of vagueness, accuracy of approximation, roughness of approximation and dependency degree and state formulas to conveniently compute these measures for inverse flow graphs based on flow graph contexts.
7. To illustrate that these propositions are indeed feasible and applicable in real situations, experiments on the POSN data set and other benchmark data set are conducted.

1.3 Scope of the Study

The scope of the research is as follows:

1. Only complete paths and connections in a flow graph are studied.
2. This study is carried out for information distribution flow graphs not physical flow graphs.

1.4 Results

The benefits of the research are as follows:

1. Fuzzy sets flow graphs and their coefficients for real world data analysis are proposed. Experimental results are also included to demonstrate the applicability of this construction.
2. Pawlak's graphical flow graphs in matrix forms are presented. They allow more flexible computation and broader application.
3. Proposed algorithms to computing three coefficients of the connection using the matrix representation are polynomial in l , where l is a number of layers of a flow graph. While traditional algorithms for computing these three coefficients of the connection are exponential in l . These algorithms can also calculate lack coefficients of one connected layer.
4. According to newly defined matrix form flow graphs, this research offers formulas for calculating all coefficients using properties of matrix directly with an empirical study.
5. A composition method of heterogeneous data sources for forming a single flow graph and formulas for calculating certainty, coverage and strength coefficients in this form are given. To illustrate, several case studies are provided.
6. Two new propositions that reveal the relationship between flow graphs and association rules are discovered.
7. Definitions and properties of rough set approximations, accuracy and roughness of approximation 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.

1.5 Research Methodology

The study proceed as follows:

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

1. Study and review literature on theoretical concepts of flow graphs, rough set theory, fuzzy set theory and their relationship.
2. Devise new method to calculate three coefficients of decision rules from fuzzy data set and construct its corresponding flow graph.
3. Study flow graphs, association rules, decision rules, matrix theory and related works.
4. Propose mathematical connections between association rules and decision rules induced by flow graphs.
5. Invent a new way to present the throughflow, certainty, coverage and strength coefficients of a flow graph in matrix form. Propose formulas to calculate coefficients of a flow graph and a combined flow graph both directly and using properties of matrix.
6. Introduce the composition of flow graphs from heterogeneous data sources and derive mathematical formulas to calculate certainty, coverage and strength coefficients.
7. Define definitions and properties of rough set approximations, accuracy and roughness of approximation which are defined in terms of a flow graph.
8. Perform real-world experiments on the POSN data set and other benchmark data sets for demonstration.
9. Refine the study and conclude, make suggestions for further work and write the thesis.

1.6 Schedule of Process of the Study

Process of the study	Month										
	1-4	5-8	9-12	13-16	15-20	21-24	25-28	27-32	31-36	37-40	41-44
Step 1											
Step 2											
Step 3											
Step 4											
Step 5											
Step 6											
Step 7											
Step 8											
Step 9											

CHAPTER 2

LITERATURE REVIEW

In this chapter, we consider both theoretical and practical literature review.

Pawlak introduced rough sets theory in 1982 [23]. The crux of rough set theory is the novel use of approximations to cope with uncertainty. The rough set approach seems to be of fundamental importance to decision analysis, knowledge discovery from databases, logic, inductive reasoning, machine learning, expert systems pattern recognition and etc. The basic ideas of rough set theory and its extensions have found many interesting applications [32, 33, 34].

In 2002, Pawlak introduced a mathematical *flow graph* which is a significant extension of rough set theory [25]. It is different from the *flow networks* introduced by Ford *et al.* that do not analyze optimal flow [11]. A flow graph represents the mathematical model of information flow from the given data set [25]. The branches of a flow graph can be constructed as decision rules, while the entire flow graph describes a decision algorithm. With every decision rule, there are three associated coefficients: *strength*, *certainty* and *coverage* [27]. During the years 2002–2006, Pawlak continued present his studies about flow graphs such as [25, 26, 27, 28, 29, 30, 31].

In 2003, Pawlak revealed the relationship between flow graphs and probability. From concept of modus ponens and modus tollens, Pawlak replaced the truth values by their corresponding probabilities and obtained *rough modus ponens* and *rough modus tollens* [26]. Later, Pawlak discovered that certainty and coverage in a flow graph satisfy Bayes' theorem, without referring to either prior or posterior probabilities [27, 28]. In 2005, 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 [29]. Pawlak also proposed a new approach to data mining and knowledge discovery based on information flow distribution [30]. There is a unique flow graph for every decision tree, which can be constructed by removing the root while its nodes are labeled by the same attribute [31].

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. A salient feature of flow graph is its capable of deriving models such as decision rules and association rules. In 2002, Greco *et al.* relaxed the concept of mutual exclusion of decision rules and showed that the properties of decision algorithms still hold. Moreover, they stated that the total probability theorem is related with modus ponens and modus

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

tollens inference rules when decision rules are true in degree of the certainty factor [12]. In 2006, Butz *et al.* showed that the traditional rough set flow graph inference algorithm has exponential time complexity. They proposed a new rough set flow graph inference algorithm that exploits the factorization [2]. This year, Pattaraintakorn *et al.* represented rough sets rule learning for ordinal prediction based on rough graphical representation of the rules [20]. In 2007, Suraj and Pancarz used flow graph as a tool for rule mining in consecutive time windows of a temporal information system [38]. This year, Chan and Tsumoto presented a rule learning algorithm for certain and boundary rules from flow graphs [3]. In 2009, Pattaraintakorn proposed flow graph analysis based on entropy computation and showed a new relationship between flow graphs and entropy, which can be applied to decision tree generation [22]

Several interesting works on flow graph applications are also conducted. In 2004, Czyzewski and Kostek applied flow graphs to knowledge extraction from the Compact Disc Database (CDDb) [10]. Since Pawlak's flow graph is a quantificational graph, i.e., it represents relations among nodes using quantity of flow, it can not exactly describe the characters of decision systems. In 2006 Sun *et al.* introduced an extension of flow graphs to solve this problem [37]. In 2008, the authors developed a model for an extended Pawlak flow graph by using granular computing [13]. In 2009, Ou Yang *et al.* combined rough set theory with flow graphs and formal concept analysis (FCA) to analyze the potential for business aviation through an empirical study [40]. The results demonstrate that the combined approaches are well suited for analyzing the market potential for business aviation and the needs of business aviation customers prior to the industrys deregulation.

Fuzzy set theory was introduced by Lofti Zadeh in 1965 [41]. A fuzzy set is a class of objects with a continuum of grader of membership. The characteristic of *membership function* of a fuzzy set is permitted to have value in the interval $[0,1]$. From original Pawlak's flow graphs, they cannot represent fuzzy attributes within this type of data sets. To solve this problem, Mieszkowicz-Rolka and Rolka introduced concept of fuzzy flow graphs [18]. They defined strength certainty and coverage of fuzzy flow graph. In 2008, Matusiewicz and Pancarz presented how to combine rough set flow graphs and max product composition of fuzzy relation equations in state prediction problems [16]. Later, form max product composition, they used eigen fuzzy sets in the prediction problems solving by means of temporal rough set flow graphs [19].

This thesis is organized as follows.

In Chapter 3, firstly we recall preliminary definitions of flow graphs, since flow graph is a useful approach to discover hidden knowledge in several formals of data in artificial intelligence and soft computing. Nevertheless, research about the relationship between flow

This material is reserved for educational use only, not allowed for commercial use.

graph and association rule is scarce, thus in this chapter we also propose data analysis based on Pawlak's flow graphs and extend them to handle association rule extraction.

Pawlak's flow graph is a useful tool for knowledge discovery and has been successfully applied in many areas. Nevertheless, representing a flow graph in matrix form is still an open problem. For these reasons, we introduce new formulas to calculate such coefficients by using matrix properties directly in Chapter 4. Moreover, in real-world databases, data sets may reside at different sources (heterogeneous data sources) which require special treatment. To analyze data in this format, data integration combines data residing at different sources, and provides the user with a unified view of these data. Thus, we introduce a new composition of flow graphs and derive formulas to calculate their coefficients in this chapter and we end up with experimental results in the last section.

In real-world data analysis, if collected data is in the form of fuzzy attributes, then it can be represented in the decision table with fuzzy attributes. To analyze data in this format, in Chapter 5, we present an alternative approach to construct flow graphs, decision rules and perform data analysis for decision table with fuzzy attributes. We end up with experimental results in the last section.

Flow distribution in flow graphs can be exploited for reasoning from large scale data set. In Chapter 6, we define fundamental definitions for rough sets: four categories of vagueness, accuracy of approximation, roughness of approximation and dependency degree. 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. Finally, a summary is provided in the last chapter along with future research directions. Complete details of the data sets used in our studies are in Appendices A and B.

CHAPTER 3

FLOW GRAPHS AND THEIR EXTENSION TO ASSOCIATION RULES

The most important aspect of data analysis is to capture all necessary semantic information embedded in the data. Along this line, the novel idea to distil hidden relationships between flow graphs and association rules will be developed theoretically. In Sections 3.1 - 3.3, we recall some concepts of flow graph, normalized flow graphs, paths, connections and dependencies [12, 25, 26, 27, 28, 29, 30, 31], respectively. Section 3.4 contains an illustrative example of flow graph construction. In Section 3.5, we introduce two new propositions which state the relationship between flow graphs and association rules. Experimental designs and settings of the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN) data are in Section 3.6. Proof of concept and experimental results are reported in Section 3.7 with summarization.

3.1 Flow Graphs and Normalized Flow Graphs

A *flow graph*, proposed by Pawlak [25], is a *directed, acyclic, finite graph* $G = (N, B, \varphi)$, where N is a set of nodes, $B \subseteq N \times N$ is a set of *directed branches*, $\varphi: B \rightarrow R^+$ is a *flow function* and R^+ is the set of non-negative real numbers.

If $(x, y) \in B$ then x is an *input* of node y denoted by $I(y)$ and y is an *output* of node x denoted by $O(x)$. Next, *input* and *output* of a flow graph G are defined respectively by $I(G) = \{x \in N \mid I(x) = \emptyset\}$ and $O(G) = \{x \in N \mid O(x) = \emptyset\}$. These inputs and outputs of G are called *external nodes* of G whereas other nodes are called *internal nodes* of G . If $(x, y) \in B$ then we call $\varphi(x, y)$ a *throughflow* from x to y . We will assume in what follows that $\varphi(x, y) \neq 0$ for every $(x, y) \in B$.

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 [29], the above equations can be considered as *flow conservation equations* [11].

Flow Graphs and Normalized Flow Graphs

In order to demonstrate interesting relationships between flow graphs and other disciplines, we consider the normalized version of flow graphs (consider σ in $[0, 1]$).

A *normalized flow graph* [30] 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 $\sigma: B \rightarrow [0, 1]$ is a *normalized flow* of (x, y) . The *strength* of (x, y) is given by

$$\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)}, \quad (3.1)$$

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

For simplicity, from now we will use normalized flow graphs only. With every node x of a flow graph G , associated *normalized inflow* and *outflow* are defined respectively as: $\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(y, x)$ and $\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x, y)$. For any internal node x , it holds that $\sigma_+(x) = \sigma_-(x) = \sigma(x)$, where $\sigma(x)$ is a *normalized throughflow* of x . 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 $\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(G)} \sigma_+(x)$. It also holds that $\sigma_+(G) = \sigma_-(G) = \sigma(G) = 1$. With every branch (x, y) of a flow graph G , the *certainty* and the *coverage* of (x, y) are defined respectively as:

$$cer(x, y) = \frac{\sigma(x, y)}{\sigma(x)}, \quad (3.2)$$

$$cov(x, y) = \frac{\sigma(x, y)}{\sigma(y)}, \quad (3.3)$$

where $\sigma(x)$ and $\sigma(y) \neq 0$.

In accordance with the previous definitions, the following are some consequence properties introduced by Pawlak in [25, 26, 27, 28, 29, 30]:

$$\sum_{y \in O(x)} cer(x, y) = 1 \quad (3.4)$$

$$\sum_{x \in I(y)} cov(x, y) = 1 \quad (3.5)$$

$$\sigma(x) = \sum_{y \in O(x)} cer(x, y) \sigma(x) = \sum_{y \in O(x)} \sigma(x, y) \quad (3.6)$$

$$\sigma(y) = \sum_{x \in I(y)} cov(x, y) \sigma(y) = \sum_{x \in I(y)} \sigma(x, y) \quad (3.7)$$

$$cer(x, y) = \frac{cov(x, y) \sigma(y)}{\sigma(x)} \quad (3.8)$$

$$cov(x, y) = \frac{cer(x, y)\sigma(x)}{\sigma(y)} \quad (3.9)$$

The above properties have a probabilistic flavor, e.g., equations (3.6) and (3.7) are the form of total probability theorem, whereas formulas (3.8) and (3.9) are Bayes rules [27, 28, 29, 30, 31].

3.2 Paths and Connections

In a flow graph G , if we concentrate on relationships of a sequence of nodes in a flow graph, we can find these relationships by use concept of directed simple path.

A (directed) *path* [30] from x to y ($x \neq y$) in G , denoted by $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ 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* $[x_1 \dots x_n]$ are defined as:

$$cer[x_1 \dots x_n] = \prod_{i=1}^{n-1} cer(x_i, x_{i+1}), \quad (3.10)$$

$$cov[x_1 \dots x_n] = \prod_{i=1}^{n-1} cov(x_i, x_{i+1}), \quad (3.11)$$

$$\sigma[x \dots y] = \sigma(x)cer[x \dots y] = \sigma(y)cov[x \dots y]. \quad (3.12)$$

Remark 1. If $cer(y, z) = 1$, we immediately obtain $\varphi[xyz] = \varphi(x)cer(x, y)$.

Remark 2. If $cov(x, y) = 1$, we immediately obtain $\varphi[xyz] = \varphi(z)cov(y, z)$.

Next, since flow graph are composed of external nodes (inputs and outputs) and internal nodes. If we are interested in relationships between input and output, then we use concept of connection to find them. The set of all paths from x to y ($x \neq y$) in G , denoted by $\langle x, y \rangle$, is a *connection* [30] of G determined by nodes x and y . For every connection $\langle x, y \rangle$, its associate *certainty*, *coverage* and *strength of the connection* $\langle x, y \rangle$ are defined as:

$$cer \langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} cer[x \dots y], \quad (3.13)$$

$$cov \langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} cov[x \dots y], \quad (3.14)$$

$$\sigma \langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} \sigma[x \dots y] = \sigma(x)cer \langle x, y \rangle = \sigma(y)cov \langle x, y \rangle. \quad (3.15)$$

If $[x \dots y]$ is a path such that x and y are input and output of the graph G , respectively, then $[x \dots y]$ 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 (respectively) of a graph G with a single branch $\langle x, y \rangle$ 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.

3.3 Dependencies

In this section, we show that the dependency factor of flow graphs has a close relationship to statistical measurements. For $(x, y) \in B$, nodes x and y are *independent* of each other if

$$\sigma(x, y) = \sigma(x)\sigma(y).$$

Thus, we have

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

$$\frac{\sigma(x, y)}{\sigma(y)} = cov(x, y) = \sigma(x).$$

Moreover, if

$$cer(x, y) > \sigma(y) \quad \text{or} \quad cov(x, y) > \sigma(x),$$

then x and y *depend positively* on each other. On the other hand, if

$$cer(x, y) < \sigma(y) \quad \text{or} \quad cov(x, y) < \sigma(x),$$

then x and y *depend negatively* on each other.

Furthermore, there is a factor that expresses the degree of dependency in the flow graph. For every $(x, y) \in B$ we define a *dependency factor*, $\eta(x, y)$, as

$$\eta(x, y) = \frac{cer(x, y) - \sigma(y)}{cer(x, y) + \sigma(y)} = \frac{cov(x, y) - \sigma(x)}{cov(x, y) + \sigma(x)}. \quad (3.16)$$

The interpretation is: if $\eta(x, y) = 0$, then x and y are independent on each other, if $-1 < \eta(x, y) < 0$, then x and y are negatively dependent on each other and if $0 < \eta(x, y) < 1$ then x and y are positively dependent on each other.

3.4 Illustrative Example for Flow Graph Construction from Descriptive Attributes

In this section, we illustrate the above ideas by means of a simple example. It is a voting analysis problem of flow graph construction (modification of [30]) as depicted in Fig. 3.1.

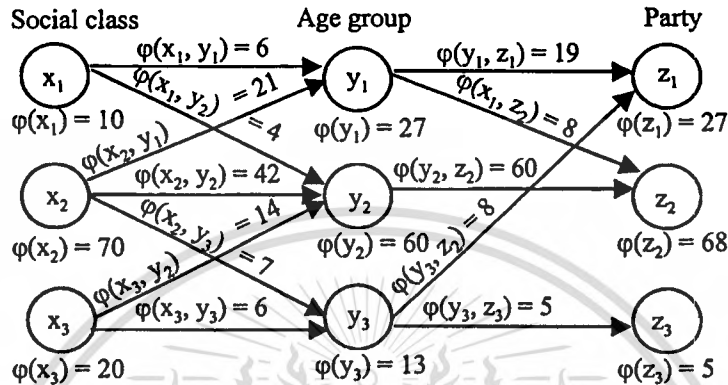


Figure 3.1 Flow graph initial data of voting problem [4].

Suppose we are given three disjoint age groups of voters: y_1 (old), y_2 (middle aged) and y_3 (young). They are belonging to three social classes x_1 (high), x_2 (middle) and x_3 (low). The voters can vote for three political parties z_1 , z_2 and z_3 . Assume that voters are grouped according to presence of the social classes, age groups and political parties as shown in Fig. 3.1. We can describe this flow graph as follows. Inputs of this flow graph are nodes x_1 , x_2 and x_3 , whereas the outputs are nodes z_1 , z_2 and z_3 . Nodes y_1 , y_2 and y_3 are internal nodes. Next, if we consider bold branches in Fig. 3.1, the throughflow of the branch (x_2, y_1) is $\varphi(x_2, y_1) = 21$. It replies that 21 voters are middle class and old. Inflow of node y_1 is $\varphi_+(y_1) = 6 + 21 = 27$. Outflow of node y_1 is $\varphi_-(y_1) = 19 + 8 = 27$. In addition, inflow of this flow graph is $\varphi_+(G) = \varphi(x_1) + \varphi(x_2) + \varphi(x_3) = 10 + 70 + 20 = 100$, and outflow of this flow graph is $\varphi_-(G) = \varphi(z_1) + \varphi(z_2) + \varphi(z_3) = 27 + 68 + 5 = 100$. Please note that $\varphi_+(G) = \varphi_-(G) = 100$.

The normalized flow graph version of Fig. 3.1 is depicted in Fig. 3.2. There are three layers in this flow graph. The first layer consists of nodes x_1 , x_2 and x_3 , the second layer consists of nodes y_1 , y_2 and y_3 and the third layer consists of nodes z_1 , z_2 and z_3 .

To illustrate, we present some examples of how to compute three coefficients of this flow graph. We can calculate strength as shown in Fig. 3.2 by using (3.1). e.g., the bold branches indicate that $\sigma(x_2, y_1) = \frac{\varphi(x_2, y_1)}{100} = \frac{21}{100} = 0.21$ which means that there are 21% of voters who are middle class and old. We can calculate certainty by using (3.2), e.g., $cer(y_1, z_1) = \frac{\sigma(y_1, z_1)}{\sigma(y_1)} = \frac{0.19}{0.27} = 0.7$ and $cer(y_1, z_2) = \frac{\sigma(y_1, z_2)}{\sigma(y_1)} = \frac{0.08}{0.27} = 0.3$, they can

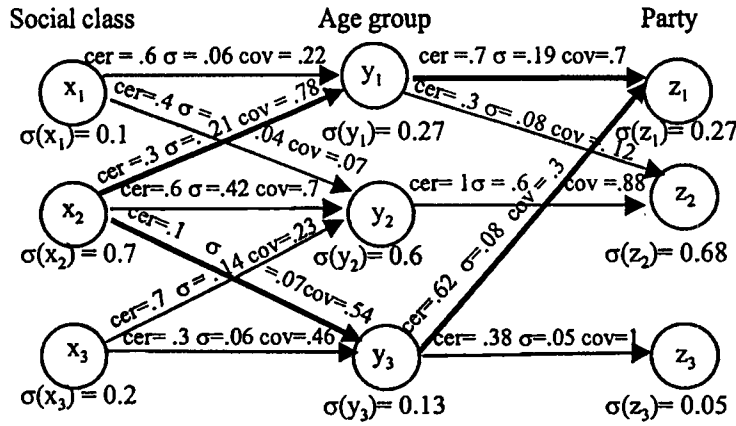


Figure 3.2 Social class, age and party votes normalized flow graph.

be interpreted as 70 % of old age group (y_1) vote for party z_1 and the rest (30 %) of old age group (y_1) vote for party z_2 . Finally, we can calculate coverage by using (3.3), e.g., $cov(y_1, z_2) = \frac{\sigma(y_1, z_2)}{\sigma(z_2)} = \frac{0.08}{0.68} = 0.12$ and $cov(y_2, z_2) = \frac{\sigma(y_2, z_2)}{\sigma(z_2)} = \frac{0.6}{0.68} = 0.88$, they can be interpreted as 12 % of party z_1 votes come from old age group (y_1) and the rest (88 %) of votes come from middle age group (y_2).

Next, if we focus to vote distribution between parties and social classes, then we construct its combined normalized flow graph. The combined normalized flow graph in Fig. 3.3 comes from all of complete connections between parties and social classes in Fig. 3.2. We can calculate certainty, coverage and strength of all connections in Fig. 3.3 by using (3.13), (3.14) and (3.15), respectively.

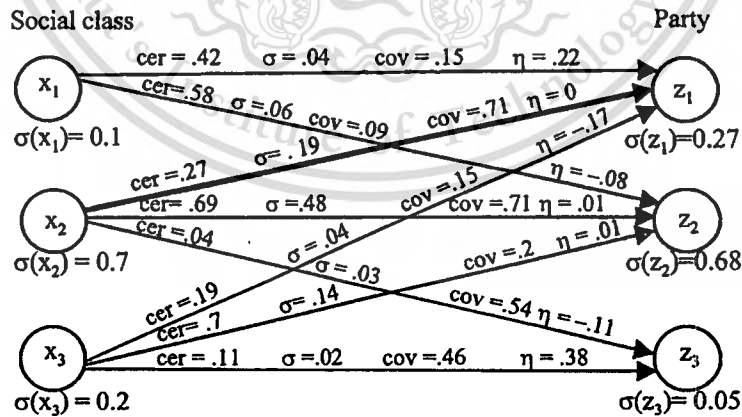


Figure 3.3 Combined normalized flow graph of Fig. 3.2.

For instance, the connection $\langle x_2, z_1 \rangle$ (bold branch in Fig. 3.3) consists of paths $[x_2 y_1 z_1]$ and $[x_2 y_3 z_1]$ (bold paths in Fig. 3.2). Then we can calculate certainty, coverage and strength of $\langle x_2, z_1 \rangle$ as:

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

$$\begin{aligned}
cer \langle x_2, z_1 \rangle &= cer(x_2, y_1) \times cer(y_1, z_1) + cer(x_2, y_3) \times cer(y_3, z_1) \\
&= (0.3 \times 0.7) + (0.1 \times 0.62) = 0.27, \\
cov \langle x_2, z_1 \rangle &= cov(x_2, y_1) \times cov(y_1, z_1) + cov(x_2, y_3) \times cov(y_3, z_1) \\
&= (0.78 \times 0.7) + (0.54 \times 1) = 0.71 \text{ and} \\
\sigma \langle x_2, z_1 \rangle &= \sigma(x_2)cer \langle x_2, z_1 \rangle = \sigma(z_1)cov \langle x_2, z_1 \rangle \\
&= 0.7 \times 0.27 = 0.27 \times 0.71 = 0.19.
\end{aligned}$$

It can be interpreted as 27% of middle class votes party z_1 , 71% of party z_1 votes come from middle class and 19% of voters who are middle class votes party z_1 , respectively.

Moreover, if we consider dependency factors between parties and social classes, then we use (3.16) to compute dependency factors and result was shown in Fig. 3.3. For instance, from (3.16) we compute dependency factors between social classes middle and party z_1 (bold branch in Fig. 3.3) as:

$$\eta(x_2, z_1) = \frac{cer(x_2, z_1) - \sigma(z_1)}{cer(x_2, z_1) + \sigma(z_1)} = \frac{0.27 - 0.27}{0.27 + 0.27} = 0.$$

It can be interpreted as social classes middle and party z_1 are independent.

3.5 Association Rules

In a decision analysis, an *association rule* is a popular and well researched method for discovering interesting relations between variables in a given data set. An association rule identifies frequently occurring patterns of information in a given set of data [36]. The application of association rule mining is market basket analysis. The association rule is an implication of the form $X \rightarrow Y$. A pattern X can be an item X_i or a conjunctive item set $X_i \wedge \dots \wedge X_j$ where $X_i, \dots, X_j$ are sets of data, $1 \leq i, j \leq n, i \neq j$. X is called the *antecedence*, while Y is called the *consequence* of the rule.

The rule $X \rightarrow Y$ can be interpreted as “if itemset X occurs in a transaction, then itemset Y will also likely occur in the same transaction”. For example, in the sales data of a supermarket, if we have the association rule, $\{pok\ bun, dumpling\} \rightarrow \{Pepsi\}$, then it indicates that if a customer buys pork bun dumpling together, their will also buy Pepsi. This association rule can be used to draw decisions about marketing activities such as promotional pricing and product placements.

A rule has two associated coefficients, called support and confidence [36]. The *support* (*sup*) of the rule $X \rightarrow Y$ is the ratio of the number of data satisfy X and Y to the number of the total data. The *confidence* (*conf*) of the rule is the ratio of the number of data which satisfy X and Y to the size of the set X . Moreover, *sup* and *conf* of the rule $X \rightarrow Y$ satisfy $P(X, Y)$ and $P(Y|X)$ [1], respectively.

From meaning of support and confident as above, we have the following relations.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Remark 3. $\sup(x, y) = \sigma(x, y)$ and $\text{conf}(x, y) = \text{cer}(x, y)$.

Remark 4. $\sup[xyz] = \frac{\varphi[xyz]}{\varphi(G)}$ and $\text{conf}[xyz] = \frac{\varphi[xyz]}{\varphi(x, y)}$.

For instance, the branch (x_2, y_1) in Fig. 3.3, we obtain association rule by using Remark 3 as: $x_2 \rightarrow y_1$, $\sup = 0.21$ and $\text{conf} = 0.30$.

We next come to our proposed propositions to reveal relationship between association rules and flow graphs.

Proposition 1. Let $G = (N, B, \sigma)$ be a flow graph. For every path $[xyz]$, $\text{cer}(y, z) = 1$, an association rule: $(x \wedge y) \rightarrow z$, where x, y , are antecedences and z is a consequence has

- (1) $\sup[xyz] = \sigma(x, y)$,
- (2) $\sup[xyz] = \sigma[xyz]$,
- (3) $\text{conf}[xyz] = 1$.

Proof. (1) From Remark 4, we have $\sup[xyz] = \frac{\varphi[xyz]}{\varphi(G)}$. By using Remark 1, (3.1) and (3.2), then $\sup[xyz] = \frac{\varphi(x)\text{cer}(x, y)}{\varphi(G)} = \sigma(x)\text{cer}(x, y) = \sigma(x, y)$.

(2) From (1), $\sup[xyz] = \sigma(x, y)$. We next multiply by $\frac{\sigma(x)}{\sigma(x)}$ and use (3.2) to obtain $\sup[xyz] = \frac{\sigma(x)\sigma(x, y)}{\sigma(x)} = \sigma(x)\text{cer}(x, y)$. If $\text{cer}(y, z) = 1$, then $\sup[xyz] = \sigma(x)\text{cer}(x, y)\text{cer}(y, z)$. Finally, (3.10) and (3.12) give us $\sup[xyz] = \sigma[xyz]$.

(3) By Remarks 1, 4 and (3.2), $\text{conf}[xyz] = \frac{\varphi[xyz]}{\varphi(x, y)} = \frac{\varphi(x)\text{cer}(x, y)}{\varphi(x, y)} = \frac{\text{cer}(x, y)}{\text{cer}(x, y)} = 1$. $\square$

For example, by using Fig. 3.3, since $\text{cer}(y_2, z_2) = 1$, we obtain three association rules by using our presented propositions as:

- $(x_1 \wedge y_2) \rightarrow z_2$, $\sup = 0.04$ and $\text{conf} = 1$,
- $(x_2 \wedge y_2) \rightarrow z_2$, $\sup = 0.42$ and $\text{conf} = 1$,
- $(x_3 \wedge y_2) \rightarrow z_2$, $\sup = 0.14$ and $\text{conf} = 1$.

Proposition 2. Let $G = (N, B, \sigma)$ be a flow graph. For every path $[xyz]$, $\text{cov}(x, y) = 1$, an association rule: $(x \wedge y) \rightarrow z$, where x, y , are antecedences and z is a consequence has

- (1) $\sup[xyz] = \sigma(y, z)$,
- (2) $\sup[xyz] = \sigma[xyz]$,
- (3) $\text{conf}[xyz] = \text{cer}(y, z)$.

Proof. (1) From Remark 4, we have $\sup[xyz] = \frac{\varphi[xyz]}{\varphi(G)}$. By using Remark 2, (3.1) and (3.3), then $\sup[xyz] = \frac{\varphi(z)\text{cov}(y, z)}{\varphi(G)} = \sigma(z)\text{cov}(y, z) = \sigma(y, z)$.

(2) From (1), $\sup[xyz] = \sigma(y, z)$. We next multiply by $\frac{\sigma(z)}{\sigma(z)}$ and use (3.3) to obtain $\sup[xyz] = \frac{\sigma(z)\sigma(y, z)}{\sigma(z)} = \sigma(z)\text{cov}(y, z)$. If $\text{cov}(x, y) = 1$, then $\sup[xyz] = \sigma(z)\text{cov}(x, y)\text{cov}(y, z)$. Finally, (3.11) and (3.12) give us $\sup[xyz] = \sigma[xyz]$.

(3) By Remarks 2, 4 and (3.3), $\text{conf}[xyz] = \frac{\varphi[xyz]}{\varphi(x, y)} = \frac{\varphi(z)\text{cov}(y, z)}{\varphi(x, y)} = \frac{\sigma(y, z)}{\sigma(x, y)}$. Since $\text{cov}(x, y) = 1$ and by (3.3), then $\text{conf}[xyz] = \frac{\sigma(y, z)}{\sigma(y)} = \text{cer}(y, z)$. $\square$

3.6 Experimental Setting (POSN data set)

To illustrate the effectiveness of the proposed ideas and their operations, consider the analysis task represented by the POSN data set of Table 3.1. This data set consists of 100 records of students collected from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN) between 2006–2007. The Olympiad mathematics' students' analysis will be explored as a real-world experiment later.

Table 3.1. Description of the POSN data

	Attribute Sets	Values
1	preference subjects	Geometry (<i>prefer_Geo</i>), Number theory (<i>prefer_Num</i>), Algebra (<i>prefer_Alg</i>), Combinatorics (<i>prefer_Comb</i>), Inequality (<i>prefer_I.E.</i>),
2	practiced subjects	Geometry (<i>prac_Geo</i>), Number theory (<i>prac_Num</i>), Algebra (<i>prac_Alg</i>), Combinatorics (<i>prac_Comb</i>), Inequality (<i>prac_I.E.</i>),
4	least preference subjects	Geometry (<i>leastprefer_Geo</i>), Number theory (<i>leastprefer_Num</i>), Algebra (<i>leastprefer_Alg</i>), Combinatorics (<i>leastprefer_Comb</i>), Inequality (<i>leastprefer_I.E.</i>),
5	least practiced subjects	Geometry (<i>leastprac_Geo</i>), Number theory (<i>leastprac_Num</i>), Algebra (<i>leastprac_Alg</i>), Combinatorics (<i>leastprac_Comb</i>), Inequality (<i>leastprac_I.E.</i>),
6	score of Geometry (<i>score_Geo</i>)	<i>high, low</i> (or G_H, G_L) ¹
7	score of Number theory (<i>score_Num</i>)	<i>high, low</i> (or N_H, N_L)
8	score of Algebra (<i>score_Alg</i>)	<i>high, low</i> (or A_H, A_L)
9	score of Combinatorics (<i>score_Comb</i>)	<i>high, low</i> (or C_H, C_L)
10	score of Inequality (<i>score_I.E.</i>)	<i>high, low</i> (or I_H, I_L)
11	total score (<i>total_score</i>)	<i>high, low</i> (or z_i)

¹For simplicity of flow graph graphical representation, we use these notations interchangeably.

There are two sources of data collection: questionnaires from students and their test results. Here the decision attribute *total_score*, which can have values high or low, is to be predicted based on condition (other) attributes. The first five attributes in Table 3.1 are condition attributes from questionnaires while the rest is from test results.

We consider the POSN data set as two independent data sets:

(i) POSN 1: preference subject, practiced subject, best score in subject and total score (see more in Section 3.7)

(ii) POSN 2: preference subject, practiced subject, score of Geometry, score of Number theory, score of Algebra, score of Combinatorics, score of Inequality and total score (see more in Section 5.5).

In addition, we consider two heterogeneous data sources according to the gender of students as listed in Section 4.3.

Our approaches in this thesis can also be applied to a wide range of data sets and domains - both benchmark and actual data sets. Complete details of the data sets used in this research are in Appendices A and B .

3.7 Experimental Results

We describe initial experimental results from the scores of Olympiad mathematics' students' analysis. To create association rules from a flow graph we consider a portion of the POSN data set contains 75 records (see Appendix A).

In the first two cases, it consists of five disjoint preference subjects: *prefer_Geo*, *prefer_Num*, *prefer_Alg*, *prefer_Comb* and *prefer_I.E.* denoted as x_1, x_2, x_3, x_4 and x_5 which belong to five most practiced subjects: *prac_Geo*, *prac_Num*, *prac_Alg*, *prac_Comb* and *prac_I.E.* denoted as y_1, y_2, y_3, y_4 and y_5 . We concentrate on what affect the students' total score, z_i^2 . In the first case study, derived from Fig. 3.4, these association rules come up with z_1 and z_2 stand for total score ≥ 82 and ≤ 81 , respectively. First of all, in Fig. 3.4, if to check what subject which its preference affects practiced subjects, we consider from the flow graph directly (between first and second layers). One can see that most students practice their preferred subjects. For example, the branch (x_5, y_5) represents an association rule $x_5 \rightarrow y_5$ with $sup = 0.04$ and $conf = \frac{0.04}{0.05} = 0.80$. In this graph we can draw three conclusions as: (i) most of students who practice geometry or inequality, will have high total score, (ii) practicing number theory (or combinatorics), does not affect total score and (iii) most of students who practice algebra, will have low total score.

²The thresholds for association rule extraction are $sup > 0.04$ and $conf > 0.50$.

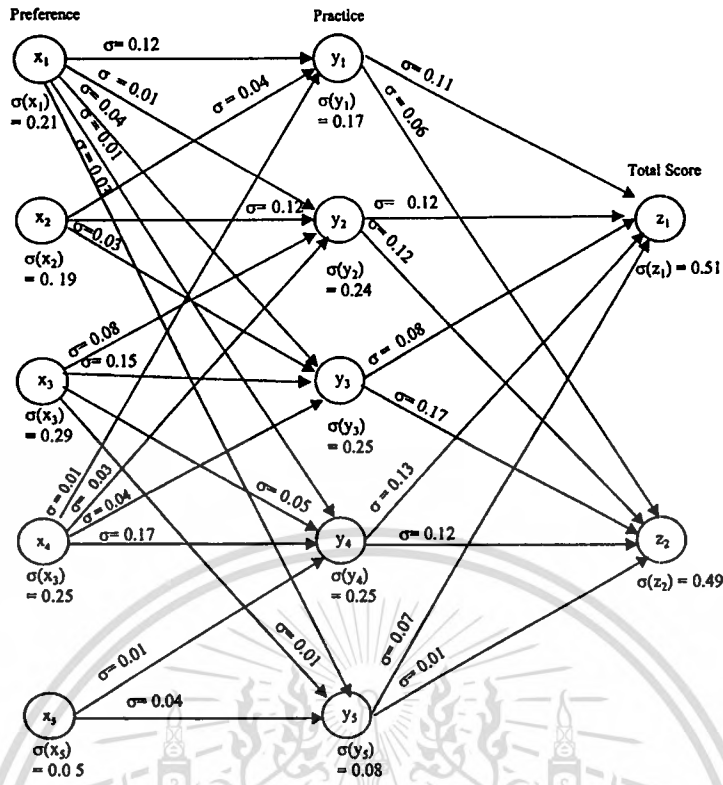


Figure 3.4 Flow graph of initial Olympiad mathematics' students score from POSN.

Case 1. If z_1 (total score ≥ 82) and z_2 (total score ≤ 81), then the selected association rules are:

Rules	Sup	Conf
IF <i>prefer_I.E.</i> and <i>prac_I.E.</i> THEN <i>total_score</i> ≥ 82	0.04	1 ³
IF <i>prefer_I.E.</i> THEN <i>total_score</i> ≥ 82	0.05	1
⋮		
IF <i>prac_Num</i> THEN <i>total_score</i> ≥ 82	0.12	0.50

with average sup = 0.10, average conf = 0.67 and accuracy = 67 %.

Case 2. If z_1 (total score ≥ 93) and z_2 (total score ≤ 92), then the selected association rules are:

Rules	Sup	Conf
IF <i>prefer_I.E.</i> and <i>prac_I.E.</i> THEN <i>total_score</i> ≥ 93	0.04	1
IF <i>prefer_Alg</i> and <i>prac_Alg</i> THEN <i>total_score</i> ≤ 92	0.13	0.91
⋮		
IF <i>prefer_Geo</i> THEN <i>total_score</i> ≥ 93 ,	0.12	0.50,

with average sup = 0.10, average conf = 0.66 and accuracy = 68 %.

³Due to round off errors in computations, the equations presented in earlier sections may be not satisfied exactly in our examples.

From these selected association rules in cases 1 and 2, we can distil knowledge from this flow graph: if students prefer or practice inequality, they will have high total scores.

In cases 3 and 4, it consists of five disjoint least preference subjects: *leastprefer_Geo*, *leastprefer_Num*, *leastprefer_Alg*, *leastprefer_Comb* and *leastprefer_I.E.* denoted as x_1, x_2, x_3, x_4 and x_5 which belong to five least practiced subjects: *leastprac_Geo*, *leastprac_Num*, *leastprac_Alg*, *leastprac_Comb* and *leastprac_I.E.* denoted as y_1, y_2, y_3, y_4 and y_5 . We concentrate on what affect the students' total score, z_i .

Case 3. If z_1 (total score ≥ 82) and z_2 (total score ≤ 81), then the selected association rule are:

Rules	Sup	Conf
IF <i>leastprefer_I.E.</i> and <i>leastprac_Geo</i> THEN <i>total_score</i> ≤ 81	0.07	1
IF <i>leastprefer_Geo</i> and <i>leastprac_I.E.</i> THEN <i>total_score</i> ≤ 81	0.04	1
⋮		
IF <i>leastprefer_I.E.</i> and <i>leastprac_I.E.</i> THEN <i>total_score</i> ≤ 81	0.24	0.56

with average *sup* = 0.12, average *conf* = 0.75 and accuracy = 68 %.

Case 4. If z_1 (total score ≥ 93) and z_2 (total score ≤ 92), then the selected association rule are:

Rules	Sup	Conf
IF <i>leastprefer_I.E.</i> and <i>leastprac_Geo</i> THEN <i>total_score</i> ≤ 92	0.07	1
IF <i>leastprac_Comb</i> THEN <i>total_score</i> ≤ 92	0.09	0.78,
⋮		
IF <i>leastprefer_Alg.</i> THEN <i>total_score</i> ≤ 92	0.04	0.50

with average *sup* = 0.13 and average *conf* = 0.68 and accuracy = 73 %.

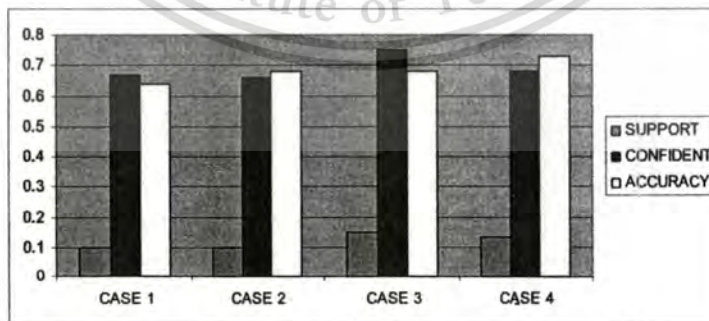


Figure 3.5 Comparison of *sup* and *conf* from four cases of POSN data set.

From these selected association rules in cases 3 and 4, we can distil knowledge from this flow graph: if students do not like inequality (or geometry) or do not learn inequality, they will have low total scores.

In cases 1 - 4, we can distil knowledge as: inequality preference and practicing affect the students' total score. That are if students prefer and practice inequality, they will have high total score and if students do not prefer or lest practice inequality, they will have low total score. In case 3, we obtain maximum average confidence and case 4, we obtain maximum average support as shown in Fig. 3.5. The highest accuracy to predict students' score is 73 % from case 4.

In this chapter, an extension of rough set theory, namely flow graphs, is considered. We present two new propositions that reveal the relationship between flow graphs and association rules. In addition surveying the research in this area, we also performed an experiment on real-world data to demonstrate the utility of flow graphs with its evaluation.



CHAPTER 4

MATRIX FORMS OF FLOW GRAPHS AND THEIR HETEROGENEOUS DATA SOURCES APPLICATIONS

Pawlak's flow graph is a useful tool for knowledge discovery and has been successfully applied in many areas. Nevertheless, representing a flow graph in matrix form is still an open problem. From a practical point of view, matrix notation can be useful in processing of large data set. Thus, in this thesis we define throughflow of a flow graph and three coefficients of a normalized and a combined normalized flow graph in matrix form. Consequently, matrix properties are employed to calculate coefficients and discover new properties of such flow graphs. Traditional algorithms for computing three coefficients of the connection are exponential in l whereas an algorithm using our matrix representation is polynomial in l , where l is a number of layers of a flow graph.

More importantly, in real-world databases, data sets may reside at different sources (heterogeneous data sources) which require special treatment. To analyze data in this format, data integration combines data residing at different sources, and provides the user with a unified view of these data. The problem of designing data integration systems is important in current real-world applications and is characterized by a number of issues that are interesting from a theoretical point of view [15].

The rest of this chapter is organized as follows. Section 4.1 contains our definitions for representing the throughflow of a flow graph and three coefficients related to flow graphs in matrix forms. In this section, we also introduce new formulas to calculate such coefficients by using matrix properties directly. We introduce the composition of flow graphs from heterogeneous data sources and formulas to calculate the corresponding coefficients in Section 4.2. The actual data set from the POSN and the experimental results are presented and discussed in Section 4.3.

4.1 Flow Graphs' Representation in Matrix Forms

The study of matrices has established theories and algorithms to solve or analyze problems and applications in many areas (e.g., systems of linear equations, differential equations, economics, graph theory, communication networks, electrical networks etc. [35]). Nevertheless, research on presenting flow graphs in matrix form is scarce. Thus, in this section we first define throughflow of a flow graph and a normalized flow graph with their

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

coefficients in matrix form. Then, we apply known results on matrices in order to simplify and analyze three coefficients of these flow graphs at the end of this section.

In Definitions 1 - 5, we introduce new methods for representing throughflow and three coefficients in matrix form directly from an initial flow graph.

Definition 1. Let $G = (N, B, \varphi)$ be a flow graph, let X be an attribute with n values: $x_1, x_2, \dots, x_n$, and let Y be an attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n$, $1 \leq j \leq m$. The throughflow of G is represented in matrix form as

$$G_{\varphi_G(x_i, y_j)} = [\varphi_{i,j}]_{n \times m}, \varphi_{i,j} = \varphi(x_i, y_j),$$

where $1 \leq i \leq n$ and $1 \leq j \leq m$.

Definition 2. Let $G = (N, B, \sigma)$ be a normalized flow graph, let X be an attribute with n values: $x_1, x_2, \dots, x_n$, and let Y be an attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n$, $1 \leq j \leq m$. The strength, certainty and coverage of G are represented in matrix form as:

1. $G_{\sigma_G(x_i, y_j)} = [\sigma_{i,j}]_{n \times m}$, $\sigma_{i,j} = \sigma(x_i, y_j)$,
2. $G_{cer_G(x_i, y_j)} = [cer_{i,j}]_{n \times m}$, $cer_{i,j} = cer(x_i, y_j)$,
3. $G_{cov_G(x_i, y_j)} = [cov_{i,j}]_{n \times m}$, $cov_{i,j} = cov(x_i, y_j)$,

where $1 \leq i \leq n$ and $1 \leq j \leq m$.

Definitions 1-2 are the representations of throughflow and the strength, certainty and coverage of directed branches in connected layers in matrix form. Let us give a brief observation.

- For matrices $G_{\varphi_G(x_i, y_j)}$, $G_{\sigma_G(x_i, y_j)}$, $G_{cer_G(x_i, y_j)}$ and $G_{cov_G(x_i, y_j)}$ the number of rows, n , is equal to the number of attributes in X and the number of columns, m , is equal to the number of attributes in Y .
- For $G_{\varphi_G(x_i, y_j)}$, we see that the summation of the i^{th} row is an outflow of x_i while summation of the j^{th} column is an inflow of y_j , for all i, j .
- The summation of all elements in $G_{\varphi_G(x_i, y_j)}$ equals the throughflow of the flow graph.
- For a matrix $G_{\sigma_G(x_i, y_j)}$, the summation of the i^{th} row is a normalized outflow of x_i while the summation of the j^{th} column is a normalized inflow of y_j , $1 \leq i \leq n$, $1 \leq j \leq m$ and the summation of all elements in the matrix is 1.
- The summation of any row in $G_{cer_G(x_i, y_j)}$ is 1 and the summation of any column in $G_{cov_G(x_i, y_j)}$ is 1.

To illustrate, from Fig. 3.1, we can represent the throughflow from x_i to y_j , $1 \leq i, j \leq 3$, for all sets of connected branches between the first and the second layers in the matrix form denoted by $G_{\varphi_G(x_i, y_j)}$ as: $G_{\varphi_G(x_i, y_j)} = \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix}$.

Similarly, the strength, certainty and coverage of (x_i, y_j) , denoted by $G_{\sigma_G(x_i, y_j)}$, $G_{cer_G(x_i, y_j)}$ and $G_{cov_G(x_i, y_j)}$, of the normalized flow graph in Fig. 3.2 in matrix form are:

$$\begin{aligned} G_{\sigma_G(x_i, y_j)} &= \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix}, \\ G_{cer_G(x_i, y_j)} &= \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix} \text{ and} \\ G_{cov_G(x_i, y_j)} &= \begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix}. \end{aligned}$$

After considering matrices representing throughflow, strength, certainty and coverage for connected layers, we extend to a flow graph with more than two layers. Definitions 3 and 4 generalize the previous definition to a flow graph and a normalized flow graph with $n - 1$ condition attributes and one decision attribute. We use $\diamond$ to denote a (binary) concatenation operator¹ for any connected layers.

Definition 3. Let $G = (N, B, \varphi)$ be a flow graph, X^i be a condition attribute with k_i values that are $x_1^i, x_2^i, \dots, x_{j_1}^i, \dots, x_{k_i}^i$, $1 \leq i \leq n - 1$, and X^n be a decision attribute with k_n values that are $x_1^n, x_2^n, \dots, x_{j_n}^n, \dots, x_{k_n}^n$ such that $(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \in B$, $1 \leq i \leq n - 1$, $1 \leq j_i \leq k_i$. Then throughflow of G is represented in matrix form as:

$$\begin{aligned} G_{\varphi_G} &= [\varphi_{j_1, j_2}^{1,2}]_{k_1 \times k_2} \diamond [\varphi_{j_2, j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [\varphi_{j_{n-1}, j_n}^{n-1, n}]_{k_{n-1} \times k_n}; \\ \varphi_{j_1, j_2}^{1,2} &= \varphi(x_{j_1}^1, x_{j_2}^2), \varphi_{j_2, j_3}^{2,3} = \varphi(x_{j_2}^2, x_{j_3}^3), \dots, \varphi_{j_{n-1}, j_n}^{n-1, n} = \varphi(x_{j_{n-1}}^{n-1}, x_{j_n}^n), \text{ where } 1 \leq j_i \leq k_i. \end{aligned}$$

Definition 4. Let $G = (N, B, \sigma)$ be a normalized flow graph, X^i be a condition attribute with k_i values that are $x_1^i, x_2^i, \dots, x_{j_1}^i, \dots, x_{k_i}^i$, $1 \leq i \leq n - 1$, and X^n be a decision attribute with k_n values that are $x_1^n, x_2^n, \dots, x_{j_n}^n, \dots, x_{k_n}^n$ such that $(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \in B$, $1 \leq i \leq n - 1$, $1 \leq j_i \leq k_i$. Then strength, certainty and coverage of G are represented in matrix form as follows:

$$\begin{aligned} 1. \quad G_{\sigma_G} &= [\sigma_{j_1, j_2}^{1,2}]_{k_1 \times k_2} \diamond [\sigma_{j_2, j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [\sigma_{j_{n-1}, j_n}^{n-1, n}]_{k_{n-1} \times k_n}; \\ \sigma_{j_1, j_2}^{1,2} &= \sigma(x_{j_1}^1, x_{j_2}^2), \sigma_{j_2, j_3}^{2,3} = \sigma(x_{j_2}^2, x_{j_3}^3), \dots, \sigma_{j_{n-1}, j_n}^{n-1, n} = \sigma(x_{j_{n-1}}^{n-1}, x_{j_n}^n), \\ 2. \quad G_{cer_G} &= [cer_{j_1, j_2}^{1,2}]_{k_1 \times k_2} \diamond [cer_{j_2, j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [cer_{j_{n-1}, j_n}^{n-1, n}]_{k_{n-1} \times k_n}; \\ cer_{j_1, j_2}^{1,2} &= cer(x_{j_1}^1, x_{j_2}^2), cer_{j_2, j_3}^{2,3} = cer(x_{j_2}^2, x_{j_3}^3), \dots, cer_{j_{n-1}, j_n}^{n-1, n} = cer(x_{j_{n-1}}^{n-1}, x_{j_n}^n), \end{aligned}$$

¹ $\diamond$ is similar to a concatenation operator, $\|$, in computer networks.

$$3. G_{covG} = [cov_{j_1 j_2}^{1,2}]_{k_1 \times k_2} \diamond [cov_{j_2 j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [cov_{j_{n-1} j_n}^{n-1,n}]_{k_{n-1} \times k_n};$$

$$cov_{j_1 j_2}^{1,2} = cov(x_{j_1}^1, x_{j_2}^2), cov_{j_2 j_3}^{2,3} = cov(x_{j_2}^2, x_{j_3}^3), \dots, cov_{j_{n-1} j_n}^{n-1,n} = cov(x_{j_{n-1}}^{n-1}, x_{j_n}^n),$$

where $1 \leq j_n \leq k_n$.

For example, we can represent the throughflow of the entire flow graph in Fig. 3.1, and the strength, certainty and coverage of the entire normalized flow graph in Fig. 3.2 respectively as:

$$G_{\varphi_G} = \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix} \diamond \begin{bmatrix} 19 & 8 & 0 \\ 0 & 60 & 0 \\ 8 & 0 & 5 \end{bmatrix},$$

$$G_{\sigma_G} = \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix},$$

$$G_{cerG} = \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.30 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.62 & 0.00 & 0.38 \end{bmatrix} \text{ and}$$

$$G_{covG} = \begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.12 & 0.00 \\ 0.00 & 0.88 & 0.00 \\ 0.30 & 0.00 & 1.00 \end{bmatrix}.$$

By Definitions 3 and 4, an entire flow graph and an entire normalized flow graph can be presented in very compact forms which are convenient to read and analyze. Two matrices joined by $\diamond$ illustrates 3 layers within the flow graph. The first and second layers are presented in the leftmost matrix while the second and third layers are presented in the rightmost matrix.

- When considering G_{φ_G} , we have $\varphi_+(y_1) = \varphi(x_1, y_1) + \varphi(x_2, y_1) + \varphi(x_3, y_1) = 6 + 21 + 0 = 27$ (leftmost matrix), and $\varphi_-(y_1) = \varphi(y_1, z_1) + \varphi(y_1, z_2) + \varphi(y_1, z_3) = 19 + 8 + 0 = 27$ (rightmost matrix).
- Similarly, in G_{σ_G} , $\sigma_+(y_1) = \sigma(x_1, y_1) + \sigma(x_2, y_1) + \sigma(x_3, y_1) = 0.06 + 0.21 + 0 = 0.27$ (leftmost matrix), and $\sigma_-(y_1) = \sigma(y_1, z_1) + \sigma(y_1, z_2) + \sigma(y_1, z_3) = 0.19 + 0.08 + 0 = 0.27$ (rightmost matrix).

Next, in Definition 5, we define the representations of the three coefficients of a combined flow graph in matrix form. The formulas for computing these coefficients directly from a normalized flow graph will be established later.

Definition 5. Let $G = (N, B, \sigma)$ be a combined normalized flow graph, X^1 be a condition attribute (input of G) with k_1 values that are $x_1^1, x_2^1, \dots, x_{k_1}^1$ and X^n be a decision attribute (output of G) with k_n values that are $x_1^n, x_2^n, \dots, x_{k_n}^n$. The strength, certainty and coverage of a combined normalized flow graph G can be represented in matrix form as

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

1. $G_{\sigma_G} \langle x_{j_1}^1, x_{j_n}^n \rangle = [\sigma_{j_1, j_n}^{1, n}]_{k_1 \times k_n}$ where $\sigma_{j_1, j_n}^{1, n} = \sigma \langle x_{j_1}^1, x_{j_n}^n \rangle$,
2. $G_{cer_G} \langle x_{j_1}^1, x_{j_n}^n \rangle = [cer_{j_1, j_n}^{1, n}]_{k_1 \times k_n}$ where $cer_{j_1, j_n}^{1, n} = cer \langle x_{j_1}^1, x_{j_n}^n \rangle$,
3. $G_{cov_G} \langle x_{j_1}^1, x_{j_n}^n \rangle = [cov_{j_1, j_n}^{1, n}]_{k_1 \times k_n}$ where $cov_{j_1, j_n}^{1, n} = cov \langle x_{j_1}^1, x_{j_n}^n \rangle$,

where $1 \leq j_1 \leq k_1, 1 \leq j_n \leq k_n$.

For example, the strength, certainty and coverage of a combined normalized flow graph in Fig. 3.3 are

$$\begin{aligned}
 G_{\sigma_G} \langle x_{j_1}^1, x_{j_3}^3 \rangle &= \begin{bmatrix} 0.04 & 0.06 & 0.00 \\ 0.19 & 0.48 & 0.03 \\ 0.04 & 0.14 & 0.02 \end{bmatrix}, \\
 G_{cer_G} \langle x_{j_1}^1, x_{j_3}^3 \rangle &= \begin{bmatrix} 0.42 & 0.58 & 0.00 \\ 0.27 & 0.69 & 0.04 \\ 0.18 & 0.70 & 0.12 \end{bmatrix} \text{ and} \\
 G_{cov_G} \langle x_{j_1}^1, x_{j_3}^3 \rangle &= \begin{bmatrix} 0.15 & 0.09 & 0.00 \\ 0.71 & 0.71 & 0.54 \\ 0.14 & 0.20 & 0.46 \end{bmatrix}.
 \end{aligned}$$

Next, we define three types of matrices related to flow graphs. We will prove later that these matrices, in fact, can be included in the new formulas to compute the three coefficients of each branch and connection of a flow graph. Definitions 7 and 8 present the diagonal matrix of the throughflow and the inverse throughflow of an attribute set.

Definition 6. For a flow graph $G = (N, B, \sigma)$, let A and B be matrices representing coefficients of G . The matrix multiplication, $A \times B^2$, of matrices $A = [a_{i,k}]_{m \times n}$ and $B = [b_{k,j}]_{n \times p}$ is the matrix $C = [c_{i,j}]_{m \times p}$, where $c_{i,j} = \sum_{k=1}^n a_{i,k} b_{k,j}$.

Definition 7. Let $G = (N, B, \sigma)$ be a normalized flow graph, and X be an attribute with n values: $x_1, x_2, \dots, x_n$. The diagonal matrix of normalized throughflow, denoted by $G_{\sigma(x_i)}$, $1 \leq i \leq n$ is represented by the matrix

$$G_{\sigma(x_i)} = [t_{i,j}]_{n \times n}, \text{ where } t_{i,j} = \begin{cases} \sigma(x_i), & i = j \\ 0 & , i \neq j \end{cases}.$$

Definition 8. Let $G = (N, B, \sigma)$ be a normalized flow graph, X be an attribute with n values: $x_1, x_2, \dots, x_n$. The diagonal matrix of inverse normalized throughflow, denoted by $G_{\frac{1}{\sigma(x_i)}}$, $1 \leq i \leq n$, is represented by the matrix

$$G_{\frac{1}{\sigma(x_i)}} = [f_{i,j}]_{n \times n}, \text{ where } f_{i,j} = \begin{cases} \frac{1}{\sigma(x_i)}, & i = j \\ 0 & , i \neq j \end{cases}.$$

²For simplicity, we use AB instead of $A \times B$ in further discussion.

We compute these matrices from the normalized flow graph in Fig. 3.2. Its normalized throughflows are $\sigma(x_1) = 0.1$, $\sigma(x_2) = 0.7$ and $\sigma(x_3) = 0.2$, and the diagonal matrix of the normalized throughflow and the diagonal matrix of the inverse normalized throughflow are: $G_{\sigma(x_i)} = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.7 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}$ and $G_{\frac{1}{\sigma(x_i)}} = \begin{bmatrix} \frac{1}{0.1} & 0 & 0 \\ 0 & \frac{1}{0.7} & 0 \\ 0 & 0 & \frac{1}{0.2} \end{bmatrix}$.

Remark 5. From the traditional matrix inversion property, we obtain $G_{\frac{1}{\sigma(x_i)}}^{-1} = G_{\sigma(x_i)}$, where $1 \leq i \leq k_i$.

So far we have introduced a new method for representing throughflow and three coefficients of normalized and combined normalized flow graphs in matrix form. Below, we prove a new method for calculating three coefficients of each branch in Theorems 1 and 2 and further extend to connections in Theorems 3 and 4 by using properties of matrices.

When we focus on the strength of each branch of a flow graph in matrix form, then we can apply (3.1) for calculating the strength of each branch solely by using properties of matrices. This is stated precisely in Theorem 1.

Theorem 1. Let $G = (N, B, \varphi)$ be a flow graph, let X be an attribute with n values: $x_1, x_2, \dots, x_n$, and let Y be an attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n$, $1 \leq j \leq m$. Then

$$G_{\sigma_G(x_i, y_j)} = \frac{1}{\varphi(G)} G_{\varphi_G(x_i, y_j)}.$$

Proof. Consider $G_{\sigma_G(x_i, y_j)} = [\sigma_{i,j}]_{n \times m}$, where $\sigma_{i,j} = \sigma(x_i, y_j)$, $1 \leq i \leq n$ and $1 \leq j \leq m$. From (3.1), $G_{\sigma_G(x_i, y_j)} = \left[\frac{\varphi_{i,j}}{\varphi(G)} \right]_{n \times m}$, where $\varphi_{i,j} = \varphi(x_i, y_j)$, for all i, j . By scalar matrix multiplication, $G_{\sigma_G(x_i, y_j)} = \frac{1}{\varphi(G)} [\varphi_{i,j}]_{n \times m}$. By Definition 1, then $G_{\sigma_G(x_i, y_j)} = \frac{1}{\varphi(G)} G_{\varphi_G(x_i, y_j)}$. $\square$

Example 1 Recall the flow graph given in Fig. 3.1. We represent the throughflow of this flow graph as $G_{\varphi_G} = \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix} \diamond \begin{bmatrix} 19 & 8 & 0 \\ 0 & 60 & 0 \\ 8 & 0 & 5 \end{bmatrix}$.

Given the fact that $\varphi(G) = 100$ and by Theorem 1, we can find the throughflow of each branch as:

$$\begin{aligned} G_{\sigma_G} &= \left(\frac{1}{100} \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix} \right) \diamond \left(\frac{1}{100} \begin{bmatrix} 19 & 8 & 0 \\ 0 & 60 & 0 \\ 8 & 0 & 5 \end{bmatrix} \right) \\ &= \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix}. \end{aligned}$$

This result corresponds to all strengths of the normalized flow graph given in Fig. 3.2. The leftmost matrix corresponds to strengths between the first and the second layers

$(\sigma(x_i, y_j), 1 \leq i, j \leq 3)$ while the rightmost matrix corresponds to strengths between the second and the third layers $(\sigma(y_j, z_k), 1 \leq j, k \leq 3)$. Thus, starting from a flow graph G with its throughflow matrix, Theorem 1 gives a formula to compute strength of the corresponding normalized flow graph directly. The following theorem gives formulas for calculating the certainty and coverage from the matrix form of flow graphs.

Theorem 2. Let $G = (N, B, \sigma)$ be a normalized flow graph, X be an attribute with n values: $x_1, x_2, \dots, x_n$, and Y be a attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n, 1 \leq j \leq m$. Then

$$G_{cerG(x_i, y_j)} = G_{\frac{1}{\sigma(x_i)}} G_{\sigma_G(x_i, y_j)} \text{ and } G_{covG(x_i, y_j)} = G_{\sigma_G(x_i, y_j)} G_{\frac{1}{\sigma(y_i)}}.$$

Proof. Consider $G_{\frac{1}{\sigma(x_i)}} G_{\sigma_G(x_i, y_j)}$, then by Definitions 2(1), 6, 8, we obtain

$$\begin{aligned} G_{\frac{1}{\sigma(x_i)}} G_{\sigma_G(x_i, y_j)} &= \begin{bmatrix} \frac{1}{\sigma(x_1)} & 0 & \dots & 0 \\ 0 & \frac{1}{\sigma(x_2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\sigma(x_n)} \end{bmatrix} \begin{bmatrix} \sigma(x_1, y_1) & \sigma(x_1, y_2) & \dots & \sigma(x_1, y_m) \\ \sigma(x_2, y_1) & \sigma(x_2, y_2) & \dots & \sigma(x_2, y_m) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma(x_n, y_1) & \sigma(x_n, y_2) & \dots & \sigma(x_n, y_m) \end{bmatrix} \\ &= \begin{bmatrix} \frac{\sigma(x_1, y_1)}{\sigma(x_1)} & \frac{\sigma(x_1, y_2)}{\sigma(x_1)} & \dots & \frac{\sigma(x_1, y_m)}{\sigma(x_1)} \\ \frac{\sigma(x_2, y_1)}{\sigma(x_2)} & \frac{\sigma(x_2, y_2)}{\sigma(x_2)} & \dots & \frac{\sigma(x_2, y_m)}{\sigma(x_2)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma(x_n, y_1)}{\sigma(x_n)} & \frac{\sigma(x_n, y_2)}{\sigma(x_n)} & \dots & \frac{\sigma(x_n, y_m)}{\sigma(x_n)} \end{bmatrix} \end{aligned}$$

By (5.2) and Definition 2(2), then $G_{cerG(x_i, y_j)} = G_{\frac{1}{\sigma(x_i)}} G_{\sigma_G(x_i, y_j)}$.

Next, consider $G_{\sigma_G(x_i, y_j)} G_{\frac{1}{\sigma(y_i)}}$, then by Definitions 2(2), 6, 8, we obtain

$$\begin{aligned} G_{\sigma_G(x_i, y_j)} G_{\frac{1}{\sigma(y_i)}} &= \begin{bmatrix} \sigma(x_1, y_1) & \sigma(x_1, y_2) & \dots & \sigma(x_1, y_m) \\ \sigma(x_2, y_1) & \sigma(x_2, y_2) & \dots & \sigma(x_2, y_m) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma(x_n, y_1) & \sigma(x_n, y_2) & \dots & \sigma(x_n, y_m) \end{bmatrix} \begin{bmatrix} \frac{1}{\sigma(y_1)} & 0 & \dots & 0 \\ 0 & \frac{1}{\sigma(y_2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\sigma(y_m)} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\sigma(x_1, y_1)}{\sigma(y_1)} & \frac{\sigma(x_1, y_2)}{\sigma(y_2)} & \dots & \frac{\sigma(x_1, y_m)}{\sigma(y_m)} \\ \frac{\sigma(x_2, y_1)}{\sigma(y_1)} & \frac{\sigma(x_2, y_2)}{\sigma(y_2)} & \dots & \frac{\sigma(x_2, y_m)}{\sigma(y_m)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma(x_n, y_1)}{\sigma(y_1)} & \frac{\sigma(x_n, y_2)}{\sigma(y_2)} & \dots & \frac{\sigma(x_n, y_m)}{\sigma(y_m)} \end{bmatrix} \end{aligned}$$

By (5.3) then $G_{covG(x_i, y_j)} = G_{\sigma_G(x_i, y_j)} G_{\frac{1}{\sigma(y_i)}}$. □

We can generalize (3.13), (3.14) and (3.15) for calculating the three coefficients by matrix properties as we presented in the earlier theorems for connections. Theorem 3 is the case of a flow graph with two condition attributes and one decision attribute whereas Theorem 4 is a general form of a flow graph with $n - 1$ condition attributes and one decision attribute.

Theorem 3. Let $G = (N, B, \varphi)$ be a normalized flow graph, X^1 be a condition attribute with k_1 values: $x_1^1, x_2^1, \dots, x_{k_1}^1$, X^2 be a condition attribute with k_2 values: $x_1^2, x_2^2, \dots, x_{k_2}^2$, and X^3 be a decision attribute with k_3 values: $x_1^3, x_2^3, \dots, x_{k_3}^3$ such that $(x_{j_i}^1, x_{j_i+1}^2) \in B$,

$1 \leq i \leq 2, 1 \leq j_i \leq k_i$. For any connection $\langle x_{j_1}^1, x_{j_3}^3 \rangle, 1 \leq j_1 \leq k_1, 1 \leq j_3 \leq k_3$, in a normalized flow graph G , the certainty, coverage and strength of all connections can be calculated as follows:

$$\begin{aligned} (1) \quad G_{cerG\langle x_{j_1}^1, x_{j_3}^3 \rangle} &= G_{cer(x_{j_1}^1, x_{j_2}^2)} G_{cer(x_{j_2}^2, x_{j_3}^3)}, \\ (2) \quad G_{covG\langle x_{j_1}^1, x_{j_3}^3 \rangle} &= G_{cov(x_{j_1}^1, x_{j_2}^2)} G_{cov(x_{j_2}^2, x_{j_3}^3)}, \\ (3)(a) \quad G_{\sigma\langle x_{j_1}^1, x_{j_3}^3 \rangle} &= G_{\sigma(x_{j_1}^1)} G_{cerG\langle x_{j_1}^1, x_{j_3}^3 \rangle}, \\ (3)(b) \quad G_{\sigma\langle x_{j_1}^1, x_{j_3}^3 \rangle} &= G_{covG\langle x_{j_1}^1, x_{j_3}^3 \rangle} G_{\sigma(x_{j_3}^3)}. \end{aligned}$$

Proof. (1) Definitions 2(2), 5(2) and 6, respectively, give

$$\begin{aligned} &G_{cer(x_{j_1}^1, x_{j_2}^2)} G_{cer(x_{j_2}^2, x_{j_3}^3)} \\ &= \begin{bmatrix} cer(x_1^1, x_1^2) & cer(x_1^1, x_2^2) & \dots & cer(x_1^1, x_{k_2}^2) \\ cer(x_2^1, x_1^2) & cer(x_2^1, x_2^2) & \dots & cer(x_2^1, x_{k_2}^2) \\ \vdots & \vdots & \ddots & \vdots \\ cer(x_{k_1}^1, x_1^2) & cer(x_{k_1}^1, x_2^2) & \dots & cer(x_{k_1}^1, x_{k_2}^2) \end{bmatrix} \begin{bmatrix} cer(x_2^2, x_1^3) & cer(x_2^2, x_2^3) & \dots & cer(x_2^2, x_{k_3}^3) \\ cer(x_2^2, x_1^3) & cer(x_2^2, x_2^3) & \dots & cer(x_2^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ cer(x_{k_2}^2, x_1^3) & cer(x_{k_2}^2, x_2^3) & \dots & cer(x_{k_2}^2, x_{k_3}^3) \end{bmatrix} \\ &= \begin{bmatrix} \sum_{j_2=1}^{k_2} cer(x_1^1, x_{j_2}^2) cer(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} cer(x_1^1, x_{j_2}^2) cer(x_{j_2}^2, x_2^3) & \dots & \sum_{j_2=1}^{k_2} cer(x_1^1, x_{j_2}^2) cer(x_{j_2}^2, x_{k_3}^3) \\ \sum_{j_2=1}^{k_2} cer(x_2^1, x_{j_2}^2) cer(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} cer(x_2^1, x_{j_2}^2) cer(x_{j_2}^2, x_2^3) & \dots & \sum_{j_2=1}^{k_2} cer(x_2^1, x_{j_2}^2) cer(x_{j_2}^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j_2=1}^{k_2} cer(x_{k_1}^1, x_{j_2}^2) cer(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} cer(x_{k_1}^1, x_{j_2}^2) cer(x_{j_2}^2, x_2^3) & \dots & \sum_{j_2=1}^{k_2} cer(x_{k_1}^1, x_{j_2}^2) cer(x_{j_2}^2, x_{k_3}^3) \end{bmatrix} \\ &= \begin{bmatrix} cer\langle x_1^1, x_1^3 \rangle & cer\langle x_1^1, x_2^3 \rangle & \dots & cer\langle x_1^1, x_{k_3}^3 \rangle \\ cer\langle x_2^1, x_1^3 \rangle & cer\langle x_2^1, x_2^3 \rangle & \dots & cer\langle x_2^1, x_{k_3}^3 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ cer\langle x_{k_1}^1, x_1^3 \rangle & cer\langle x_{k_1}^1, x_2^3 \rangle & \dots & cer\langle x_{k_1}^1, x_{k_3}^3 \rangle \end{bmatrix} \\ &= G_{cerG\langle x_{j_1}^1, x_{j_3}^3 \rangle}. \end{aligned}$$

(2) Definitions 2(3), 5(3) and 6, respectively, give

$$\begin{aligned} &G_{cov(x_{j_1}^1, x_{j_2}^2)} G_{cov(x_{j_2}^2, x_{j_3}^3)} \\ &= \begin{bmatrix} cov(x_1^1, x_1^2) & cov(x_1^1, x_2^2) & \dots & cov(x_1^1, x_{k_2}^2) \\ cov(x_2^1, x_1^2) & cov(x_2^1, x_2^2) & \dots & cov(x_2^1, x_{k_2}^2) \\ \vdots & \vdots & \ddots & \vdots \\ cov(x_{k_1}^1, x_1^2) & cov(x_{k_1}^1, x_2^2) & \dots & cov(x_{k_1}^1, x_{k_2}^2) \end{bmatrix} \begin{bmatrix} cov(x_2^2, x_1^3) & cov(x_2^2, x_2^3) & \dots & cov(x_2^2, x_{k_3}^3) \\ cov(x_2^2, x_1^3) & cov(x_2^2, x_2^3) & \dots & cov(x_2^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ cov(x_{k_2}^2, x_1^3) & cov(x_{k_2}^2, x_2^3) & \dots & cov(x_{k_2}^2, x_{k_3}^3) \end{bmatrix} \\ &= \begin{bmatrix} \sum_{j_2=1}^{k_2} cov(x_1^1, x_{j_2}^2) cov(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} cov(x_1^1, x_{j_2}^2) cov(x_{j_2}^2, x_2^3) & \dots & \sum_{j_2=1}^{k_2} cov(x_1^1, x_{j_2}^2) cov(x_{j_2}^2, x_{k_3}^3) \\ \sum_{j_2=1}^{k_2} cov(x_2^1, x_{j_2}^2) cov(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} cov(x_2^1, x_{j_2}^2) cov(x_{j_2}^2, x_2^3) & \dots & \sum_{j_2=1}^{k_2} cov(x_2^1, x_{j_2}^2) cov(x_{j_2}^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j_2=1}^{k_2} cov(x_{k_1}^1, x_{j_2}^2) cov(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} cov(x_{k_1}^1, x_{j_2}^2) cov(x_{j_2}^2, x_2^3) & \dots & \sum_{j_2=1}^{k_2} cov(x_{k_1}^1, x_{j_2}^2) cov(x_{j_2}^2, x_{k_3}^3) \end{bmatrix} \\ &= \begin{bmatrix} cov\langle x_1^1, x_1^3 \rangle & cov\langle x_1^1, x_2^3 \rangle & \dots & cov\langle x_1^1, x_{k_3}^3 \rangle \\ cov\langle x_2^1, x_1^3 \rangle & cov\langle x_2^1, x_2^3 \rangle & \dots & cov\langle x_2^1, x_{k_3}^3 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ cov\langle x_{k_1}^1, x_1^3 \rangle & cov\langle x_{k_1}^1, x_2^3 \rangle & \dots & cov\langle x_{k_1}^1, x_{k_3}^3 \rangle \end{bmatrix} \\ &= G_{covG\langle x_{j_1}^1, x_{j_3}^3 \rangle}. \end{aligned}$$

(3)(a) Definitions 5(2), 6, and 7, respectively give

$$G_{\sigma(x_{j_1}^1)} G_{cer_G \langle x_{j_1}^1, x_{j_3}^3 \rangle} = \begin{bmatrix} \sigma(x_1^1) cer \langle x_1^1, x_1^3 \rangle & \sigma(x_1^1) cer \langle x_1^1, x_2^3 \rangle & \dots & \sigma(x_1^1) cer \langle x_1^1, x_{k_3}^3 \rangle \\ \sigma(x_2^1) cer \langle x_2^1, x_1^3 \rangle & \sigma(x_2^1) cer \langle x_2^1, x_2^3 \rangle & \dots & \sigma(x_2^1) cer \langle x_2^1, x_{k_3}^3 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \sigma(x_{k_1}^1) cer \langle x_{k_1}^1, x_1^3 \rangle & \sigma(x_{k_1}^1) cer \langle x_{k_1}^1, x_2^3 \rangle & \dots & \sigma(x_{k_1}^1) cer \langle x_{k_1}^1, x_{k_3}^3 \rangle \end{bmatrix}.$$

By (3.15) and Definition 5(1), then $G_{\sigma \langle x_{j_1}^1 \dots x_{j_n}^n \rangle} = G_{\sigma(x_{j_1}^1)} G_{cer_G \langle x_{j_1}^1 \dots x_{j_n}^n \rangle}$.

(3)(b) Definition 5(3), 6 and 7, respectively give

$$G_{cov_G \langle x_{j_1}^1, x_{j_n}^n \rangle} G_{\sigma(x_{j_n}^n)} = \begin{bmatrix} cov \langle x_1^1, x_1^3 \rangle \sigma(x_1^3) & cov \langle x_1^1, x_2^3 \rangle \sigma(x_2^3) & \dots & cov \langle x_1^1, x_{k_3}^3 \rangle \sigma(x_{k_3}^3) \\ cov \langle x_2^1, x_1^3 \rangle \sigma(x_1^3) & cov \langle x_2^1, x_2^3 \rangle \sigma(x_2^3) & \dots & cov \langle x_2^1, x_{k_3}^3 \rangle \sigma(x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ cov \langle x_{k_1}^1, x_1^3 \rangle \sigma(x_1^3) & cov \langle x_{k_1}^1, x_2^3 \rangle \sigma(x_2^3) & \dots & cov \langle x_{k_1}^1, x_{k_3}^3 \rangle \sigma(x_{k_3}^3) \end{bmatrix}.$$

By (3.3) and Definition 5(1), then $G_{\sigma \langle x_{j_1}^1 \dots x_{j_n}^n \rangle} = G_{cov_G \langle x_{j_1}^1 \dots x_{j_n}^n \rangle} G_{\sigma(x_{j_n}^n)}$. $\square$

Example 2 Recall that $G_{\sigma_G} = \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix}$ from Example 1.

Since $\sigma(x_1) = 0.1$, $\sigma(x_2) = 0.7$ and $\sigma(x_3) = 0.2$, by Definition 8, then $G_{\frac{1}{\sigma(x_1)}} = \begin{bmatrix} \frac{1}{0.1} & 0 & 0 \\ 0 & \frac{1}{0.7} & 0 \\ 0 & 0 & \frac{1}{0.2} \end{bmatrix}$

and similarly, $G_{\frac{1}{\sigma(x_2)}} = \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.60} & 0 \\ 0 & 0 & \frac{1}{0.13} \end{bmatrix}$. By the formulas given in Theorem 2, we obtain

$$\begin{aligned} G_{cer_G} &= \begin{bmatrix} \frac{1}{0.1} & 0 & 0 \\ 0 & \frac{1}{0.7} & 0 \\ 0 & 0 & \frac{1}{0.2} \end{bmatrix} \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.60} & 0 \\ 0 & 0 & \frac{1}{0.13} \end{bmatrix} \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix} \\ &= \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.30 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.62 & 0.00 & 0.38 \end{bmatrix}. \end{aligned}$$

Consequently, we can compute

$$\begin{aligned} G_{cov_G} &= \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.60} & 0 \\ 0 & 0 & \frac{1}{0.13} \end{bmatrix} \diamond \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix} \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.60} & 0 \\ 0 & 0 & \frac{1}{0.05} \end{bmatrix} \\ &= \begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.12 & 0.00 \\ 0.00 & 0.88 & 0.00 \\ 0.30 & 0.00 & 1.00 \end{bmatrix}. \end{aligned}$$

This also corresponds to the certainty and coverage of the flow graph given in Fig. 3.2.

Next, for a combined normalized flow graph we use Theorem 3 for calculating certainty, coverage and strength of the connection, respectively as:

$$\begin{aligned} G_{cer_G \langle x, z \rangle} &= \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix} \begin{bmatrix} 0.70 & 0.30 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.62 & 0.00 & 0.38 \end{bmatrix} = \begin{bmatrix} 0.42 & 0.58 & 0.00 \\ 0.27 & 0.69 & 0.04 \\ 0.19 & 0.70 & 0.11 \end{bmatrix}, \\ G_{cov_G \langle x, z \rangle} &= \begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix} \begin{bmatrix} 0.70 & 0.12 & 0.00 \\ 0.00 & 0.88 & 0.00 \\ 0.30 & 0.00 & 1.00 \end{bmatrix} = \begin{bmatrix} 0.15 & 0.09 & 0.00 \\ 0.71 & 0.71 & 0.54 \\ 0.14 & 0.20 & 0.46 \end{bmatrix} \text{ and} \end{aligned}$$

$$G_{\sigma_G \langle x, z \rangle} = \begin{bmatrix} 0.10 & 0.00 & 0.00 \\ 0.00 & 0.70 & 0.00 \\ 0.00 & 0.00 & 0.20 \end{bmatrix} \begin{bmatrix} 0.42 & 0.58 & 0.00 \\ 0.27 & 0.69 & 0.10 \\ 0.18 & 0.70 & 0.12 \end{bmatrix} = \begin{bmatrix} 0.04 & 0.06 & 0.00 \\ 0.19 & 0.48 & 0.03 \\ 0.04 & 0.14 & 0.02 \end{bmatrix}.$$

This matrix form can simplify computation as described below. Butz *et al.* showed that to compute certainty and coverage traditional flow graph algorithms have $O(3n^3)$ time complexity, where n is the number of attribute values for each layer [2]. In Theorem 3, we can employ well-known matrix multiplication algorithms to compute these coefficients. The time complexities of the Simple algorithm, Strassen algorithm and Coppersmith-Winograd algorithm of matrix multiplication are $O(n^3)$, $O(n^{2.807})$ and $O(n^{2.376})$, respectively [9]. Below, we come to a general formula to compute coefficients of connections.

Theorem 4. Let $G = (N, B, \sigma)$ be a normalized flow graph, X^i be a condition attribute with k_i values: $x_1^i, x_2^i, \dots, x_{k_i}^i$, $1 \leq i \leq n-1$, and X^n be a decision attribute with k_n values: $x_1^n, x_2^n, \dots, x_{k_n}^n$ such that $(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \in B$, $1 \leq i \leq n-1$, $1 \leq j_i \leq k_i$. For any connection $\langle x_{j_1}^1, x_{j_n}^n \rangle$, $1 \leq j_1 \leq k_1$, $1 \leq j_n \leq k_n$. The certainty, coverage and strength of the connection are

$$\begin{aligned} (1) \quad G_{cerG \langle x_{j_1}^1, x_{j_n}^n \rangle} &= \prod_{i=1}^{n-1} G_{cer(x_{j_i}^i, x_{j_{i+1}}^{i+1})}, \\ (2) \quad G_{covG \langle x_{j_1}^1, x_{j_n}^n \rangle} &= \prod_{i=1}^{n-1} G_{cov(x_{j_i}^i, x_{j_{i+1}}^{i+1})}, \\ (3)(a) \quad G_{\sigma \langle x_{j_1}^1, x_{j_n}^n \rangle} &= G_{\sigma(x_{j_1}^1)} G_{cerG \langle x_{j_1}^1, x_{j_n}^n \rangle}, \\ (3)(b) \quad G_{\sigma \langle x_{j_1}^1, x_{j_n}^n \rangle} &= G_{covG \langle x_{j_1}^1, x_{j_n}^n \rangle} G_{\sigma(x_{j_n}^n)}. \end{aligned}$$

Proof. We prove this by mathematical induction for $n \geq 2$. The case $n = 2$ is shown in Theorem 3. For $n > 2$, it can be proven in a straightforward way according to

- (1) Definitions 2(2), 5(2), 6 and Theorem 3(1),
- (2) Definitions 2(3), 5(3), 6 and Theorem 3(2),
- (3)(a) Definitions 5(1), (2), 6, (6) and Theorem 3(3)(a),
- (3)(b) Definitions 5(1), (3), 6, (6) and Theorem 3(3)(b). □

Butz *et al.* showed that traditional algorithms for computing these formulas have $O(ln^l)$ time complexity, where l is a number layers of a flow graph and n is a number of attribute values for each layer [2]. In Theorem 4, the flow graph in matrix form improves this complexity: for the Gauss-Jordan elimination algorithm, Strassen algorithm and Coppersmith-Winograd algorithm of matrix multiplication to $O((l-2)n^3)$, $O((l-2)n^{2.807})$ and $O((l-2)n^{2.376})$, respectively [9]. While traditional algorithms are exponential in l , an algorithm using our matrix representation is polynomial in l .

In accordance with the previous definitions and theorems, if any one connected layer lacks coefficients, then we can calculate these coefficients by using Theorem 5.

Theorem 5. Let $G = (N, B, \sigma)$ be a normalized flow graph, X^i be a condition attribute with k_i values: $x_1^i, x_2^i, \dots, x_{k_i}^i$, $1 \leq i \leq n-1$, and X^n be a decision attribute with k_n values: $x_1^n, x_2^n, \dots, x_{k_n}^n$ such that $(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \in B$, $1 \leq i \leq n-1$, $1 \leq j_i \leq k_i$. For any branch $(x_{j_l}^l, x_{j_{l+1}}^{l+1})$, $1 \leq l \leq n-1$, $1 \leq j_l \leq k_l$. The certainty, coverage and strength of branches are

$$(1) G_{\text{cer}_G(x_{j_l}^l, x_{j_{l+1}}^{l+1})} = \left(\prod_{i=1}^{l-1} G_{\text{cer}(x_{j_i}^i, x_{j_{i+1}}^{i+1})} \right)^{-1} G_{\text{cer}_G < x_1^1 \dots x_{j_n}^n >} \left(\prod_{i=l+1}^{n-1} G_{\text{cer}(x_{j_i}^i, x_{j_{i+1}}^{i+1})} \right)^{-1},$$

where $\prod_{i=1}^{l-1} G_{\text{cer}(x_{j_i}^i, x_{j_{i+1}}^{i+1})}$ and $\prod_{i=l+1}^{n-1} G_{\text{cer}(x_{j_i}^i, x_{j_{i+1}}^{i+1})}$ are invertible matrices.

$$(2) G_{\text{cov}_G(x_{j_l}^l, x_{j_{l+1}}^{l+1})} = \left(\prod_{i=1}^{l-1} G_{\text{cov}(x_{j_i}^i, x_{j_{i+1}}^{i+1})} \right)^{-1} G_{\text{cov}_G < x_1^1 \dots x_{j_n}^n >} \left(\prod_{i=l+1}^{n-1} G_{\text{cov}(x_{j_i}^i, x_{j_{i+1}}^{i+1})} \right)^{-1},$$

where $\prod_{i=1}^{l-1} G_{\text{cov}(x_{j_i}^i, x_{j_{i+1}}^{i+1})}$ and $\prod_{i=l+1}^{n-1} G_{\text{cov}(x_{j_i}^i, x_{j_{i+1}}^{i+1})}$ are invertible matrices.

$$(3)(a) G_{\sigma(x_{j_l}^l, x_{j_{l+1}}^{l+1})} = G_{\sigma(x_{j_l}^l)} G_{\text{cer}_G(x_{j_l}^l, x_{j_{l+1}}^{l+1})},$$

$$(3)(b) G_{\sigma(x_{j_l}^l, x_{j_{l+1}}^{l+1})} = G_{\text{cov}_G(x_{j_l}^l, x_{j_{l+1}}^{l+1})} G_{\sigma(x_{j_{l+1}}^{l+1})},$$

Proof. This can be easily verified by using a multiplicative property of the inverse matrix³ and using

- (1) Theorem 4(1),
- (2) Theorem 4(2),
- (3)(a) Theorem 2 and Remark 5,
- (3)(b) Theorem 2 and Remark 5.

□

In this theorem, we can utilize the Gauss-Jordan elimination, Strassen algorithm or Coppersmith-Winograd algorithm to compute inverse matrix which require time complexities $O(n^3)$, $O(n^{2.807})$ and $O(n^{2.376})$, respectively [9]. Theorem 5 can alleviate missing coefficient problem as demonstrated in this example.

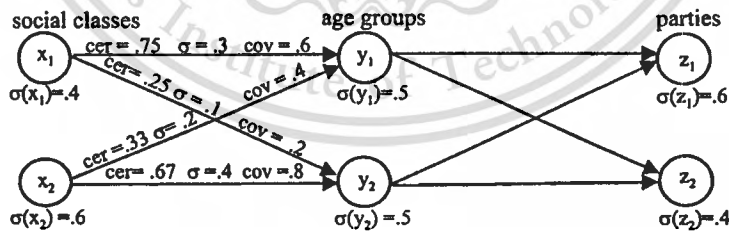


Figure 4.1 A normalized flow graph with missing coefficients.

Example 3 Suppose we are given two disjoint age groups of voters: y_1 (old) and y_2 (middle aged). They belong to two social classes x_1 (high) and x_2 (middle). The voters can vote for two political parties z_1 and z_2 . Assume that voters are grouped according to social classes, age groups and political parties as shown in Fig. 4.1. In this case, this flow graph has some missing coefficients within the layer of age groups and parties. If we also have additional

³Let A , B and C be matrices and $AB = C$. If A is an invertible matrix, then $B = A^{-1}C$.

combined normalized flow graph of Fig. 4.1 as depicted in Fig. 4.2, then we can compute the missing coefficients by using Theorem 5.

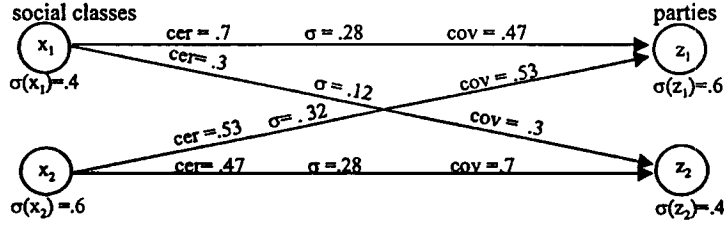


Figure 4.2 Combined normalized flow graph of Fig. 4.1.

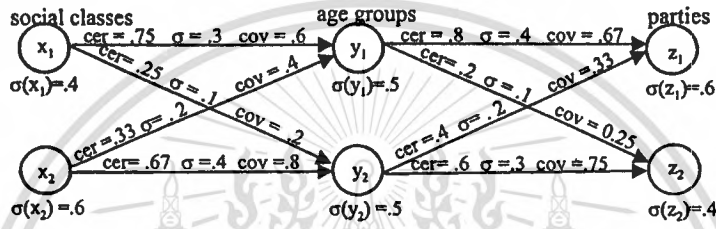


Figure 4.3 A normalized flow graph of Fig. 4.1 with filled coefficients.

From Fig. 4.1, the certainty, coverage and strength of the branches (y_j, z_k) , $1 \leq j, k \leq 2$, are missing. Then by using Theorem 5, we have that:

$$G_{cer_G(y_j, z_k)} = \begin{bmatrix} 0.75 & 0.25 \\ 0.33 & 0.67 \end{bmatrix}^{-1} \begin{bmatrix} 0.70 & 0.30 \\ 0.53 & 0.47 \end{bmatrix} = \begin{bmatrix} 0.80 & 0.20 \\ 0.40 & 0.60 \end{bmatrix},$$

$$G_{cov_G(y_j, z_k)} = \begin{bmatrix} 0.60 & 0.20 \\ 0.40 & 0.80 \end{bmatrix}^{-1} \begin{bmatrix} 0.47 & 0.30 \\ 0.53 & 0.70 \end{bmatrix} = \begin{bmatrix} 0.67 & 0.25 \\ 0.33 & 0.75 \end{bmatrix} \text{ and}$$

$$G_{\sigma_G(y_j, z_k)} = \begin{bmatrix} 0.50 & 0.00 \\ 0.00 & 0.50 \end{bmatrix} \begin{bmatrix} 0.80 & 0.20 \\ 0.40 & 0.60 \end{bmatrix} = \begin{bmatrix} 0.40 & 0.10 \\ 0.20 & 0.30 \end{bmatrix}.$$

Thus, we are able to fill in the missing coefficients of Fig 4.1 as shown in Fig 4.3.

4.2 Flow Graph Composition

In knowledge discovery disciplines, Pawlak's flow graph is a mathematical model to present relationships between attribute sets. Each layer denotes a singleton set of attribute values and each branch describes flow distribution between nodes. In the earlier sections, all data sets which were used to construct Pawlak's flow graph come from a single data set format. However, in some situations, data sets may be from heterogeneous data sources⁴. In data integration [15], we desire to analyze and extract some useful patterns of flow graphs

⁴Heterogeneous data sources in this paper is a special case where each data source stores the same condition and decision attributes.

stored in this format. To do this, we require special treatment to integrate flow graphs from distinct sources to form a single flow graph and formulas for calculating their certainty, coverage and strength coefficients.

Theorem 6. Let $N^i, i = 1, 2, \dots, n$, be data sets with n^i elements and let the certainty, coverage and strength of data sets N^i be denoted by $cer(x^i, y^i)$, $cov(x^i, y^i)$ and $\sigma(x^i, y^i)$. Then the certainty, coverage and strength of a composed normalized flow graph $N^1 \cup N^2 \cup \dots \cup N^n$, denoted by $cer_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y)$, $cov_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y)$ and $\sigma_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y)$ are respectively computed as

$$\begin{aligned} (1) \quad cer_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y) &= \frac{\sum_{i=1}^n n^i \sigma(x^i) cer(x^i, y^i)}{\sum_{i=1}^n n^i \sigma(x^i)}, \\ (2) \quad cov_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y) &= \frac{\sum_{i=1}^n n^i \sigma(y^i) cov(x^i, y^i)}{\sum_{i=1}^n n^i \sigma(y^i)}, \\ (3) \quad \sigma_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y) &= \frac{\sum_{i=1}^n n^i \sigma(x^i, y^i)}{\sum_{i=1}^n n^i}. \end{aligned}$$

Proof. (1) Consider $\frac{\sum_{i=1}^n n^i \sigma(x^i) cer(x^i, y^i)}{\sum_{i=1}^n n^i \sigma(x^i)}$, the definition of $\sigma(x)$, (3.1), (5.2) and $\varphi(G_{N^i}) = n^i$, then $\frac{\sum_{i=1}^n n^i \sigma(x^i) cer(x^i, y^i)}{\sum_{i=1}^n n^i \sigma(x^i)} = \frac{\sum_{i=1}^n \varphi(x^i, y^i)}{\sum_{i=1}^n \varphi(x^i)} = cer_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y)$.

(2) Consider $\frac{\sum_{i=1}^n n^i \sigma(y^i) cov(x^i, y^i)}{\sum_{i=1}^n n^i \sigma(y^i)}$, the definition of $\sigma(y)$, (3.1), (5.3) and $\varphi(G_{N^i}) = n^i$, then $\frac{\sum_{i=1}^n n^i \sigma(y^i) cov(x^i, y^i)}{\sum_{i=1}^n n^i \sigma(y^i)} = \frac{\sum_{i=1}^n \varphi(x^i, y^i)}{\sum_{i=1}^n \varphi(y^i)} = cov_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y)$.

(3) Consider $\frac{\sum_{i=1}^n n^i \sigma(x^i, y^i)}{\sum_{i=1}^n n^i}$, (3.1) and $\varphi(G^i) = n^i$, then $\frac{\sum_{i=1}^n n^i \sigma(x^i, y^i)}{\sum_{i=1}^n n^i} = \frac{\sum_{i=1}^n \varphi(x^i, y^i)}{\sum_{i=1}^n \varphi(G^i)} = \sigma_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y)$. $\square$

Example 4 Suppose we have three disjoint age groups of voters x_1 (old), x_2 (middle aged) and x_3 (young). They can vote for three political parties z_1 , z_2 and z_3 . Assume that voters are grouped according to social classes, age groups and political parties as shown in Fig(s). 4.4 - 4.5 where Fig. 4.4 is a normalized flow graph of voting initial data, N^1 , for 40 voters and Fig. 4.5 is a normalized flow graph of voting initial data, N^2 , of 60 voters. Then we can compute three coefficients of the composed normalized flow graph $N^1 \cup N^2$ by using Theorem 6 for 100 voters as shown in Fig. 4.6.

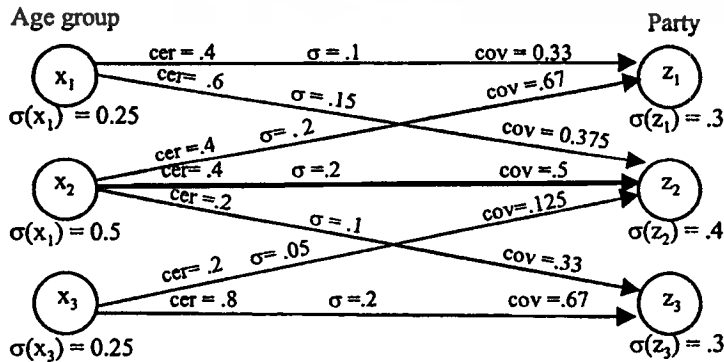


Figure 4.4 Flow graph of voting data, N^1 , with 40 voters.

We can use our new formulas to compute the certainty, coverage and strength of this new flow graph $N^1 \cup N^2$ accurately. Below, a sample computation of bold branches is described,

$$\begin{aligned}
 cer_{N^1 \cup N^2}(x_2, z_2) &= \frac{n^1 \sigma(x_2^1) cer(x_2^1, z_2^1) + n^2 \sigma(x_2^2) cer(x_2^2, z_2^2)}{n^1 \sigma(x_2^1) + n^2 \sigma(x_2^2)} \\
 &= \frac{(40)(0.5)(0.4) + (60)(0.25)(0.8)}{(40)(0.5) + (60)(0.25)} = 0.57, \\
 cov_{N^1 \cup N^2}(x_2, z_2) &= \frac{n^1 \sigma(x_2^1) cov(x_2^1, z_2^1) + n^2 \sigma(x_2^2) cov(x_2^2, z_2^2)}{n^1 \sigma(x_2^1) + n^2 \sigma(x_2^2)} \\
 &= \frac{(40)(0.4)(0.5) + (60)(0.4)(0.5)}{(40)(0.4) + (60)(0.4)} = 0.5, \\
 \sigma_{N^1 \cup N^2}(x_2, z_2) &= \frac{n^1 \sigma(x_2^1, z_2^1) + n^2 \sigma(x_2^2, z_2^2)}{n^1 + n^2} \\
 &= \frac{(40)(0.2) + (60)(0.2)}{40 + 60} = 0.2.
 \end{aligned}$$

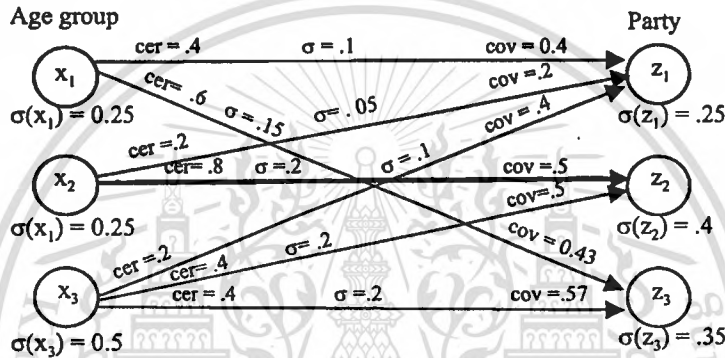


Figure 4.5 Flow graph of voting data, N^2 , with 60 voters.

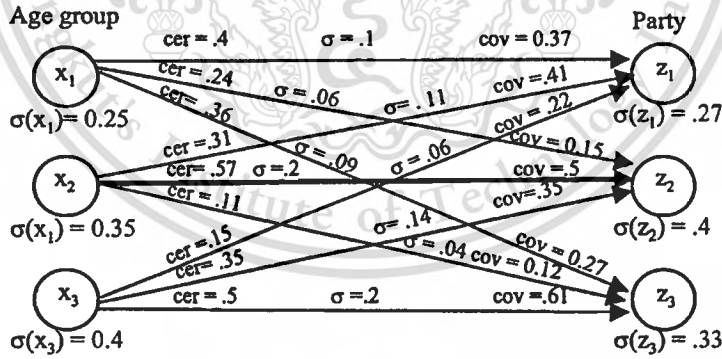


Figure 4.6 Flow graph of voting data of $N^1 \cup N^2$.

We can distil new patterns from this flow graph e.g., 57% of the middle aged group votes for party z_2 , 50% of party z_2 votes come from middle aged voters and 20% of voters are middle aged and vote for party z_2 .

4.3 Experimental Results

To demonstrate that the idea and formulas presented earlier are applicable, we conduct an experiment with real-world data. It uses the scores of Olympiad mathematics students from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN), Thailand. The data set contains 100 records (N). We consider two distributed databases according to the genders of students: N^1 is 20 records for females and N^2 is 80 records for males as tabulated in Tables 4.1(a), (b).

In this experiment, we study the strength, certainty and coverage for condition attribute sets which are the *score for Geometry* (G), *score for number theory* (N), *score of algebra* (A), *score for combinatorics* (C) and *score for inequality* (I) and a decision attribute, the *total score* (T). Possible values of condition and decision attributes are transformed to score being high (H) and low (L) for the flow graph construction.

Table 4.1. (a) A POSN decision table of N^1

	G	N	A	C	I	T
f_1	5	1	25	20	7	58
f_2	8	2	40	4	5	59
f_3	10	0	39	9	3	61
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
f_{20}	26	5	36	34	22	123

(b) A POSN decision table of N^2

	G	N	A	C	I	T
m_1	6	6	26	16	4	58
m_2	11	13	28	11	5	58
m_3	9	7	20	16	6	58
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{80}	31	46	44	30	45	196

From the data in Tables 4.1(a), (b), we can represent the corresponding flow graphs and their throughflows as depicted in Fig. 4.7 and Fig. 4.8, respectively. In these figures there are six layers. The first five layers are condition attributes whereas the last layer is a decision attribute.

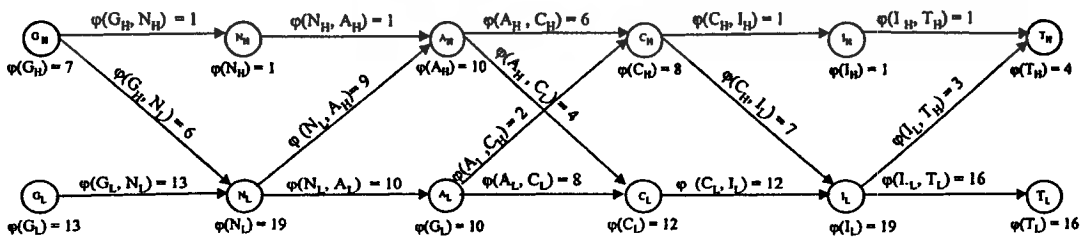


Figure 4.7 Flow graph of initial data for N^1 .

Below, we represent these flow graphs in matrix form. There are five matrices concatenated in $G_{\varphi N^1}$ and $G_{\varphi N^2}$ for representing throughflows of the corresponding flow graphs N^1 and N^2 , respectively. The leftmost matrices represent sets of connected branches

(G_i, N_j) whereas the second, the third, the fourth and the rightmost matrices represent sets of connected branches (N_i, A_j) , (A_i, C_j) , (C_i, I_j) and (I_i, T_j) , for $i, j \in \{H, L\}$, respectively.

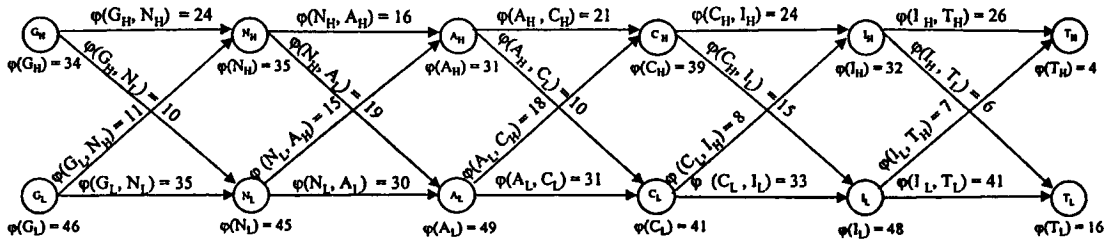


Figure 4.8 Flow graph of initial data for N^2 .

$$G_{\varphi_{N^1}} = \begin{bmatrix} 1 & 6 \\ 0 & 13 \end{bmatrix} \diamond \begin{bmatrix} 1 & 0 \\ 9 & 10 \end{bmatrix} \diamond \begin{bmatrix} 6 & 4 \\ 2 & 8 \end{bmatrix} \diamond \begin{bmatrix} 1 & 7 \\ 0 & 12 \end{bmatrix} \diamond \begin{bmatrix} 1 & 0 \\ 3 & 16 \end{bmatrix} \text{ and}$$

$$G_{\varphi_{N^2}} = \begin{bmatrix} 24 & 10 \\ 11 & 35 \end{bmatrix} \diamond \begin{bmatrix} 16 & 19 \\ 15 & 30 \end{bmatrix} \diamond \begin{bmatrix} 21 & 10 \\ 18 & 31 \end{bmatrix} \diamond \begin{bmatrix} 24 & 15 \\ 8 & 33 \end{bmatrix} \diamond \begin{bmatrix} 26 & 6 \\ 7 & 41 \end{bmatrix}.$$

Similarly, we can construct the corresponding normalized flow graphs of N^1 and N^2 . Moreover from $G_{\varphi_{N^1}}$ and $G_{\varphi_{N^2}}$, we use Theorems 1 and 2 to derive the matrix forms of the strength, certainty and coverage respectively as:

$$\begin{aligned} G_{\sigma_{N^1}} &= \begin{bmatrix} 0.05 & 0.30 \\ 0.00 & 0.65 \end{bmatrix} \diamond \begin{bmatrix} 0.05 & 0.00 \\ 0.45 & 0.50 \end{bmatrix} \diamond \begin{bmatrix} 0.30 & 0.20 \\ 0.10 & 0.40 \end{bmatrix} \diamond \begin{bmatrix} 0.05 & 0.35 \\ 0.00 & 0.60 \end{bmatrix} \diamond \begin{bmatrix} 0.05 & 0.00 \\ 0.15 & 0.80 \end{bmatrix}, \\ G_{\sigma_{N^2}} &= \begin{bmatrix} 0.30 & 0.13 \\ 0.14 & 0.44 \end{bmatrix} \diamond \begin{bmatrix} 0.20 & 0.24 \\ 0.19 & 0.38 \end{bmatrix} \diamond \begin{bmatrix} 0.26 & 0.13 \\ 0.23 & 0.39 \end{bmatrix} \diamond \begin{bmatrix} 0.30 & 0.19 \\ 0.10 & 0.41 \end{bmatrix} \diamond \begin{bmatrix} 0.33 & 0.08 \\ 0.09 & 0.51 \end{bmatrix}, \\ G_{cer_{N^1}} &= \begin{bmatrix} 0.14 & 0.86 \\ 0.00 & 1.00 \end{bmatrix} \diamond \begin{bmatrix} 1.00 & 0.00 \\ 0.47 & 0.53 \end{bmatrix} \diamond \begin{bmatrix} 0.60 & 0.40 \\ 0.20 & 0.80 \end{bmatrix} \diamond \begin{bmatrix} 0.13 & 0.87 \\ 0.00 & 1.00 \end{bmatrix} \diamond \begin{bmatrix} 1.00 & 0.00 \\ 0.16 & 0.84 \end{bmatrix}, \\ G_{cer_{N^2}} &= \begin{bmatrix} 0.70 & 0.30 \\ 0.24 & 0.76 \end{bmatrix} \diamond \begin{bmatrix} 0.46 & 0.54 \\ 0.33 & 0.67 \end{bmatrix} \diamond \begin{bmatrix} 0.68 & 0.32 \\ 0.37 & 0.63 \end{bmatrix} \diamond \begin{bmatrix} 0.62 & 0.38 \\ 0.20 & 0.80 \end{bmatrix} \diamond \begin{bmatrix} 0.81 & 0.19 \\ 0.15 & 0.85 \end{bmatrix}, \\ G_{cov_{N^1}} &= \begin{bmatrix} 1.00 & 0.32 \\ 0.00 & 0.68 \end{bmatrix} \diamond \begin{bmatrix} 0.10 & 0.00 \\ 0.90 & 1.00 \end{bmatrix} \diamond \begin{bmatrix} 0.75 & 0.33 \\ 0.25 & 0.67 \end{bmatrix} \diamond \begin{bmatrix} 1.00 & 0.37 \\ 0.00 & 0.63 \end{bmatrix} \diamond \begin{bmatrix} 0.25 & 0.00 \\ 0.75 & 1.00 \end{bmatrix}, \\ G_{cov_{N^2}} &= \begin{bmatrix} 0.69 & 0.22 \\ 0.31 & 0.78 \end{bmatrix} \diamond \begin{bmatrix} 0.52 & 0.39 \\ 0.48 & 0.61 \end{bmatrix} \diamond \begin{bmatrix} 0.54 & 0.24 \\ 0.46 & 0.76 \end{bmatrix} \diamond \begin{bmatrix} 0.75 & 0.31 \\ 0.25 & 0.69 \end{bmatrix} \diamond \begin{bmatrix} 0.79 & 0.13 \\ 0.21 & 0.87 \end{bmatrix}. \end{aligned}$$

To illustrate, we can describe some results of comparisons between female (N^1) and male (N^2) students' patterns for the rightmost matrices as follows:

- In $G_{\sigma_{N^1}}$ and $G_{\sigma_{N^2}}$, 80% of female and 51% of male students have low *score for inequality* and low *total score*, respectively ($\sigma_{N^1}(I_L, T_L) = 0.80$, $\sigma_{N^2}(I_L, T_L) = 0.51$).
- In $G_{cer_{N^1}}$ and $G_{cer_{N^2}}$, 84% of female and 85% of male students who have low *score for inequality* will also have low *total score*, respectively ($cer_{N^1}(I_L, T_L) = 0.84$, $cer_{N^2}(I_L, T_L) = 0.85$).

This material is for personal use only, not allowed for commercial use.

- In $G_{cov_{N^1}}$ and $G_{cov_{N^2}}$, all female and 87% of male students who have low *total score* are students who have low *score for inequality* ($cov_{N^1}(I_L, T_L) = 1.00$, $cov_{N^2}(I_L, T_L) = 0.87$).

The rest of the matrices can be interpreted by the same approach e.g., the leftmost matrix in $G_{\sigma_{N^1}}$ indicates that there are only 5% of female students who have high *score for geometry* and high *score for number theory* ($\sigma_{N^1}(G_H, N_H) = 0.05$), 30% of female students have high *score for geometry* and low *score of Number theory* ($\sigma_{N^1}(G_H, N_L) = 0.3$), 65% of female students have both low *score of Geometry* and low *score for number theory* ($\sigma_{N^1}(G_L, N_L) = 0.65$) but there is no female student who has low *score of Geometry* and high *score of Number theory* ($\sigma_{N^1}(G_L, N_H) = 0$) and so on.

Next, we use Theorem 6 for calculating the strength, certainty and coverage of the composed flow graph: $N^1 \cup N^2$, the results in matrix form are

$$\begin{aligned} G_{\sigma_{N^1 \cup N^2}} &= \begin{bmatrix} 0.25 & 0.16 \\ 0.11 & 0.48 \end{bmatrix} \diamond \begin{bmatrix} 0.17 & 0.19 \\ 0.24 & 0.40 \end{bmatrix} \diamond \begin{bmatrix} 0.27 & 0.14 \\ 0.20 & 0.39 \end{bmatrix} \diamond \begin{bmatrix} 0.25 & 0.22 \\ 0.08 & 0.45 \end{bmatrix} \diamond \begin{bmatrix} 0.27 & 0.06 \\ 0.10 & 0.57 \end{bmatrix}, \\ G_{cer_{N^1 \cup N^2}} &= \begin{bmatrix} 0.61 & 0.39 \\ 0.19 & 0.81 \end{bmatrix} \diamond \begin{bmatrix} 0.47 & 0.53 \\ 0.38 & 0.62 \end{bmatrix} \diamond \begin{bmatrix} 0.66 & 0.34 \\ 0.34 & 0.66 \end{bmatrix} \diamond \begin{bmatrix} 0.53 & 0.47 \\ 0.15 & 0.85 \end{bmatrix} \diamond \begin{bmatrix} 0.82 & 0.18 \\ 0.15 & 0.85 \end{bmatrix}, \\ G_{cov_{N^1 \cup N^2}} &= \begin{bmatrix} 0.69 & 0.25 \\ 0.31 & 0.75 \end{bmatrix} \diamond \begin{bmatrix} 0.41 & 0.32 \\ 0.59 & 0.68 \end{bmatrix} \diamond \begin{bmatrix} 0.57 & 0.26 \\ 0.43 & 0.74 \end{bmatrix} \diamond \begin{bmatrix} 0.76 & 0.33 \\ 0.24 & 0.67 \end{bmatrix} \diamond \begin{bmatrix} 0.73 & 0.10 \\ 0.27 & 0.90 \end{bmatrix}. \end{aligned}$$

From the above matrices, we can interpret some results for students' patterns ($N^1 \cup N^2$) of the rightmost matrices as follows:

- In $G_{\sigma_{N^1 \cup N^2}}$, 57% of students have low *score for inequality* and low *total score* ($\sigma_{N^1 \cup N^2}(I_L, T_L) = 0.57$).
- In $G_{cer_{N^1 \cup N^2}}$, 85% of students who have low *score for inequality* will also have low *total score* ($cer_{N^1 \cup N^2}(I_L, T_L) = 0.85$).
- In $G_{cov_{N^1 \cup N^2}}$, 90% of students who have low *total score* come from students who have low *score for inequality*. ($cov_{N^1 \cup N^2}(I_L, T_L) = 0.90$).

Here we focus on geometry and total scores. We can construct the combined normalized flow graphs for female (N^1) and male (N^2) students and apply Theorem 4 to compute all coefficients of the corresponding connections:

$$\begin{aligned} G_{\sigma_{N^1}}(G_i, T_j) &= \begin{bmatrix} 0.07 & 0.28 \\ 0.13 & 0.52 \end{bmatrix}, G_{cer_{N^1}}(G_i, T_j) = \begin{bmatrix} 0.21 & 0.79 \\ 0.20 & 0.80 \end{bmatrix}, G_{cov_{N^1}}(G_i, T_j) = \begin{bmatrix} 0.36 & 0.35 \\ 0.64 & 0.65 \end{bmatrix}, \\ G_{\sigma_{N^2}}(G_i, T_j) &= \begin{bmatrix} 0.18 & 0.24 \\ 0.24 & 0.34 \end{bmatrix}, G_{cer_{N^2}}(G_i, T_j) = \begin{bmatrix} 0.42 & 0.58 \\ 0.41 & 0.59 \end{bmatrix}, G_{cov_{N^2}}(G_i, T_j) = \begin{bmatrix} 0.43 & 0.42 \\ 0.57 & 0.58 \end{bmatrix}. \end{aligned}$$

Note that we calculate $G_{cov_{N^1}}(G_i, T_j)$ and $G_{cer_{N^1}}(G_i, T_j)$ by using Theorem 4 with 32 multiplications and 16 additions whereas traditional formulas require 130 multiplications and 22 additions. Similarly, we compute $G_{cov_{N^2}}(G_i, T_j)$ and $G_{cer_{N^2}}(G_i, T_j)$ with 32 multipli-

This material is reserved for educational use only; not allowed for commercial use.

cations and 16 additions whereas traditional formulas require 320 multiplications and 60 additions.

Finally, we can use Theorem 6 to directly calculate three coefficients of connections for the composed normalized flow graph $N^1 \cup N^2$ as

$$G_{\sigma_{N^1 \cup N^2}}(G_i, T_j) = \begin{bmatrix} 0.16 & 0.25 \\ 0.22 & 0.38 \end{bmatrix},$$

$$G_{cer_{N^1 \cup N^2}}(G_i, T_j) = \begin{bmatrix} 0.39 & 0.61 \\ 0.37 & 0.63 \end{bmatrix} \text{ and}$$

$$G_{cov_{N^1 \cup N^2}}(G_i, T_j) = \begin{bmatrix} 0.42 & 0.40 \\ 0.58 & 0.60 \end{bmatrix}.$$

Sensitivity, Specificity and Accuracy Analysis and Model Selection

From $G_{cer_{N^1}}$, $G_{cer_{N^2}}$ and $G_{cer_{N^1 \cup N^2}}$ above, we extract association rules and decision rules (or models) from the row of each matrix by selecting rules which give maximum certainty. We consider five models as illustrated in Tables 4.2 - 4.4. We compare the sensitivity, specificity and accuracy of each model.

Table 4.2. Models for N^1

	Classification models of N^1				
	Model 1: G and N	Model 2: N and A	Model 3: A and C	Model 4: C and I	Model 5: I and T
	IF G is high, THEN N is low. IF G is low, THEN N is low.	IF N is high, THEN A is high. IF N is low, THEN A is low.	IF A is high, THEN C is high. IF A is low, THEN C is low.	IF C is high, THEN I is low. IF C is low, THEN I is low.	IF I is high, THEN T is high. IF I is low, THEN T is low.
Sensitivity	0%	11.00%	75.00%	0%	100%
Specificity	100%	100%	67.00%	100%	84.21%
Accuracy	95.00%	60.00%	70.00%	95.00%	85.00%

Table 4.3. Models for N^2

	Classification models of N^2				
	Model 1: G and N	Model 2: N and A	Model 3: A and C	Model 4: C and I	Model 5: I and T
	IF G is high, THEN N is high. IF G is low, THEN N is low.	IF N is high, THEN A is low. IF N is low, THEN A is low.	IF A is high, THEN C is high. IF A is low, THEN C is low.	IF C is high, THEN I is high. IF C is low, THEN I is low.	IF I is high, THEN T is high. IF I is low, THEN T is low.
Sensitivity	68.57%	0%	52.63%	75.00%	76.67%
Specificity	78.78%	68.06%	73.81%	68.75%	89.13%
Accuracy	73.75%	45.57%	63.75%	71.25%	83.75%

In Tables 4.2 - 4.4, “high” is considered as a positive case and “low” is considered as a negative case. We can interpret as follows:

Table 4.4. Models for $N^1 \cup N^2$

	Classification models of $N^1 \cup N^2$				
	Model 1: G and N	Model 2: N and A	Model 3: A and C	Model 4: C and I	Model 5: I and T
	IF G is high, THEN N is high. IF G is low, THEN N is low.	IF N is low, THEN A is low. IF N is high, THEN A is low.	IF A is high, THEN C is high. IF A is low, THEN C is low.	IF C is high, THEN I is high. IF C is low, THEN I is low.	IF I is high, THEN T is high. IF I is low, THEN T is low.
Sensitivity	69.44%	0%	57.44%	75.76%	72.97%
Specificity	75.00%	100%	73.59%	67.16%	90.48%
Accuracy	73.00%	59.00%	66.00%	70.00%	84.00%

- Model 5 for N^1 has maximum sensitivity that is 100%. This means that we can use this model to certainly recognize all actual positives.
- Models 1, 2 and 4 for N^1 and 2 for $N^1 \cup N^2$ have maximum specificity (100%). For example, model 2 for N^1 tells us that all female student who has low *score for algebra* is identified as having low *score for algebra*.
- Models 1 and 4 for N^1 have maximum accuracy equal to 95%. That is, by using these models we can classify female students' *score for number theory* and *score for inequality* with 95 % accuracy.

In conclusion, we see that some models have high sensitivity but low accuracy or have both high sensitivity and specificity but low accuracy. By the meaning of sensitivity, specificity and accuracy, the most desirable model for classification should have high sensitivity specificity and accuracy. For this data set, models 5 for N^1 , N^2 and $N^1 \cup N^2$ are appropriate for classification.

CHAPTER 5

FLOW GRAPHS CONSTRUCTION FROM FUZZY SETS

In real-world data analysis, if collected data is in the form of fuzzy attributes, then it can be represented in a decision table with fuzzy attributes. We can analyze this decision table by applying concepts of fuzzy set. Rolka *et al.* introduced a concept of fuzzy flow graphs in [18]. This approach allowed computation not only for descriptive attributes but also for fuzzy attributes in flow graphs. In this thesis, we introduce an alternative approach to construct flow graphs, decision rules and perform data analysis for decision table with fuzzy attributes.

This chapter is organized as follows. In Sections 5.1, we provide a glimpse of fuzzy set theory [41], respectively. In Section 5.2, we recall flow graphs with fuzzy attributes. Section 5.3 contains an example on data analysis. We end up with experimental results in the last section.

5.1 Fuzzy Set Theory

The following fuzzy set theory concept is taken from [41]. Let X be a space of objects and x be a generic element of X . A *fuzzy set* A in X is characterized by a *membership function*, $\mu_A(x) \rightarrow [0,1]$. The nearer the value of $\mu_A(x)$ to 1, the higher the grade of membership of x in A . A crucial issue for fuzzy sets applications is to express real-world problem by fuzzy sets attributes in decision table and perform data analysis. In further discussions, we consider decision table with fuzzy attributes in the form introduced in [17, 18].

Let $(U, C \cup D)$ be a decision table with fuzzy attributes, $U = \{x_1, x_2, \dots, x_N\}$ be a finite non-empty set of the universe with N elements, $C = \{c_1, c_2, \dots, c_n\}$ be a set of condition attributes and $D = \{d_1, d_2, \dots, d_m\}$ be a set of decision attributes. The domain of the family of linguistic values of condition attribute c_i denoted by $V_i = \{V_{i_1}, V_{i_2}, \dots, V_{i_{n_i}}\}$, where n_i is the number of the linguistic values of i -th conditions, $i = 1, 2, \dots, n$. The domain of the family of linguistic values of decision attribute d_j denoted by $W_j = \{W_{j_1}, W_{j_2}, \dots, W_{j_{m_j}}\}$, where m_j is the number of the linguistic values of j -th conditions, $j = 1, 2, \dots, m$.

In order to describe an element $x \in U$ with fuzzy attributes, its membership degree should be determined. This process is called fuzzification state. In this state, for any element x , fuzzy value of condition attribute c_i and decision attribute d_j denoted by $V_i(x)$ and $W_j(x)$ are defined as

$$V_i(x) = \left\{ \mu_{v_{i_1}}(x)/V_{i_1}, \mu_{v_{i_2}}(x)/V_{i_2}, \dots, \mu_{v_{i_{n_i}}}(x)/V_{i_{n_i}} \right\}$$

and

$$W_j(x) = \left\{ \mu_{w_{j_1}}(x)/W_{j_1}, \mu_{w_{j_2}}(x)/W_{j_2}, \dots, \mu_{w_{j_{m_j}}}(x)/W_{j_{m_j}} \right\}^1.$$

Furthermore, for any $x \in U$ we assume that values of $V_i(x)$ and $W_j(x)$ satisfy $power(V_i(x)) = \sum_{k=1}^{n_i} \mu_{V_{i_k}}(x) = 1$ and $power(W_j(x)) = \sum_{k=1}^{m_j} \mu_{W_{j_k}}(x) = 1$, $i = 1, 2, \dots, n$, $j = 1, 2, \dots, m$. An example of fuzzy values $V_i(x)$ of c_i and fuzzy values $W_j(x)$ of d_j is in Table 5.1.

Table 5.1. Decision table with fuzzy attributes

	c_1	c_2	...	c_n	d_1	d_2	...	d_m
x_1	$V_1(x_1)$	$V_2(x_1)$	...	$V_n(x_1)$	$W_1(x_1)$	$W_2(x_1)$	...	$W_m(x_1)$
x_2	$V_1(x_2)$	$V_2(x_2)$	...	$V_n(x_2)$	$W_1(x_2)$	$W_2(x_2)$	...	$W_m(x_2)$
$\vdots$								$\vdots$
x_N	$V_1(x_N)$	$V_2(x_N)$	...	$V_n(x_N)$	$W_1(x_N)$	$W_2(x_N)$	...	$W_m(x_N)$

5.2 Flow Graphs with Fuzzy Attributes

Flow graphs with fuzzy attributes represent data in form of fuzzy attributes and can analyze decision table with fuzzy attributes. They have several advantages as listed in literature [18]. In further discussions, we employ flow graphs in the form introduced in [18]. All possible decision rules with fuzzy attributes can be generated by using the *Cartesian product* of the set of attribute values with $r = \prod_{i=1}^n n_i \prod_{j=1}^m m_j$ as a possible number of rules. Let R_k be the k -th decision rule. It can be expressed by

$$\begin{aligned} R_k: \text{ IF } & c_1 \text{ is } V_1^k, c_2 \text{ is } V_2^k, \dots \text{ and } c_n \text{ is } V_n^k, \\ \text{ THEN } & d_1 \text{ is } W_1^k, d_2 \text{ is } W_2^k, \dots \text{ and } d_m \text{ is } W_m^k, \end{aligned}$$

where $k = 1, 2, \dots, r$, and $V_i^k \in V_i$, $i = 1, 2, \dots, n$, $W_j^k \in W_j$, $j = 1, 2, \dots, m$.

Thus, Cartesian products $C^k = V_1^k \times V_2^k \times \dots \times V_n^k$ and $D^k = W_1^k \times W_2^k \times \dots \times W_m^k$ are the k -th decision rules expressed in the form of fuzzy implication, $C^k \rightarrow D^k$.

Next, we apply a *T-norm* operator to determine the *confirmation degree* of fuzzy decision rule. Let $cda(x, k)$ and $cdc(x, k)$ be the confirmation degrees of the antecedent and consequent of decision rules and $cd(x, k)$ be the confirmation degrees of the k -th decision rules by the element $x \in U$. They are defined respectively as:

¹If the values of all attributes are singletons or disjoint intervals on the original domain of attributes, then we obtain a classical crisp decision table.

$$cda(x, k) = T(\mu_{V_1^k}(x), \mu_{V_2^k}(x), \dots, \mu_{V_n^k}(x)),$$

$$cdc(x, k) = T(\mu_{W_1^k}(x), \mu_{W_2^k}(x), \dots, \mu_{W_m^k}(x)),$$

$$cd(x, k) = T(cda(x, k), cdc(x, k)).$$

In fuzzy set domain, the *support* (*sup*) of decision rule's antecedent, consequent and itself are defined respectively as:

$$sup(cda(x, k)) = \{cda(x_1, k)/x_1, cda(x_2, k)/x_2, \dots, cda(x_N, k)/x_N\},$$

$$sup(cdc(x, k)) = \{cdc(x_1, k)/x_1, cdc(x_2, k)/x_2, \dots, cdc(x_N, k)/x_N\},$$

$$sup(R_k) = \{cd(x_1, k)/x_1, cd(x_2, k)/x_2, \dots, cd(x_N, k)/x_N\}.$$

Moreover, when consider the fuzzy universe, we can define the *throughflow*, $\varphi(x, y)$ of the branch (x, y) in G to the cardinality of the product of fuzzy sets, $\tilde{x}$ and $\tilde{y}$. The *strength* of (x, y) is thus given by

$$\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)}, \quad (5.1)$$

where $\varphi(x, y) = \text{power}(\tilde{x} \cap \tilde{y})$ and $\varphi(G) = N$.

With every node x of a flow graph G , associated *inflow* and *outflow* are defined respectively [18] as:

$$\varphi_-(x) = \text{power}\tilde{x} = \sum_{y \in O(x)} \varphi(x, y) = \sum_{y \in O(x)} \text{power}(\tilde{x} \cap \tilde{y}), \quad (5.2)$$

$$\varphi_+(x) = \text{power}\tilde{x} = \sum_{y \in I(x)} \varphi(x, y) = \sum_{y \in I(x)} \text{power}(\tilde{x} \cap \tilde{y}), \quad (5.3)$$

where $\text{power}\tilde{x}$ denotes fuzzy cardinality.

For any internal node x , it holds that $\varphi_-(x) = \varphi_+(x) = \varphi(x)$.

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

$$cer(x, y) = \frac{\text{Power}(\tilde{x} \cap \tilde{y})}{\text{Power}(\tilde{x})} = \frac{\sigma(x, y)}{\sigma(x)}, \quad (5.4)$$

$$cov(x, y) = \frac{\text{Power}(\tilde{x} \cap \tilde{y})}{\text{Power}(\tilde{y})} = \frac{\sigma(x, y)}{\sigma(y)}. \quad (5.5)$$

² $\tilde{x}$ is a fuzzy set on the universe U , which describes membership degree of particular elements $x_i \in U$ in the linguistic value represented by x . The membership degrees of all x_i in the set $\tilde{x}$ can be found in a respective column of the considered decision table [18].

A *path* from x to y ($x \neq y$) in G , denoted by $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ such that $x_1 = x$ and $x_n = y$ and $(x_i, x_{i+1}) \in B$ for every i , $1 \leq i \leq n-1$. We obtain the *certainty*, *coverage* and *strength* coefficients of the path $[x_1 \dots x_n]$ as:

$$cer[x_1 \dots x_n] = \prod_{i=1}^{n-1} cer(x_1 \dots x_i, x_{i+1}), \quad (5.6)$$

where $cer(x_1 \dots x_i, x_{i+1}) = \frac{power(\tilde{x}_1 \cap \tilde{x}_2 \cap \dots \cap \tilde{x}_{i+1})}{power(\tilde{x}_1 \cap \tilde{x}_2 \cap \dots \cap \tilde{x}_i)}$,

$$cov[x_1 \dots x_n] = \prod_{i=1}^{n-1} cov(x_1 \dots x_i, x_{i+1}), \quad (5.7)$$

where $cov(x_1 \dots x_i, x_{i+1}) = \frac{power(\tilde{x}_1 \cap \tilde{x}_2 \cap \dots \cap \tilde{x}_{i+1})}{power(\tilde{x}_{i+1})}$,

$$\sigma[x_1 \dots x_n] = \sigma(x_1)cer[x_1 \dots x_n] = \sigma(x_n)cov[x_1 \dots x_n]. \quad (5.8)$$

Next, a *path*, $[x_1 \dots x_n]$, is a sequence of decision rules $x_1 \rightarrow x_2$, $x_2 \rightarrow x_3$, $\dots$, $x_{n-1} \rightarrow x_n$. This sequence of decision rules can be interpreted as a single decision rule $x_1 x_2 \dots x_{n-1} \rightarrow x_n$, in short $x^* \rightarrow x_n$, where $x^* = x_1 x_2 \dots x_{n-1}$.

The decision rule R_k is represented by a branch (x^*, x_n) . Power of the support of the rule R_k is equal to the flow between the nodes x^* and x_n as follows:

$$\varphi(x^*, x_n) = power(sup(R_k)). \quad (5.9)$$

Moreover, with every decision rule R_k , $x^* \rightarrow x_n$, where x^* is condition and x_n is decision can be characterized by three coefficients:

$$cer(x^*, x_n) = cer(R_k) = \frac{power(sup(R_k))}{power(sup(cda(x, k)))}, \quad (5.10)$$

$$cov(x^*, x_n) = cov(R_k) = \frac{power(sup(R_k))}{power(sup(cdc(x, k)))}, \quad (5.11)$$

$$\sigma(x^*, x_n) = strength(R_k) = \frac{power(sup(R_k))}{card(U)}. \quad (5.12)$$

5.3 Illustrative Example for Flow Graph Construction from Fuzzy Attributes

In this section, we present an example of how to compute coefficients: *certainty*, *coverage* and *strength* for flow graphs constructed from fuzzy attributes. To compute these coefficients for flow graph constructed from descriptive attributes please refer to Section 4.3.

Table 5.2. An example decision table

	<i>N</i>	<i>A</i>	<i>D</i>		<i>N</i>	<i>A</i>	<i>D</i>
x_1	32	46	178	x_6	16	31	81
x_2	9	40	102	x_7	2	37	74
x_3	8	35	101	x_8	0	36	73
x_4	11	31	88	x_9	10	29	73
x_5	12	27	87	x_{10}	4	28	70

Table 5.2 shows a decision table $S = (U, A)$, where $U = \{x_1, x_2, \dots, x_{10}\}$, condition attributes are score of Number theory (N) and score of Algebra (A). The decision attribute (target function) is total score (D).

We begin by using triangular membership functions below to the data in fuzzification state. The result is a decision table with fuzzy attributes as shown in Table 5.3.

$\mu_{N_1} = \begin{cases} 1 & , 0 \leq x < 5 \\ (11-x)/2, 5 \leq x < 11 \\ 0 & , 11 \leq x \leq 45 \end{cases}$	$\mu_{N_2} = \begin{cases} 0 & , 0 \leq x < 5 \\ (x-5)/6, 5 \leq x < 11 \\ (40-x)/29, 11 \leq x < 40 \\ 0 & , 40 \leq x \leq 45 \end{cases}$	$\mu_{N_3} = \begin{cases} 0 & , 0 \leq x < 11 \\ (x-11)/29, 11 \leq x < 40 \\ 1 & , 40 \leq x \leq 45 \end{cases}$
$\mu_{A_1} = \begin{cases} 1 & , 0 \leq x < 20 \\ (30-x)/10, 20 \leq x < 30 \\ 0 & , 30 \leq x \leq 46 \end{cases}$	$\mu_{A_2} = \begin{cases} 0 & , 0 \leq x < 20 \\ (x-20)/10, 20 \leq x < 30 \\ (36-x)/6, 30 \leq x < 36 \\ 0 & , 36 \leq x \leq 46 \end{cases}$	$\mu_{A_3} = \begin{cases} 0 & , 0 \leq x < 30 \\ (x-30)/6, 30 \leq x < 36 \\ 1 & , 36 \leq x \leq 46 \end{cases}$
$\mu_{D_1} = \begin{cases} 1 & , 0 \leq x < 69 \\ (85-x)/16, 69 \leq x < 85 \\ 0 & , 85 \leq x \leq 195 \end{cases}$	$\mu_{D_2} = \begin{cases} 0 & , 0 \leq x < 69 \\ (x-69)/16, 69 \leq x < 85 \\ (110-x)/25, 85 \leq x < 110 \\ 0 & , 110 \leq x \leq 195 \end{cases}$	$\mu_{D_3} = \begin{cases} 0 & , 0 \leq x < 85 \\ (x-85)/25, 85 \leq x < 110 \\ 1 & , 110 \leq x \leq 195. \end{cases}$

Table 5.3. Decision table with fuzzy attributes from Table 5.2

	N_i			A_j			D_k		
	N_1	N_2	N_3	A_1	A_2	A_3	D_1	D_2	D_3
x_1	0	0.276	0.724	0	0	1	0	0	1
x_2	0.333	0.667	0	0	0	1	0	0.320	0.680
x_3	0.500	0.500	0	0	0.167	0.833	0	0.360	0.640
x_4	0	1	0	0	0.833	0.167	0	0.880	0.120
x_5	0	0.965	0.035	0.300	0.700	0	0	0.920	0.080
x_6	0	0.828	0.172	0	0.833	0.167	0.250	0.750	0
x_7	1	0	0	0	0	1	0.688	0.312	0
x_8	1	0	0	0	0	1	0.750	0.250	0
x_9	0.167	0.833	0	0.100	0.900	0	0.75	0.250	0
x_{10}	1	0	0	0.200	0.800	0	0.938	0.062	0

Afterwards, we can compute the strength coefficient, $\sigma(N_i, A_j)$, where $i, j = 1, 2, 3$, of two condition attributes in Table 5.3 by (5.1) and the result is in Table 5.4. For example, since

$$\widetilde{N_1} = \{0/x_1, 0.333/x_2, 0.5/x_3, 0/x_4, 0/x_5, 0/x_6, 1/x_7, 1/x_8, 0.167/x_9, 1/x_{10}\} \text{ and}$$
$$\widetilde{A_1} = \{0/x_1, 0/x_2, 0/x_3, 0/x_4, 0.3/x_5, 0/x_6, 0/x_7, 0/x_8, 0.1/x_9, 0.2/x_{10}\},$$

we get

$$\widetilde{N_1} \cap \widetilde{A_1} = \widetilde{N_1} \cap \widetilde{A_1} = \{0 \cdot 0/x_1, 0.333 \cdot 0/x_2, 0.5 \cdot 0/x_3, 0 \cdot 0/x_4, 0 \cdot 0.3/x_5, 0 \cdot 0/x_6, \\ 1 \cdot 0/x_7, 1 \cdot 0/x_8, 0.167 \cdot 0.1/x_9, 1 \cdot 0.2/x_{10}\}.$$

Therefore, we can write

$$\begin{aligned} \widetilde{N_1} \widetilde{A_1} &= \widetilde{N_1} \cap \widetilde{A_1} = (0 \times 0) + (0.33 \times 0) + (0.5 \times 0) + (0 \times 0) + (0 \times 0.3) + \\ &\quad (0 \times 0) + (1 \times 0) + (1 \times 0) + (0.167 \times 0.1) + (1 \times 0.2) \\ &= 0.2167. \end{aligned}$$

Then $\varphi(N_1, A_1) = Power(\widetilde{N_1} \cap \widetilde{A_1}) = 0.2167$ and $\sigma(N_1, A_1) = \frac{\varphi(N_1, A_1)}{\varphi(G)} = \frac{0.2167}{10} = 0.0217$ (first left cell). The interpretation is 2.17% of students have low scores of Number theory and Algebra.

Table 5.4. Strength between score of Number theory and Algebra ($\sigma(N_i, A_j)$)

	A_1	A_2	A_3	Σ
N_1	0.0217	0.1033	0.2750	$\sigma_+(N_1) = 0.4000$
N_2	0.0373	0.3032	0.1664	$\sigma_+(N_2) = 0.5069$
N_3	0.0010	0.0168	0.0753	$\sigma_+(N_3) = 0.0931$
Σ	$\sigma_-(A_1) = 0.0600$	$\sigma_-(A_2) = 0.4233$	$\sigma_-(A_3) = 0.5167$	$\sigma(G) = 1$

Table 5.5. Strength between score of Algebra and total score ($\sigma(A_j, D_k)$)

	D_1	D_2	D_3	Σ
A_1	0.0263	0.0314	0.0024	$\sigma_+(A_1) = 0.0600$
A_2	0.1633	0.2337	0.0263	$\sigma_+(A_2) = 0.4233$
A_3	0.1479	0.1454	0.2233	$\sigma_+(A_3) = 0.5167$
Σ	$\sigma_-(D_1) = 0.3375$	$\sigma_-(D_2) = 0.4105$	$\sigma_-(D_3) = 0.2520$	$\sigma(G) = 1$

Moreover, in Table 5.4, last row depicts inflow of each A_j , and last column depicts outflow of each N_i . For example,

$$\begin{aligned} \sigma_-(A_1) &= \sum_{i=1}^3 \sigma(N_i, A_1) \\ &= \sigma(N_1, A_1) + \sigma(N_2, A_1) + \sigma(N_3, A_1) \\ &= 0.0217 + 0.0373 + 0.00103 = 0.06. \end{aligned}$$

We can interpret as the inflow of A_1 constitutes 6% of the total flow.

This material is for personal use only. It is not to be used for commercial use.
Forbidden to modify the content, and cite the document when use.

Similarly,

$$\begin{aligned}\sigma_+(N_1) &= \sum_{j=1}^3 \sigma(N_1, A_j) \\ &= \sigma(N_1, A_1) + \sigma(N_1, A_2) + \sigma(N_1, A_3) \\ &= 0.0217 + 0.1033 + 0.2750 = 0.4.\end{aligned}$$

We can interpret this result as the outflow of N_1 constitutes 40% of the total flow.

Consequently, Table 5.5 shows strength between score of Algebra and total score ($\sigma(A_j, D_k)$ where $j, k = 1, 2, 3$) by similar step as in Table 5.4. Please note that the last row from Table 5.4 and the last column from Table 5.5 show that $\sigma_-(A_j) = \sigma_+(A_j)$, where $j = 1, 2, 3$.

Table 5.6 Strength between score of Number theory, Algebra and total score ($\sigma(N_i A_j, D_k)$)

	D_1	D_2	D_3	Σ
$N_1 A_1$	0.0200	0.0017	0	$\sigma_+(N_1 A_1) = 0.0217$
$N_1 A_2$	0.0863	0.0118	0.0053	$\sigma_+(N_1 A_2) = 0.1033$
$\vdots$				$\vdots$
$N_3 A_3$	0.0007	0.0022	0.0724	$\sigma_+(N_3 A_3) = 0.0752$
Σ	$\sigma_-(D_1) = 0.3375$	$\sigma_-(D_2) = 0.4105$	$\sigma_-(D_3) = 0.2520$	$\sigma(G) = 1$

Table 5.6 tabulates strength between scores of Number theory, Algebra and total score ($\sigma(N_i A_j, D_k)$) by using (5.12). For example,

$$\begin{aligned}\widetilde{N}_1 \cap \widetilde{A}_1 \cap \widetilde{D}_1 &= \{(0 \cdot 0 \cdot 0)/x_1, (0.333 \cdot 0 \cdot 0)/x_2, (0.5 \cdot 0 \cdot 0)/x_3, (0 \cdot 0 \cdot 0)/x_4, \\ &\quad (0 \cdot 0.3 \cdot 0)/x_5, (0 \cdot 0 \cdot 0.25)/x_6, (1 \cdot 0 \cdot 0.688)/x_7, (1 \cdot 0 \cdot 0.75)/x_8, \\ &\quad (0.167 \cdot 0.1 \cdot 0.75)/x_9, (1 \cdot 0.2 \cdot 0.938)/x_{10}\},\end{aligned}$$

which yields $\varphi(\widetilde{N}_1 \cap \widetilde{A}_1 \cap \widetilde{D}_1) = \text{Power}(\widetilde{N}_1 \cap \widetilde{A}_1 \cap \widetilde{D}_1) = 0.2$.

Then $\sigma(N_1 A_1, D_1) = \frac{\varphi(N_1 \cap A_1 \cap D_1)}{\varphi(G)} = \frac{0.2}{10} = 0.02$. (i.e., 20% of students who have low score of Number theory and Algebra will have low total score). Furthermore, last row depicts inflow of D_k and last column depicts an outflow of $N_i A_j$.

Table 5.7. Certainty and coverage coefficients between N_i and A_j

	A_1		A_2		A_3	
	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>
N_1	0.0542	0.3612	0.2583	0.2440	0.6875	0.5322
N_2	0.0736	0.6217	0.5982	0.7163	0.3282	0.3220
N_3	0.0111	0.0171	0.1802	0.0397	0.8087	0.1458

Above we described a procedure for computing strength coefficient given a set of data. Next, we discuss the computation for certainty and coverage coefficients. Table 5.7 shows the certainty and coverage coefficients of N_i and A_j , where $i, j = 1, 2, 3$. They are computed by (5.4), as $cer(N_i, A_j) = \frac{\sigma(N_i, A_j)}{\sigma(N_i)}$ and by (5.5) as $cov(N_i, A_j) = \frac{\sigma(N_i, A_j)}{\sigma(A_j)}$. For example,

$$cer(N_1, A_1) = \frac{\sigma(N_1, A_1)}{\sigma(N_1)} = \frac{0.0217}{0.4} = 0.0542 \text{ (top left cell) and}$$

$$cov(N_1, A_1) = \frac{\sigma(N_1, A_1)}{\sigma(A_1)} = \frac{0.02167}{0.06} = 0.3612.$$

We can describe that 5.42% of students who have low score of Number theory will have low score of Algebra and 36.12% of students who have low score of Algebra come from students who have low score of Number theory.

Table 5.8 shows certainty and coverage coefficients between A_j and D_k , where $j, k = 1, 2, 3$. They are computed by the same approach as in Table 5.7.

Table 5.8. Certainty and coverage coefficients between A_i and D_k

	D_1		D_2		D_3	
	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>
A_1	0.4375	0.0778	0.5225	0.0764	0.0400	0.0095
A_2	0.3858	0.4840	0.5521	0.5694	0.0620	0.1042
A_3	0.2863	0.4383	0.2815	0.3542	0.4323	0.8862

Table 5.9. Certainty and coverage coefficients between N_i and D_k

	D_1		D_2		D_3	
	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>
N_1	0.6250	0.7407	0.2383	0.2322	0.1367	0.2169
N_2	0.1641	0.2465	0.5900	0.7285	0.2459	0.4946
N_3	0.0463	0.0128	0.173	0.0392	0.7807	0.2885

Finally, Table 5.9 shows certainty and coverage coefficients of N_i and D_k , where $i, k = 1, 2, 3$. They are computed by the same approach as in Table 5.7.

Next, by (5.10) and (5.11), we can find certainty and coverage coefficients of N_i, A_j and D_k , where $i, j, k = 1, 2, 3$. The results are shown in Table 5.10. For example

$$cer(N_1 A_1, D_1) = \frac{Power(N_1 \cap A_1 \cap D_1)}{Power(N_1 \cap A_1)}$$

$$= \frac{\sigma(N_1 A_1, D_1)}{\sigma(N_1, A_1)} = \frac{0.02}{0.0217} = 0.9229 \text{ (top left cell) and}$$

$$cov(N_1 A_1, D_1) = \frac{Power(N_1 \cap A_1 \cap D_1)}{Power(D_1)}$$

$$= \frac{\sigma(N_1 A_1, D_1)}{\sigma(D_1)} = \frac{0.02}{0.3375} = 0.0593.$$

Table 5.10. Certainty and coverage coefficients of N_i , A_j and D_k

	D_1		D_2		D_3	
	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>	<i>cer</i>	<i>cov</i>
N_1A_1	0.9229	0.0593	0.0769	0.0041	0	0
N_1A_2	0.8347	0.2556	0.1137	0.0286	0.0516	0.0212
$\vdots$						$\vdots$
N_3A_3	0.0095	0.0021	0.0286	0.0053	0.9618	0.2874

We can describe that 92.29% of students who have low scores of both Number theory and Algebra will have low total score and 6% of students who have low total score come from students who have low scores of both Number theory and Algebra.

5.4 Experimental Results

In this section, we describe experimental results from the scores of Olympiad mathematics' students' analysis as described in Section 3.6. We study decision rules between condition score of Geometry (*score_Geo*), score of Number theory (*score_Num*), score of Algebra (*score_Alg*), score of Inequality (*score_I.E.*), score of Combinatorics (*score_Comb*) and decision total score (*total_score*). Data contains 75 records (see Appendix B). We transform POSN data set to be in fuzzy attributes form by triangular membership functions (Fig. 5.1). Several case studies are then presented according to the combination of these attributes. The threshold for decision rule extraction is $cer > 0.6$.

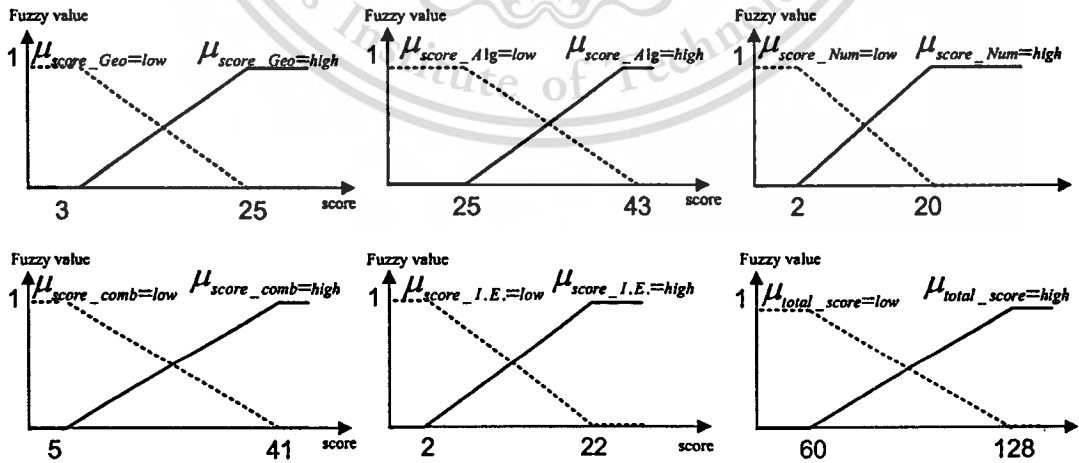


Figure 5.1 Triangular membership functions.

Case 1. If we consider decision rules between score of any subject and total score. Then the selected decision rule is:

Rules	cer	cov
IF <i>score_Num</i> is high THEN <i>total_score</i> is high	0.627	0.504

with $cer = 0.627$ and $cov = 0.504$.

Case 2. If we consider decision rules between score of any two subjects and total score. Then the selected decision rules are:

Rules	Sup	Conf
IF <i>score_Geo</i> and <i>I.E.</i> are low THEN <i>total_score</i> is low	0.651	0.409
IF <i>score_Geo</i> and <i>I.E.</i> are high THEN <i>total_score</i> is high	0.68	0.301
⋮		
IF <i>score_Alg</i> and <i>I.E.</i> are high THEN <i>total_score</i> is high	0.618	0.226

with average $cer = 0.656$ and average $cov = 0.343$.

Case 3. If we consider decision rules between score of any five subjects and total score. Then the selected decision rules are:

Rules	cer	cov
IF <i>score_Geo</i> , <i>Num</i> , <i>Alg</i> , <i>Com</i> and <i>score_I.E.</i> are low THEN <i>total_score</i> is low	0.782	0.139
IF <i>score_Geo</i> , <i>Num</i> , <i>Alg</i> , and <i>Com</i> are low, <i>I.E.</i> is high THEN <i>total_score</i> is low	0.645	0.09
⋮		
IF <i>score_Geo</i> , <i>Num</i> , <i>Alg</i> , <i>Comb</i> and <i>I.E.</i> are high THEN <i>total_score</i> is high	0.81	0.024

with average $cer = 0.691$ and average $cov = 0.047$.

Case 4. If we consider decision rules between practiced and preferred subjects, score of practiced and preferred subject and total score. The selected decision rules are:

Rules	cer	cov
IF <i>Pref</i> and <i>Prac_Geo</i> and <i>score_Geo</i> is low THEN <i>total_score</i> is low	0.638	0.719
IF <i>Pref</i> and <i>Prac_Num</i> and <i>score_Num</i> is high THEN <i>total_score</i> is high	0.834	0.6

with average $cer = 0.736$ and average $cov = 0.660$.

Case 5. If we consider decision rules between practiced subjects, score of practiced subjects and total score. The selected decision rules are:

Rules	cer	cov
IF <i>Prac_Geo</i> and <i>score_Geo</i> is low THEN <i>total_score</i> is low	0.602	0.731
IF <i>Prac_I.E.</i> and <i>score_I.E.</i> is low THEN <i>total_score</i> is low	0.665	0.623

with average $cer = 0.634$ and average $cov = 0.677$.

Case 6. If we consider decision rules between preferred subjects, score of preferred subjects and total score. The selected decision rules are:

Rules	cer	cov
IF <i>Pref_Geo</i> and <i>score_Geo</i> is high THEN <i>total_score</i> is high	0.671	0.541
IF <i>Pref_Num</i> and <i>score_Num</i> is high THEN <i>total_score</i> is high	0.633	0.602
IF <i>Pref_I.E.</i> and <i>score_I.E.</i> is low THEN <i>total_score</i> is low	0.829	0.72
IF <i>Pref_I.E.</i> and <i>score_I.E.</i> is high THEN <i>total_score</i> is low	0.831	0.28

with average $cer = 0.741$ and average $cov = 0.536$.

From these selected decision rules, we can distil useful knowledge from these flow graphs as follows. Number theory affects total score substantially. Because in Case 1, 62.7% of students with high score of Number theory also have high total score. Moreover, 50.4% of students which have high total score are from the group in which Number theory score is high. Obviously, students who get high score from any two (or more) subjects will successively get high total score as shown in Cases 2 and 3. Interestingly, the combinations of either preferred subject or practiced subject and high score for such subject resulted in high total scores from Cases 4, 5 and 6. In Case 6, we obtain maximum certainty of selected decision rules as one can see in Fig. 5.2.

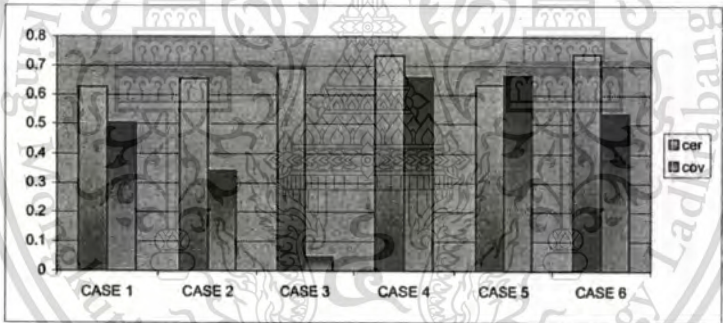


Figure 5.2 Comparison of cer , cov of POSN data.

In Fig. 5.2, the results are desirable values of cer and cov with an average $cer = 0.681$ and $cov = 0.461$. In this chapter, an extension of mathematical fuzzy set theory, namely flow graphs, is considered. New certainty and coverage coefficients are introduced. This approach allows computation not only for descriptive attributes but also for fuzzy attributes in flow graph. They are successively applied to the POSN data set.

CHAPTER 6

AN EXTENSION OF ROUGH SET APPROXIMATION TO FLOW GRAPH BASED DATA ANALYSIS

Flow distribution in flow graphs can be exploited for approximation and reasoning in data analysis. 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.

This chapter is organized as follows. In Section 6.1, we present the basic concepts of rough sets. In Section 6.2, we recall preliminary definitions of flow graphs. In Section 6.3, we present a new bridge between rough sets and flow graphs with an example throughout, followed by a conclusion in the last section

6.1 Rough Set Theory

The following rough sets preliminary is taken from [24]. Rough sets are based on an *information system*. 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 .

If we partition an information system into 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 $C \cap D = \emptyset$. 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 . Equivalence classes of the relation $I(B)$ are referred to as *B-elementary sets* or *B-elementary granules* denote by $B(X)$, i.e., $B(X)$ describes X in the terms of attribute values from B [29]. Below, we recall key feature definitions of approximations in rough sets.

Definition 9. [32] 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 respectively as

$$\underline{B}(X) = \bigcup_{x \in U} \{B(X) \mid B(X) \subseteq X\},$$

- $\overline{B}(X) = \bigcup_{x \in U} \{B(x) \mid B(x) \cap X \neq \emptyset\}$ and
- $BN_B(X) = \overline{B}(X) - \underline{B}(X)$.

If the boundary region of X is the empty set (i.e., $BN_B(x) = \emptyset$), then X is *crisp*. On the contrary, if $BN_B(X) \neq \emptyset$, then X is *rough*. In what follows we recall four basic classes of rough sets, i.e., four categories of vagueness.

Definition 10. [32] 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*.

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

Definition 11. [32] 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

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

where $\text{card}(X)$ 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 12. Let $S = (U, A)$ be an information system and $F = \{X_1, X_2, \dots, X_n\}$ be a partition of the universe U . For $B \subseteq A$, F depends on B to a degree $k_B(F)$ defined by

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

Definitions 10 - 12 will be stated in the context of flow graphs in Section 6.2.

6.2 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 $G = (A, B, \sigma)$, where $A = \{A_{i_1}, A_{i_2}, \dots, A_{i_n}\}$ is a set of attributes¹, 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_{i_k} 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_{i_k} contains n_{i_k} values, we say that it contains n_{i_k} nodes.

We now consider how to approximate an attribute value $Y \in A_{i_{k+1}}$ from attribute values of A_{i_k} where $A_{i_k} = \{X_1, X_2, \dots, X_{n_{i_k}}\}$, to indicate lower approximation, upper approximation and boundary region of Y . In Definition 13, we recall Pawlak's definitions of lower approximation, upper approximation and boundary region for flow graphs.

Definition 13. [29] Let $G = (A, B, \sigma)$ be a normalized flow graph, $A_{i_k} = \{X_1, X_2, \dots, X_{n_{i_k}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{i_{k+1}}$. For any branch (X_j, Y) , $j \in \{1, \dots, n_{i_k}\}$, of the flow graph G , the union of all inputs X_j of Y is the upper approximation of Y (denoted $\overline{A_{i_k}}(Y)$), the union of all inputs X_j of Y , such that $\text{cer}(X_j, Y) = 1$, is the lower approximation of Y (denoted $\underline{A_{i_k}}(Y)$). Moreover, the union of all inputs X_j of Y , such that $\text{cer}(X_j, Y) < 1$, is the boundary region of Y (denoted $A_{i_k}N_{A_{i_k}}(Y)$).

In Definition 14, we state four categories of rough sets mentioned in Definition 10 in terms of flow graph.

Definition 14. Let $G = (A, B, \sigma)$ be a flow graph, $A_{i_k} = \{X_1, X_2, \dots, X_{n_{i_k}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{i_{k+1}}$. For any branch (X_j, Y) , $j \in \{1, \dots, n_{i_k}\}$, of G , we define four categories of vagueness as

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

From the definition we obtain the following interpretation:

- if Y is roughly A_{i_k} -definable, this means that we are able to decide for some elements of U whether they belong to Y or $-Y^2$, using A_{i_k} ,

¹In what follows, we regard $\mathcal{N}$ as A for simplicity.

²Where $-Y = U - Y$.

- 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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$, $2 \leq i \leq k$, be an attribute in layer i and W be a node in $A_{l_{i-1}}$. For any branch (X_j, W) , $j \in \{1, \dots, n_{l_i}\}$ in the inverse flow graph of G , the union of all output X_j of W in flow graph G is the upper approximation of W , the union of all outputs X_j of W in a flow graph G , such that $\text{cov}(W, X_j) = 1$, is the lower approximation of W . Moreover, the union of all outputs X_i of W , such that $\text{cov}(W, X_j) < 1$, 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 13. $\square$

Example 5 Suppose we are given the flow graph for the preference analysis problem depicted in Fig. 6.1, that 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 $Y = \{Y_1, Y_2, Y_3\}$.

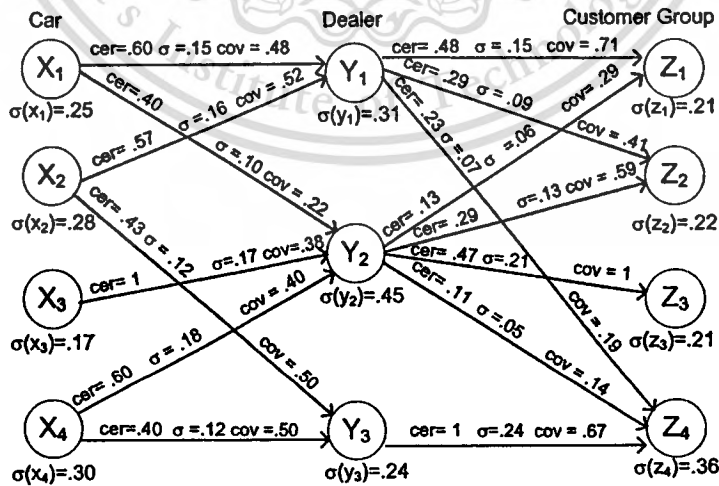


Figure 6.1 A normalized flow graph.

By Definition 13, 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 14, we conclude that Z_1 is internally Y -indefinable. In Fig. 6.1 (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.

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 14 do not provide approximations quantitatively. Hence, in Definitions 15 and 16, we define two measures for flow graphs, the accuracy of approximation and the roughness of approximation.

Definition 15. Let $G=(A,B,\sigma)$ be a flow graph, $A_{l_i}=\{X_1, X_2, \dots, X_{n_{l_i}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j, Y) , $j \in \{1, \dots, n_{l_i}\}$, of G , the accuracy of approximation, $\alpha_{A_{l_i}}(Y)$, is defined as:

$$\alpha_{A_{l_i}}(Y) = \frac{\text{card}(A_{l_i}(Y))}{\text{card}(A_{l_i})}.$$

We can use the accuracy of approximation to specify the quality of an approximation. Obviously, $0 \leq \alpha_B(X) \leq 1$. 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 16. Let $G=(A,B,\sigma)$ be a flow graph, $A_{l_i}=\{X_1, X_2, \dots, X_{n_{l_i}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_i, Y) , $i \in \{1, \dots, n_{l_i}\}$, of G , the roughness of approximation, $\gamma_{A_{l_i}}(Y)$, is defined as:

$$\gamma_{A_{l_i}}(Y) = 1 - \alpha_{A_{l_i}}(Y) = \frac{\text{card}(\overline{A_{l_i}}(Y)) - \text{card}(A_{l_i}(Y))}{\text{card}(A_{l_i}(Y))}.$$

Property 2. Let $G=(A,B,\sigma)$ be a flow graph, $A_{l_i}=\{X_1, X_2, \dots, X_{n_{l_i}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j, Y) , $j \in \{1, \dots, n_{l_i}\}$, of G , we have

$$(1) \alpha_{A_{l_i}}(Y) = \frac{\sum_{\text{cer}(X_j, Y)=1} \sigma(X_j)}{\sum_{X_j \in I(Y)} \sigma(X_j)} \text{ and}$$

$$(2) \gamma_{A_{l_i}}(Y) = \frac{\sum_{\text{cer}(X_j, Y)<1} \sigma(X_j)}{\sum_{X_j \in I(Y)} \sigma(X_j)}.$$

This manuscript is for personal use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Proof. (1) From Definition 13, we have $\text{card}(\underline{A}_{l_i}(Y)) = \sum_{\text{cer}(X_j, Y)=1} \text{card}(X_j)$ and $\text{card}(\overline{A}_{l_i}(Y)) = \sum_{X_j \in I(Y)} \text{card}(X_j)$. Since $\text{card}(X_j) = \varphi(X_j) = \sigma(X_j)\varphi(G) = \sigma(X_j)\varphi(U)$ and by Definition 15, then $\alpha_B(Y) = \frac{\sum_{\text{cer}(X_j, Y)=1} \sigma(X_j)}{\sum_{X_j \in I(Y)} \sigma(X_j)}$.

(2) It can be proved similarly to (1). $\square$

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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$, $2 \leq i \leq k$, be an attribute in layer i and W be a node in $A_{l_{i-1}}$. For any branch (X_j, W) , $j \in \{1, \dots, n_{l_j}\}$ in the inverse flow graph of G , we have

$$(1) \alpha_{A_{l_j}}(W) = \frac{\sum_{\text{cov}(W, X_j)=1} \sigma(X_j)}{\sum_{X_j \in O(W)} \sigma(X_j)} \text{ and}$$

$$(2) \gamma_{A_{l_j}}(W) = \frac{\sum_{\text{cov}(W, X_j) \leq 1} \sigma(X_j)}{\sum_{X_j \in O(W)} \sigma(X_j)}.$$

Proof. (1) From Property 1, we have $\text{card}(\underline{A}_{l_j}(W)) = \sum_{\text{cov}(X_j, W)=1} \text{card}(X_j)$ and $\text{card}(\overline{A}_{l_j}(W)) = \sum_{X_j \in O(W)} \text{card}(X_j)$. Since $\text{card}(X_j) = \varphi(X_j) = \sigma(X_j)\varphi(G) = \sigma(X_j)\varphi(U)$ and by Definition 15, then $\alpha_{A_{l_j}}(W) = \frac{\sum_{\text{cer}(X_j, W)=1} \sigma(X_j)}{\sum_{X_j \in O(W)} \sigma(X_j)}$.

(2) It can be proved similarly to (1). $\square$

Example 5 (Cont.) Consider the branches between dealer and customer group in Fig. 6.1. 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 ($\text{cer}(Y_3, Z_4) = 1$). There is only one branch (Y_3, Z_4) with $\text{cer}(Y_3, Z_4) = 1$ ³. Thus, by Property 2(1), we have $\alpha_Y(Z_1) = \alpha_Y(Z_2) = \alpha_Y(Z_3) = 0$ and $\alpha_Y(Z_4) = \frac{\sum_{\text{cer}(Y_i, Z_4)=1} \sigma(Y_i)}{\sum_{Y_i \in I(Z_4)} \sigma(Y_i)} = \frac{\sigma(Y_3)}{\sigma(Y_1) + \sigma(Y_2) + \sigma(Y_3)} = 0.24$.

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 17.

³By employing the approach presented in Chapter 3, 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.

Definition 17. Let $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$ and $A_{l_{i+1}} = \{Y_1, Y_2, \dots, Y_{n_{l_{i+1}}}\}$, $1 \leq i \leq k$, be any two adjacent layers. $A_{l_{i+1}}$ depends on A_{l_i} to a degree

$$k_{A_{l_i}}(A_{l_{i+1}}) = \frac{\sum_{l=1}^{n_{l_{i+1}}} \text{card}(A_{l_i}(Y_l))}{\text{card}(U)}.$$

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 [30]. 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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$ and $A_{l_{i+1}} = \{X_1, X_2, \dots, X_{n_{l_{i+1}}}\}$, $1 \leq i \leq k$, be any two adjacent layers. $A_{l_{i+1}}$ depends on A_{l_i} to a degree

$$k_{A_{l_i}}(A_{l_{i+1}}) = \sum_{\text{cer}(X_i, X_j)=1} \sigma(X_i).$$

Proof. From Definition 13, $\sum_{j=1}^{n_{l_{i+1}}} \text{card}(A_{l_i}(X_j)) = \sum_{j=1}^{n_{l_{i+1}}} \sum_{\text{cer}(X_i, Y_j)=1} \text{card}(X_i)$. Since $X_n \cap X_m = \emptyset$, $1 \leq n \neq m \leq n_{l_i}$, then $A_{l_i}(X_n) \cap A_{l_i}(X_m) = \emptyset$. Thus $\sum_{j=1}^n \text{card}(A_{l_i}(X_j)) = \sum_{\text{cer}(X_i, Y_j)=1} \text{card}(X_i)$. Since $\varphi(X_i) = \sigma(X_i)\varphi(G) = \sigma(X_i)\varphi(U)$ and by Definition 17, we can write $\gamma_B(D) = \sum_{\text{cer}(X_i, X_j)=1} \sigma(X_i)$. $\square$

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

$$k_{A_{l_j}}(A_{l_{j-1}}) = \sum_{\text{cov}(X_i, X_j)=1} \sigma(X_i).$$

Proof. It can be proved similarly as Property 4. $\square$

Example 5 (Cont.) Consider model of car and dealer in the flow graph G in Fig. 6.1. By Property 4, dealer depends on model of car to a degree $\gamma_X(Y) = \sum_{\text{cer}(X_i, Y_j)=1} \sigma(X_i) = \sigma(X_3) = 0.17$. 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 $\gamma_Z(Y) = \sum_{\text{cov}(Y_i, Z_j)=1} \sigma(Z_i) = \sigma(Z_3) = 0.21$. 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*.

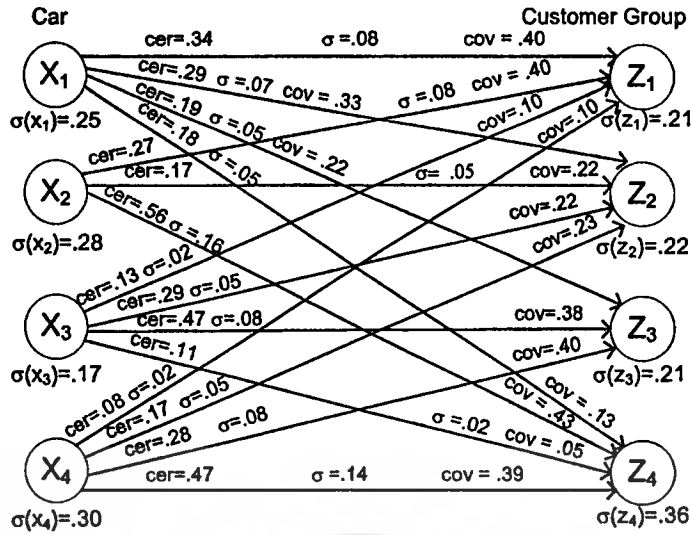


Figure 6.2 A combined flow graph.

Example 5 (Cont.) For model of car and customer group in Fig. 6.1, we give a combined flow graph in Fig. 6.2. By Definition 13, 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 14, 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*.

CHAPTER 7

CONCLUSION AND RECOMMENDATION

7.1 Conclusion

During the years 2002 - 2010, flow graphs applications became larger, but no more complex or systematic. Research in theoretical aspects of flow graphs, explored the complementary nature of their properties with other mathematical theories, are scarce. Our initial contributions accomplished in this thesis can be summarized below.

The analysis in Chapter 3 indicates that useful patterns and knowledge can be extracted from a flow graph in association rule format. We establish two new propositions that reveal the relationship between flow graphs and association rules.

From computational viewpoint, representing flow graph in matrix, array and the like generalizes flow graph to many advancing research. In Chapter 4, we represent Pawlak's graphical flow graphs in matrix forms which allow more flexible computation which is a significant contribution. The matrix form can simplify computation, improve time complexity (an algorithm using our matrix representation is polynomial in l , where l is a number of layers of a flow graph), alleviate problems due to missing coefficients and hence help to widen the applications of flow graphs. We also derive formulas to calculate coefficients by using properties of matrix and using flow graph directly. In addition, we introduce a flow graph composition from heterogeneous data sources and its formulas to calculate coefficients.

In Chapter 5, an alternative approach to construct flow graphs, decision rules and performs data analysis for decision table with fuzzy attributes are formulated.

In Chapter 6, 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.

In the end of Chapters 3 - 5, we initially demonstrate the utility of flow graphs to real-world data collected from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN). The obtained mathematical properties and formulas can be employed to analyze this data successfully. The preliminary results illustrate a desirable outcome as reported.

7.2 Recommendations

In our point of view, our future works are derivation of mathematical formulas, properties and convenient expression in matrix forms for inverse flow graphs and formulas for missing value coefficients. Another challenge is to formulate theoretical properties of flow graphs in graph theory aspect. In addition, idea to relax concept of mutual exclusion [12] might yield less restrictive analysis of flow graph but more valuable results. Last but not least important, we should also pay attention to logically flow graphs by studying relationship between inference rules and decision rules in the context of decision algorithm to induce patterns within data.



REFERENCES

- [1] Bowes J., Neufeld E., Greer J. E. and Cooke J. 2000. "A comparison of association rule discovery and bayesian network causal inference algorithms to discover relationships in discrete Data." 326 – 336. **Proc. of the 13th Biennial Conference of the Canadian Society on Computational Studies of Intelligence.** Canada: Quebec.
- [2] Butz C., Yan J., W. and Yang B. 2006. "An Efficient Algorithm for Inference in Rough Set Flow Graphs." Transaction on Rough Sets V, in: **Lect. Notes. Artif. Int. Springer-Verlag, Berlin.** 4100: 102 – 122.
- [3] Chan C. and Tsumoto S. 2007. "On Learning Decision Rules from Flow Graphs." 655 – 658. **Proc. of North American Fuzzy Information Processing Society Society, NAFIPS '07.** Annual Meeting of the North American.
- [4] Chitcharoen D. and Pattaraintakorn P. 2008. "Knowledge Discovery by Rough Sets Mathematical Flow Graphs and its Extension." 340 – 345. **Proc. of the IASTED International Conference on Artificial Intelligence and Applications.** Austria: Innsbruck.
- [5] Chitcharoen D. and Pattaraintakorn P. 2008. "Towards Theories of Fuzzy Set and Rough Set to Flow Graphs." 1675 – 1682. **Proc. of the 2008 IEEE World Congress on Computational Intelligence.** Hong Kong.
- [6] Chitcharoen D. and Pattaraintakorn P. 2010. "An Extension of Rough Set Approximation to Flow Graph Based Data Analysis." 418 – 427. **Proc. of the Seventh International Conference on Rough Sets and Current Trends in Computing (RSCTC 2010).** Poland: Warsaw.
- [7] Chitcharoen D. and Pattaraintakorn P. 2010. "Novel Matrix Forms of Flow Graphs with Applications to Data Integration." **Comput. Math. Appl.** 60(10), 2880 – 2897.
- [8] Chitcharoen D. and Pattaraintakorn P. Inverse Flow Graphs and their Application to Data Analysis, (submitted).
- [9] Corman T., Leiserson C., Rivest R., Stein C. Introduction to Algorithms, Second Edition. **Press and McGraw-Hill,** 2001.

- [10] Czyzewski A. and Kostek. B. 2004. "Musical Metadata Retrieval with Flow Graphs." in: S. Tsumoto et al. (Eds), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 3066: 691 – 698.
- [11] Ford L. R. and Fulkerson D. R. 1962. "Flows in Networks." **Princeton University Press, Princeton**. New Jersey.
- [12] Greco S., Pawlak Z. and Slowinski R. 2002. "Generalized Decision Algorithms, Rough Inference Rules, and Flow Graphs." in: J. J. Alpigini et al. (Eds), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 2475: 93 – 104.
- [13] Huawen L., Jigui, S. and Huijie, Z, 2008. "Interpretation of Extended Pawlak Flow Graphs Using Granular Computing." in: J. F. Peters and A. Skowron (Eds.), *Transactions on Rough Sets VIII*, in: *Lect. Notes Comp. Sci.* **Springer-Verlag, Berlin** 5084: 93 – 115.
- [14] Jigui S., Huawen L. and Huijie Z. 2006. "An Extension of Pawlak's Flow Graphs." in: G. Wang (Eds), *Rough Sets and Knowledge Technology*, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 4062: 191 – 199.
- [15] Lenzerini, M., 2002. "Data Integration: A Theoretical Perspective." 233 – 246. **Proc. of the 21th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems**. USA: Madison, Winsconsin.
- [16] Matusiewicz Z. and Pancerz K. 2008. "Rough Set Flow Graphs and Max -* Fuzzy Relation Equations in State Prediction Problems." in: C. C. Chan et al. (Eds), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 5306: 359 – 368.
- [17] Mieszkowicz-Rolka A and Rolka L. 2004. "Fuzzy Implication Operators in Variable Precision Fuzzy Rough Sets Model." Rutkowski, L. (Eds.) *Proc. 7th International Conference Artificial Intelligence and Soft Computing*, Zakopane, Poland, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 3037: 498 – 503.
- [18] Mieszkowicz-Rolka A and Rolka L. 2006. "Flow Graph and Decision Tables with Fuzzy Attributes." *Proc. of the 8th International Conference Artificial Intelligence and Soft Computing*, Zakopane, Poland, in: *Lect. Notes. Artif. Int.*, vol 4029, **Springer-Verlag, Berlin**. 268 – 277.
- [19] Pancerz K. and Matusiewicz Z. 2010. "Prediction with Temporal Rough Set Flow Graphs: the Eigen Fuzzy Sets Perspective." 744 – 748. **Proc. of the 2010 IEEE Human System Interactions (HSI)**. Russian.

- [20] Pattaraintakorn P., Cercone N. and Naruedomkul K. 2006. "Rule learning: ordinal prediction based on rough set and soft-computing." **Appl. Math. Lett.** 19 (12) 1300 – 1307.
- [21] Pattaraintakorn P., Cercone N. and Naruedomkul K. 2006 "Rule Analysis with Rough Sets Theory." 582 – 585. **Proc. of the 2006 IEEE International Conference on Granular Computing**. USA: Atlanta.
- [22] Pattaraintakorn P., 2009. "Entropy Measures of Flow Graphs with Applications to Decision Trees." in: P. Wen et al. (Eds.), *Rough Sets and Knowledge Technology*, in: *Lect. Notes Comp. Sci.*, vol 5589, **Springer-Verlag, Berlin**. 618 – 625.
- [23] Pawlak Z. 1982. "Rough sets." **Int. J. Inf. Comput. Sci.** 11 (5): 341 – 356.
- [24] Pawlak Z. 1991. "Rough Sets: Theoretical Aspects of Reasoning about Data." **Kluwer Academic Publishers**. Dordrecht.
- [25] Pawlak Z. 2002. "Rough Sets, Decision Algorithms and Bayes' Theorem." **Eur. J. of Oper. Res.** 136: 181 – 189.
- [26] Pawlak Z. 2003. "Probability, Truth and Flow Graphs." **Proc. of Rough Sets in Knowledge Discovery and Soft Computing**. 1 – 9. Poland: Warsaw.
- [27] Pawlak Z. 2003 "Flow Graphs and Decision Algorithms." in: G. Wang et al. (Eds.), *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 2639: 1 – 10.
- [28] Pawlak Z. 2003. "Decision Algorithms and Flow Graphs: A Rough Set Approach." **J. of Telecom. and Inform. Tech.** 3: 98 – 101.
- [29] Pawlak Z. 2005. "Rough Set and Flow Graphs" in: D. Slezak et al. (Eds.), *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 3641: 1 – 11.
- [30] Pawlak Z. 2005. "Flow Graphs and Data Mining" in: J.F. Peters, A. Skowron (Eds.), *Transactions on Rough Sets III*, in: *Lect. Notes Comp. Sci.* **Springer-Verlag, Berlin**. 3400: 1 – 36.
- [31] Pawlak Z. 2006. "Decision Trees and Flow Graphs, in: S. Greco (Eds.), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.* **Springer-Verlag, Berlin**. 4259: 1 – 11.
- [32] Pawlak, Z. 2007. "Rudiments of rough set" **Inf. Sci.** 177(1): 3 – 27.
- [33] Pawlak, Z. 2007. "Rough sets: Some extensions" **Inf. Sci.** 177(1): 28 – 40.

- [34] Pawlak, Z. 2007. "Rough sets and Boolean reasoning" **Inf. Sci.** 177(1): 41 – 73.
- [35] Poole D. 2006. "Linear Algebra" **A model introduction**. Sccond edition, Thomson.
- [36] Srikant R. and Agrawal R. 1995. "Mining generalized association rules." 407 – 419. **Proc. of the 21st International Conference on Very Large Data Bases**, Swizerland: Zurich.
- [37] Sun J., Liu H. and Zhang H. 2006. "An Extension of Pawlak's Flow Graphs." 191 – 199. **Proc. of the 1st International Conference on Rough Sets and Knowledge Technology** Chongqing.
- [38] Suraj Z. and Pancercz K. 2007. "Flow Graphs as a tool for Mining Prediction Rule of Changes of Components in: temporal Information Systems." in: J. T Yao (Eds.), **Verlag, Berlin**. 2475: 468 – 475.
- [39] Sun J., Liu H., Qi C. and Zhang H. 2006. "An Interpretation of Flow Graphs by Granular Computing." in: S. Greco (Eds.), **Rough Sets and Current Trends in Computing**, in: Lect. Notes Comp. Sci. **Springer-Verlag, Berlin**. 4259: 448 – 457.
- [40] Ou Yang Y. P., Tzeng G. H., Yen L. and Chan C. C. 2009. "Combined rough sets with flow graph and formal concept analysis for business aviation decision-making." **J. Intell. Inf. Syst.** DOI 10.1007/s10844-009-0110-y
- [41] Zadeh, L.A. 1965. "Fuzzy set." **Information and Control**, 8: 338 – 353.

APPENDIX A

DATA SET USED IN CHAPTER 3

A data set in Chapter 3 contains 75 records collected from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN). The complete list of this POSN data set is tabulated below.

Table A.1 A portion of POSN decision table used in Chapter 3

	<i>pref</i>	<i>leastpref</i>	<i>pra</i>	<i>leastpra</i>	<i>total_score</i>
1	Geo	Comb	I.E.	Comb	195
2	Geo	Comb	Alg	Comb	180
3	Geo	Comb	Geo	Alg	178
4	Geo	I.E.	Geo	Comb	162
5	Comb	I.E.	Comb	I.E.	149
6	Alg	I.E.	Comb	Num	146
7	Comb	I.E.	Comb	I.E.	146
8	Alg	I.E.	Comb	I.E.	145
9	Alg	Num	Num	Alg	136
10	Num	Geo	Num	Alg	133
11	Comb	I.E.	Num	Alg	129
12	Comb	I.E.	Comb	I.E.	128
13	I.E.	Comb	I.E.	Comb	123
14	Alg	I.E.	Num	Comb	121
15	Geo	Comb	Geo	Num	119
16	Alg	Num	Comb	Geo	116
17	Comb	I.E.	Comb	I.E.	115
18	Num	Geo	Alg	Geo	111
19	I.E.	Comb	I.E.	Geo	110
20	Num	Geo	Num	Alg	102
21	Geo	Comb	Geo	Comb	102

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Table A.1 (Cont.) A portion of POSN decision table used in Chapter 3

	<i>pref</i>	<i>leastpref</i>	<i>pra</i>	<i>leastpra</i>	<i>total_score</i>
22	I.E.	Comb	I.E.	Comb	101
23	Geo	I.E.	Geo	I.E.	100
24	Comb	I.E.	Alg	I.E.	100
25	Alg	I.E.	Num	Num	99
26	Alg	Geo	Alg	I.E.	95
27	Num	I.E.	Geo	I.E.	94
28	Geo	I.E.	Num	I.E.	94
29	Comb	I.E.	Comb	I.E.	94
30	Comb	Geo	Comb	I.E.	91
31	Alg	I.E.	Alg	I.E.	91
32	Alg	I.E.	Num	I.E.	90
33	Num	I.E.	Num	I.E.	90
34	I.E.	Geo	Comb	I.E.	88
35	Geo	I.E.	Geo	Num	88
36	Geo	Num	I.E.	Num	88
37	Comb	I.E.	Alg	I.E.	87
38	Num	Comb	Geo	Comb	85
39	Alg	I.E.	Alg	I.E.	81
40	Alg	I.E.	Alg	I.E.	79
41	Alg	Comb	Num	I.E.	78
42	Num	Comb	Alg	Geo	77
43	Comb	I.E.	Num	I.E.	76
44	Comb	I.E.	Comb	I.E.	75
45	Geo	I.E.	Geo	I.E.	74
46	Comb	I.E.	Comb	Alg	74
47	Alg	I.E.	Alg	Geo	74
48	Comb	I.E.	Comb	I.E.	74
49	Alg	I.E.	Alg	I.E.	73
50	Num	Comb	Num	Comb	73

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Table A.1 (Cont.) A portion of POSN decision table used in Chapter 3

	<i>pref</i>	<i>leastpref</i>	<i>pra</i>	<i>leastpra</i>	<i>total_score</i>
51	Geo	I.E.	Geo	I.E.	73
52	Geo	Alg	Geo	I.E.	73
53	Num	I.E.	Num	I.E.	71
54	Comb	I.E.	Comb	Alg	71
55	Alg	I.E.	Alg	I.E.	70
56	Alg	I.E.	Alg	Geo	70
57	Comb	Num	Alg	I.E.	69
58	Alg	I.E.	Alg	I.E.	69
59	Comb	Geo	Comb	Geo	67
60	Num	I.E.	Num	Alg	65
61	Geo	I.E.	Comb	Num	64
62	Alg	I.E.	Alg	I.E.	63
63	Num	I.E.	Geo	Num	63
64	Alg	I.E.	Num	Geo	63
65	Num	I.E.	Num	I.E.	61
66	Comb	I.E.	Comb	Geo	61
67	Comb	I.E.	Geo	I.E.	60
68	Alg	I.E.	Comb	Geo	60
69	Num	Geo	Num	Geo	59
70	Num	I.E.	Num	I.E.	59
71	Alg	I.E.	Alg	I.E.	59
72	Alg	Comb	I.E.	Geo	58
73	Geo	Comb	Alg	I.E.	58
74	Comb	I.E.	Comb	I.E.	58
75	Geo	I.E.	Alg	I.E.	58

APPENDIX B

DATA SET USED IN CHAPTER 5

A data set in Chapter 5 contains 75 records collected from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN). The complete list of this POSN data set is tabulated below.

Table B.1 A portion of POSN decision table used in Chapter 5

	<i>pref</i>	<i>pra</i>	<i>score_Geo</i>	<i>score_Num</i>	<i>score_Alg</i>	<i>score_Comb</i>	<i>score_I.E.</i>	<i>total_score</i>
1	Geo	I.E.	30	45	42	33	45	195
2	Geo	Alg	29	34	38	34	45	180
3	Geo	Geo	40	32	46	25	35	178
4	Geo	Geo	18	25	31	42	46	162
5	Comb	Comb	23	28	33	42	23	149
6	Alg	Comb	25	24	43	33	21	146
7	Comb	Comb	28	20	38	43	17	146
8	Alg	Comb	22	23	44	16	40	145
9	Alg	Num	23	20	41	30	22	136
10	Num	Num	10	17	26	37	43	133
11	Comb	Num	19	14	44	29	23	129
12	Comb	Comb	19	27	35	28	19	128
13	I.E.	I.E.	26	5	36	34	22	123
14	Alg	Num	16	20	37	33	15	121
15	Geo	Geo	25	12	36	24	22	119
16	Alg	Comb	15	11	44	22	24	116
17	Comb	Comb	22	10	38	36	9	115
18	Num	Alg	5	21	40	26	19	111
19	I.E.	I.E.	21	10	29	21	29	110
20	Num	Num	7	11	37	40	7	102
21	Geo	Geo	16	9	40	30	7	102

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Table B.1 (Cont.) A portion of POSN decision table used in Chapter 5

	<i>pref</i>	<i>pra</i>	<i>score_Geo</i>	<i>score_Num</i>	<i>score_Alg</i>	<i>score_Comb</i>	<i>score_I.E.</i>	<i>total_score</i>
22	I.E.	I.E.	14	8	35	22	22	101
23	Geo	Geo	25	10	37	21	7	100
24	Comb	Alg	16	15	40	23	6	100
25	Alg	Num	7	10	37	31	14	99
26	Alg	Alg	10	10	37	24	14	95
27	Num	Geo	14	3	34	33	10	94
28	Geo	Num	26	3	38	21	6	94
29	Comb	Comb	15	12	32	26	9	94
30	Comb	Comb	8	6	39	36	2	91
31	Alg	Alg	16	4	42	19	10	91
32	Alg	Num	15	6	39	22	8	90
33	Num	Num	15	15	32	21	7	90
34	I.E.	Comb	13	10	38	14	13	88
35	Geo	Geo	15	12	34	24	3	88
36	Geo	I.E.	12	11	31	27	7	88
37	Comb	Alg	13	12	27	25	10	87
38	Num	Geo	8	1	31	17	28	85
39	Alg	Alg	10	16	31	19	5	81
40	Alg	Alg	24	3	32	9	11	79
41	Alg	Num	5	1	40	20	12	78
42	Num	Alg	18	17	30	7	5	77
43	Comb	Num	6	11	23	28	8	76
44	Comb	Comb	7	11	35	17	5	75
45	Geo	Geo	10	2	37	19	6	74
46	Comb	Comb	7	10	31	20	6	74
47	Alg	Alg	10	1	30	30	3	74
48	Comb	Comb	14	12	28	18	2	74
49	Alg	Alg	11	0	36	18	8	73
50	Num	Num	7	8	33	18	7	73

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Table B.1 (Cont.) A portion of POSN decision table used in Chapter 5

	<i>pref</i>	<i>pra</i>	<i>score_Geo</i>	<i>score_Num</i>	<i>score_Alg</i>	<i>score_Comb</i>	<i>score_I.E.</i>	<i>total_score</i>
51	Geo	Geo	13	10	29	12	9	73
52	Geo	Geo	8	20	29	13	3	73
53	Num	Num	10	8	37	16	0	71
54	Comb	Comb	3	16	31	16	5	71
55	Alg	Alg	15	8	27	14	6	70
56	Alg	Alg	11	4	28	26	1	70
57	Comb	Alg	10	16	32	3	8	69
58	Alg	Alg	9	10	32	16	2	69
59	Comb	Comb	4	13	26	17	7	67
60	Num	Num	5	5	32	22	1	65
61	Geo	Comb	6	10	33	13	2	64
62	Alg	Alg	8	9	27	15	4	63
63	Num	Geo	12	2	32	16	1	63
64	Alg	Num	6	3	26	24	4	63
65	Num	Num	18	1	29	6	7	61
66	Comb	Comb	5	2	26	26	2	61
67	Comb	Geo	9	1	30	13	7	60
68	Alg	Comb	3	2	30	19	6	60
69	Num	Num	5	2	30	18	4	59
70	Num	Num	12	2	27	11	7	59
71	Alg	Alg	8	2	40	4	5	59
72	Alg	I.E.	5	1	25	20	7	58
73	Geo	Alg	11	3	28	11	5	58
74	Comb	Comb	9	7	20	16	6	58
75	Geo	Alg	3	1	37	16	1	58

APPENDIX C

LIST OF PUBLICATIONS

1. Chitcharoen D. and Pattaraintakorn P. 2008. "Knowledge Discovery by Rough Sets Mathematical Flow Graphs and its Extension." 340 – 345. **Proc. of the IASTED International Conference on Artificial Intelligence and Applications**. Austria: Innsbruck.
2. Chitcharoen D. and Pattaraintakorn P. 2008. "Towards Theories of Fuzzy Set and Rough Set to Flow Graphs." 1675 – 1682. **Proc. of the 2008 IEEE World Congress on Computational Intelligence**. Hong Kong.
3. Chitcharoen D. and Pattaraintakorn P. 2010 "An Extension of Rough Set Approximation to Flow Graph Based Data Analysis." 418 – 427. **Proc. of the Seventh International Conference on Rough Sets and Current Trends in Computing (RSCTC 2010)**. Poland: Warsaw.
4. Chitcharoen D. and Pattaraintakorn P. 2010. "Novel Matrix Forms of Flow Graphs with Applications to Data Integration." **Comput. Math. Appl** 60 (10), 2880 – 2897.

KNOWLEDGE DISCOVERY BY ROUGH SETS MATHEMATICAL FLOW GRAPHS AND ITS EXTENSION

Doungnat Chitcharoen

Department of Mathematics and Computer Science,
 Faculty of Science, King Mongkut's Institute of
 Technology Ladkrabang, Bangkok, Thailand 10520
 email: s0067111@kmitl.ac.th

Puntip Pattaraintakorn

Department of Mathematics and Computer Science,
 Faculty of Science, King Mongkut's Institute of
 Technology Ladkrabang, Bangkok, Thailand 10520
 email: kppuntip@kmitl.ac.th

ABSTRACT

Mathematical rough set theory has attracted both practical and theoretical researchers. A significant extension of rough set theory is called *flow graphs*. It is a knowledge representation in the form of information flow. Flow graph is a promising approach to analyze data flow, decision trees, decision rules, probability learning, etc. In this article, we present their connections to pertinent techniques and propose a new extension to association rules. Two new propositions are used to reveal the relationship between flow graphs and association rules. We conduct experiment on real-world data collected from POSN with the evaluation. We discuss some important properties of flow graphs, with examples throughout.

KEY WORDS

Flow graphs, rough set theory and association rules.

1 Introduction

Rough set theory was introduced by Zdzislaw Pawlak in the early 1980's [1]. The crux of rough sets is the novel use of approximations to cope with uncertainty. A significant extension of rough set theory is called *flow graphs*, such that the union of all inputs x of y are interpreted as the *upper approximation* of y and the union of all inputs x of y with $cer(x, y) \equiv 1$ are interpreted as the *lower approximation* of y [2]. Pawlak's flow graphs are different to *flow networks* introduced by Ford et al. that do not analyze optimal flow [3].

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, while the entire flow graph describes a decision algorithm [4]. With every decision rule, there are three associated coefficients: *strength*, *certainty* and *coverage*. These coefficients satisfy Bayes' theorem. We can discover dependency and correlation within a data set without reference to its probabilistic nature by using flow graphs. In [5], the authors attached objects to the flow graphs' nodes. Hence, more precise relationships among nodes were considered. They also proposed the extended flow graphs and algorithms to handle redundancy and consistency. A decision tree can be constructed as a unique flow graph by removing the root while its nodes

are labeled by the same attribute [6]. The author concluded that such a flow graph provides better structure of data compared to a decision tree [6]. The mathematical model of flow graph in the decision algorithm can build an efficient search engines for the CDDB database [7]. Flow graph is a useful approach to discover hidden knowledge in the data for artificial intelligence and soft computing (e.g., preference analysis [2], conflict analysis [8], rule analysis [9], propositional calculus [10], data mining [11, 6] and granular computing [5] were analyzed by flow graphs efficiently). For these reasons, we propose data analysis based on Pawlak's flow graphs and extend them to handle association rule extraction.

This paper is organized as follows. Sect. 2 presents basic concept of rough sets. In Sect. 3, we introduce preliminary definitions and properties of flow graphs followed by two new propositions which reveal of connection flow graphs and association rule. Sect. 4 contains an example on data analysis. The actual data set from POSN, experimental results and their evaluation are presented in Sect. 5. Finally we conclude in the last section.

2 Rough Set Theory

The following rough sets preliminary is taken from the study of Pawlak [1]. This mathematical theory formed the basis of flow graphs later in the early 2000s [2, 4, 6, 10, 11, 12]. Rough sets are based on an *information system* (data table). 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 *B-gramules*, for more granular computing studies please refer to [2]). If we distinguish an information system to two disjoint classes of attributes, called *condition* (C) and *decision attributes* (D), then the information system will be called a *decision system*.

Suppose we are given an information system $S =$

(U, A) , $X \subseteq U$, and $B \subseteq A$. We can approximate X using the information in B . We define two operations assigning to every $X \subseteq U$ two sets $B_*(X)$ and $B^*(X)$ called *B-lower* and *B-upper approximations* of X , respectively, and defined as follows:

$$B_*(X) = \bigcup_{x \in U} \{B(x) \mid B(x) \in X\}, \quad (1)$$

$$B^*(X) = \bigcup_{x \in U} \{B(x) \mid B(x) \cap X \neq \emptyset\}. \quad (2)$$

In addition, the set

$$BN_B(X) = B^*(X) - B_*(X) \quad (3)$$

will be referred to as *B-boundary region* of X . 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*. More studies involving rough sets are in [2, 4, 6, 8, 10, 11, 12].

3 Flow Graphs

In this section we present basic definitions of flow graphs from the studies of Pawlak [2, 4, 6, 10, 11, 12].

3.1 Normalized Flow Graphs

A flow graph is an alternative representation of the knowledge, more specifically the information flow, in the given data. Several studies on traditional flow graphs were in [2, 4, 6, 10, 11, 12]. 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 $\varphi: B \rightarrow R^+$ is a flow function, $\varphi(G)$ is a *throughflow* of flow graph G and $\sigma: B \rightarrow [0, 1]$ is a *normalized flow* of (x, y) . The *strength* of (x, y) is given by

$$\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)} \quad (4)$$

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 , associated *normalized inflow* and *outflow* are defined respectively as:

$$\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(y, x), \quad (5)$$

$$\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x, y). \quad (6)$$

For any internal node x , it holds that $\sigma_+(x) = \sigma_-(x) = \sigma(x)$, where $\sigma(x)$ is a *normalized throughflow* of x . Similarly, *normalized inflow* and *outflow* for the flow graph G are defined as:

$$\sigma_+(G) = \frac{\varphi_+(G)}{\varphi(G)} = \sum_{x \in I(G)} \sigma_-(x), \quad (7)$$

$$\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(G)} \sigma_+(x). \quad (8)$$

It also holds that $\sigma_+(G) = \sigma_-(G) = \sigma(G) = 1$. With every branch (x, y) of a flow graph G , the *certainty* and the *coverage* of (x, y) are defined respectively as:

$$cer(x, y) = \frac{\sigma(x, y)}{\sigma(x)}, \quad (9)$$

$$cov(x, y) = \frac{\sigma(x, y)}{\sigma(y)}, \quad (10)$$

where $\sigma(x) \neq 0$ and $\sigma(y) \neq 0$. In accordance with the previous definitions, the following are some consequence properties:

$$\sum_{y \in O(x)} cer(x, y) = 1 \text{ and } \sum_{y \in I(x)} cov(x, y) = 1, \quad (11)$$

$$\sigma(x) = \sum_{y \in O(x)} cer(x, y) \sigma(y) = \sum_{y \in I(x)} \sigma(y), \quad (12)$$

$$\sigma(y) = \sum_{x \in I(y)} cov(x, y) \sigma(x) = \sum_{x \in O(y)} \sigma(x), \quad (13)$$

$$cer(x, y) = \frac{cov(x, y) \sigma(y)}{\sigma(x)} \quad (14)$$

$$cov(x, y) = \frac{cer(x, y) \sigma(x)}{\sigma(y)}. \quad (15)$$

Equations (11)-(13) have some probabilistic relations. (11) have the form of a total probability theorem, while (12)-(13) are Bayes' rules [4, 6, 10, 11, 12].

3.2 Paths and Connections

A (directed) *path* from x to y ($x \neq y$) in G , denoted by $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ 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 $[x_1 \dots x_n]$ are defined respectively as:

$$cer[x_1 \dots x_n] = \prod_{i=1}^{n-1} cer(x_i, x_{i+1}), \quad (16)$$

$$cov[x_1 \dots x_n] = \prod_{i=1}^{n-1} cov(x_i, x_{i+1}) \quad (17)$$

$$\sigma[x \dots y] = \sigma(x) cer[x \dots y] = \sigma(y) cov[x \dots y]. \quad (18)$$

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 nodes x and y . For every connection $\langle x, y \rangle$, its associate *certainty*, *coverage* and *strength* of the connection $\langle x, y \rangle$ are defined as:

$$cer\langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} cer[x \dots y], \quad (19)$$

$$cov\langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} cov[x \dots y], \quad (20)$$

$$\begin{aligned} \sigma\langle x, y \rangle &= \sum_{[x \dots y] \in \langle x, y \rangle} \sigma[x \dots y] \\ &= \sigma(x)cer\langle x, y \rangle \\ &= \sigma(y)cov\langle x, y \rangle. \end{aligned} \quad (21)$$

If $[x \dots y]$ is a path such that x and y are input and output of the graph G , respectively, then $[x \dots y]$ will be referred to as *complete*. For simplicity, from now we 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 $\langle x, y \rangle$ such that $\sigma\langle x, y \rangle = \sigma(x, y)$, $cer\langle x, y \rangle = cer(x, y)$ and $cov\langle x, y \rangle = cov(x, y)$ 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.

3.3 Dependencies

In this section, we show that the dependency factor of flow graphs has a close relationship to statistical measurements. For $\langle x, y \rangle \in B$, nodes x and y are *independent* of each other if

$$\sigma\langle x, y \rangle = \sigma(x)\sigma(y). \quad (22)$$

Thus, we have

$$\frac{\sigma\langle x, y \rangle}{\sigma(x)} = cer\langle x, y \rangle = \sigma(y). \quad (23)$$

Moreover, if

$$cer\langle x, y \rangle > \sigma(y) \quad \text{or} \quad cov\langle x, y \rangle > \sigma(x), \quad (24)$$

then x and y *depend positively* on each other. On the other hand, if

$$cer\langle x, y \rangle < \sigma(y) \quad \text{or} \quad cov\langle x, y \rangle < \sigma(x), \quad (25)$$

then x and y *depend negatively* on each other.

Furthermore, there is a factor that expresses the degree of dependency in the flow graph. For every $\langle x, y \rangle \in B$ we define a *dependency factor*, $\eta(x, y)$, as

$$\eta(x, y) = \frac{cer\langle x, y \rangle - \sigma(y)}{cer\langle x, y \rangle + \sigma(y)} = \frac{cov\langle x, y \rangle - \sigma(x)}{cov\langle x, y \rangle + \sigma(x)}. \quad (26)$$

The interpretation is: if $\eta(x, y) = 0$, then x and y are independent on each other, if $-1 < \eta(x, y) < 0$, then x and y are negatively dependent on each other and if $0 < \eta(x, y) < 1$ then x and y are positively dependent on each other.

3.4 Decision Algorithm

In this section, we present flow graphs that can be interpreted as a *decision algorithm*, taken from [4, 11].

Let us assume that the set of nodes of a graph is interpreted as a set of *formulas*, denoted by Φ, Ψ . The formulas can be viewed as *propositional functions* in logic [4]. Then every branch $\langle \Phi, \Psi \rangle$ can be understood as a *decision rule*, $\Phi \rightarrow \Psi$, Φ will be referred to as a *condition*, whereas Ψ is the *decision*. Such a rule is characterized by three numbers, $\sigma(\Phi, \Psi)$, $cer(\Phi, \Psi)$ and $cov(\Phi, \Psi)$. Thus, every path $[\Phi_1 \dots \Phi_n]$ determines a sequence of decision rules $\Phi_1 \rightarrow \Phi_2, \Phi_2 \rightarrow \Phi_3, \dots, \Phi_{n-1} \rightarrow \Phi_n$. As shown in [4], this sequence of decision rules can be interpreted as a single decision rule $\Phi_1 \Phi_2 \dots \Phi_{n-1} \rightarrow \Phi_n$, in short $\Phi^* \rightarrow \Phi_n$, where $\Phi^* = \Phi_1 \wedge \Phi_2 \wedge \dots \wedge \Phi_{n-1}$, characterized by three coefficients:

$$cer(\Phi^*, \Phi_n) = cer[\Phi_1 \dots \Phi_n], \quad (27)$$

$$cov(\Phi^*, \Phi_n) = cov[\Phi_1 \dots \Phi_n], \quad (28)$$

$$\begin{aligned} \sigma(\Phi^*, \Phi_n) &= \sigma(\Phi_1)cer[\Phi_1 \dots \Phi_n] \\ &= \sigma(\Phi_n)cov[\Phi_1 \dots \Phi_n] \end{aligned} \quad (29)$$

where $\sigma(\Phi)$ can be considered as the truth value of the formula Φ and $\sigma(\Phi, \Psi)$ is the strength of the decision rule $\Phi \rightarrow \Psi$. Similarly, every *connection* $\langle \Phi, \Psi \rangle$ can be interpreted as a single decision rule $\Phi \rightarrow \Psi$ such that

$$cer\langle \Phi, \Psi \rangle = cer(\Phi, \Psi), \quad (30)$$

$$cov\langle \Phi, \Psi \rangle = cov(\Phi, \Psi), \quad (31)$$

$$\sigma\langle \Phi, \Psi \rangle = \sigma(\Phi)cer\langle \Phi, \Psi \rangle = \sigma(\Psi)cov\langle \Phi, \Psi \rangle. \quad (32)$$

The set of all decision rules associated with all complete paths will be called a *decision algorithm* induced by the flow graph.

3.5 Association Rule

An *association rule* identifies frequently occurring patterns of information in a given set of data [13].

An association rule is an expression $X \rightarrow Y$. A pattern X can be an item X_i or a conjunctive item set $X_1 \wedge \dots \wedge X_j$ where $X_1, \dots, X_j$ are sets of data, $1 \leq i, j \leq n, i \neq j$. X is called the *antecedence*, while Y is called the *consequence* of the rule. A rule has two associated coefficients, called *support* and *confidence* [13]. The *support* (*sup*) of the rule $X \rightarrow Y$ is the ratio of the number of data satisfy X and Y to the number of the total data. The *confidence* (*conf*) of the rule is the ratio of the number of data which satisfy X and Y to the size of the set X . Moreover, *sup* and *conf* of the

3.5 Association Rule

An *association rule* identifies frequently occurring patterns of information in a given set of data [13]. An association rule is an expression $X \rightarrow Y$. A pattern X can be an item X_i or a conjunctive item set $X_i \wedge \dots \wedge X_j$ where $X_i, \dots, X_j$ are sets of data, $1 \leq i, j \leq n, i \neq j$. X is called the *antecedence*, while Y is called the *consequence* of the rule. A rule has two associated coefficients, called support and confidence [13]. The *support* (sup) of the rule $X \rightarrow Y$ is the ratio of the number of data satisfy X and Y to the number of the total data. The *confidence* ($conf$) of the rule is the ratio of the number of data which satisfy X and Y to the size of the set X . Moreover, sup and $conf$ of the rule $X \rightarrow Y$ satisfy $P(X, Y)$ and $P(X|Y)$, respectively [15]. One can see the following relations.

Remark 2a. $sup[xyz] = \frac{\varphi(xyz)}{\varphi(G)}$.

2b. $conf[xyz] = \frac{\varphi(xyz)}{\varphi(x \wedge y)}$.

We next come to our proposed propositions to reveal relationship between association rules and flow graphs.

Proposition 1. Let $G = (N, B, \sigma)$ be a flow graph. The strength of (x, y) is given by $\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)}$ where $0 \leq \sigma(x, y) \leq 1$. A path from x to y ($x \neq y$) in G , denoted by $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ such that $x_1 = x$ and $x_n = y$ and $(x_i, x_{i+1}) \in B$ for every $i, 1 \leq i \leq n-1$. For every path $[xyz]$, $cer(y, z) = 1$, an association rule: $(x \wedge y) \rightarrow z$, where x, y , are antecedences and z is a consequence has

$$sup[xyz] = \sigma[xyz] \quad (33)$$

where $0 \leq sup[xyz] \leq 1$.

Proof By Remark 2a, $sup[xyz] = \frac{\varphi(xyz)}{\varphi(G)}$. Form Remark 1 and (4), we get $sup[xyz] = \frac{\varphi(x, y)}{\varphi(G)} = \sigma(x, y)$, respectively. We next multiply by $\frac{\sigma(z)}{\sigma(x)}$ and use (9) to obtain $sup[xyz] = \frac{\sigma(z)\sigma(x, y)}{\sigma(x)} = \sigma(x)cer(x, y)$. If $cer(y, z) = 1$, then $sup[xyz] = \sigma(x)cer(x, y)cer(y, z)$. Finally, (16) and (18) give us $sup[xyz] = \sigma[xyz]$. Q.E.D.

Proposition 2. Let $G = (N, B, \sigma)$ be a flow graph. The strength of (x, y) is given by $\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)}$ where $0 < \sigma(x, y) \leq 1$. A path from x to y ($x \neq y$) in G , denoted by $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ such that $x_1 = x$ and $x_n = y$ and $(x_i, x_{i+1}) \in B$ for every $i, 1 \leq i \leq n-1$. For every path $[xyz]$, $cer(y, z) = 1$, an association rule: $(x \wedge y) \rightarrow z$, where x, y , are antecedences and z is a consequence has

$$conf[xyz] = 1 \quad (34)$$

where $0 \leq conf[xyz] \leq 1$.

Proof By Remarks 2b and 1, $conf[xyz] = \frac{\varphi(xyz)}{\varphi(x \wedge y)} = \frac{\varphi(x \wedge y)}{\varphi(x \wedge y)} = 1$. Q.E.D.

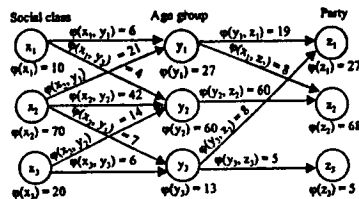


Figure 1. Flow graph initial data of voting problem.

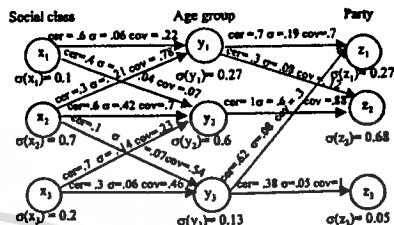


Figure 2. Social class, age and party votes flow graph.

4 Illustrative Example

In this section we illustrate the above ideas by means of a simple example (modification of the example given in [11]) concerning a voting analysis as depicted in Fig. 1. Suppose we have 3 disjoint age groups of voters y_1 (old), y_2 (middle aged) and y_3 (young) belonging to 3 social classes x_1 (high), x_2 (middle) and x_3 (low). The voters voted for 3 political parties z_1, z_2 and z_3 . In this population, e.g., certainties of input layer indicate that 10 % of voters has high social class, 70 % has middle social class and 20 % has low social class. Similarly, certainties of hidden layer, age group, and the output layer, party, have their values as shown in Fig. 2.

We can interpret its coverages that, e.g., party z_1 obtains 27 % of total votes, which 70 % is from old age group y_1 and 30 % is from young age group y_3 . One can see that the lower approximation of the party z_1 ($cer = 1$)

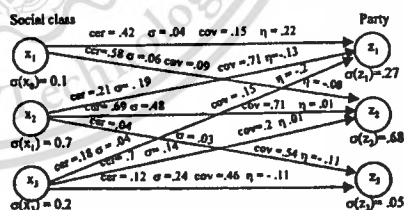


Figure 3. Combined flow graph.

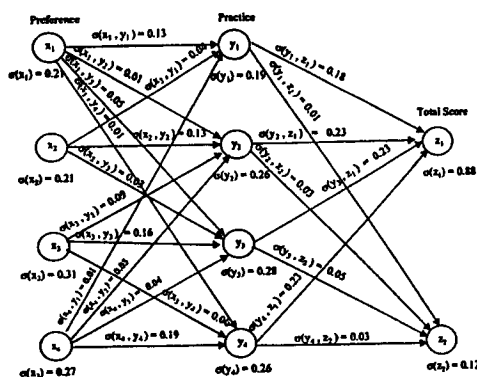


Figure 4. Flow graph of initial Olympiad mathematics' students score from POSN.

is an empty set and upper approximation of the party z_1 ($cer \neq 1$) is $(x_1 \cap y_1) \cup (x_2 \cap y_1) \cup (x_3 \cap y_1) \cup (x_4 \cap y_1)$. From Fig. 2, since $cer(y_2, z_2) = 1$, we obtain three association rules by using our presented propositions as:

$$\begin{aligned} (x_1 \wedge y_2) &\rightarrow z_2, \text{sup} = 0.04 \text{ and } \text{conf} = 1, \\ (x_2 \wedge y_2) &\rightarrow z_2, \text{sup} = 0.42 \text{ and } \text{conf} = 1, \\ (x_3 \wedge y_2) &\rightarrow z_2, \text{sup} = 0.14 \text{ and } \text{conf} = 1. \end{aligned}$$

Next, if we want to know how votes are distributed between parties with respect to social classes, we can compute the combined of the corresponding graph. Employing the algorithm presented in [10, 11], we obtain the results as in Fig. 3. For example, we can see that the party z_3 obtains 5 % of total vote, of which 54 % is from the social class x_2 and 46 % is from the social class x_3 . When we consider the dependency factor, the negative dependency between high social class and party z_2 is $\eta(x_1, z_2) = -0.08$ with strength $\sigma = 0.06$. The negative dependency reveals that when one node is increasing, its pair is decreasing as well.

5 Experimental Results

In this section, we describe experimental results from the scores of Olympiad mathematics' students' analysis. We depict an initial data flow graph in Fig. 4. It contains 68 records collected from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN). It consists of four disjoint preference subjects: x_1 (Geometry), x_2 (Number theory), x_3 (Algebra) and x_4 (Combinatorics) which belong to four most practiced subjects: y_1 (Geometry), y_2 (Number theory), y_3 (Algebra) and y_4 (Combinatorics). The decision attribute is the students' total score, z , that we will compare 4 discretization cases presented below. The thresholds for association rule extraction are $\text{sup} > 0.1$ and $\text{conf} > 0.35$.

Case 1. If z_1 (score ≥ 103), z_2 ($80 \leq \text{score} \leq 102$), z_3 ($70 \leq \text{score} \leq 79$) and z_4 (score ≤ 69), then the selected

association rule is:

IF Proc_Comb THEN score ≥ 103 , $\text{sup} = 0.10$, $\text{conf} = 0.39$ ¹.

with $\text{sup} = 0.10$, $\text{conf} = 0.39$ and $\text{accuracy} = 97.6\%$.

Case 2. If z_1 (score ≥ 147), z_2 ($118 \leq \text{score} \leq 146$), z_3 ($88 \leq \text{score} \leq 117$) and z_4 (score ≤ 87), then the selected association rules are:

IF Prefer_Alg and Proc_Alg THEN score ≥ 87 , $\text{sup} = 0.12$, $\text{conf} = 0.73$,
IF Prefer_Alg THEN score ≤ 87 , $\text{sup} = 0.13$, $\text{conf} = 0.64$,
IF Prefer_Comb THEN score ≤ 87 , $\text{sup} = 0.16$, $\text{conf} = 0.59$,
IF Prefer_Number THEN score ≤ 87 , $\text{sup} = 0.17$, $\text{conf} = 0.57$,
IF Proc_Alg THEN score ≤ 87 , $\text{sup} = 0.2$, $\text{conf} = 0.74$,

with average $\text{sup} = 0.16$ average $\text{conf} = 0.65$ and $\text{accuracy} = 86.76\%$.

Case 3. If z_1 (score ≥ 94), z_2 ($73 \leq \text{score} \leq 93$) and z_3 ($58 \leq \text{score} \leq 72$), then the selected association rules are:

IF Prefer_Comb THEN score ≥ 94 , $\text{sup} = 0.10$, $\text{conf} = 0.37$,
IF Proc_Alg THEN $73 \leq \text{score} \leq 93$, $\text{sup} = 0.10$, $\text{conf} = 0.37$,
IF Proc_Number THEN score ≥ 94 , $\text{sup} = 0.10$, $\text{conf} = 0.379$,
IF Prefer_Geo THEN score ≥ 94 , $\text{sup} = 0.10$, $\text{conf} = 0.5$,
IF Proc_Alg THEN $58 \leq \text{score} \leq 72$, $\text{sup} = 0.12$, $\text{conf} = 0.44$,
IF Proc_Comb THEN ≥ 94 , $\text{sup} = 0.12$, $\text{conf} = 0.42$,

with average $\text{sup} = 0.11$, average $\text{conf} = 0.42$ and $\text{accuracy} = 80.88\%$.

Case 4. If z_1 (score ≥ 61) and z_2 (score ≤ 60), then the selected association rules are:

IF Prefer_Geo and Proc_Geo THEN score ≥ 61 , $\text{sup} = 0.13$, $\text{conf} = 1$,
IF Prefer_Number and Proc_Number THEN score ≥ 61 , $\text{sup} = 0.13$, $\text{conf} = 0.78$,
IF Prefer_Alg and Proc_Alg THEN score ≥ 61 , $\text{sup} = 0.15$, $\text{conf} = 0.90$,
IF Prefer_Comb and Proc_Comb THEN score ≥ 61 , $\text{sup} = 0.18$, $\text{conf} = 0.92$,
IF Prefer_Geo THEN score ≥ 61 , $\text{sup} = 0.17$, $\text{conf} = 0.86$,
IF Prefer_Alg THEN score ≤ 60 , $\text{sup} = 0.17$, $\text{conf} = 0.88$,
IF Proc_Geo THEN score ≤ 60 , $\text{sup} = 0.18$, $\text{conf} = 0.92$,
IF Proc_Alg THEN score ≤ 60 , $\text{sup} = 0.23$, $\text{conf} = 0.84$,
IF Proc_Num THEN score ≤ 60 , $\text{sup} = 0.24$, $\text{conf} = 0.89$,
IF Proc_Comb THEN score ≤ 60 , $\text{sup} = 0.25$, $\text{conf} = 0.89$,
IF Prefer_Comb THEN score ≤ 60 , $\text{sup} = 0.28$, $\text{conf} = 0.90$,
IF Prefer_Alg THEN score ≤ 60 , $\text{sup} = 0.28$, $\text{conf} = 0.90$,

with average $\text{sup} = 0.19$, average $\text{conf} = 0.89$ and $\text{accuracy} = 51.47\%$.

From these selected association rules, we can distill knowledge from this flow graph: if students practice their preferred subjects, then they will have high scores. In Case 4, we obtain maximum average support and confidence as shown in Fig. 5. The highest average accuracy to predict students' score is 97.6 % from Case 1 as shown in Fig. 5.

6 Conclusion

In this paper, an extension of rough set theory, namely flow graphs, is considered. In addition surveying the research

¹Due to round off errors in computations, the equation may be not satisfied exactly in our examples.

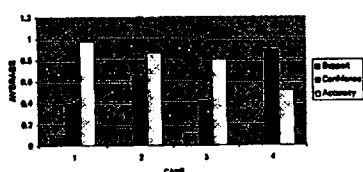


Figure 5. Comparison of *sup*, *conf* and *accuracy/100* from four cases of POSN data set.

in this area, we also performed an experiment on real-world data to demonstrate the utility of flow graphs with its evaluation. Further, we presented two new propositions that reveal the relationship between flow graphs and association rules.

The following are some possible extensions to enhance the utilities of flow graphs. (i) New representations of nodes, branches and their associate coefficients (e.g., *cer*, *cov*) are required to be able to interpret flow graphs easier. Dynamic approaches to update, delete or add some new information are open issues. (ii) In order to generate flow graphs and provide interactive results to user, algorithms and software implementation that consider scalability are necessary. (iii) Hybrid approaches of flow graphs and other pertinent techniques to increase the prediction ability are open issues.

Acknowledgements

This paper was supported by KMITL and NRCT grants. Thanks also to POSN, G.M. Zaverucha and anonymous reviewers.

References

- [1] Z. Pawlak, *Rough Sets. Theoretical aspects of reasoning about data*. (Dordrecht: Kluwer Academic Publishers, Dordrecht, 1991).
- [2] Z. Pawlak, Rough set and flow graphs, in D. Slezak et al. (Eds.) *Rough sets, fuzzy sets, data mining and granular computing*, *LNAI 3641* (Berlin: Springer-Verlag, 2005) 1-11.
- [3] L.R. Ford, D.R. Fulkerson, *Flows in networks*. (New Jersey: Princeton University Press, Princeton, 1962).
- [4] Z. Pawlak, Decision algorithms and flow graphs: a rough set approach, *J. of Telecom. and Inform. Tech.* 3, 2003, 98-101.
- [5] J. Sun, H. Liu, C. Qi, H. Zhang, An interpretation of flow graphs by granular computing, in S. Greco et al. (Eds.) *Rough Sets and Current Trends in Computing*, *LNCS 4259* (Berlin: Springer-Verlag, 2006) 448-457.
- [6] Z. Pawlak, Decision trees and flow graphs, in S. Greco et al. (Eds.) *Rough sets and current trends in computing*, *LNAI 4259* (Berlin: Springer-Verlag, 2006) 1-11.
- [7] B. Kostek, A. Czdrezej, Processing of musical meta-data employing Pawlak's flow graphs, in J.F Peters, A. Skowron et al. (Eds.) *Transactions on Rough Sets I*, *LNAI 3100* (Berlin: Springer-Verlag, 2004) 279-298.
- [8] Z. Pawlak, Some remarks on conflict analysis. *European J. of Oper. Res.* 166, 2005, 649-654.
- [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, Probability, truth and flow graphs, *Proc of rough sets in knowledge discovery and soft computing*, Warsaw, Poland, 2003, 1-9.
- [11] Z. Pawlak, Flow graphs and data mining, in J. F. Peters, A. Skowron et al. (Eds.) *Transactions on rough sets III*, *LNCS 3400* (Berlin: Springer-Verlag, 2005) 1-36.
- [12] Z. Pawlak, Flow graphs and decision algorithms, in G. Wang et al. (Eds.) *Rough sets, fuzzy sets, data mining and granular computing*, *LNAI 2639* (Berlin: Springer-Verlag, 2003) 1-10.
- [13] R. Srikant, R. Agrawal, Mining generalized association rules. *Proc. of the 21st International Conference on Very Large Data Bases*, Zurich Switzerland, 1995, 407-419.
- [14] C.J. Butz, W. Yan, B. Yang, The computational complexity of inference using rough set flow graphs, in D. Slezak, et al. (Eds.) *Rough sets, fuzzy sets, data mining and granular computing*, *LNAI 364* (Berlin: Springer-Verlag, 2005) 335-344.
- [15] J. Bowes, E. Neufeld, J.E. Greer, J. Cooke, A comparison of association rule discovery and bayesian network causal inference algorithms to discover relationships in discrete Data. *Proc. of the 13th Biennial Conference of the Canadian Society on Computational Studies of Intelligence*, Quebec Canada, 2000, 326-336.

Towards Theories of Fuzzy Set and Rough Set to Flow Graphs

Doungnat Chitcharoen and Puntip Pattaraintakorn

Abstract—Mathematical rough set theory and fuzzy set theory have attracted both practical and theoretical researchers from their efficiently and effectively to analyze real-world data. A novel and significant extension is called *flow graphs*. In this paper, we introduced how to calculate *certainty*, *coverage* and *strength* coefficients of decision rules from fuzzy attributes in a flow graph. Furthermore, we relax concept of mutual exclusion and introduced four new propositions of certainty and coverage coefficients for decision rules extracted from flow graph. An example calculation of these coefficients is provided. We also demonstrate real-world experiment on POSN data set. Several case studies illustrate a desirable outcome.

Index Terms—Fuzzy set theory, rough set theory, flow graphs.

I. INTRODUCTION

ROUGH set theory was introduced by Zdzislaw Pawlak in the early 1980's [1]. The crux of rough set theory is the novel use of approximations to cope with uncertainty. A significant extension of rough set theory is called *flow graphs*. Pawlak's flow graphs are different to *flow networks* introduced by Ford *et al.* that do not analyze optimal flow [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]. 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]. In [5], Pawlak showed that a decision tree can be constructed as a flow graph. Rough sets also closely related with *modus ponens* (MP) and *modus tollens* (MT) inference rules [6]. Preference analysis [7], rule analysis [8], propositional calculus [6], data mining [5], [9] and granular computing [10] were analyzed by flow graphs efficiently.

Fuzzy set theory was introduced by Lofti Zadeh in 1965 [11]. A fuzzy set is a class of objects with a continuum of grader of membership. The characteristic of *membership function* of a fuzzy set is permitted to have value in the interval $[0, 1]$. If A is a classical set, its membership function can take on only two values 0 and 1. It can be understood as a vague boundary set theory. Recently, fuzzy set theory and its applications were developed extensively and fascinated attention [11]. In real-world data analysis, if collected data is in the form of fuzzy attributes, then it can be represented in the decision table with fuzzy attributes. We can analyze

this decision table by applying concepts of fuzzy set. Alicja *et al.* introduced a concept of fuzzy flow graphs in [12]. This approach allowed computation not only for descriptive attributes but also for fuzzy attributes in flow graphs. In the present paper, we introduce an alternative approach to construct flow graphs, decision rules and perform data analysis for decision table with fuzzy attributes. Moreover, we relax mutual exclusion and find four new coefficients within flow graphs.

This article is organized as follows. Sections II, III contain rudiment of rough set and fuzzy set theories. We introduce preliminary definitions and properties of flow graphs in Section IV. Next section contains an example on data analysis. The actual data set from POSN and experimental results are presented in Section VI follows by conclusion of the paper.

II. ROUGH SET THEORY

The following rough sets preliminary is taken from [1]. This mathematical theory formed the basis of flow graphs later in the early 2000s [3], [5], [7], [13]. Rough sets are based on an *information system* (*data table*). 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 *B-granules*, for more *granular computing* studies please refer to [7], [10]). 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 $B_*(X) = \bigcup_{x \in U} \{B(x) | B(x) \subseteq X\}$, $B^*(X) = \bigcup_{x \in U} \{B(x) | B(x) \cap X \neq \emptyset\}$, respectively. In addition, the set $BN_B(X) = B^*(X) - B_*(X)$ will be referred to as *B-boundary region* of X . 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*. More studies involve rough sets are in [1], [3], [5], [6], [7], [9], [13].

Doungnat Chitcharoen and Puntip Pattaraintakorn are with Department of Mathematics and Computer Science, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520, email: s0067111@kmitl.ac.th, kppuntip@kmitl.ac.th.

1675

978-1-4244-1819-0/08/\$25.00©2008 IEEE

III. FUZZY SET THEORY

The following fuzzy set theory concept is taken from [11]. Let X be a space of objects and x be a generic element of X . A fuzzy set A in X is characterized by a membership function, $\mu_A(x) \rightarrow [0,1]$. The nearer the value of $\mu_A(x)$ to 1, the higher the grade of membership of x in A . A crucial issue for fuzzy sets applications is to express real-world problem by fuzzy sets attributes in decision table and perform data analysis. In further discussions, we consider decision table with fuzzy attributes in the form introduced in [12], [14].

Let $(U, C \cup D)$ be a decision table with fuzzy attributes, $U = \{x_1, x_2, \dots, x_N\}$ be a finite non-empty set of the universe with N elements, $C = \{c_1, c_2, \dots, c_n\}$ be a set of condition attributes and $D = \{d_1, d_2, \dots, d_m\}$ be a set of decision attributes. The domain of the family of linguistic values of condition attribute c_i denoted by $V_i = \{V_{i1}, V_{i2}, \dots, V_{in_i}\}$, where n_i is the number of the linguistic values of i -th conditions, $i = 1, 2, \dots, n$. The domain of the family of linguistic values of decision attribute d_i denoted by $W_j = \{W_{j1}, W_{j2}, \dots, W_{jm_j}\}$, where m_j is the number of the linguistic values of j -th conditions, $j = 1, 2, \dots, m$.

In order to describe an element $x \in U$ with fuzzy attributes, its membership degree should be determined. This process is called fuzzification state. In this state, for any element x , fuzzy value of condition attribute c_i and decision attribute d_j denoted by $V_i(x)$ and $W_j(x)$ are defined as $V_i(x) = \{\frac{\mu_{v_{i1}}(x)}{V_{i1}}, \frac{\mu_{v_{i2}}(x)}{V_{i2}}, \dots, \frac{\mu_{v_{in_i}}(x)}{V_{in_i}}\}$ and $W_j(x) = \{\frac{\mu_{w_{j1}}(x)}{W_{j1}}, \frac{\mu_{w_{j2}}(x)}{W_{j2}}, \dots, \frac{\mu_{w_{jm_j}}(x)}{W_{jm_j}}\}$.

Furthermore, for any $x \in U$ we assume that values of $V_i(x)$ and $W_j(x)$ satisfy $power(V_i(x)) = \sum_{k=1}^{n_i} \mu_{v_{ik}}(x) = 1$ and $power(W_j(x)) = \sum_{k=1}^{m_j} \mu_{w_{jk}}(x) = 1$, $i = 1, 2, \dots, n$, $j = 1, 2, \dots, m$. An example of fuzzy values $V_i(x)$ of c_i and fuzzy values $W_j(x)$ of d_j is in Table 1.

TABLE 1
DECISION TABLE WITH FUZZY ATTRIBUTES

	c_1	c_2	...	c_n	d_1	d_2	...	d_m
x_1	$V_1(x_1)$	$V_2(x_1)$	...	$V_n(x_1)$	$W_1(x_1)$	$W_2(x_1)$	...	$W_m(x_1)$
x_2	$V_1(x_2)$	$V_2(x_2)$	...	$V_n(x_2)$	$W_1(x_2)$	$W_2(x_2)$	...	$W_m(x_2)$
...	...	...	...	...	...	...	...	...
x_N	$V_1(x_N)$	$V_2(x_N)$	...	$V_n(x_N)$	$W_1(x_N)$	$W_2(x_N)$	...	$W_m(x_N)$

IV. FLOW GRAPHS

In this section we present basic definitions and some properties of flow graphs from [3], [5], [7], [9], [13].

A. 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

¹If the values of all attributes are singletons or disjoint intervals on the original domain of attributes, then we obtain a classical crisp decision table.

in [3], [5], [7], [9], [13] and for fuzzy set theory [12], [14]. 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 $\varphi: B \rightarrow R^+$ is a flow function, $\varphi(G)$ is a throughflow of flow graph G and $\sigma: B \rightarrow [0,1]$ 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: $\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(x, y)$ and $\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x, y)$. For any internal node x , $\sigma_+(x) = \sigma_-(x) = \sigma(x)$, where $\sigma(x)$ is a normalized throughflow of x . Similarly, normalized inflow and outflow for the flow graph G are defined as: $\sigma_+(G) = \sum_{x \in I(G)} \sigma_-(x)$ and $\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(G)} \sigma_+(x)$.

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 $\sum_{x \in I(y)} cov(x, y) = 1$, 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 $\sigma(y) = \sum_{x \in I(y)} cov(x, y)\sigma(x) = \sum_{x \in I(y)} \sigma(x, y)$ are Bayes' rules [3], [5], [7], [9], [13]. 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 $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ 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 $[x_1 \dots x_n]$ are defined as: $cer[x_1 \dots x_n] = \prod_{i=1}^{n-1} cer(x_i, x_{i+1})$, $cov[x_1 \dots x_n] = \prod_{i=1}^{n-1} cov(x_i, x_{i+1})$ and $\sigma[x \dots y] = \sigma(x)cer[x \dots y] = \sigma(y)cov[x \dots y]$. 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: $cer(x, y) = \sum_{[x \dots y] \in \langle x, y \rangle} cer[x \dots y]$, $cov(x, y) = \sum_{[x \dots y] \in \langle x, y \rangle} cov[x \dots y]$ and $\sigma(x, y) = \sum_{[x \dots y] \in \langle x, y \rangle} \sigma[x \dots y] = \sigma(x)cer(x, y) = \sigma(y)cov(x, y)$. If $[x \dots y]$ is a path such that x and y are input and output of the graph G , respectively, then $[x \dots y]$ 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 $\sigma(x, y) = \sigma(x, y)$, $cer(x, y) = cer(x, y)$ and $cov(x, y) = cov(x, y)$ 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.

B. Decision Rules, Decision Algorithm and Flow Graph

In this section, basic concepts of decision rules and decision algorithms that can be used as classifiers in logical form [4], [7] are provided. Let $S = (U, A)$ be an information system, for any subset B of A , there is an associate set of formulas $For(B)$. $For(B)$ is built up from attribute value pairs (a, v) where $a \in B$ and $v \in V_a$. For any $\Phi \in For(B)$, let $\|\Phi\|$ be the set of all objects from the universe U satisfy Φ in S . By means of logical connectives AND ($\wedge$), OR ($\vee$) and NOT ($\neg$), Pawlak defined some inductively properties as follows: $\|\Phi \vee \Psi\| = \|\Phi\| \cup \|\Psi\|$, $\|\Phi \wedge \Psi\| = \|\Phi\| \cap \|\Psi\|$ and $\|\neg\Phi\| = U - \|\Phi\|$ [4].

A decision rule in S is an expression in the form $\Phi \rightarrow \Psi$, Φ and Ψ are referred to as a condition and decision of the rule, respectively. For every decision rules, there are three associated coefficients: $\sigma(\Phi, \Psi) = \frac{card(\|\Phi\| \cap \|\Psi\|)}{card(\|\Phi\|)}$, $cer(\Phi, \Psi) = \frac{card(\|\Phi\| \cap \|\Psi\|)}{card(\|\Phi\|)}$ and $cov(\Phi, \Psi) = \frac{card(\|\Phi\| \cap \|\Psi\|)}{card(\|\Psi\|)}$ [3]. A set of all decision rules $Dec(\Phi, \Psi) = \{\Phi_i \rightarrow \Psi_i\}_{i=1}^n$, $n \geq 2$, will be called a decision algorithm if all decision rules are satisfied conditions: (1) *Mutually exclusive*: for every $\Phi_i \rightarrow \Psi_i$ and $\Phi_j \rightarrow \Psi_j$, $\|\Phi_i\| \cap \|\Phi_j\| = \emptyset$ and $\|\Psi_i\| \cap \|\Psi_j\| = \emptyset$ for $i \neq j$, (2) *Admissible*: $\|\Phi \wedge \Psi\| \neq \emptyset$, (3) *Covering*: $\bigcup_{i=1}^n \|\Phi_i\| = U$ and $\bigcup_{i=1}^n \|\Psi_i\| = U$ [3], [4].

A flow graph can be interpreted as decision algorithms [3], [9] and will be described as follows. Let us assume that a set of nodes of a flow graph is interpreted as a set of formulas, denoted by Φ, Ψ . Then every branch (Φ, Ψ) can be understood as a decision rule $\Phi \rightarrow \Psi$. Φ will be referred to as condition, whereas Ψ is decision. Such a rule is characterized by three coefficients i.e., strength, certainty and coverage as: $\sigma(\Phi, \Psi) = \frac{card(\|\Phi\| \cap \|\Psi\|)}{card(\|\Phi\|)}$, $cer(\Phi, \Psi) = \frac{card(\|\Phi\| \cap \|\Psi\|)}{card(\|\Phi\|)}$ and $cov(\Phi, \Psi) = \frac{card(\|\Phi\| \cap \|\Psi\|)}{card(\|\Psi\|)}$ [3].

Next, a path, $\{\Phi_1 \dots \Phi_n\}$, is a sequence of decision rules $\Phi_1 \rightarrow \Phi_2, \Phi_2 \rightarrow \Phi_3, \dots, \Phi_{n-1} \rightarrow \Phi_n$. As shown in [3], this sequence of decision rules can be interpreted as a single decision rule $\Phi_1 \Phi_2 \dots \Phi_{n-1} \rightarrow \Phi_n$, in short $\Phi^* \rightarrow \Phi_n$, where $\Phi^* = \Phi_1 \wedge \Phi_2 \wedge \dots \wedge \Phi_{n-1}$. It is characterized by three coefficients: $\sigma(\Phi^*, \Phi_n) = \sigma(\Phi_1)cer[\Phi_1 \dots \Phi_n] = \sigma(\Phi_n)cov[\Phi_1 \dots \Phi_n]$, $cer(\Phi^*, \Phi_n) = cer[\Phi_1 \dots \Phi_n]$ and $cov(\Phi^*, \Phi_n) = cov[\Phi_1 \dots \Phi_n]$, where $\sigma(\Phi)$ can be understood as the truth value of the formula Φ . Let $\{\Phi_1 \dots \Phi_n\}$ be a path such that Φ_1 is an input and Φ_n is an output of flow graph G , respectively, then the $\{\Phi_1 \dots \Phi_n\}$ will be referred as complete. The set of all paths from Φ_1 to Φ_n denoted by $\langle \Phi_1 \dots \Phi_n \rangle$ is called complete paths. The set of all decision rules associated with all complete paths will be called a decision algorithm induced by the flow graph.

C. New Certainty and Coverage Coefficients for Decision Rules from Relaxed Mutually Exclusive

In this section, we propose relaxation of the mutual exclusion property of decision algorithm and find its new coverage and coverage coefficients.

²The formulas can be viewed as propositional functions in logic [3].

Let $\|\Phi_i\|$ and $\|\Psi_i\|$, $i = 1, 2, \dots, n$ be sets of all objects from the universe U which satisfy these condition and decision attributes.

From mutual exclusive of a decision algorithm, there are assumptions that for every decision rules $\Phi_i \rightarrow \Psi_i$ and $\Phi_j \rightarrow \Psi_j$, [3], [4]

$$\|\Phi_i\| \cap \|\Phi_j\| = \emptyset \text{ and } \|\Psi_i\| \cap \|\Psi_j\| = \emptyset$$

for $i \neq j$.

In this paper, we propose relaxation of this mutual exclusion property as follows. If we allow at least two decision rules $\Phi_i \rightarrow \Psi$ and $\Phi \rightarrow \Psi_j$ to have

$$\|\Phi_{i_m}\| \cap \|\Phi_{i_k}\| \neq \emptyset \text{ or } \|\Psi_{i_m}\| \cap \|\Psi_{i_k}\| \neq \emptyset$$

for $i_m \neq j_k$, then we can find new certainty and coverage coefficients as presented in Propositions 1–4.

Let us first give a brief explanation of the usefulness of this relax mutual exclusion. For example, if we have two decision rules:

IF *score_of_Algebra* is high THEN *total_score* is high and

IF *score_of_Number_Theory* is high THEN *total_score* is high,

in a decision algorithm, then there may be some elements satisfy both conditions ($\|\Phi_{i_m}\| \cap \|\Phi_{i_k}\| \neq \emptyset$). In this case, mutual exclusive condition is failed but we can apply our relaxation to compute certainty and coverage coefficients.

Before we come to Propositions 1–4, let us introduce Remarks 1, 2 and 3 that are important to use to find certainty and coverage coefficients for relaxed mutually exclusive. If we connect any set of conditions (set of decisions) by logical connectives AND ($\wedge$) or OR ($\vee$), then the following Remarks 1 and 2 are holded by Pawlak's definitions of certainty and coverage coefficients [3], [4], [7], [9].

Remark 1: Let Φ and Ψ_i , $i = 1, 2, \dots, n$ be the set of condition and decision attributes, then

- 1) $cer(\Phi, \Psi_1 \vee \dots \vee \Psi_n) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cup \dots \cup \|\Psi_n\|))}{card(\|\Phi\|)}$,
- 2) $cer(\Phi, \Psi_1 \wedge \dots \wedge \Psi_n) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cap \dots \cap \|\Psi_n\|))}{card(\|\Phi\|)}$,
- 3) $cov(\Phi, \Psi_1 \vee \dots \vee \Psi_n) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cup \dots \cup \|\Psi_n\|))}{card(\|\Psi_1 \vee \dots \vee \Psi_n\|)}$,
- 4) $cov(\Phi, \Psi_1 \wedge \dots \wedge \Psi_n) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cap \dots \cap \|\Psi_n\|))}{card(\|\Psi_1 \wedge \dots \wedge \Psi_n\|)}$.

Remark 2: Let Φ_i and Ψ , $i = 1, 2, \dots, n$ be the set of condition and decision attributes, then

- 1) $cov(\Phi_1 \vee \dots \vee \Phi_n, \Psi) = \frac{card((\|\Phi_1\| \cup \dots \cup \|\Phi_n\|) \cap \|\Psi\|)}{card(\|\Psi\|)}$,
- 2) $cov(\Phi_1 \wedge \dots \wedge \Phi_n, \Psi) = \frac{card((\|\Phi_1\| \cap \dots \cap \|\Phi_n\|) \cap \|\Psi\|)}{card(\|\Psi\|)}$,
- 3) $cer(\Phi_1 \vee \dots \vee \Phi_n, \Psi) = \frac{card((\|\Phi_1\| \cup \dots \cup \|\Phi_n\|) \cap \|\Psi\|)}{card(\|\Phi_1 \vee \dots \vee \Phi_n\|)}$,
- 4) $cer(\Phi_1 \wedge \dots \wedge \Phi_n, \Psi) = \frac{card((\|\Phi_1\| \cap \dots \cap \|\Phi_n\|) \cap \|\Psi\|)}{card(\|\Phi_1 \wedge \dots \wedge \Phi_n\|)}$.

Remark 3 is the property obtained from set theory.

Remark 3: Let Φ_i , $i = 1, 2, \dots, n$ be a set of attributes, then $card(\|\Phi_1\| \cup \|\Phi_2\| \cup \dots \cup \|\Phi_n\|)$

$$= \sum_{i=1}^n card(\|\Phi_i\|) - \sum_{i=1}^{n-1} \sum_{j=i+1}^n card(\|\Phi_i\| \cap \|\Phi_j\|) + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^n card(\|\Phi_i\| \cap \|\Phi_j\| \cap \|\Phi_k\|) - \dots + (-1)^{n-1} card(\|\Phi_1\| \cap \|\Phi_2\| \cap \dots \cap \|\Phi_n\|).$$

We come to relaxation of mutual exclusion property. If there exist at least two decisions $\|\Psi_i\|$ and $\|\Psi_j\|$ such that $\|\Psi_i\| \cap \|\Psi_j\| \neq \emptyset$, we can relax mutually exclusive for decisions (as in Propositions 1, 2). If there exist at least two conditions $\|\Phi_i\|$ and $\|\Phi_j\|$ such that $\|\Phi_i\| \cap \|\Phi_j\| \neq \emptyset$, we can relax mutually exclusive for conditions (as in Propositions 3, 4).

Proposition 1: If we are given a set of decision rules, $\{\Phi \rightarrow \Psi_i\}_n$, such that $\|\Psi_{i_m}\| \cap \|\Psi_{i_k}\| \neq \emptyset$, for any $i_m \neq i_k$. Then $cer(\Phi, \Psi_1 \vee \Psi_2 \vee \dots \vee \Psi_n)$

$$= \sum_{i=1}^n cer(\Phi, \Psi_i) - \sum_{i_1=1}^{n-1} \sum_{i_2=i_1+1}^n cer(\Phi, \Psi_{i_1} \wedge \Psi_{i_2}) \\ + \sum_{i_1=1}^{n-2} \sum_{i_2=i_1+1}^{n-1} \sum_{i_3=i_2+1}^n cer(\Phi, \Psi_{i_1} \wedge \Psi_{i_2} \wedge \Psi_{i_3}) \\ - \dots + (-1)^{n-1} cer(\Phi, \Psi_1 \wedge \Psi_2 \wedge \dots \wedge \Psi_n).$$

Proof: If we denote $cer(\Phi, \Psi_1 \vee \Psi_2 \vee \dots \vee \Psi_n)$ by $cer(\Phi, \Psi_i)$, then by Remark 1(1) we have $cer(\Phi, \Psi_i) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cup \|\Psi_2\| \cup \dots \cup \|\Psi_n\|))}{card(\|\Phi\|)}$. Next, by left-distributive over union, we obtain

$$cer(\Phi, \Psi_i) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cup (\|\Phi\| \cap \|\Psi_2\|) \cup \dots \cup (\|\Phi\| \cap \|\Psi_n\|)))}{card(\|\Phi\|)} \\ = \sum_{i=1}^n \frac{card(\|\Phi\| \cap \|\Psi_i\|)}{card(\|\Phi\|)} \\ - \sum_{i_1=1}^{n-1} \sum_{i_2=i_1+1}^n \frac{card(\|\Phi\| \cap \|\Psi_{i_1}\| \cap \|\Psi_{i_2}\|)}{card(\|\Phi\|)} \\ + \sum_{i_1=1}^{n-2} \sum_{i_2=i_1+1}^{n-1} \sum_{i_3=i_2+1}^n \frac{card(\|\Phi\| \cap \|\Psi_{i_1}\| \cap \|\Psi_{i_2}\| \cap \|\Psi_{i_3}\|)}{card(\|\Phi\|)} \\ - \dots + (-1)^{n-1} \frac{card(\|\Phi\| \cap \|\Psi_1\| \cap \|\Psi_2\| \cap \dots \cap \|\Psi_n\|)}{card(\|\Phi\|)} \text{ (by Remark 3)} \\ = \sum_{i=1}^n cer(\Phi, \Psi_i) - \sum_{i_1=1}^{n-1} \sum_{i_2=i_1+1}^n cer(\Phi, \Psi_{i_1} \wedge \Psi_{i_2}) \\ + \sum_{i_1=1}^{n-2} \sum_{i_2=i_1+1}^{n-1} \sum_{i_3=i_2+1}^n cer(\Phi, \Psi_{i_1} \wedge \Psi_{i_2} \wedge \Psi_{i_3}) \\ - \dots + (-1)^{n-1} cer(\Phi, \Psi_1 \wedge \Psi_2 \wedge \dots \wedge \Psi_n) \text{ (by Remark 1(2))}. \blacksquare$$

Proposition 2: If we are given a set of decision rules, $\{\Phi \rightarrow \Psi_i\}_n$, such that $\|\Psi_{i_m}\| \cap \|\Psi_{i_k}\| \neq \emptyset$, for any $i_m \neq i_k$. Then $cov(\Phi, \Psi_1 \vee \Psi_2 \vee \dots \vee \Psi_n) < \sum_{i=1}^n cov(\Phi, \Psi_i)$.

Proof: If we denote $cov(\Phi, \Psi_1 \vee \Psi_2 \vee \dots \vee \Psi_n)$ by $cov(\Phi, \Psi_i)$, then by Remark 1(3) we have $cov(\Phi, \Psi_i) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cup \|\Psi_2\| \cup \dots \cup \|\Psi_n\|))}{card(\|\Psi_1\| \cup \|\Psi_2\| \cup \dots \cup \|\Psi_n\|)}$. Next, by left-distributive over union and inequality property, we obtain

$$cov(\Phi, \Psi_i) = \frac{card(\|\Phi\| \cap (\|\Psi_1\| \cup (\|\Phi\| \cap \|\Psi_2\|) \cup \dots \cup (\|\Phi\| \cap \|\Psi_n\|)))}{card(\|\Psi_1\| \cup \|\Psi_2\| \cup \dots \cup \|\Psi_n\|)} \\ < \sum_{i=1}^n \frac{card(\|\Phi\| \cap \|\Psi_i\|)}{card(\|\Psi_1\| \cup \|\Psi_2\| \cup \dots \cup \|\Psi_n\|)} \\ < \sum_{i=1}^n \frac{card(\|\Phi\| \cap \|\Psi_i\|)}{card(\|\Psi_i\|)}.$$

Then $(\Phi, \Psi_i) < \sum_{i=1}^n cov(\Phi, \Psi_i)$ (by Remark 1(3)). $\blacksquare$

Proposition 3: If we are given a set of decision rules, $\{\Phi_i \rightarrow \Psi\}_n$, such that $\|\Phi_{i_m}\| \cap \|\Phi_{i_k}\| \neq \emptyset$, for any $i_m \neq i_k$. Then $cov(\Phi_1 \vee \Phi_2 \vee \dots \vee \Phi_n, \Psi)$

$$= \sum_{i=1}^n cov(\Phi_i, \Psi) - \sum_{i_1=1}^{n-1} \sum_{i_2=i_1+1}^n cov(\Phi_{i_1} \wedge \Phi_{i_2}, \Psi) \\ + \sum_{i_1=1}^{n-2} \sum_{i_2=i_1+1}^{n-1} \sum_{i_3=i_2+1}^n cov(\Phi_{i_1} \wedge \Phi_{i_2} \wedge \Phi_{i_3}, \Psi) \\ - \dots + (-1)^{n-1} cov(\Phi_1 \wedge \Phi_2 \wedge \dots \wedge \Phi_n, \Psi).$$

Proof: If we denote $cov(\Phi_1 \vee \Phi_2 \vee \dots \vee \Phi_n, \Psi)$ by $cov(\Phi_i, \Psi)$, then by Remark 2(1) we have $cov(\Phi_i, \Psi) = \frac{card(\|\Phi_i\| \cap (\|\Psi_1\| \cup \dots \cup \|\Psi_n\|) \cap \|\Psi\|)}{card(\|\Psi\|)}$. Next, by right-distributive over union, we obtain

$$cov(\Phi_i, \Psi) = \frac{card(\|\Phi_i\| \cap (\|\Psi_1\| \cup (\|\Phi_2\| \cap \|\Psi\|) \cup \dots \cup (\|\Phi_n\| \cap \|\Psi\|)))}{card(\|\Psi\|)} \\ = \sum_{i=1}^n \frac{card(\|\Phi_i\| \cap \|\Psi\|)}{card(\|\Psi\|)} \\ - \sum_{i_1=1}^{n-1} \sum_{i_2=i_1+1}^n \frac{card(\|\Phi_{i_1}\| \cap \|\Phi_{i_2}\| \cap \|\Psi\|)}{card(\|\Psi\|)} \\ + \sum_{i_1=1}^{n-2} \sum_{i_2=i_1+1}^{n-1} \sum_{i_3=i_2+1}^n \frac{card(\|\Phi_{i_1}\| \cap \|\Phi_{i_2}\| \cap \|\Phi_{i_3}\| \cap \|\Psi\|)}{card(\|\Psi\|)} \\ - \dots + (-1)^{n-1} \frac{card(\|\Phi_1\| \cap \|\Phi_2\| \cap \dots \cap \|\Phi_n\| \cap \|\Psi\|)}{card(\|\Psi\|)} \text{ (by Remark 3)}$$

$$= \sum_{i=1}^n cov(\Phi_i, \Psi) - \sum_{i_1=1}^{n-1} \sum_{i_2=i_1+1}^n cov(\Phi_{i_1} \wedge \Phi_{i_2}, \Psi) \\ + \sum_{i_1=1}^{n-2} \sum_{i_2=i_1+1}^{n-1} \sum_{i_3=i_2+1}^n cov(\Phi_{i_1} \wedge \Phi_{i_2} \wedge \Phi_{i_3}, \Psi) \\ - \dots + (-1)^{n-1} cov(\Phi_1 \wedge \Phi_2 \wedge \dots \wedge \Phi_n, \Psi) \text{ (by Remark 2(1))}. \blacksquare$$

Proposition 4: If we are given a set of decision rules, $\{\Phi_i \rightarrow \Psi\}_n$, such that $\|\Phi_{i_m}\| \cap \|\Phi_{i_k}\| \neq \emptyset$, for any $i_m \neq i_k$. Then $cer(\Phi_1 \vee \Phi_2 \vee \dots \vee \Phi_n, \Psi) < \sum_{i=1}^n cer(\Phi_i, \Psi)$.

Proof: If we denote $cer(\Phi_1 \vee \Phi_2 \vee \dots \vee \Phi_n, \Psi)$ by $cer(\Phi_i, \Psi)$, then by Remark 2(3) we have $cer(\Phi_i, \Psi) = \frac{card(\|\Phi_i\| \cap (\|\Psi_1\| \cup \dots \cup \|\Psi_n\|) \cap \|\Psi\|)}{card(\|\Phi_i\| \cap (\|\Psi_1\| \cup \dots \cup \|\Psi_n\|))}$. Next, by right-distributive over union and inequality property, we obtain

$$cer(\Phi_i, \Psi) = \frac{card(\|\Phi_i\| \cap (\|\Psi_1\| \cap \|\Psi\|) \cup (\|\Phi_2\| \cap \|\Psi\|) \cup \dots \cup (\|\Phi_n\| \cap \|\Psi\|))}{card(\|\Phi_i\| \cap (\|\Psi_1\| \cup \dots \cup \|\Psi_n\|))} \\ < \sum_{i=1}^n \frac{card(\|\Phi_i\| \cap \|\Psi\|)}{card(\|\Phi_i\| \cap (\|\Psi_1\| \cup \dots \cup \|\Psi_n\|))} \\ < \sum_{i=1}^n \frac{card(\|\Phi_i\| \cap \|\Psi\|)}{card(\|\Phi_i\|)}.$$

Then $cer(\Phi_i, \Psi) < \sum_{i=1}^n cer(\Phi_i, \Psi)$ (by Remark 2(1)). $\blacksquare$

D. Flow Graphs with Fuzzy Attributes

Flow graphs with fuzzy attributes have several advantages (e.g., represent data in form of fuzzy attributes) [12]. In further discussions, we apply flow graphs to represent and analyze decision table with fuzzy attributes in the form introduced in [12]. All possible decision rules with fuzzy attributes can be generated by using the *Cartesian product* of the set of attribute values with $r = \prod_{i=1}^n m_i$ as a possible number of rules. Let R_k be the k -th decision rule. It can be expressed by

R_k : IF c_1 is V_1^k , c_2 is V_2^k , ... and c_n is V_n^k ,
 THEN d_1 is W_1^k , d_2 is W_2^k , ... and d_m is W_m^k ,
 where $k = 1, 2, \dots, r$, and $V_i^k \in V_i$, $i = 1, 2, \dots, n$, $W_j^k \in W_j$,
 $j = 1, 2, \dots, m$.

Thus, Cartesian products $C^k = V_1^k \times V_2^k \times \dots \times V_n^k$ and $D^k = W_1^k \times W_2^k \times \dots \times W_m^k$ are the k -th decision rules expressed in the form of fuzzy implication, $C^k \rightarrow D^k$.

Next, we apply a T -norm operator to determine the *confirmation degree* of fuzzy decision rule. Let $cda(x, k)$ and $cdc(x, k)$ be the confirmation degrees of the antecedent and consequent of decision rules and $cd(x, k)$ be the confirmation degrees of the k -th decision rules by the element $x \in U$. They are defined respectively as

$$cda(x, k) = T(\mu_{V_1^k}(x), \mu_{V_2^k}(x), \dots, \mu_{V_n^k}(x)) \quad (1)$$

$$cdc(x, k) = T(\mu_{W_1^k}(x), \mu_{W_2^k}(x), \dots, \mu_{W_m^k}(x)) \quad (2)$$

$$cd(x, k) = T(cda(x, k), cdc(x, k)). \quad (3)$$

By using (1), (2) and (3) in fuzzy set domain, the *support* (*sup*) of decision rule's antecedent, consequent and itself are defined respectively as

$$\text{sup}(cda(x, k)) = \{cda(x_1, k)/x_1, \dots, cda(x_N, k)/x_N\},$$

$$\text{sup}(cdc(x, k)) = \{cdc(x_1, k)/x_1, \dots, cdc(x_N, k)/x_N\} \text{ and}$$

$$\text{sup}(R_k) = \{cd(x_1, k)/x_1, \dots, cd(x_N, k)/x_N\}.$$

Moreover, when consider the fuzzy universe, we can define the *throughflow*, $\varphi(x, y)$ of the branch (x, y) in G to the cardinality of the product of fuzzy sets, $\tilde{x}$ and $\tilde{y}$. The *strength* of (x, y) is thus given by

$$\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)}. \quad (4)$$

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

$$\varphi_-(x) = \text{power}\tilde{x} = \sum_{y \in O(x)} \varphi(x, y) = \sum_{y \in O(x)} \text{power}(\tilde{x} \cap \tilde{y}) \quad (5)$$

$$\varphi_+(x) = \text{power}\tilde{x} = \sum_{y \in I(x)} \varphi(x, y) = \sum_{y \in I(x)} \text{power}(\tilde{x} \cap \tilde{y}) \quad (6)$$

where $\text{power}\tilde{x}$ denotes fuzzy cardinality. For any internal node x , it holds that $\varphi_-(x) = \varphi_+(x) = \varphi(x)$.

A (directed) *path* from x to y ($x \neq y$) in G , denoted by $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ such that $x_1 = x$ and $x_n = y$ and $(x_i, x_{i+1}) \in B$, $i, 1 \leq i \leq n-1$. We obtain the *certainty*, *coverage* and *strength* coefficients of the path $[x_1 \dots x_n]$ [12], [14] as: $\text{cer}[x_1 \dots x_n] = \prod_{i=1}^{n-1} \text{cer}(x_1 \dots x_i, x_{i+1})$, where $\text{cer}(x_1 \dots x_i, x_{i+1}) = \frac{\text{power}(x_1 \cap x_2 \cap \dots \cap x_{i+1})}{\text{power}(x_1 \cap x_2 \cap \dots \cap x_i)}$, $\text{cov}[x_1 \dots x_n] = \prod_{i=1}^{n-1} \text{cov}(x_1 \dots x_i, x_{i+1})$, where $\text{cov}(x_1 \dots x_i, x_{i+1}) = \frac{\text{power}(x_1 \cap x_2 \cap \dots \cap x_{i+1})}{\text{power}(x_1 \cap x_2 \cap \dots \cap x_i)}$, and $\sigma[x_1 \dots x_n] = \sigma(x_1) \text{cer}[x_1 \dots x_n] = \sigma(x_n) \text{cov}[x_1 \dots x_n]$.

The decision rule R_k , $\Phi^* \rightarrow \Phi_n$, where Φ^* is condition and Φ_n is decision can be characterized by three coefficients:

$$\text{cer}(\Phi^*, \Phi_n) = \text{cer}(R_k) = \frac{\text{power}(\text{sup}(R_k))}{\text{power}(\text{sup}(cda(x, k)))} \quad (7)$$

$$\text{cov}(\Phi^*, \Phi_n) = \text{cov}(R_k) = \frac{\text{power}(\text{sup}(R_k))}{\text{power}(\text{sup}(cdc(x, k)))} \quad (8)$$

$$\sigma(\Phi^*, \Phi_n) = \text{strength}(R_k) = \frac{\text{power}(\text{sup}(R_k))}{\text{card}(U)}. \quad (9)$$

We will illustrate an example of these introduced notions and experimental results in Sections V, VI.

V. ILLUSTRATIVE EXAMPLE

In this section, we present an example of how to compute coefficients: *certainty*, *coverage* and *strength* for flow graphs with fuzzy attributes. To compute these coefficients for flow graphs by the notion of rough set theory please refer to [2], [7], [9].

Table II shows a decision table $S = (U, A)$, where $U = \{x_1, x_2, \dots, x_{10}\}$, condition attributes are score of Number theory (N) and score of Algebra (A). The decision attribute is the total score (D).

TABLE II
AN EXAMPLE DECISION TABLE

	N	A	D
x_1	32	46	178
x_2	9	40	102
x_3	8	35	101
x_4	11	31	88
x_5	12	27	87
x_6	16	31	81
x_7	2	37	74
x_8	0	36	73
x_9	10	29	73
x_{10}	4	28	70

We next use triangular membership functions below to the data (Table II) in fuzzification state.

$\mu_{N_1} = \begin{cases} 1, & \text{if } 0 \leq x < 5 \\ (11-x)/6, & \text{if } 5 \leq x < 11 \\ 0, & \text{if } 11 \leq x \leq 45 \end{cases}$	$\mu_{N_2} = \begin{cases} 0, & \text{if } 0 \leq x < 5 \\ (x-5)/6, & \text{if } 5 \leq x < 11 \\ (40-x)/29, & \text{if } 11 \leq x < 40 \\ 0, & \text{if } 40 \leq x \leq 45 \end{cases}$
$\mu_{N_3} = \begin{cases} 0, & \text{if } 0 \leq x < 11 \\ (x-11)/29, & \text{if } 11 \leq x < 40 \\ 1, & \text{if } 40 \leq x \leq 45 \end{cases}$	$\mu_{A_1} = \begin{cases} 1, & \text{if } 0 \leq x < 20 \\ (30-x)/10, & \text{if } 20 \leq x < 30 \\ 0, & \text{if } 30 \leq x \leq 46 \end{cases}$
$\mu_{A_2} = \begin{cases} 0, & \text{if } 0 \leq x < 20 \\ (x-20)/10, & \text{if } 20 \leq x < 30 \\ (36-x)/6, & \text{if } 30 \leq x < 36 \\ 0, & \text{if } 36 \leq x \leq 46 \end{cases}$	$\mu_{A_3} = \begin{cases} 0, & \text{if } 0 \leq x < 30 \\ (x-30)/6, & \text{if } 30 \leq x < 36 \\ 1, & \text{if } 36 \leq x \leq 46 \end{cases}$
$\mu_{D_1} = \begin{cases} 1, & \text{if } 0 \leq x < 60 \\ (85-x)/16, & \text{if } 60 \leq x < 85 \\ 0, & \text{if } 85 \leq x \leq 195 \end{cases}$	$\mu_{D_2} = \begin{cases} 0, & \text{if } 0 \leq x < 60 \\ (x-60)/16, & \text{if } 60 \leq x < 85 \\ (110-x)/25, & \text{if } 85 \leq x < 110 \\ 0, & \text{if } 110 \leq x \leq 195 \end{cases}$
$\mu_{D_3} = \begin{cases} 0, & \text{if } 0 \leq x < 85 \\ (x-85)/25, & \text{if } 85 \leq x < 110 \\ 1, & \text{if } 110 \leq x \leq 195 \end{cases}$	

The result is a decision table with fuzzy attributes as shown in Table III. Afterwards, we can compute strength $\sigma(N_i, A_j)$ of condition attributes in Table III by (4) and the result is in Table IV.

In Table IV, last row depicts inflow of each A_j , and last column depicts outflow of each N_i . For example, since throughflow $\varphi(N_1, A_1)$ is sum of the cardinality of product of fuzzy set $\tilde{N}_1$ and $\tilde{A}_1$, we get $\tilde{N}_1 \tilde{A}_1 = \tilde{N}_1 \cap \tilde{A}_1 = (0 \times 0) + (0.33 \times 0) + (0.5 \times 0) + (0 \times 0) + (0 \times 0.3) + (0 \times 0) + (1 \times 0) + (1 \times 0) + (0.167 \times 0.1) + (1 \times 0.2) = 0.2167$. Then by (4), $\sigma(N_1, A_1) = \frac{\varphi(N_1, A_1)}{10} = 0.02167$.

TABLE III
DECISION TABLE WITH FUZZY ATTRIBUTES

	N_i			A_j			D_k		
	N_1	N_2	N_3	A_1	A_2	A_3	D_1	D_2	D_3
x_1	0	0.276	0.724	0	0	1	0	0	1
x_2	0.333	0.667	0	0	0	1	0	0.320	0.680
x_3	0.500	0.500	0	0	0.167	0.833	0	0.360	0.640
x_4	0	1	0	0	0.833	0.167	0	0.680	0.120
x_5	0	0.963	0.033	0.300	0.700	0	0	0.920	0.080
x_6	0	0.828	0.172	0	0.833	0.167	0.250	0.750	0
x_7	1	0	0	0	0	1	0.688	0.312	0
x_8	1	0	0	0	0	1	0.750	0.250	0
x_9	0.167	0.833	0	0.100	0.900	0	0.75	0.250	0
x_{10}	1	0	0	0.200	0.800	0	0.938	0.062	0

TABLE IV
STRENGTH $\sigma(N_i, A_j)$ BETWEEN CONDITION AND DECISION ATTRIBUTE

	A_1	A_2	A_3	$\sum$
N_1	0.02167	0.10333	0.27500	0.4000
N_2	0.03730	0.30322	0.16638	0.5069
N_3	0.00103	0.01678	0.07329	0.0931
$\sum$	0.06000	0.42330	0.51667	1

Next, we can use (4) and (5) to compute the strength of inflow of A_1 and obtain $\sigma_-(A_1) = \frac{\text{power}(A_1)}{N} = \sum_{i=1}^3 \sigma(N_i, A_1) = \sigma(N_1, A_1) + \sigma(N_2, A_1) + \sigma(N_3, A_1) = 0.02167 + 0.0373 + 0.00103 = 0.06$ (bottom left cell of Table IV).

TABLE V
STRENGTH $\sigma(A_j, D_k)$ BETWEEN CONDITION AND DECISION ATTRIBUTE

	D_1	D_2	D_3	$\sum$
A_1	0.02625	0.03135	0.00240	0.06
A_2	0.16333	0.23373	0.02627	0.42333
A_3	0.14792	0.14542	0.22333	0.51667
$\sum$	0.33750	0.41050	0.2520	1

Table V shows strength $\sigma(A_j, D_k)$ between condition and decision attributes. Last row depicts inflow of each D_k and last column depicts outflow of each A_j . If we use (4) and (6) to compute the strength of outflow of A_1 , we obtain $\sigma_+(A_1) (= \sigma_-(A_1)) = \frac{\text{power}(A_1)}{N} = \sum_{j=1}^3 \sigma(A_j, D_1) = \sigma(A_1, D_1) + \sigma(A_2, D_1) + \sigma(A_3, D_1) = 0.02625 + 0.03135 + 0.0024 = 0.06$ (top right cell). Please note that, last row from Table IV and last column from Table V show that $\sigma_-(A_j) = \sigma_+(A_j)$, where $j = 1, 2, 3$.

TABLE VI
CERTAINTY AND COVERAGE COEFFICIENTS OF N_i AND A_j

	A_1		A_2		A_3	
	cer	cov	cer	cov	cer	cov
N_1	0.0542	0.3612	0.2383	0.2440	0.6875	0.5322
N_2	0.0736	0.6217	0.5982	0.7163	0.3282	0.3220
N_3	0.0111	0.0171	0.1802	0.0397	0.8087	0.1438

Table VI shows the certainty and coverage coefficients of N_i and A_j , where $i, j = 1, 2, 3$. They are computed by (7), as $\text{cer}(N_i, A_j) = \frac{\text{power}(\tilde{N}_i \cap \tilde{A}_j)}{\text{power}(\tilde{N}_i)} = \frac{\sigma(N_i, A_j)}{\sigma(N_i)}$ and by (8)

as $\text{cov}(N_i, A_j) = \frac{\text{power}(\tilde{N}_i \cap \tilde{A}_j)}{\text{power}(\tilde{A}_j)} = \frac{\sigma(N_i, A_j)}{\sigma(A_j)}$. For example, $\text{cer}(N_1, A_1) = \frac{\sigma(N_1, A_1)}{\sigma(N_1)} = \frac{0.02167}{0.4} = 0.0542$ (top left cell) and $\text{cov}(N_1, A_1) = \frac{\sigma(N_1, A_1)}{\sigma(A_1)} = \frac{0.02167}{0.06} = 0.3612$ (next cell).

TABLE VII
CERTAINTY AND COVERAGE COEFFICIENTS OF A_i AND D_k

	D_1		D_2		D_3	
	cer	cov	cer	cov	cer	cov
A_1	0.4375	0.0778	0.5225	0.0764	0.0400	0.0095
A_2	0.3838	0.4840	0.5321	0.5694	0.0620	0.1042
A_3	0.2863	0.4383	0.2815	0.3542	0.4323	0.8862

Table VII shows certainty and coverage coefficients of A_j and D_k , where $j, k = 1, 2, 3$. They are computed by the same approach as in Table VI.

TABLE VIII
VALUE OF NORMALIZED STRENGTH BETWEEN CONDITION AND DECISION ATTRIBUTES

	D_1	D_2	D_3	$\sum$
$N_1 A_1$	0.0200	0.0017	0	0.02167
$N_1 A_2$	0.0863	0.0118	0.0053	0.10333
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$N_3 A_3$	0.0007	0.0022	0.0724	0.07529
$\sum$	0.3375	0.4105	0.2520	1

Table VIII shows the value of normalized strength between condition and decision attributes, $\sigma(N_i, A_j, D_k)$ where $i, j, k = 1, 2, 3$ by (9). Last row depicts inflow of D_k and last column depicts outflow of $N_i A_j$.

Next, by (4), (7) and (8), we can find certainty and coverage coefficients of N_i, A_j and D_k . The results are shown in Table IX, e.g., $\text{cer}(N_1 A_1, D_1) = \frac{0.02}{0.02167} = 0.9229$ (top left cell) and $\text{cov}(N_1 A_1, D_1) = \frac{0.02}{0.3375} = 0.0593$ (next cell).

TABLE IX
CERTAINTY AND COVERAGE COEFFICIENTS OF N_i, A_j AND D_k

	D_1		D_2		D_3	
	cer	cov	cer	cov	cer	cov
$N_1 A_1$	0.9229	0.0593	0.0769	0.0041	0	0
$N_1 A_2$	0.8347	0.2556	0.1137	0.0286	0.0516	0.0212
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$N_3 A_3$	0.0095	0.0021	0.0286	0.0053	0.9618	0.2874

TABLE X
CERTAINTY AND COVERAGE COEFFICIENTS OF N_i AND D_k

	D_1		D_2		D_3	
	cer	cov	cer	cov	cer	cov
N_1	0.6250	0.7407	0.2383	0.2322	0.1367	0.2169
N_2	0.1641	0.2465	0.5900	0.7285	0.2439	0.4946
N_3	0.0463	0.0128	0.173	0.0392	0.7807	0.2885

Table X shows certainty and coverage coefficients of N_i and D_k , where $i, k = 1, 2, 3$. They are computed by the same approach as in Table VI.

Finally, if we want to relax mutual exclusion of condition, then from Propositions 3 and 4 we can find $cer(N_i \vee A_j, D_k)$ and $cov(N_i \vee A_j, D_k)$, as shown in Table XI. For example, $cer(N_1 \vee A_2, D_2) < cer(N_1, D_2) + cer(A_2, D_2) = 0.2383 + 0.5521 = 0.7904$ and $cov(N_1 \vee A_2, D_2) = cov(N_1, D_2) + cov(A_2, D_2) - cov(N_1 \wedge A_2, D_2) = 0.2322 + 0.5694 - 0.0286 = 0.7730$.

TABLE XI
CERTAINTY AND COVERAGE COEFFICIENTS OF $N_i \vee A_j$ AND D_k

	D_1		D_2		D_3	
	cer	cov	cer	cov	cer	cov
$N_1 \vee A_1$	<1.0000	0.7407	<0.7608	0.3045	<0.1767	0.2264
$N_1 \vee A_2$	<1.0000	0.9691	<0.7904	0.7730	<0.1987	0.2993
$\vdots$						
$N_3 \vee A_2$	<0.3326	0.4490	<0.4343	0.4343	<1.0000	0.8873

From Table XI, we applied Propositions 3 and 4 to calculate certainty and coverage coefficients for relax mutual exclusion of conditions effectively. The antecedent part of decision rules can then be connected with OR ($\vee$) which is useful in some situations.

VI. EXPERIMENTAL RESULTS

In this section, we describe experimental results from the scores of Olympiad mathematics' students' analysis. Data contains 75 records collected from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN) in 2006–2007. In this paper, we study decision rules between condition and decision attributes as shown below:

Condition	Symbol	Condition	Symbol
score of Geometry	score.Geo	preference of Algebra	Prof.Alg
score of Number theory	score.Num	preference of Combinatorics	Prof.Com
score of Algebra	score.Alg	preference of Inequality	Prof.I.E
score of Combinatorics	score.Com	practiced of Geometry	Prac.Geo
score of Inequality	score.I.E	practiced of Number theory	Prac.Num
preference of Geometry	Prof.Geo	practiced of Algebra	Prac.Alg
preference of Number theory	Prof.Num	practiced of Combinatorics	Prac.Com
practiced of Inequality	Prac.I.E		
Decision	Symbol		
total score	total_score		

We next come to our experiment. In Cases 1–2, we consider condition and decision attributes in POSN data set as descriptive attributes. Possible values of first five condition attributes and decision attribute are low score and high score. These two case studies are then presented according to the combination of these attributes. The threshold for decision rule extraction is $cer > 0.6$.

Case 1. Decision rules between score of any subject and total score. The selected decision rules are:

IF Score.Num is low THEN total_score is low, cer = 0.867, cov = 0.812, IF Score.Geo is low THEN total_score is low, cer = 0.792, cov = 0.775, : IF Score.I.E is high THEN total_score is high, cer = 0.956, cov = 0.814,

with average cer = 0.791 and average cov = 0.731.

³Due to round off errors in computations, the equation may be not satisfied exactly in our examples.

Case 2. Decision rules between score of any two subjects and total score. Then the selected decision rules are:

IF score.Geo and I.E. are low THEN total_score is low, cer = 0.633, cov = 0.581, IF score.Geo and Alg are low THEN total_score is low, cer = 0.622, cov = 0.571, : IF score.Alg and I.E. are low THEN total_score is high, cer = 0.625, cov = 0.523,
--

with average cer = 0.620 and average cov = 0.556.

From Cases 1–2, we can summarize that, *If score of one or two subjects are low, then total score tends to be low too and If score of one or two subjects are high, then total score tends to be high as well.* We can see that decision rules obtained from Cases 1–2 depend on only two possible values of attributes (high and low). Thus, we next consider cases that scores are high and low to some degrees i.e., consider fuzzy attributes.

In Cases 3–8, we transform POSN data set to be in fuzzy attributes form by triangular membership functions. Several case studies are then presented according to the combination of these attributes.

Case 3. Decision rules between score of any subject and total score. Then the selected decision rule is:

IF Score.Num is high THEN total_score is high, cer = 0.627, cov = 0.504

with cer = 0.627 and cov = 0.504.

Case 4. Decision rules between score of any two subjects and total score. Then the selected decision rules are:

IF score.Geo and I.E. are low THEN total_score is low, cer = 0.651, cov = 0.409, IF score.Geo and I.E. are high THEN total_score is high, cer = 0.68, cov = 0.301, : IF score.Alg and I.E. are high THEN total_score is high, cer = 0.618, cov = 0.226,

with average cer = 0.656 and average cov = 0.343.

Case 5. Decision rules between score of any five subjects and total score. Then the selected decision rules are:

IF score.Geo, Num, Alg, Com and score.I.E are low THEN total_score is low, cer = 0.782, cov = 0.138, IF score.Geo, Num, Alg, and Com are low, I.E is high THEN total_score is low, cer = 0.645, cov = 0.09, : IF score.Geo, Num, Alg, Comb and I.E. are high THEN total_score is high, cer = 0.81, cov = 0.024,

with average cer = 0.691 and average cov = 0.047.

Case 6. Decision rules between practiced and preferred subjects, score of practiced and preferred subject and total score. The selected decision rules are:

IF Prof and Prac.Geo and score.Geo is low THEN total_score is low, cer = 0.638, cov = 0.719, IF Prof and Prac.Num and score.Num is high THEN total_score is high, cer = 0.834, cov = 0.6,
--

with average cer = 0.736 and average cov = 0.660.

Case 7. Decision rules between practiced subjects, score of practiced subjects and total score. The selected decision rules are:

```
IF Prac.Geo and score.Geo is low
THEN total_score is low, cer = 0.802, cov = 0.731,
IF Prac.J.E. and score.J.E. is low
THEN total_score is low, cer = 0.665, cov = 0.623,
```

with average cer = 0.634 and average cov = 0.677.

Case 8. Decision rules between preferred subjects, score of preferred subjects and total score. The selected decision rules are:

```
IF Pref.Geo and score.Geo is high
total_score is high, cer = 0.671, cov = 0.541,
IF Pref.Num and score.Num is high
THEN total_score is high, cer = 0.633, cov = 0.602,
IF Pref.J.E. and score.J.E. is low
THEN total_score is low, cer = 0.829, cov = 0.72,
IF Pref.J.E. and score.J.E. is high
THEN total_score is low, cer = 0.831, cov = 0.28,
```

with average cer = 0.741 and average cov = 0.536.

From these selected decision rules (Cases 3–8), we can distil useful knowledge from these flow graphs as follows. Number theory affects total score substantially. Because in Case 3, 62.7% of students with high score of Number theory also have high total score. Moreover, 50.4% of students which have high total score are from the group in which Number theory score is high. Obviously, students who get high score from any two (or more) subjects will successively get high total score as shown in Cases 4 and 5. Interestingly, the combinations of either preferred subject or practiced subject and high score for such subject resulted in high total scores from Cases 6, 7 and 8. In Case 8, we obtain maximum certainty of selected decision rules as one can see in Fig. 1. The results are desirable values of cer and cov with an average cer = 0.681 and cov = 0.461.

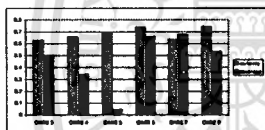


Fig. 1. Comparison of cer, cov of POSN data.

Finally, in Case 9 we relax mutual exclusion of conditions attributes. We can then illustrate that our introduced notions in Section IV are indeed feasible and applicable in real application. We thus use Propositions 3 and 4 to compute certainty and coverage coefficients of any one or two subjects and total score. The antecedent parts of the rules are connected by OR which is different from traditional rules as shown below.

Case 9. Decision rules between one or two subjects and total score. The selected decision rules are:

```
IF score.Geo or J.E. are low
THEN total_score is high, cer < 0.828, cov = 0.729,
IF score.Geo or Cov are low
THEN total_score is high, cer < 0.882, cov = 0.823,
...
IF score.Alg or J.E. are high
THEN total_score is low, cer < 0.870 cov = 0.601,
```

with average cov = 0.754.

As one can see, Proposition 4 gives range of certainty coefficients of the decision rules while Proposition 3 gives the values of coverage coefficients successfully.

VII. CONCLUSION

In this paper, an extension of mathematical fuzzy set theory and rough set theory, namely flow graphs, is considered. New certainty and coverage coefficients are introduced. They are successively applied to POSN data set efficiency. Our future work is to combine rough sets and fuzzy sets flow graphs for better suit specific type of data set and invent new method to allow extensively computation for large data set.

ACKNOWLEDGMENTS

This paper was supported by KMITL and NRCT grants. Thanks also to POSN, G.M. Zaverucha.

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.* 3, 2003, 98–101.
- [4] Z. Pawlak, Rough Sets, Decision Algorithms and Bayes' Theorem, *Eur. J. of Oper. Res.* 136, 2002, 181–189.
- [5] Z. Pawlak, Decision Trees and Flow Graphs, in S. Greco (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 J.J. Alpigini et al. (Eds.) *Rough Sets and Current Trends in Computing*, LNAI, 2475 (Berlin: Springer-Verlag, 2002) 93–104.
- [7] Z. Pawlak, Rough Set and Flow Graphs, in D. Slezak et al. (Eds.) *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, LNAI 3641 (Berlin: Springer-Verlag, 2005) 1–11.
- [8] P. Pattaraintakorn, N. Cercone, K. Naruemonkul, Rule Learning: Ordinal Prediction based on Rough Set and Soft-computing, *Appl. Math. Lett.* 19(12), 2006, 1300–1307.
- [9] Z. Pawlak, Flow Graphs and Data Mining, in J.F. Peters, A. Skowron (Eds.) *Transactions on Rough Sets III, LNCS 3400* (Berlin: Springer-Verlag, 2005) 1–36.
- [10] J. Sun, H. Liu, C. Qi, H. Zhang, An Interpretation of Flow Graphs by Granular Computing in S. Greco (Eds.) *Rough Sets and Current Trends in Computing*, LNCS 4259 (Berlin: Springer-Verlag 2006) 448–457.
- [11] L.A. Zadeh, Fuzzy set. *Information and Control* 8, 1965, 338–353.
- [12] M. Alicja, M. Leszek, Flow Graph and Decision Tables with Fuzzy Attributes, *Proc. 8th International Conference Artificial Intelligence and Soft Computing*, Zakopane, Poland, LNAI 4029, 2006, 268–277.
- [13] Z. Pawlak, Flow Graphs and Decision Algorithms, in G. Wang et al. (Eds.) *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, LNAI, 2639 (Berlin: Springer-Verlag 2003) 1–10.
- [14] M. Alicja, M. Leszek, Fuzzy Implication Operators in Variable Precision Fuzzy Rough Sets Model, in L. Rutkowski et al. (Eds.) *Proc. 7th International Conference Artificial Intelligence and Soft Computing*, Zakopane, Poland, LNAI 3037, 2004, 498–503.
- [15] C.J. Butz, W. Yan, B. Yang, The Computational Complexity of Inference using Rough Set Flow Graphs, in D. Slezak et al. (Eds.) *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, LNAI, 364 (Berlin: Springer-Verlag, 2005) 335–344.

An Extension of Rough Set Approximation to Flow Graph Based Data Analysis

Doungtrat Chitcharoen and Puntip Pattaraintakorn

School of Mathematics, Faculty of Science,
King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
s0067111@kmitl.ac.th, kppuntip@kmitl.ac.th

Abstract. This paper concerns some aspects of mathematical flow graph based data analysis. In particular, taking a flow graph view on rough sets' categories and measures leads to a new methodology of inductively reasoning from data. This perspective shows interesting relationships and properties among rough set, flow graphs and inverse flow graphs. A possible car dealer application is outlined and discussed. Evidently, our new categories and measures assist and alleviate some limitations in flow graphs to discover new patterns and explanations.

Keywords: Flow graphs, rough sets and decision rules.

1 Introduction

A mathematical flow graph, invented by Pawlak in 2002, is an extension of *rough set theory* [9]. A flow graph represents the information flow from the given data set [10,11,12,13,14]. The branches of a flow graph can be constructed as decision rules, with every decision rule, there are three associated coefficients: *strength*, *certainty* and *coverage* [14]. These coefficients satisfy Bayes' theorem. Inference in flow graphs has polynomial time and flow conservation comes with probabilistic conditional independencies in the problem domain [1]. Flow graphs have led to many interesting applications and extensions such as preference analysis [11], decision tree [13], survival analysis [7], association rule [3], data mining [14], search engines [2], fuzzy set [4,6], entropy measures [8] and granular computing [5]. More studies involving rough sets are discussed and provided in [15].

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.

This paper is organized as follows. In Section 2, we present the basic concepts of rough sets. In Section 3, we recall preliminary definitions of flow graphs. In Section 4, we present a new bridge between rough sets and flow graphs with an example throughout, followed by a conclusion in the last section.

M. Szczuka et al. (Eds.): RSCTC 2010, LNAI 6086, pp. 418–427, 2010.
© Springer-Verlag Berlin Heidelberg 2010

2 Rough Set Theory

The following rough sets preliminary is taken from [9]. Rough sets are based on an *information system*. 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 .

If we partition an information system into 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 $C \cap D = \emptyset$. 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 . Equivalence classes of the relation $I(B)$ are referred to as *B-elementary sets* or *B-elementary granules* denote by $B(X)$, i.e., $B(X)$ describes X in the terms of attribute values from B [11]. Below, we recall key feature definitions of approximations in rough sets.

Definition 1. [15] 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 $\underline{B}(X) = \bigcup_{x \in U} \{B(X) \mid B(X) \subseteq X\}$, $\overline{B}(X) = \bigcup_{x \in U} \{B(X) \mid B(X) \cap X \neq \emptyset\}$ and $BN_B(X) = \overline{B}(X) - \underline{B}(X)$, respectively.

If the boundary region of X is the empty set (i.e., $BN_B(x) = \emptyset$), then X is *crisp*. On the contrary, if $BN_B(X) \neq \emptyset$, then X is *rough*. In what follows we recall four basic classes of rough sets, i.e., four categories of vagueness.

Definition 2. [15] 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*.

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

Definition 3. [15] 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 $\alpha_B(X) = \frac{\text{card}(\underline{B}(X))}{\text{card}(\overline{B}(X))}$ and $\gamma_B(X) = 1 - \alpha_B(X) = 1 - \frac{\text{card}(\underline{B}(X))}{\text{card}(\overline{B}(X))}$, where $\text{card}(X)$ 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 $F = \{X_1, X_2, \dots, X_n\}$ be a partition of the universe U . For $B \subseteq A$, F depends on B to a degree $k_B(F) = \frac{\sum_{i=1}^n \text{card}(\underline{B}(X_i))}{\text{card}(U)}$.

Definitions 2 – 4 will be stated in the context of flow graphs in Section 4.

3 Flow Graphs

In this section, we recall some concepts of flow graphs which were introduced by Pawlak in [10,11,12,13,14].

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*, $\varphi: \mathcal{B} \rightarrow R^+$ 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 [11], the above equations can be considered as *flow conservation equations* (or *pairwise consistent* [1]).

Normalized Flow Graphs, Paths and Connections

In order to demonstrate interesting relationships between flow graphs and other disciplines (e.g., statistics), we come to the normalized version of flow graphs.

A *normalized flow graph* is a *directed, acyclic, finite graph* $G = (\mathcal{N}, \mathcal{B}, \sigma)$, where $\mathcal{N}$ is a set of nodes, $\mathcal{B} \subseteq \mathcal{N} \times \mathcal{N}$ is a set of *directed branches* and $\sigma: \mathcal{B} \rightarrow [0, 1]$ is a *normalized flow function* of (x, y) . The *strength* of (x, y) is $\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)}$. With every node x of a flow graph G , the associated *normalized inflow* and *outflow* are defined as: $\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(y, x)$, $\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x, y)$. For any internal node x , it holds that $\sigma_+(x) = \sigma_-(x) = \sigma(x)$, where $\sigma(x)$ is a *normalized throughflow* of x . Similarly, *normalized inflow* and *outflow* for the flow graph G are defined as: $\sigma_+(G) = \frac{\varphi_+(G)}{\varphi(G)} = \sum_{x \in I(G)} \sigma_-(x)$, $\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(G)} \sigma_+(x)$. It also holds that $\sigma_+(G) = \sigma_-(G) = \sigma(G) = 1$. With every branch (x, y) of a flow graph G , the *certainty* and the *coverage* of (x, y) are defined respectively as: $cer(x, y) = \frac{\sigma(x, y)}{\sigma(x)}$, $cov(x, y) = \frac{\sigma(x, y)}{\sigma(y)}$, where $\sigma(x), \sigma(y) \neq 0$. Properties of these coefficients were studied by Pawlak in [10,11,12,13,14].

Next, if we focus on sequence of nodes in a flow graph, we can find them by using the concept of a directed simple path. A (directed) *path* from x to y ($x \neq y$) in G , denoted by $[x \dots y]$, is a sequence of nodes $x_1, \dots, x_n$ such that $x_1 = x$ and $x_n = y$ and $(x_i, x_{i+1}) \in \mathcal{B}$ for every i , $1 \leq i \leq n - 1$. The *certainty*, *coverage* and *strength of the path* $[x_1 \dots x_n]$ are defined respectively as:

$$\text{cer}[x_1 \dots x_n] = \prod_{i=1}^{n-1} \text{cer}(x_i, x_{i+1}), \text{cov}[x_1 \dots x_n] = \prod_{i=1}^{n-1} \text{cov}(x_i, x_{i+1}), \sigma[x \dots y] = \sigma(x)\text{cer}[x \dots y] = \sigma(y)\text{cov}[x \dots y].$$

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 nodes x and y . For every connection $\langle x, y \rangle$, the associated *certainty*, *coverage* and *strength of the connection* $\langle x, y \rangle$ are defined as: $\text{cer} \langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} \text{cer}[x \dots y]$, $\text{cov} \langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} \text{cov}[x \dots y]$, $\sigma \langle x, y \rangle = \sum_{[x \dots y] \in \langle x, y \rangle} \sigma[x \dots y] = \sigma(x)\text{cer} \langle x, y \rangle = \sigma(y)\text{cov} \langle x, y \rangle$. If $[x \dots y]$ is a path such that x and y are the input and output of G , then $[x \dots y]$ will be referred to as a *complete path*. The set of complete paths from x to y will be called a *complete connection* from x to y in G .

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 $\sigma(x, y) = \sigma \langle x, y \rangle$, $\text{cer}(x, y) = \text{cer} \langle x, y \rangle$ and $\text{cov}(x, y) = \text{cov} \langle x, y \rangle$ then we have a new flow graph G' with the property: $\sigma(G) = \sigma(G')$. G' is called a *combined flow graph*.

Starting from a flow graph, if we invert the direction of all branches in G , then the resulting graph G^{-1} will be called the *inverted graph* of G (or the *inverse flow graph* of G) [14]. Essentially, three coefficients of an inverse flow graph can be computed from its flow graph as follows: $\sigma_{G^{-1}}(y, x) = \sigma_G(x, y)$, $\text{cer}_{G^{-1}}(y, x) = \text{cov}_G(x, y)$ and $\text{cov}_{G^{-1}}(y, x) = \text{cer}_G(x, y)$.

4 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 $G = (A, B, \sigma)$, where $A = \{A_{i_1}, A_{i_2}, \dots, A_{i_n}\}$ is a set of attributes¹, 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_{i_k} 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_{i_k} contains n_{i_k} values, we say that it contains n_{i_k} nodes.

We now consider how to approximate an attribute value $Y \in A_{i_{k+1}}$ from attribute values of A_{i_k} where $A_{i_k} = \{X_1, X_2, \dots, X_{n_{i_k}}\}$, 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. [11] Let $G = (A, B, \sigma)$ be a normalized flow graph, $A_{i_k} = \{X_1, X_2, \dots, X_{n_{i_k}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{i_{k+1}}$. For any branch (X_j, Y) , $j \in \{1, \dots, n_{i_k}\}$, of the flow graph G , the union of all inputs X_j of Y is the upper approximation of Y (denoted $\overline{A_{i_k}}(Y)$), the union of all inputs X_j of Y , such that $\text{cer}(X_j, Y) = 1$, is the lower approximation

¹ In what follows, we regard $\mathcal{N}$ as A for simplicity.

of Y (denoted $A_{l_i}(Y)$). Moreover, the union of all inputs X_j of Y , such that $\text{cer}(X_j, Y) < 1$, 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 flow graph.

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

- $\exists X_j [\text{cer}(X_j, Y) = 1]$ and $\exists X_j [X_j \notin I(Y)]$ iff Y is roughly A_{l_i} -definable,
- $\forall X_j [\text{cer}(X_j, Y) \neq 1]$ and $\exists X_j [X_j \notin I(Y)]$ iff Y is internally A_{l_i} -indefinable,
- $\exists X_j [\text{cer}(X_j, Y) = 1]$ and $\forall X_j [X_j \in I(Y)]$ iff Y is externally A_{l_i} -definable,
- $\forall X_j [\text{cer}(X_j, Y) \neq 1]$ 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 $\neg Y^2$, 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 $\neg Y$, 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 $\neg Y$ 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 $\neg Y$, using A_{l_i} .

Property 1. Let $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$, $2 \leq i \leq k$, be an attribute in layer i and W be a node in $A_{l_{i-1}}$. For any branch (X_j, W) , $j \in \{1, \dots, n_{l_i}\}$ in the inverse flow graph of G , the union of all output X_j of W in flow graph G is the upper approximation of W , the union of all outputs X_j of W in a flow graph G , such that $\text{cov}(W, X_j) = 1$, is the lower approximation of W . Moreover, the union of all outputs X_i of W , such that $\text{cov}(W, X_j) < 1$, 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. $\square$

Example. Suppose we are given the flow graph for the preference analysis problem depicted in Fig. 1, that 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

² Where $\neg Y = U - Y$.

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.

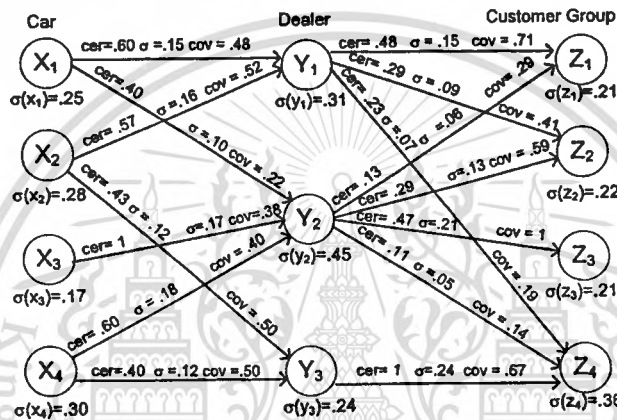


Fig. 1. A normalized flow graph

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 quantitatively. 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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j, Y) , $j \in \{1, \dots, n_{l_i}\}$, of G , the accuracy of approximation, $\alpha_{A_{l_i}}(Y)$, is defined as: $\alpha_{A_{l_i}}(Y) = \frac{\text{card}(A_{l_i}(Y))}{\text{card}(A_{l_i}(Y))}$.

We can use the accuracy of approximation to specify the quality of an approximation. Obviously, $0 \leq \alpha_B(X) \leq 1$. 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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_i, Y) , $i \in \{1, \dots, n_{l_i}\}$, of G , the roughness of approximation, $\gamma_{A_{l_i}}(Y)$, is defined as: $\gamma_{A_{l_i}}(Y) = 1 - \alpha_{A_{l_i}}(Y) = \frac{\text{card}(\overline{A_{l_i}}(Y)) - \text{card}(A_{l_i}(Y))}{\text{card}(\overline{A_{l_i}}(Y))}$.

Property 2. Let $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$, $1 \leq i \leq k-1$, be an attribute in layer i and Y be a node in $A_{l_{i+1}}$. For any branch (X_j, Y) , $j \in \{1, \dots, n_{l_i}\}$, of G , we have

$$(1) \alpha_{A_{l_i}}(Y) = \frac{\sum_{\text{cer}(X_j, Y)=1} \sigma(X_j)}{\sum_{X_j \in I(Y)} \sigma(X_j)} \text{ and } (2) \gamma_{A_{l_i}}(Y) = \frac{\sum_{\text{cer}(X_j, Y)<1} \sigma(X_j)}{\sum_{X_j \in I(Y)} \sigma(X_j)}.$$

Proof. (1) From Definition 5, we have $\text{card}(A_{l_i}(Y)) = \sum_{\text{cer}(X_j, Y)=1} \text{card}(X_j)$ and $\text{card}(\overline{A_{l_i}}(Y)) = \sum_{X_j \in I(Y)} \text{card}(X_j)$. Since $\text{card}(X_j) = \varphi(X_j) = \sigma(X_j)\varphi(G) = \sigma(X_j)\varphi(U)$ and by Definition 7, then $\alpha_B(Y) = \frac{\sum_{\text{cer}(X_j, Y)=1} \sigma(X_j)}{\sum_{X_j \in I(Y)} \sigma(X_j)}$.

(2) It can be proved similarly to (1). $\square$

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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$, $2 \leq i \leq k$, be an attribute in layer i and W be a node in $A_{l_{i-1}}$. For any branch (X_j, W) , $j \in \{1, \dots, n_{l_i}\}$ in the inverse flow graph of G , we have

$$(1) \alpha_{A_{l_i}}(W) = \frac{\sum_{\text{cov}(W, X_j)=1} \sigma(X_j)}{\sum_{X_j \in O(W)} \sigma(X_j)} \text{ and } (2) \gamma_{A_{l_i}}(W) = \frac{\sum_{\text{cov}(W, X_j)<1} \sigma(X_j)}{\sum_{X_j \in O(W)} \sigma(X_j)}.$$

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

(2) It can be proved similarly to (1). $\square$

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 ($\text{cer}(Y_3, Z_4) = 1$). There is only one branch (Y_3, Z_4) with $\text{cer}(Y_3, Z_4) = 1$ ³. Thus, by Property 2(1), we have $\alpha_Y(Z_1) = \alpha_Y(Z_2) = \alpha_Y(Z_3) = 0$ and $\alpha_Y(Z_4) = \frac{\sum_{\text{cer}(Y_i, Z_4)=1} \sigma(Y_i)}{\sum_{Y_i \in I(Z_4)} \sigma(Y_i)} = \frac{\sigma(Y_3)}{\sigma(Y_1) + \sigma(Y_2) + \sigma(Y_3)} = 0.24$.

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

³ By employing the approach presented in our previous study [3], 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.

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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$ and $A_{l_{i+1}} = \{Y_1, Y_2, \dots, Y_{n_{l_{i+1}}}\}$, $1 \leq i \leq k$, be any two adjacent layers. $A_{l_{i+1}}$ depends on A_{l_i} to a degree $k_{A_{l_i}}(A_{l_{i+1}}) = \frac{\sum_{j=1}^{n_{l_{i+1}}} \text{card}(A_{l_i}(Y_j))}{\text{card}(U)}$.

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 [14]. 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 $G = (A, B, \sigma)$ be a flow graph, $A_{l_i} = \{X_1, X_2, \dots, X_{n_{l_i}}\}$ and $A_{l_{i+1}} = \{X_1, X_2, \dots, X_{n_{l_{i+1}}}\}$, $1 \leq i \leq k$, be any two adjacent layers. $A_{l_{i+1}}$ depends on A_{l_i} to a degree $k_{A_{l_i}}(A_{l_{i+1}}) = \sum_{\text{cer}(X_i, X_j)=1} \sigma(X_i)$.

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

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

Proof. It can be proved similarly as Property 4 $\square$

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 $\gamma_X(Y) = \sum_{\text{cer}(X_i, Y_j)=1} \sigma(X_i) = \sigma(X_3) = 0.17$. 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 $\gamma_Z(Y) = \sum_{\text{cov}(Y_i, Z_j)=1} \sigma(Z_i) = \sigma(Z_3) = 0.21$. These results give a conclusion that dealers depend on customer groups more than models of cars.

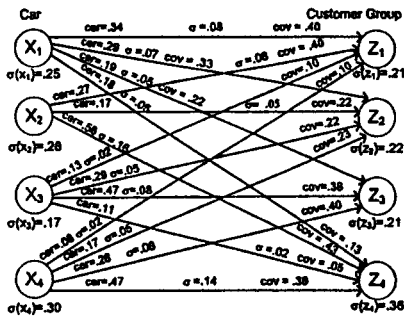


Fig. 2. A combined flow graph

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*.

5 Conclusion

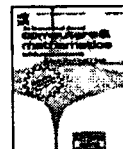
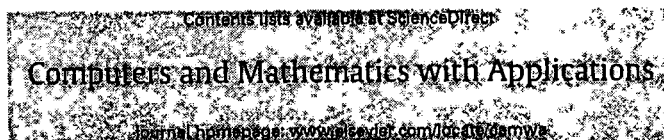
In this paper, we introduce definitions and properties of rough set approximations, accuracy and roughness of approximation which are defined in terms of a flow graph. They can be useful when the initial data is in the form of flow graph and contains some limitations. We illustrate a car dealer preference analysis to support our propositions.

Acknowledgments

This paper was supported by the grant MRG5180071 from the Thailand Research Fund and the Commission on Higher Education and King Mongkut's Institute of Technology Ladkrabang. Thanks also to G.M. Zaverucha.

References

1. Butz, C.J., Yen, W., Yang, B.: The Computational Complexity of Inference Using Rough Set Flow Graphs. In: Ślęzak, D., Wang, G., Szczuka, M.S., Düntsch, I., Yao, Y. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3641, pp. 335–344. Springer, Heidelberg (2005)
2. Czyzewski, A., Kostek, B.: Musical Metadata Retrieval with Flow Graphs. In: Tsumoto, S., Słowiński, R., Komorowski, J., Grzymała-Busse, J.W. (eds.) RSCTC 2004. LNCS (LNAI), vol. 3066, pp. 691–698. Springer, Heidelberg (2004)
3. Chitcharone, D., Pattaraintakorn, P.: Knowledge Discovery by Rough Sets Mathematical Flow Graphs and its Extension. In: Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria, pp. 340–345 (2008)
4. Chitcharone, D., Pattaraintakorn, P.: Towards Theories of Fuzzy Set and Rough Set to Flow Graphs. In: The 2008 IEEE World Congress on Computational Intelligence, pp. 1675–1682. IEEE Press, Hong Kong (2008)
5. Liu, H., Sun, J., Zhang, H.: Interpretation of Extended Pawlak Flow Graphs Using Granular Computing. In: Peters, J.F., Skowron, A. (eds.) Transactions on Rough Sets VIII. LNCS, vol. 5084, pp. 93–115. Springer, Heidelberg (2008)
6. Matusiewicz, Z., Pancerz, K.: Rough Set Flow Graphs and Max $\rightarrow$ Fuzzy Relation Equations in State Prediction Problems. In: Chan, C.-C., Grzymała-Busse, J.W., Ziarko, W.P. (eds.) RSCTC 2008. LNCS (LNAI), vol. 5306, pp. 359–368. Springer, Heidelberg (2008)
7. Pattaraintakorn, P., Cercone, N., Naruedomkul, K.: Rule Learning: Ordinal Prediction Based on Rough Set and Soft-Computing. Appl. Math. Lett. 19(12), 1300–1307 (2006)
8. Pattaraintakorn, P.: Entropy Measures of Flow Graphs with Applications to Decision Trees. In: Wen, P., Li, Y., Polkowski, L., Yao, Y., Tsumoto, S., Wang, G. (eds.) RSKT 2009. LNCS, vol. 5589, pp. 618–625. Springer, Heidelberg (2009)
9. Pawlak, Z.: Rough Sets. Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
10. Pawlak, Z.: Rough Sets, Decision Algorithms and Bayes' Theorem. European J. of Oper. Res. 136, 181–189 (2002)
11. Pawlak, Z.: Rough Set and Flow Graphs. In: Ślęzak, D., Wang, G., Szczuka, M.S., Düntsch, I., Yao, Y. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3641, pp. 1–11. Springer, Heidelberg (2005)
12. Pawlak, Z.: Inference Rules and Decision Rules. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 102–108. Springer, Heidelberg (2004)
13. Pawlak, Z.: Decision Trees and Flow Graphs. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNCS (LNAI), vol. 4259, pp. 1–11. Springer, Heidelberg (2006)
14. Pawlak, Z.: Flow Graphs and Data Mining. In: Peters, J.F., Skowron, A. (eds.) Transactions on Rough Sets III. LNCS, vol. 3400, pp. 1–36. Springer, Heidelberg (2005)
15. Pawlak, Z., Skowron, A.: Rudiments of Rough Sets. Inform. Sciences 177, 3–20 (2007)



Novel matrix forms of rough set flow graphs with applications to data integration

Doungtrat Chitcharoen, Puntip Pattaraintakorn*

Department of Mathematics, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, 10520, Thailand

ARTICLE INFO

Article history:
Received 27 March 2009
Received in revised form 11 August 2010
Accepted 27 September 2010

Keywords:
Flow graphs
Rough sets
Matrix
Data integration

ABSTRACT

Pawlak's flow graphs have attracted both practical and theoretical researchers because of their ability to visualize information flow. In this paper, we invent a new schema to represent throughflow of a flow graph and three coefficients of both normalized and combined normalized flow graphs in matrix form. Alternatively, starting from a flow graph with its throughflow matrix, we reform Pawlak's formulas to calculate these three coefficients in flow graphs by using matrix properties. While traditional algorithms for computing these three coefficients of the connection are exponential in l , an algorithm using our matrix representation is polynomial in l , where l is the number of layers of a flow graph. The matrix form can simplify computation, improve time complexity, alleviate problems due to missing coefficients and hence help to widen the applications of flow graphs.

Practically, data sets often reside at different sources (heterogeneous data sources). Their individual analysis at each source is inadequate and requires special treatment. Hence, we introduce a composition method for flow graphs and corresponding formulas for calculating their coefficients which can omit some data sharing. We provide a real-world experiment on the Promotion of Academic Olympiads and Development of Science Education Foundation (POSNE) data set which illustrates a desirable outcome and the advantages of the proposed matrix forms and the composition method.

© 2010 Elsevier Ltd. All rights reserved.

0. Introduction

Rough set theory was introduced by Zdzislaw Pawlak in 1982 [1]. The crux of rough set theory is the novel use of approximations to cope with uncertainty. In 2002, Pawlak introduced a mathematical flow graph (rough set flow graph) which is a significant extension of rough set theory [2]. 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 $\text{cer}(x, y) = 1$ as the lower approximation of y [3]. This is different from the flow networks introduced by Ford et al. that do not analyze optimal flow [4]. A flow graph represents the mathematical model of information flow from the given data set. The branches of a flow graph can be interpreted as decision rules [2]. In 2003, Pawlak revealed the relationship between flow graphs and probability. From the concepts of modus ponens and modus tollens, Pawlak replaced the truth values by their corresponding probabilities and obtained rough modus ponens and rough modus tollens [5]. Later, Pawlak discovered that certainty and coverage in a flow graph satisfy Bayes' theorem, without referring to either prior or posterior probabilities [6]. Pawlak also proposed a new approach to data mining and knowledge discovery based on information flow distribution [7]. There is a unique flow graph for every decision tree, which can be constructed by removing the root while its nodes are labeled with the same attribute [8].

Within the past decade, there has been much research on 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 wider in scope and more complex. In 2002, Greco et al. relaxed the concept of mutual exclusion of decision rules

* Corresponding author.

E-mail addresses: s0067111@kmitl.ac.th (D. Chitcharoen), kppuntip@kmitl.ac.th, puntip@cs.dal.ca (P. Pattaraintakorn).

and showed that the properties of decision algorithms still hold [9]. In 2004, Czyzewski and Kostek applied flow graphs to knowledge extraction from the Compact Disc Database (CDDb) [10]. In 2006, Butz et al. showed that the traditional rough set flow graph inference algorithm has exponential time complexity. They proposed a new rough set flow graph inference algorithm that exploits the factorization [11]. This year, Pattaraintakorn et al. represented rough sets rule learning for ordinal prediction based on rough sets representation of the rules [12]. Since Pawlak's flow graph is a quantificational graph, i.e., it represents relations among nodes using quantity of flow, it cannot describe exactly the characteristics of decision systems. Sun et al. introduced an extension of flow graphs to solve this problem [13]. The concept of fuzzy flow graphs was introduced in 2006 by Mieszkowicz-Rolka and Rolka [14]. They allowed decision tables with fuzzy attributes to be analyzed and interpreted in terms of flow graphs. In 2007, Suraj and Pancerz used flow graphs as a tool for rule mining in consecutive time windows of a temporal information system [15]. Chan and Tsumoto presented a rule learning algorithm for certain boundary rules for flow graphs [16]. In 2008, Liu et al. developed a model for an extended flow graph by using granular computing [17]. In the same year, Matusiewicz and Pancerz combined flow graphs and max product composition of fuzzy relation equations in state prediction problems [18].

In our previous studies, we proposed a new extension of flow graphs to association rules in data mining. We revealed the relationship between support and confidence of association rules and flow graphs i.e., for every path xyz if $\text{cer}(y, z) = 1$, then $\text{sup}[xyz] = \sigma[xyz]$ and $\text{conf}[xyz] = 1$ [19]. We also extended a method for calculating the certainty, coverage and strength coefficients of fuzzy flow graphs under several conditions [20]. Furthermore, we proposed a relaxation of the mutual exclusion property of the decision algorithm and computed its certainty and coverage coefficients [20].

Pawlak's flow graph is a useful tool for knowledge discovery and has been successfully applied in many areas. Nevertheless, representing a flow graph in matrix form is still an open problem. From a practical point of view, matrix notation can be useful in the processing of large data sets. Thus, in this paper we define throughflow of a flow graph and three coefficients of a normalized and a combined normalized flow graph in matrix form. Consequently, matrix properties are employed to calculate coefficients and discover new properties of such flow graphs. Traditional algorithms for computing three coefficients of the connection are exponential in l whereas an algorithm using our matrix representation is polynomial in l , where l is the number of layers of a flow graph.

More importantly, in real-world databases, data sets may reside at different sources (heterogeneous data sources) which requires special treatment. To analyze data in this format, data integration combines data residing at different sources, and provides the user with a unified view of these data. The problem of designing data integration systems is important in current real-world applications and is characterized by a number of issues that are interesting from a theoretical point of view [21]. Thus, we introduce a new composition of flow graphs and derive formulas for calculating their coefficients. We conduct a real-world experiment on the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN) data set with sensitivity, specificity and accuracy analysis and model selection.

This article is organized as follows. In Section 1 we present preliminary definitions of flow graphs. Section 2 contains our definitions for representing the throughflow of a flow graph and coefficients related to flow graphs in matrix forms. In this section, we also introduce new formulas for calculating such coefficients by using matrix properties directly. We establish the composition of flow graphs from heterogeneous data sources and formulas to calculate the corresponding coefficients in Section 3. The actual data set from the POSN and the experimental results are presented in Section 4.

1. Pawlak's flow graphs

Flow graphs were introduced by Pawlak in 2002 [2]. In this section, we recall some concepts of flow graphs from [2,3,5–9].

A flow graph is a directed, acyclic, finite graph $G = (N, B, \varphi)$, where N is a set of nodes, $B \subseteq N \times N$ is a set of directed branches, $\varphi : B \rightarrow \mathbb{R}^+$ is a flow function and $\mathbb{R}^+$ is the set of non-negative real numbers. If $(x, y) \in B$ then x is an input of node y denoted by $I(y)$ and y is an output of node x denoted by $O(x)$. Next, the input and output of a flow graph G are defined respectively by $I(G) = \{x \in N | I(x) = \emptyset\}$ and $O(G) = \{x \in N | O(x) = \emptyset\}$. These inputs and outputs of G are called external nodes of G whereas other nodes are called internal nodes of G . If $(x, y) \in B$ then we call $\varphi(x, y)$ a throughflow from x to y . We will assume in what follows that $\varphi(x, y) \neq 0$ for every $(x, y) \in B$.

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 [3], the above equations can be considered as flow conservation equations [4].

1.1. Normalized flow graphs

In order to demonstrate interesting relationships between flow graphs and other disciplines, we come to the normalized version of flow graphs.

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 $\sigma : B \rightarrow [0, 1]$ is a normalized flow of (x, y) . The strength of (x, y) where $0 \leq \sigma(x, y) \leq 1$ is

$$\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)}. \quad (1)$$

With every node x of a flow graph G , the associated normalized inflow and outflow are defined as $\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(y, x)$ and $\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x, y)$. For any internal node x , it holds that $\sigma_+(x) = \sigma_-(x) = \sigma(x)$, where $\sigma(x)$ is a normalized throughflow of x . Similarly, the 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 $\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(G)} \sigma_+(x)$. It also holds that $\sigma_+(G) = \sigma_-(G) = \sigma(G) = 1$. With every branch (x, y) of a flow graph G , the certainty and the coverage of (x, y) are defined respectively as

$$\text{cer}(x, y) = \frac{\sigma(x, y)}{\sigma(x)}, \quad (2)$$

$$\text{cov}(x, y) = \frac{\sigma(x, y)}{\sigma(y)}, \quad (3)$$

where $\sigma(x), \sigma(y) \neq 0$.

In accordance with the previous notions, here are some consequential properties: $\sum_{y \in O(x)} \text{cer}(x, y) = 1$ and $\sum_{x \in I(y)} \text{cov}(x, y) = 1$. In addition, $\sigma(x) = \sum_{y \in O(x)} \text{cer}(x, y) \sigma(x) = \sum_{y \in O(x)} \sigma(x, y)$ and $\sigma(y) = \sum_{x \in I(y)} \text{cov}(x, y) \sigma(y) = \sum_{x \in I(y)} \sigma(x, y)$ have the form of a total probability theorem, whereas $\text{cer}(x, y) = \frac{\text{cov}(x, y) \sigma(y)}{\sigma(x)}$ and $\text{cov}(x, y) = \frac{\text{cer}(x, y) \sigma(x)}{\sigma(y)}$ are Bayes' rules [3,7].

1.2. Paths and connections

In a flow graph G , if we concentrate on relationships of a sequence of nodes in a flow graph, we can find these relationships using the concept of paths.

A (directed) path from x to y ($x \neq y$) in G , denoted by $\{x \dots y\}$, is a sequence of nodes $x_1, \dots, x_n$ 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 $\{x_1 \dots x_n\}$ are defined as $\text{cer}\{x_1 \dots x_n\} = \prod_{i=1}^{n-1} \text{cer}(x_i, x_{i+1})$, $\text{cov}\{x_1 \dots x_n\} = \prod_{i=1}^{n-1} \text{cov}(x_i, x_{i+1})$ and $\sigma\{x_1 \dots x_n\} = \sigma(x_1) \text{cer}\{x_1 \dots x_n\} = \sigma(x_n) \text{cov}\{x_1 \dots x_n\}$.

Next, since flow graphs are composed of internal and external nodes, if we only focus on input and output, then we require the concept of a connection.

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 nodes x and y . For every connection $\langle x, y \rangle$, its associated certainty, coverage and strength of the connection $\langle x, y \rangle$ are

$$\text{cer}\langle x, y \rangle = \sum_{\{x \dots y\} \in \langle x, y \rangle} \text{cer}\{x \dots y\}, \quad (4)$$

$$\text{cov}\langle x, y \rangle = \sum_{\{x \dots y\} \in \langle x, y \rangle} \text{cov}\{x \dots y\}, \quad (5)$$

$$\sigma\langle x, y \rangle = \sum_{\{x \dots y\} \in \langle x, y \rangle} \sigma\{x \dots y\} = \sigma(x) \text{cer}\langle x, y \rangle = \sigma(y) \text{cov}\langle x, y \rangle. \quad (6)$$

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

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 $\sigma(x, y) = \sigma\langle x, y \rangle$, $\text{cer}(x, y) = \text{cer}\langle x, y \rangle$ and $\text{cov}(x, y) = \text{cov}\langle x, y \rangle$ then we have a new flow graph with the property $\sigma(G) = \sigma(G')$. This is called the combined flow graph.

1.3. An illustrative example

In this section, we illustrate the above ideas by means of a simple example (a modification of [7]). It is a voting analysis problem as depicted in Fig. 1.

Suppose we are given three disjoint age groups of voters: y_1 (old), y_2 (middle aged) and y_3 (young). They belong to three social classes x_1 (high), x_2 (middle) and x_3 (low). The voters can vote for three political parties z_1, z_2 and z_3 . Assume that voters are grouped according to social classes, age groups and political parties as shown in Fig. 1. We can give a brief explanation as follows. Inputs of this flow graph are nodes x_1, x_2 and x_3 , whereas the outputs are nodes z_1, z_2 and z_3 . Nodes y_1, y_2 and y_3 are internal nodes. Next, if we consider the bold branch in Fig. 1, the throughflow of the branch (x_2, y_1) is $\varphi(x_2, y_1) = 21$. Therefore 21 voters are middle class and old. The inflow of node y_1 is $\varphi_+(y_1) = 27$ and outflow is $\varphi_-(y_1) = 27$.

The normalized flow graph version of Fig. 1 is depicted in Fig. 2. There are three layers in this flow graph. The first layer consists of nodes x_1, x_2 and x_3 , the second layer consists of nodes y_1, y_2 and y_3 and the third layer consists of nodes z_1, z_2 and z_3 .

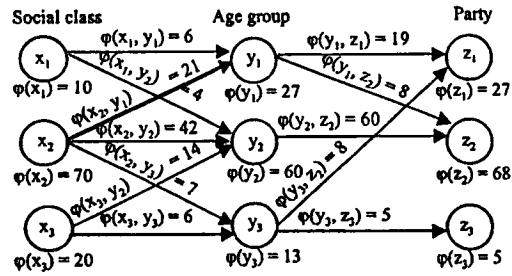


Fig. 1. Flow graph initial data for the voting problem [19].

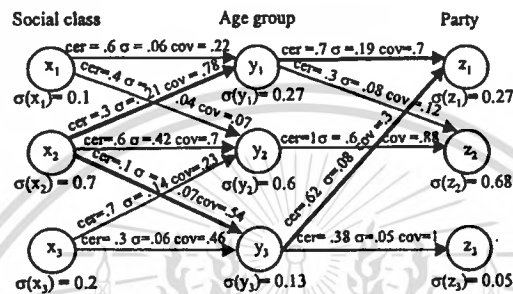


Fig. 2. Social class, age and party votes normalized flow graph.

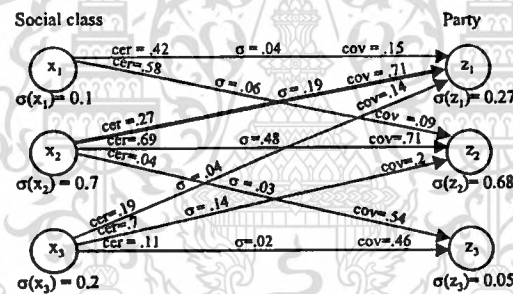


Fig. 3. Combined normalized flow graph of Fig. 2.

To illustrate, we present some examples of how to compute three coefficients of this flow graph. We can calculate strength by using (1), e.g., the bold branches indicate that $\sigma(x_2, y_1) = 0.21$ which means that 21% of the voters are middle class and old. We can calculate the certainties by using (2), e.g., $\text{cer}(y_1, z_1) = 0.7$ and $\text{cer}(y_1, z_2) = 0.3$; these can be interpreted as indicating that 70% of the old age group (y_1) vote for party z_1 and the other 30% vote for party z_2 . Finally, we can calculate the coverage by using (3), e.g., $\text{cov}(y_1, z_2) = 0.12$ and $\text{cov}(y_2, z_2) = 0.88$, which can be interpreted as indicating that 12% of the votes for party z_2 come from the old age group (y_1) and the rest (88%) come from the middle age group (y_2).

Next, to focus on vote distribution between parties and social classes, we construct the combined normalized flow graph as depicted in Fig. 3. It comes from all of the complete connections between parties and social classes in Fig. 2. We can calculate the certainty, coverage and strength of the connections in Fig. 3 by using (4)–(6), respectively. For instance, the connection (x_2, z_1) consists of paths $[x_2 y_1 z_1]$ and $[x_2 y_3 z_1]$ (bold paths in Fig. 2). Then we can calculate the certainty, coverage and strength of (x_2, z_1) : $\text{cer}(x_2, z_1) = 0.27$, $\text{cov}(x_2, z_1) = 0.71$ and $\sigma(x_2, z_1) = 0.19$. This can be interpreted as indicating that 27% of the middle class vote for party z_1 , 71% of party z_1 's votes come from the middle class and 19% of voters who are middle class vote for party z_1 , respectively.

2. Representation of flow graphs in matrix form

The study of matrices has established theories and algorithms for solving or analyzing problems and applications in many areas (e.g., systems of linear equations, differential equations, economics, graph theory, communication networks, electrical networks etc. [22]). Nevertheless, research on presenting flow graphs in matrix form is scarce. In this section, we first define throughflow of a flow graph and a normalized flow graph with their coefficients in matrix form. Then, we apply known results on matrices in order to simplify and analyze three coefficients of these flow graphs at the end of this section.

In Definitions 1–5, we offer new methods for representing throughflow and three coefficients in matrix form directly from an initial flow graph.

Definition 1. Let $G = (N, B, \varphi)$ be a flow graph, let X be an attribute with n values: $x_1, x_2, \dots, x_n$, and let Y be an attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n$, $1 \leq j \leq m$. The throughflow of G is represented in matrix form as

$$G_{\varphi G}(x_i, y_j) = [\varphi_{ij}]_{n \times m}, \quad \varphi_{ij} = \varphi(x_i, y_j),$$

where $1 \leq i \leq n$ and $1 \leq j \leq m$.

Definition 2. Let $G = (N, B, \sigma)$ be a normalized flow graph, let X be an attribute with n values: $x_1, x_2, \dots, x_n$, and let Y be an attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n$, $1 \leq j \leq m$. The strength, certainty and coverage of G are represented in matrix form as:

1. $G_{\sigma G}(x_i, y_j) = [\sigma_{ij}]_{n \times m}$, $\sigma_{ij} = \sigma(x_i, y_j)$.
2. $G_{\text{cer}G}(x_i, y_j) = [\text{cer}_{ij}]_{n \times m}$, $\text{cer}_{ij} = \text{cer}(x_i, y_j)$.
3. $G_{\text{cov}G}(x_i, y_j) = [\text{cov}_{ij}]_{n \times m}$, $\text{cov}_{ij} = \text{cov}(x_i, y_j)$.

where $1 \leq i \leq n$ and $1 \leq j \leq m$.

Definitions 1 and 2 are the representations of throughflow and the strength, certainty and coverage of directed branches in connected layers in matrix form. Let us give a brief observation.

- For matrices $G_{\varphi G}(x_i, y_j)$, $G_{\sigma G}(x_i, y_j)$, $G_{\text{cer}G}(x_i, y_j)$ and $G_{\text{cov}G}(x_i, y_j)$ the number of rows, n , is equal to the number of attribute values of X and the number of columns, m , is equal to the number of attribute values of Y .
- For $G_{\varphi G}(x_i, y_j)$, we see that the summation of the i th row is an outflow of x_i while the summation of the j th column is an inflow of y_j , for all i, j .
- The summation of all elements in $G_{\varphi G}(x_i, y_j)$ equals the throughflow of the flow graph.
- For a matrix $G_{\sigma G}(x_i, y_j)$, the summation of the i th row is a normalized outflow of x_i while the summation of the j th column is a normalized inflow of y_j , $1 \leq i \leq n$, $1 \leq j \leq m$ and the summation of all elements in the matrix is 1.
- The summation of any row in $G_{\text{cer}G}(x_i, y_j)$ is 1 and the summation of any column in $G_{\text{cov}G}(x_i, y_j)$ is 1.

To illustrate, from Fig. 1, we can represent the throughflow from x_i to y_j , $1 \leq i, j \leq 3$, for all sets of connected branches between the first and the second layers in the matrix form denoted by $G_{\varphi G}(x_i, y_j)$ as $G_{\varphi G}(x_i, y_j) = \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix}$. Similarly, the strength, certainty and coverage of (x_i, y_j) , denoted by $G_{\sigma G}(x_i, y_j)$, $G_{\text{cer}G}(x_i, y_j)$ and $G_{\text{cov}G}(x_i, y_j)$, of the normalized flow graph in Fig. 2 in matrix form are $G_{\sigma G}(x_i, y_j) = \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix}$, $G_{\text{cer}G}(x_i, y_j) = \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix}$ and $G_{\text{cov}G}(x_i, y_j) = \begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix}$.

After considering matrices representing throughflow, strength, certainty and coverage for connected layers, we extend to a flow graph with more than two layers. Definitions 3 and 4 generalize the previous definition to a flow graph and a normalized flow graph with $n-1$ condition attributes and one decision attribute. We use $\diamond$ to denote a (binary) concatenation operator for any connected layer. Note that $\diamond$ is similar to a concatenation operator, $||$, in computer networks.

Definition 3. Let $G = (N, B, \varphi)$ be a flow graph, X^i be a condition attribute with k_i values that are $x_1^i, x_2^i, \dots, x_{k_i}^i$, $1 \leq i \leq n-1$, and X^n be a decision attribute with k_n values that are $x_1^n, x_2^n, \dots, x_{k_n}^n$, such that $(x_j^i, x_{j+1}^{i+1}) \in B$, $1 \leq i \leq n-1$, $1 \leq j_1 \leq k_i$. Then throughflow of G is represented in matrix form as

$$G_{\varphi G} = [\varphi_{j_1 j_2}^{1,2}]_{k_1 \times k_2} \diamond [\varphi_{j_2 j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [\varphi_{j_{n-1} j_n}^{n-1,n}]_{k_{n-1} \times k_n};$$

$$\varphi_{j_1 j_2}^{1,2} = \varphi(x_{j_1}^1, x_{j_2}^2), \varphi_{j_2 j_3}^{2,3} = \varphi(x_{j_2}^2, x_{j_3}^3), \dots, \varphi_{j_{n-1} j_n}^{n-1,n} = \varphi(x_{j_{n-1}}^{n-1}, x_{j_n}^n),$$

where $1 \leq j_n \leq k_n$.

Definition 4. Let $G = (N, B, \sigma)$ be a normalized flow graph, X^i be a condition attribute with k_i values that are $x_1^i, x_2^i, \dots, x_{k_i}^i$, $1 \leq i \leq n-1$, and X^n be a decision attribute with k_n values that are $x_1^n, x_2^n, \dots, x_{k_n}^n$, such that $(x_j^i, x_{j+1}^{i+1}) \in B$, $1 \leq i \leq n-1$, $1 \leq j_1 \leq k_i$. Then strength, certainty and coverage of G are represented in matrix form as

follows:

1. $G_{\sigma_G} = [\sigma_{j_1 j_2}^{1,2}]_{k_1 \times k_2} \diamond [\sigma_{j_2 j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [\sigma_{j_{n-1} j_n}^{n-1,n}]_{k_{n-1} \times k_n}$;
 $\sigma_{j_1 j_2}^{1,2} = \sigma(x_{j_1}^1, x_{j_2}^2), \sigma_{j_2 j_3}^{2,3} = \sigma(x_{j_2}^2, x_{j_3}^3), \dots, \sigma_{j_{n-1} j_n}^{n-1,n} = \sigma(x_{j_{n-1}}^{n-1}, x_{j_n}^n).$
2. $G_{\text{cer}_G} = [\text{cer}_{j_1 j_2}^{1,2}]_{k_1 \times k_2} \diamond [\text{cer}_{j_2 j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [\text{cer}_{j_{n-1} j_n}^{n-1,n}]_{k_{n-1} \times k_n}$;
 $\text{cer}_{j_1 j_2}^{1,2} = \text{cer}(x_{j_1}^1, x_{j_2}^2), \text{cer}_{j_2 j_3}^{2,3} = \text{cer}(x_{j_2}^2, x_{j_3}^3), \dots, \text{cer}_{j_{n-1} j_n}^{n-1,n} = \text{cer}(x_{j_{n-1}}^{n-1}, x_{j_n}^n).$
3. $G_{\text{cov}_G} = [\text{cov}_{j_1 j_2}^{1,2}]_{k_1 \times k_2} \diamond [\text{cov}_{j_2 j_3}^{2,3}]_{k_2 \times k_3} \diamond \dots \diamond [\text{cov}_{j_{n-1} j_n}^{n-1,n}]_{k_{n-1} \times k_n}$;
 $\text{cov}_{j_1 j_2}^{1,2} = \text{cov}(x_{j_1}^1, x_{j_2}^2), \text{cov}_{j_2 j_3}^{2,3} = \text{cov}(x_{j_2}^2, x_{j_3}^3), \dots, \text{cov}_{j_{n-1} j_n}^{n-1,n} = \text{cov}(x_{j_{n-1}}^{n-1}, x_{j_n}^n).$

where $1 \leq j_n \leq k_n$.

For example, we can represent the throughflow of the entire flow graph in Fig. 1, and the strength, certainty and coverage of the entire normalized flow graph in Fig. 2 respectively as $G_{\sigma_G} = \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix} \diamond \begin{bmatrix} 19 & 8 & 0 \\ 0 & 60 & 0 \\ 8 & 0 & 5 \end{bmatrix}, G_{\sigma_G} = \begin{bmatrix} 0.08 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond$

$\begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.06 & 0.00 & 0.05 \end{bmatrix}, G_{\text{cer}_G} = \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.30 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.62 & 0.00 & 0.38 \end{bmatrix}$ and $G_{\text{cov}_G} = \begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.12 & 0.00 \\ 0.00 & 0.88 & 0.00 \\ 0.30 & 0.00 & 1.00 \end{bmatrix}.$

By Definitions 3 and 4, these entire flow graphs can be presented in very compact forms which are convenient to read and analyze. Two matrices joined by $\diamond$ illustrate three layers within the flow graph. The first and second layers are presented in the leftmost matrix while the second and third layers are presented in the rightmost matrix.

- When considering G_{σ_G} , we have $\varphi_+(y_1) = \varphi(x_1, y_1) + \varphi(x_2, y_1) + \varphi(x_3, y_1) = 6 + 21 + 0 = 27$ (leftmost matrix), and $\varphi_-(y_1) = \varphi(y_1, z_1) + \varphi(y_1, z_2) + \varphi(y_1, z_3) = 19 + 8 + 0 = 27$ (rightmost matrix).
- Similarly, in G_{σ_G} , $\sigma_+(y_1) = \sigma(x_1, y_1) + \sigma(x_2, y_1) + \sigma(x_3, y_1) = 0.06 + 0.21 + 0 = 0.27$, and $\sigma_-(y_1) = \sigma(y_1, z_1) + \sigma(y_1, z_2) + \sigma(y_1, z_3) = 0.19 + 0.08 + 0 = 0.27$.

Next, in Definition 5, we define the representations of the three coefficients of a combined flow graph in matrix form. The formulas for computing these coefficients directly from a normalized flow graph will be established later.

Definition 5. Let $G = (N, B, \sigma)$ be a combined normalized flow graph, X^1 be a condition attribute (input of G) with k_1 values that are $x_1^1, x_2^1, \dots, x_{k_1}^1$, and X^n be a decision attribute (output of G) with k_n values that are $x_1^n, x_2^n, \dots, x_{k_n}^n$. The strength, certainty and coverage of a combined normalized flow graph G can be represented in matrix form as

1. $G_{\sigma_G}(x_{j_1}^1, x_{j_n}^n) = [\sigma_{j_1 j_n}^{1,n}]_{k_1 \times k_n}$ where $\sigma_{j_1 j_n}^{1,n} = \sigma(x_{j_1}^1, x_{j_n}^n).$
2. $G_{\text{cer}_G}(x_{j_1}^1, x_{j_n}^n) = [\text{cer}_{j_1 j_n}^{1,n}]_{k_1 \times k_n}$ where $\text{cer}_{j_1 j_n}^{1,n} = \text{cer}(x_{j_1}^1, x_{j_n}^n).$
3. $G_{\text{cov}_G}(x_{j_1}^1, x_{j_n}^n) = [\text{cov}_{j_1 j_n}^{1,n}]_{k_1 \times k_n}$ where $\text{cov}_{j_1 j_n}^{1,n} = \text{cov}(x_{j_1}^1, x_{j_n}^n).$

where $1 \leq j_1 \leq k_1, 1 \leq j_n \leq k_n$.

For example, the strength, certainty and coverage of a combined normalized flow graph in Fig. 3 are $G_{\sigma_G}(x_{j_1}^1, x_{j_n}^n) =$

$$\begin{bmatrix} 0.04 & 0.06 & 0.00 \\ 0.19 & 0.48 & 0.03 \\ 0.04 & 0.14 & 0.02 \end{bmatrix}, G_{\text{cer}_G}(x_{j_1}^1, x_{j_n}^n) = \begin{bmatrix} 0.42 & 0.58 & 0.00 \\ 0.27 & 0.69 & 0.04 \\ 0.18 & 0.70 & 0.12 \end{bmatrix} \text{ and } G_{\text{cov}_G}(x_{j_1}^1, x_{j_n}^n) = \begin{bmatrix} 0.15 & 0.09 & 0.00 \\ 0.71 & 0.71 & 0.54 \\ 0.14 & 0.20 & 0.46 \end{bmatrix}.$$

Next, we define three kinds of matrices related to flow graphs. We will prove later that these matrices can, in fact, be included in the new formulas in order to compute the three coefficients of each branch and connection of a flow graph. Definitions 7 and 8 present the diagonal matrix of the throughflow and the inverse throughflow of an attribute set.

Definition 6. For a flow graph $G = (N, B, \sigma)$, let A and B be matrices representing coefficients of G . The matrix multiplication, $A \times B$, of matrices $A = [a_{ij}]_{m \times n}$ and $B = [b_{ij}]_{n \times p}$ is the matrix $C = [c_{ij}]_{m \times p}$, where $c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}$.

Definition 7. Let $G = (N, B, \sigma)$ be a normalized flow graph, and X be an attribute with n values: $x_1, x_2, \dots, x_n$. The diagonal matrix of normalized throughflow, denoted by $G_{\sigma(n)}$, $1 \leq i \leq n$, is represented by the matrix

$$G_{\sigma(n)} = [t_{ij}]_{n \times n}, \quad \text{where } t_{ij} = \begin{cases} \sigma(x_i), & i = j \\ 0, & i \neq j. \end{cases}$$

¹ For simplicity, we use AB instead of $A \times B$ in further discussion.

Definition 8. Let $G = (N, B, \sigma)$ be a normalized flow graph, and X be an attribute with n values: $x_1, x_2, \dots, x_n$. The diagonal matrix of inverse normalized throughflow, denoted by $G_{\frac{1}{\sigma(x_i)}}$, $1 \leq i \leq n$, is represented by the matrix

$$G_{\frac{1}{\sigma(x_i)}} = [f_{ij}]_{n \times n}, \quad \text{where } f_{ij} = \begin{cases} \frac{1}{\sigma(x_i)}, & i = j \\ 0, & i \neq j. \end{cases}$$

We compute these matrices from the normalized flow graph in Fig. 2. Its normalized throughflows are $\sigma(x_1) = 0.1$, $\sigma(x_2) = 0.7$ and $\sigma(x_3) = 0.2$, and the diagonal matrix of the normalized throughflow and the diagonal matrix of the inverse normalized throughflow are

$$G_{\sigma(x_i)} = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.7 & 0 \\ 0 & 0 & 0.2 \end{bmatrix} \quad \text{and} \quad G_{\frac{1}{\sigma(x_i)}} = \begin{bmatrix} \frac{1}{0.1} & 0 & 0 \\ 0 & \frac{1}{0.7} & 0 \\ 0 & 0 & \frac{1}{0.2} \end{bmatrix}.$$

Remark 1. From the traditional matrix inversion property, we obtain

$$G_{\frac{1}{\sigma(x_i)}}^{-1} = G_{\sigma(x_i)},$$

where $1 \leq i \leq k_i$.

So far we have introduced a new method for representing throughflow and three coefficients of normalized and combined normalized flow graphs in matrix form. Below, we prove a new method for calculating three coefficients of each branch in Theorems 1 and 2 and further extend to connections in Theorems 3 and 4 by using properties of matrices.

When we focus on the strength of each branch of a flow graph in matrix form, then we can apply (1) to calculate the strength of each branch solely by using properties of matrices. This is stated precisely in Theorem 1.

Theorem 1. Let $G = (N, B, \varphi)$ be a flow graph, let X be an attribute with n values: $x_1, x_2, \dots, x_n$, and let Y be an attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n$, $1 \leq j \leq m$. Then

$$G_{\sigma_G(x_i, y_j)} = \frac{1}{\varphi(G)} G_{\varphi_G(x_i, y_j)}.$$

Proof of Theorem 1. Consider $G_{\sigma_G(x_i, y_j)} = [\sigma_{ij}]_{n \times m}$, where $\sigma_{ij} = \sigma(x_i, y_j)$, $1 \leq i \leq n$ and $1 \leq j \leq m$. From (1), $G_{\sigma_G(x_i, y_j)} = \left[\frac{\varphi_{ij}}{\varphi(G)} \right]_{n \times m}$, where $\varphi_{ij} = \varphi(x_i, y_j)$, for all i, j . By scalar matrix multiplication, $G_{\sigma_G(x_i, y_j)} = \frac{1}{\varphi(G)} [\varphi_{ij}]_{n \times m}$. By Definition 1, then $G_{\sigma_G(x_i, y_j)} = \frac{1}{\varphi(G)} G_{\varphi_G(x_i, y_j)}$. $\square$

Example 1. Recall the flow graph given in Fig. 1. We represent the throughflow of this flow graph as $G_{\varphi_G} = \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix} \diamond$

$\begin{bmatrix} 19 & 8 & 0 \\ 0 & 60 & 0 \\ 8 & 0 & 5 \end{bmatrix}$. Given the fact that $\varphi(G) = 100$ and by Theorem 1, we can find the throughflow of each branch as $G_{\sigma_G} = \left(\frac{1}{100} \begin{bmatrix} 6 & 4 & 0 \\ 21 & 42 & 7 \\ 0 & 14 & 6 \end{bmatrix} \right) \diamond \left(\frac{1}{100} \begin{bmatrix} 19 & 8 & 0 \\ 0 & 60 & 0 \\ 8 & 0 & 5 \end{bmatrix} \right) = \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix}.$

This result corresponds to all strengths of the normalized flow graph given in Fig. 2. The leftmost matrix corresponds to strengths between the first and the second layers ($\sigma(x_i, y_j)$, $1 \leq i, j \leq 3$) while the rightmost matrix corresponds to strengths between the second and the third layers ($\sigma(y_j, z_k)$, $1 \leq j, k \leq 3$). Thus, starting from a flow graph G with its throughflow matrix, Theorem 1 gives a formula for computing the strength of the corresponding normalized flow graph directly. The following theorem gives formulas for calculating the certainty and coverage from the matrix form of flow graphs.

Theorem 2. Let $G = (N, B, \sigma)$ be a normalized flow graph, X be an attribute with n values: $x_1, x_2, \dots, x_n$, and Y be an attribute with m values: $y_1, y_2, \dots, y_m$, such that $(x_i, y_j) \in B$, $1 \leq i \leq n$, $1 \leq j \leq m$. Then

$$G_{\text{cer}_G(x_i, y_j)} = G_{\frac{1}{\sigma(x_i)}} G_{\sigma_G(x_i, y_j)} \quad \text{and}$$

$$G_{\text{cov}_G(x_i, y_j)} = G_{\sigma_G(x_i, y_j)} G_{\frac{1}{\sigma(y_j)}}.$$

Proof of Theorem 2. Consider $G_{\frac{1}{\sigma(y_j)}} G_{\sigma_C(x_i, y_j)}$; then by Definitions 2(1), 6 and 8, we obtain

$$\begin{aligned}
 G_{\frac{1}{\sigma(y_j)}} G_{\sigma_C(x_i, y_j)} &= \begin{bmatrix} \frac{1}{\sigma(x_1)} & 0 & \dots & 0 \\ 0 & \frac{1}{\sigma(x_2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\sigma(x_n)} \end{bmatrix} \begin{bmatrix} \sigma(x_1, y_1) & \sigma(x_1, y_2) & \dots & \sigma(x_1, y_m) \\ \sigma(x_2, y_1) & \sigma(x_2, y_2) & \dots & \sigma(x_2, y_m) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma(x_n, y_1) & \sigma(x_n, y_2) & \dots & \sigma(x_n, y_m) \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\sigma(x_1, y_1)}{\sigma(x_1)} & \frac{\sigma(x_1, y_2)}{\sigma(x_1)} & \dots & \frac{\sigma(x_1, y_m)}{\sigma(x_1)} \\ \frac{\sigma(x_2, y_1)}{\sigma(x_2)} & \frac{\sigma(x_2, y_2)}{\sigma(x_2)} & \dots & \frac{\sigma(x_2, y_m)}{\sigma(x_2)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma(x_n, y_1)}{\sigma(x_n)} & \frac{\sigma(x_n, y_2)}{\sigma(x_n)} & \dots & \frac{\sigma(x_n, y_m)}{\sigma(x_n)} \end{bmatrix}.
 \end{aligned}$$

By (2) and Definition 2(2), then $G_{\text{cer}_C(x_i, y_j)} = G_{\frac{1}{\sigma(y_j)}} G_{\sigma_C(x_i, y_j)}$.

Next, consider $G_{\sigma_C(x_i, y_j)} G_{\frac{1}{\sigma(y_j)}}$; then by Definitions 2(2), 6 and 8, we obtain

$$\begin{aligned}
 G_{\sigma_C(x_i, y_j)} G_{\frac{1}{\sigma(y_j)}} &= \begin{bmatrix} \sigma(x_1, y_1) & \sigma(x_1, y_2) & \dots & \sigma(x_1, y_m) \\ \sigma(x_2, y_1) & \sigma(x_2, y_2) & \dots & \sigma(x_2, y_m) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma(x_n, y_1) & \sigma(x_n, y_2) & \dots & \sigma(x_n, y_m) \end{bmatrix} \begin{bmatrix} \frac{1}{\sigma(y_1)} & 0 & \dots & 0 \\ 0 & \frac{1}{\sigma(y_2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\sigma(y_m)} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\sigma(x_1, y_1)}{\sigma(y_1)} & \frac{\sigma(x_1, y_2)}{\sigma(y_2)} & \dots & \frac{\sigma(x_1, y_m)}{\sigma(y_m)} \\ \frac{\sigma(x_2, y_1)}{\sigma(y_1)} & \frac{\sigma(x_2, y_2)}{\sigma(y_2)} & \dots & \frac{\sigma(x_2, y_m)}{\sigma(y_m)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma(x_n, y_1)}{\sigma(y_1)} & \frac{\sigma(x_n, y_2)}{\sigma(y_2)} & \dots & \frac{\sigma(x_n, y_m)}{\sigma(y_m)} \end{bmatrix}.
 \end{aligned}$$

By (3) then $G_{\text{cov}_C(x_i, y_j)} = G_{\sigma_C(x_i, y_j)} G_{\frac{1}{\sigma(y_j)}}$. $\square$

We can generalize (4)–(6) to calculate the three coefficients using matrix properties as we presented in the earlier theorems for connections. Theorem 3 is the case of a flow graph with two condition attributes and one decision attribute whereas Theorem 4 is a general form of a flow graph with $n - 1$ condition attributes and one decision attribute.

Theorem 3. Let $G = (N, B, \varphi)$ be a normalized flow graph, X^1 be a condition attribute with k_1 values: $x_1^1, x_2^1, \dots, x_{k_1}^1$, X^2 be a condition attribute with k_2 values: $x_1^2, x_2^2, \dots, x_{k_2}^2$, and X^3 be a decision attribute with k_3 values: $x_1^3, x_2^3, \dots, x_{k_3}^3$ such that $(x_{j_1}^1, x_{j_2}^2, x_{j_3}^3) \in B$, $1 \leq j_1 \leq k_1$, $1 \leq j_2 \leq k_2$, $1 \leq j_3 \leq k_3$, in a normalized flow graph G , the certainty, coverage and strength of all connections can be calculated as follows:

- (1) $G_{\text{cer}_G(x_{j_1}^1, x_{j_2}^2)} = G_{\text{cer}(x_{j_1}^1, x_{j_2}^2)} G_{\text{cer}(x_{j_2}^2, x_{j_3}^3)}$,
- (2) $G_{\text{cov}_G(x_{j_1}^1, x_{j_2}^2)} = G_{\text{cov}(x_{j_1}^1, x_{j_2}^2)} G_{\text{cov}(x_{j_2}^2, x_{j_3}^3)}$,
- (3) (a) $G_{\sigma(x_{j_1}^1, x_{j_2}^2)} = G_{\sigma(x_{j_1}^1)} G_{\text{cer}_G(x_{j_1}^1, x_{j_2}^2)}$,
- (b) $G_{\sigma(x_{j_1}^1, x_{j_2}^2)} = G_{\text{cov}_G(x_{j_1}^1, x_{j_2}^2)} G_{\sigma(x_{j_2}^2)}$.

Proof of Theorem 3. (1) Definitions 2(2), 5(2) and 6, respectively, give

$$\begin{aligned}
 & G_{\text{cer}(x_j^1, x_j^2)} G_{\text{cer}(x_j^2, x_j^3)} \\
 &= \begin{bmatrix} \text{cer}(x_1^1, x_1^2) & \text{cer}(x_1^1, x_2^2) & \cdots & \text{cer}(x_1^1, x_{k_2}^2) \\ \text{cer}(x_2^1, x_1^2) & \text{cer}(x_2^1, x_2^2) & \cdots & \text{cer}(x_2^1, x_{k_2}^2) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cer}(x_{k_1}^1, x_1^2) & \text{cer}(x_{k_1}^1, x_2^2) & \cdots & \text{cer}(x_{k_1}^1, x_{k_2}^2) \end{bmatrix} \begin{bmatrix} \text{cer}(x_1^2, x_1^3) & \text{cer}(x_1^2, x_2^3) & \cdots & \text{cer}(x_1^2, x_{k_3}^3) \\ \text{cer}(x_2^2, x_1^3) & \text{cer}(x_2^2, x_2^3) & \cdots & \text{cer}(x_2^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cer}(x_{k_2}^2, x_1^3) & \text{cer}(x_{k_2}^2, x_2^3) & \cdots & \text{cer}(x_{k_2}^2, x_{k_3}^3) \end{bmatrix} \\
 &= \begin{bmatrix} \sum_{j_2=1}^{k_2} \text{cer}(x_1^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} \text{cer}(x_1^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_2^3) & \cdots & \sum_{j_2=1}^{k_2} \text{cer}(x_1^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_{k_3}^3) \\ \sum_{j_2=1}^{k_2} \text{cer}(x_2^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} \text{cer}(x_2^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_2^3) & \cdots & \sum_{j_2=1}^{k_2} \text{cer}(x_2^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j_2=1}^{k_2} \text{cer}(x_{k_1}^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} \text{cer}(x_{k_1}^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_2^3) & \cdots & \sum_{j_2=1}^{k_2} \text{cer}(x_{k_1}^1, x_{j_2}^2) \text{cer}(x_{j_2}^2, x_{k_3}^3) \end{bmatrix} \\
 &= \begin{bmatrix} \text{cer}(x_1^1, x_1^3) & \text{cer}(x_1^1, x_2^3) & \cdots & \text{cer}(x_1^1, x_{k_3}^3) \\ \text{cer}(x_2^1, x_1^3) & \text{cer}(x_2^1, x_2^3) & \cdots & \text{cer}(x_2^1, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cer}(x_{k_1}^1, x_1^3) & \text{cer}(x_{k_1}^1, x_2^3) & \cdots & \text{cer}(x_{k_1}^1, x_{k_3}^3) \end{bmatrix} = G_{\text{cer}(x_j^1, x_j^3)}.
 \end{aligned}$$

(2) Definitions 2(3), 5(3) and 6, respectively, give

$$\begin{aligned}
 & G_{\text{cov}(x_j^1, x_j^2)} G_{\text{cov}(x_j^2, x_j^3)} \\
 &= \begin{bmatrix} \text{cov}(x_1^1, x_1^2) & \text{cov}(x_1^1, x_2^2) & \cdots & \text{cov}(x_1^1, x_{k_2}^2) \\ \text{cov}(x_2^1, x_1^2) & \text{cov}(x_2^1, x_2^2) & \cdots & \text{cov}(x_2^1, x_{k_2}^2) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(x_{k_1}^1, x_1^2) & \text{cov}(x_{k_1}^1, x_2^2) & \cdots & \text{cov}(x_{k_1}^1, x_{k_2}^2) \end{bmatrix} \begin{bmatrix} \text{cov}(x_1^2, x_1^3) & \text{cov}(x_1^2, x_2^3) & \cdots & \text{cov}(x_1^2, x_{k_3}^3) \\ \text{cov}(x_2^2, x_1^3) & \text{cov}(x_2^2, x_2^3) & \cdots & \text{cov}(x_2^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(x_{k_2}^2, x_1^3) & \text{cov}(x_{k_2}^2, x_2^3) & \cdots & \text{cov}(x_{k_2}^2, x_{k_3}^3) \end{bmatrix} \\
 &= \begin{bmatrix} \sum_{j_2=1}^{k_2} \text{cov}(x_1^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} \text{cov}(x_1^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_2^3) & \cdots & \sum_{j_2=1}^{k_2} \text{cov}(x_1^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_{k_3}^3) \\ \sum_{j_2=1}^{k_2} \text{cov}(x_2^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} \text{cov}(x_2^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_2^3) & \cdots & \sum_{j_2=1}^{k_2} \text{cov}(x_2^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j_2=1}^{k_2} \text{cov}(x_{k_1}^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_1^3) & \sum_{j_2=1}^{k_2} \text{cov}(x_{k_1}^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_2^3) & \cdots & \sum_{j_2=1}^{k_2} \text{cov}(x_{k_1}^1, x_{j_2}^2) \text{cov}(x_{j_2}^2, x_{k_3}^3) \end{bmatrix} \\
 &= \begin{bmatrix} \text{cov}(x_1^1, x_1^3) & \text{cov}(x_1^1, x_2^3) & \cdots & \text{cov}(x_1^1, x_{k_3}^3) \\ \text{cov}(x_2^1, x_1^3) & \text{cov}(x_2^1, x_2^3) & \cdots & \text{cov}(x_2^1, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(x_{k_1}^1, x_1^3) & \text{cov}(x_{k_1}^1, x_2^3) & \cdots & \text{cov}(x_{k_1}^1, x_{k_3}^3) \end{bmatrix} = G_{\text{cov}(x_j^1, x_j^3)}.
 \end{aligned}$$

(3)(a) Definition 5(2), 6 and 7, respectively, give

$$G_{\sigma}(x_j^1) G_{\text{cer}(x_j^1, x_j^3)} = \begin{bmatrix} \sigma(x_1^1) \text{cer}(x_1^1, x_1^3) & \sigma(x_1^1) \text{cer}(x_1^1, x_2^3) & \cdots & \sigma(x_1^1) \text{cer}(x_1^1, x_{k_3}^3) \\ \sigma(x_2^1) \text{cer}(x_2^1, x_1^3) & \sigma(x_2^1) \text{cer}(x_2^1, x_2^3) & \cdots & \sigma(x_2^1) \text{cer}(x_2^1, x_{k_3}^3) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma(x_{k_1}^1) \text{cer}(x_{k_1}^1, x_1^3) & \sigma(x_{k_1}^1) \text{cer}(x_{k_1}^1, x_2^3) & \cdots & \sigma(x_{k_1}^1) \text{cer}(x_{k_1}^1, x_{k_3}^3) \end{bmatrix}.$$

By (6) and Definition 5(1), then $G_{\sigma}(x_j^1) G_{\text{cer}(x_j^1, x_j^3)} = G_{\sigma}(x_j^1) G_{\text{cer}(x_j^1, x_j^3)}$.

(b) Definitions 5(3), 6 and 7, respectively, give

$$G_{cov_G(x_j^1, x_j^2)} G_{\sigma(x_j^3)} = \begin{bmatrix} \text{cov}(x_1^1, x_1^2) \sigma(x_1^3) & \text{cov}(x_1^1, x_2^2) \sigma(x_2^3) & \cdots & \text{cov}(x_1^1, x_k^2) \sigma(x_k^3) \\ \text{cov}(x_2^1, x_1^2) \sigma(x_1^3) & \text{cov}(x_2^1, x_2^2) \sigma(x_2^3) & \cdots & \text{cov}(x_2^1, x_k^2) \sigma(x_k^3) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(x_k^1, x_1^2) \sigma(x_1^3) & \text{cov}(x_k^1, x_2^2) \sigma(x_2^3) & \cdots & \text{cov}(x_k^1, x_k^2) \sigma(x_k^3) \end{bmatrix}.$$

By (6) and Definition 5(1), then $G_{\sigma(x_j^1 \dots x_j^2)} = G_{cov_G(x_j^1, x_j^2)} G_{\sigma(x_j^3)}$. $\square$

Example 2. Recall that $G_{\sigma_G} = \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix}$ from Example 1.

Since $\sigma(x_1) = 0.1$, $\sigma(x_2) = 0.7$ and $\sigma(x_3) = 0.2$, by Definition 8, then $G_{\frac{1}{\sigma(x_1)}} = \begin{bmatrix} \frac{1}{0.1} & 0 & 0 \\ 0 & \frac{1}{0.7} & 0 \\ 0 & 0 & \frac{1}{0.2} \end{bmatrix}$ and similarly,

$$G_{\frac{1}{\sigma(x_2)}} = \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.60} & 0 \\ 0 & 0 & \frac{1}{0.13} \end{bmatrix}. \text{ By the formulas given in Theorem 2, we obtain}$$

$$\begin{aligned} G_{cert_G} &= \begin{bmatrix} \frac{1}{0.1} & 0 & 0 \\ 0 & \frac{1}{0.7} & 0 \\ 0 & 0 & \frac{1}{0.2} \end{bmatrix} \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \diamond \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.60} & 0 \\ 0 & 0 & \frac{1}{0.13} \end{bmatrix} \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix} \\ &= \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.30 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.62 & 0.00 & 0.38 \end{bmatrix}. \end{aligned}$$

Consequently, we can compute

$$\begin{aligned} G_{cov_G} &= \begin{bmatrix} 0.06 & 0.04 & 0.00 \\ 0.21 & 0.42 & 0.07 \\ 0.00 & 0.14 & 0.06 \end{bmatrix} \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.60} & 0 \\ 0 & 0 & \frac{1}{0.13} \end{bmatrix} \diamond \begin{bmatrix} 0.19 & 0.08 & 0.00 \\ 0.00 & 0.60 & 0.00 \\ 0.08 & 0.00 & 0.05 \end{bmatrix} \begin{bmatrix} \frac{1}{0.27} & 0 & 0 \\ 0 & \frac{1}{0.68} & 0 \\ 0 & 0 & \frac{1}{0.05} \end{bmatrix} \\ &= \begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix} \diamond \begin{bmatrix} 0.70 & 0.12 & 0.00 \\ 0.00 & 0.88 & 0.00 \\ 0.30 & 0.00 & 1.00 \end{bmatrix}. \end{aligned}$$

This also corresponds to the certainty and coverage of the flow graph given in Fig. 2.

Next, for a combined normalized flow graph we use Theorem 3 to calculate the certainty, coverage and strength of the connection, respectively, as $G_{cert_G(x,z)} = \begin{bmatrix} 0.60 & 0.40 & 0.00 \\ 0.30 & 0.60 & 0.10 \\ 0.00 & 0.70 & 0.30 \end{bmatrix} \begin{bmatrix} 0.70 & 0.30 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.62 & 0.00 & 0.38 \end{bmatrix} = \begin{bmatrix} 0.42 & 0.58 & 0.00 \\ 0.27 & 0.69 & 0.04 \\ 0.19 & 0.70 & 0.11 \end{bmatrix}$, $G_{cov_G(x,z)} =$

$$\begin{bmatrix} 0.22 & 0.07 & 0.00 \\ 0.78 & 0.70 & 0.54 \\ 0.00 & 0.23 & 0.46 \end{bmatrix} \begin{bmatrix} 0.70 & 0.12 & 0.00 \\ 0.00 & 0.88 & 0.00 \\ 0.30 & 0.00 & 1.00 \end{bmatrix} = \begin{bmatrix} 0.15 & 0.09 & 0.00 \\ 0.71 & 0.71 & 0.54 \\ 0.14 & 0.20 & 0.46 \end{bmatrix} \text{ and } G_{\sigma_G(x,z)} = \begin{bmatrix} 0.10 & 0.00 & 0.00 \\ 0.00 & 0.70 & 0.00 \\ 0.00 & 0.00 & 0.20 \end{bmatrix} \begin{bmatrix} 0.42 & 0.58 & 0.00 \\ 0.27 & 0.69 & 0.10 \\ 0.18 & 0.70 & 0.12 \end{bmatrix} =$$

This matrix form can simplify the computation as described below. Butz et al. showed that for computing the certainty and coverage, traditional flow graph algorithms have $O(3n^3)$ time complexity, where n is the number of attribute values for each layer [11]. In Theorem 3, we can employ well-known matrix multiplication algorithms for computing these coefficients. The time complexities of the Simple algorithm, the Strassen algorithm and the Coppersmith–Winograd algorithm of matrix multiplication are $O(n^3)$, $O(n^{2.807})$ and $O(n^{2.376})$, respectively [23]. Below, we come to a general formula for computing coefficients of connections.

Theorem 4. Let $G = (N, B, \sigma)$ be a normalized flow graph, X^l be a condition attribute with k_l values: $x_1^l, x_2^l, \dots, x_{k_l}^l$, $1 \leq l \leq n-1$, and X^n be a decision attribute with k_n values: $x_1^n, x_2^n, \dots, x_{k_n}^n$ such that $(x_{j_l}^l, x_{j_l+1}^{l+1}) \in B$, $1 \leq l \leq n-1$, $1 \leq j_l \leq k_l$. For any connection $\langle x_{j_1}^1, x_{j_n}^n \rangle$, $1 \leq j_1 \leq k_1$, $1 \leq j_n \leq k_n$. The certainty, coverage and strength of the connection are:

- (1) $G_{\text{cer}}(x_{j_1}^1, x_{j_n}^n) = \prod_{l=1}^{n-1} G_{\text{cer}}(x_{j_l}^l, x_{j_{l+1}}^{l+1})$,
- (2) $G_{\text{cov}}(x_{j_1}^1, x_{j_n}^n) = \prod_{l=1}^{n-1} G_{\text{cov}}(x_{j_l}^l, x_{j_{l+1}}^{l+1})$,
- (3) (a) $G_{\sigma}(x_{j_1}^1, x_{j_n}^n) = G_{\sigma}(x_{j_1}^1) G_{\text{cer}}(x_{j_1}^1, x_{j_n}^n)$,
(b) $G_{\sigma}(x_{j_1}^1, x_{j_n}^n) = G_{\text{cov}}(x_{j_1}^1, x_{j_n}^n) G_{\sigma}(x_{j_n}^n)$.

Proof of Theorem 4. We prove this by mathematical induction for $n \geq 2$. The case $n = 2$ is shown in Theorem 3. For $n > 2$, it can be proven in a straightforward way according to:

- (1) Definitions 2(2), 5(2), 6 and Theorem 3(1),
- (2) Definitions 2(3), 5(3), 6 and Theorem 3(2),
- (3) (a) Definitions 5(1), (2), 6, (6) and Theorem 3(3)(a),
(b) Definitions 5(1), (3), 6, (6) and Theorem 3(3)(b). $\square$

Butz et al. showed that traditional algorithms for computing these formulas have $O(ln^l)$ time complexity, where l is a number layers of a flow graph and n is a number of attribute values for each layer [11]. In Theorem 4, the flow graph in matrix form improves this complexity: for the Gauss–Jordan elimination algorithm, Strassen algorithm and Coppersmith–Winograd algorithm of matrix multiplication to $O((l-2)n^3)$, $O((l-2)n^{2.807})$ and $O((l-2)n^{2.376})$, respectively [23]. While traditional algorithms are exponential in l , an algorithm using our matrix representation is polynomial in l .

In accordance with the previous definitions and theorems, if any one connected layer lacks coefficients, then we can calculate these coefficients by using Theorem 5.

Theorem 5. Let $G = (N, B, \sigma)$ be a normalized flow graph, X^l be a condition attribute with k_l values: $x_1^l, x_2^l, \dots, x_{k_l}^l$, $1 \leq l \leq n-1$, and X^n be a decision attribute with k_n values: $x_1^n, x_2^n, \dots, x_{k_n}^n$ such that $(x_{j_l}^l, x_{j_{l+1}}^{l+1}) \in B$, $1 \leq l \leq n-1$, $1 \leq j_l \leq k_l$. For any branch $\langle x_{j_l}^l, x_{j_{l+1}}^{l+1} \rangle$, $1 \leq l \leq n-1$, $1 \leq j_l \leq k_l$. The certainty, coverage and strength of branches are:

- (1) $G_{\text{cer}}(x_{j_l}^l, x_{j_{l+1}}^{l+1}) = \left(\prod_{i=1}^{l-1} G_{\text{cer}}(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \right)^{-1} G_{\text{cer}}(x_{j_1}^1, \dots, x_{j_n}^n) \left(\prod_{i=l+1}^{n-1} G_{\text{cer}}(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \right)^{-1}$,
where $\prod_{i=1}^{l-1} G_{\text{cer}}(x_{j_i}^i, x_{j_{i+1}}^{i+1})$ and $\prod_{i=l+1}^{n-1} G_{\text{cer}}(x_{j_i}^i, x_{j_{i+1}}^{i+1})$ are invertible matrices.
- (2) $G_{\text{cov}}(x_{j_l}^l, x_{j_{l+1}}^{l+1}) = \left(\prod_{i=1}^{l-1} G_{\text{cov}}(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \right)^{-1} G_{\text{cov}}(x_{j_1}^1, \dots, x_{j_n}^n) \left(\prod_{i=l+1}^{n-1} G_{\text{cov}}(x_{j_i}^i, x_{j_{i+1}}^{i+1}) \right)^{-1}$,
where $\prod_{i=1}^{l-1} G_{\text{cov}}(x_{j_i}^i, x_{j_{i+1}}^{i+1})$ and $\prod_{i=l+1}^{n-1} G_{\text{cov}}(x_{j_i}^i, x_{j_{i+1}}^{i+1})$ are invertible matrices.
- (3) (a) $G_{\sigma}(x_{j_l}^l, x_{j_{l+1}}^{l+1}) = G_{\sigma}(x_{j_l}^l) G_{\text{cer}}(x_{j_l}^l, x_{j_{l+1}}^{l+1})$,
(b) $G_{\sigma}(x_{j_l}^l, x_{j_{l+1}}^{l+1}) = G_{\text{cov}}(x_{j_l}^l, x_{j_{l+1}}^{l+1}) G_{\sigma}(x_{j_{l+1}}^{l+1})$.

Proof of Theorem 5. This can be easily verified by using a multiplicative property of the inverse matrix² and using:

- (1) Theorem 4(1),
- (2) Theorem 4(2),
- (3) (a) Theorem 2 and Remark 1,
(b) Theorem 2 and Remark 1. $\square$

Example 3. Suppose we are given two disjoint age groups of voters: y_1 (old) and y_2 (middle aged). They belong to two social classes x_1 (high) and x_2 (middle). The voters can vote for two political parties z_1 and z_2 . Assume that voters are grouped according to social classes, age groups and political parties as shown in Fig. 4.

In this case, this flow graph has some missing coefficients within the layers of age groups and parties. If we also have additional combined normalized flow graph of Fig. 4 as depicted in Fig. 5, then we can compute the missing coefficients by using Theorem 5.

² Let A , B and C be matrices and $AB = C$. If A is an invertible matrix, then $B = A^{-1}C$.

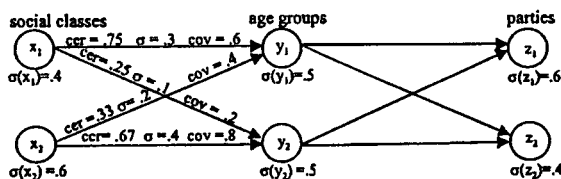


Fig. 4. A normalized flow graph with missing coefficients.

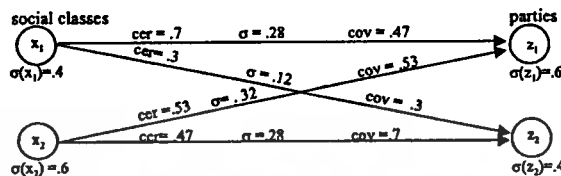


Fig. 5. Combined normalized flow graph of Fig. 4.

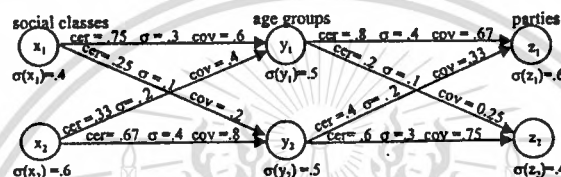


Fig. 6. A normalized flow graph of Fig. 4 with filled coefficients.

From Fig. 4, the certainty, coverage and strength of the branches (y_j, z_k) , $1 \leq j, k \leq 2$, are missing. Then by using Theorem 5, we have that

$$G_{cer_C(y_j, z_k)} = \begin{bmatrix} 0.75 & 0.25 \\ 0.33 & 0.67 \end{bmatrix}^{-1} \begin{bmatrix} 0.70 & 0.30 \\ 0.53 & 0.47 \end{bmatrix} = \begin{bmatrix} 0.80 & 0.20 \\ 0.40 & 0.60 \end{bmatrix},$$

$$G_{cov_C(y_j, z_k)} = \begin{bmatrix} 0.60 & 0.20 \\ 0.40 & 0.80 \end{bmatrix}^{-1} \begin{bmatrix} 0.47 & 0.30 \\ 0.53 & 0.70 \end{bmatrix} = \begin{bmatrix} 0.67 & 0.25 \\ 0.33 & 0.75 \end{bmatrix} \text{ and}$$

$$G_{\sigma_C(y_j, z_k)} = \begin{bmatrix} 0.50 & 0.00 \\ 0.00 & 0.50 \end{bmatrix} \begin{bmatrix} 0.80 & 0.20 \\ 0.40 & 0.60 \end{bmatrix} = \begin{bmatrix} 0.40 & 0.10 \\ 0.20 & 0.30 \end{bmatrix}.$$

Thus, we can fill in the missing coefficients of Fig. 4 as shown in Fig. 6.

3. Flow graph composition

In knowledge discovery disciplines, Pawlak's flow graph is a mathematical model for presenting relationships between attribute sets. Each layer denotes a singleton set of attribute values and each branch describes flow distribution between nodes. In the earlier sections, all data sets which were used to construct Pawlak's flow graph come from a single data set format. However, in some situations, data sets may be from heterogeneous data sources.³ In data integration [21], we desire to analyze and extract some useful patterns of flow graphs stored in this format. To do this, we require special treatment for integrating flow graphs from distinct sources to form a single flow graph and formulas for calculating their certainty, coverage and strength coefficients.

Theorem 6. Let N^i , $i = 1, 2, \dots, n$, be data sets with n^i elements and let the certainty, coverage and strength of data sets N^i be denoted by $cer(x^i, y^i)$, $cov(x^i, y^i)$ and $\sigma(x^i, y^i)$. Then the certainty, coverage and strength of a composed normalized flow graph $N^1 \cup N^2 \cup \dots \cup N^n$, are respectively computed as

$$(1) \text{cer}_{N^1 \cup N^2 \cup \dots \cup N^n}(x, y) = \frac{\sum_{i=1}^n n^i \text{cer}(x^i, y^i)}{\sum_{i=1}^n n^i \sigma(x^i)},$$

³ Heterogeneous data sources in this paper are a special case where each data source stores the same condition and decision attributes.

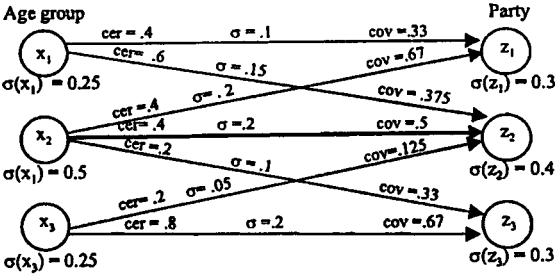


Fig. 7. Flow graph of voting data, N^1 , with 40 voters.

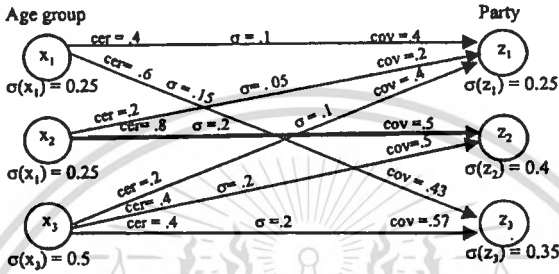


Fig. 8. Flow graph of voting data, N^2 , with 60 voters.

(2) $COV_{N^1 \cup N^2 \cup \dots \cup N^m}(x, y) = \frac{\sum_{j=1}^m n^j \sigma(y^j) cov(x^j, y^j)}{\sum_{j=1}^m n^j \sigma(y^j)}$,
(3) $\sigma_{N^1 \cup N^2 \cup \dots \cup N^m}(x, y) = \frac{\sum_{j=1}^m n^j \sigma(x^j, y^j)}{\sum_{j=1}^m n^j}$.

Proof of Theorem 6. (1) Consider $\frac{\sum_{j=1}^m n^j \sigma(x^j) cer(x^j, y^j)}{\sum_{j=1}^m n^j \sigma(x^j)}$, the definition of $\sigma(x)$, (1), (2) and $\varphi(G_N) = n^i$; then $\frac{\sum_{j=1}^m n^j \sigma(x^j) cer(x^j, y^j)}{\sum_{j=1}^m n^j \sigma(x^j)} = \frac{\sum_{j=1}^m \varphi(x^j, y^j)}{\sum_{j=1}^m \varphi(x^j)} = CER_{N^1 \cup N^2 \cup \dots \cup N^m}(x, y)$.
(2) Consider $\frac{\sum_{j=1}^m n^j \sigma(y^j) cov(x^j, y^j)}{\sum_{j=1}^m n^j \sigma(y^j)}$, the definition of $\sigma(y)$, (1), (3) and $\varphi(G_N) = n^i$; then $\frac{\sum_{j=1}^m n^j \sigma(y^j) cov(x^j, y^j)}{\sum_{j=1}^m n^j \sigma(y^j)} = \frac{\sum_{j=1}^m \varphi(x^j, y^j)}{\sum_{j=1}^m \varphi(y^j)} = COV_{N^1 \cup N^2 \cup \dots \cup N^m}(x, y)$.
(3) Consider $\frac{\sum_{j=1}^m n^j \sigma(x^j, y^j)}{\sum_{j=1}^m n^j}$, (1) and $\varphi(G) = n^i$; then $\frac{\sum_{j=1}^m n^j \sigma(x^j, y^j)}{\sum_{j=1}^m n^j} = \frac{\sum_{j=1}^m \varphi(x^j, y^j)}{\sum_{j=1}^m \varphi(G)} = \sigma_{N^1 \cup N^2 \cup \dots \cup N^m}(x, y)$. $\square$

Example 4. Suppose we have three disjoint age groups of voters x_1 (old), x_2 (middle aged) and x_3 (young). They can vote for three political parties z_1 , z_2 and z_3 . Assume that voters are grouped according to social classes, age groups and political parties as shown in Figs. 7 and 8 where Fig. 7 is a normalized flow graph of voting initial data, N^1 , for 40 voters and Fig. 8 is a normalized flow graph of voting initial data, N^2 , for 60 voters. Then we can compute three coefficients of the composed normalized flow graph $N^1 \cup N^2$ by using Theorem 6 for 100 voters as shown in Fig. 9.

We can use our new formulas to compute the certainty, coverage and strength of this new flow graph $N^1 \cup N^2$ accurately. Below, a sample computation of bold branches is described.

$$cer_{N^1 \cup N^2}(x_2, z_2) = \frac{n^1 \sigma(x_2^1) cer(x_2^1, z_2^1) + n^2 \sigma(x_2^2) cer(x_2^2, z_2^2)}{n^1 \sigma(x_2^1) + n^2 \sigma(x_2^2)} = \frac{(40)(0.5)(0.4) + (60)(0.25)(0.8)}{(40)(0.5) + (60)(0.25)} = 0.57,$$
$$cov_{N^1 \cup N^2}(x_2, z_2) = \frac{n^1 \sigma(z_2^1) cov(x_2^1, z_2^1) + n^2 \sigma(z_2^2) cov(x_2^2, z_2^2)}{n^1 \sigma(z_2^1) + n^2 \sigma(z_2^2)} = \frac{(40)(0.4)(0.5) + (60)(0.4)(0.5)}{(40)(0.4) + (60)(0.4)} = 0.5,$$
$$\sigma_{N^1 \cup N^2}(x_2, z_2) = \frac{n^1 \sigma(x_2^1, z_2^1) + n^2 \sigma(x_2^2, z_2^2)}{n^1 + n^2} = \frac{(40)(0.2) + (60)(0.2)}{40 + 60} = 0.2.$$

We can distil new patterns from this flow graph; e.g., 57% of the middle aged group votes for party z_2 , 50% of the party z_2 votes come from middle aged voters and 20% of voters are middle aged and vote for party z_2 .

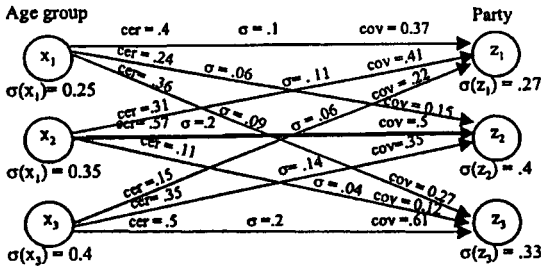


Fig. 9. Flow graph of voting data, $N^1 \cup N^2$.

Table 1(a)
A POSN decision table for N^1 .

	G	N	A	C	I	T
f_1	5	1	25	20	7	58
f_2	6	3	26	24	4	63
...	...	...	...	...	...	...
f_{20}	26	5	36	34	22	123

Table 1(b)
A POSN decision table for N^2 .

	G	N	A	C	I	T
m_1	6	6	26	16	4	58
m_2	5	2	30	15	4	58
...	...	...	...	...	...	...
m_{80}	18	25	31	42	46	162

4. Experimental results

To demonstrate that the idea and formulas presented earlier are applicable, we conduct an experiment with real-world data. It uses the scores of Olympiad mathematics students from the Promotion of Academic Olympiads and Development of Science Education Foundation (POSN), Thailand. The data set contains 100 records (N). We consider two distributed databases according to the genders of students: N^1 is 20 records for females and N^2 is 80 records for males as tabulated in Tables 1(a) and 1(b).

In this experiment, we study the strength, certainty and coverage for condition attribute sets which are the score for geometry (G), score for number theory (N), score for algebra (A), score for combinatorics (C) and score for inequalities (I) and a decision attribute, the total score (T). Possible values of condition and decision attributes are transformed to scores being high (H) and low (L) for the flow graph construction.

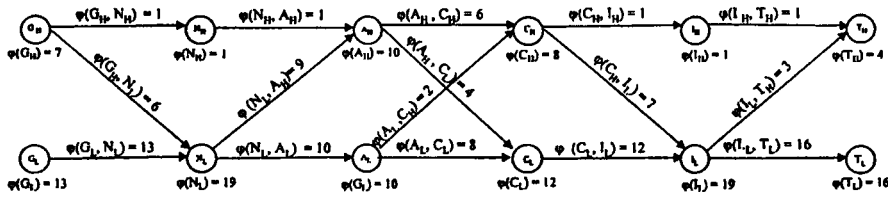
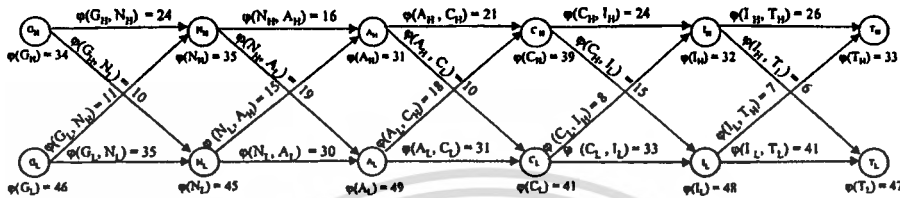
In our previous studies, we derived association rules from the entire POSN data set. We distilled some useful patterns, e.g., if students practice their preferred subjects, then they will have high scores with accuracy 97.6% [19].

We transformed the entire POSN data set to be in fuzzy attribute form and extracted decision rules for predicting total scores [20]. We discovered, e.g., number theory affects the total score substantially, and students who get high scores from any two (or more) subjects will subsequently get high total scores.

From the data in Tables 1(a) and 1(b), we can represent the corresponding flow graphs and their throughflows as depicted in Figs. 10 and 11, respectively. In these figures there are six layers. The first five layers are condition attributes whereas the last layer is a decision attribute.

Below, we represent these flow graphs in matrix form. There are five matrices concatenated in $G_{\varphi_{N^1}}$ and $G_{\varphi_{N^2}}$ to represent throughflows of the corresponding flow graphs N^1 and N^2 , respectively. The leftmost matrices represent sets of connected branches (G_i, N_j) whereas the second, the third, the fourth and the rightmost matrices represent sets of connected branches (N_i, A_j), (A_i, C_j), (C_i, I_j) and (I_i, T_j), for $i, j \in \{H, L\}$, respectively.

$$G_{\varphi_{N^1}} = \begin{bmatrix} 1 & 6 \\ 0 & 13 \end{bmatrix} \diamond \begin{bmatrix} 1 & 0 \\ 9 & 10 \end{bmatrix} \diamond \begin{bmatrix} 6 & 4 \\ 2 & 8 \end{bmatrix} \diamond \begin{bmatrix} 1 & 7 \\ 0 & 12 \end{bmatrix} \diamond \begin{bmatrix} 1 & 0 \\ 3 & 16 \end{bmatrix} \quad \text{and} \\ G_{\varphi_{N^2}} = \begin{bmatrix} 24 & 10 \\ 11 & 35 \end{bmatrix} \diamond \begin{bmatrix} 16 & 19 \\ 15 & 30 \end{bmatrix} \diamond \begin{bmatrix} 21 & 10 \\ 18 & 31 \end{bmatrix} \diamond \begin{bmatrix} 24 & 15 \\ 8 & 33 \end{bmatrix} \diamond \begin{bmatrix} 26 & 6 \\ 7 & 41 \end{bmatrix}.$$

Fig. 10. Flow graph of initial data for N^1 .Fig. 11. Flow graph of initial data for N^2 .

Similarly, we can construct the corresponding normalized flow graphs of N^1 and N^2 . Moreover from $G_{\varphi_{N^1}}$ and $G_{\varphi_{N^2}}$, we use Theorems 1 and 2 to derive the matrix forms of the strength, certainty and coverage respectively as

$$\begin{aligned}
 G_{\sigma_{N^1}} &= \begin{bmatrix} 0.05 & 0.30 \\ 0.00 & 0.65 \end{bmatrix} \diamond \begin{bmatrix} 0.05 & 0.00 \\ 0.45 & 0.50 \end{bmatrix} \diamond \begin{bmatrix} 0.30 & 0.20 \\ 0.10 & 0.40 \end{bmatrix} \diamond \begin{bmatrix} 0.05 & 0.35 \\ 0.00 & 0.60 \end{bmatrix} \diamond \begin{bmatrix} 0.05 & 0.00 \\ 0.15 & 0.80 \end{bmatrix}, \\
 G_{\sigma_{N^2}} &= \begin{bmatrix} 0.30 & 0.13 \\ 0.14 & 0.44 \end{bmatrix} \diamond \begin{bmatrix} 0.20 & 0.24 \\ 0.19 & 0.38 \end{bmatrix} \diamond \begin{bmatrix} 0.26 & 0.13 \\ 0.23 & 0.39 \end{bmatrix} \diamond \begin{bmatrix} 0.30 & 0.19 \\ 0.10 & 0.41 \end{bmatrix} \diamond \begin{bmatrix} 0.33 & 0.08 \\ 0.09 & 0.51 \end{bmatrix}, \\
 G_{cer_{N^1}} &= \begin{bmatrix} 0.14 & 0.86 \\ 0.00 & 1.00 \end{bmatrix} \diamond \begin{bmatrix} 1.00 & 0.00 \\ 0.47 & 0.53 \end{bmatrix} \diamond \begin{bmatrix} 0.60 & 0.40 \\ 0.20 & 0.80 \end{bmatrix} \diamond \begin{bmatrix} 0.13 & 0.87 \\ 0.00 & 1.00 \end{bmatrix} \diamond \begin{bmatrix} 1.00 & 0.00 \\ 0.16 & 0.84 \end{bmatrix}, \\
 G_{cer_{N^2}} &= \begin{bmatrix} 0.70 & 0.30 \\ 0.24 & 0.76 \end{bmatrix} \diamond \begin{bmatrix} 0.46 & 0.54 \\ 0.33 & 0.67 \end{bmatrix} \diamond \begin{bmatrix} 0.68 & 0.32 \\ 0.37 & 0.63 \end{bmatrix} \diamond \begin{bmatrix} 0.62 & 0.38 \\ 0.20 & 0.80 \end{bmatrix} \diamond \begin{bmatrix} 0.81 & 0.19 \\ 0.15 & 0.85 \end{bmatrix}, \\
 G_{cov_{N^1}} &= \begin{bmatrix} 1.00 & 0.32 \\ 0.00 & 0.68 \end{bmatrix} \diamond \begin{bmatrix} 0.10 & 0.00 \\ 0.90 & 1.00 \end{bmatrix} \diamond \begin{bmatrix} 0.75 & 0.33 \\ 0.25 & 0.67 \end{bmatrix} \diamond \begin{bmatrix} 1.00 & 0.37 \\ 0.00 & 0.63 \end{bmatrix} \diamond \begin{bmatrix} 0.25 & 0.00 \\ 0.75 & 1.00 \end{bmatrix}, \\
 G_{cov_{N^2}} &= \begin{bmatrix} 0.69 & 0.22 \\ 0.31 & 0.78 \end{bmatrix} \diamond \begin{bmatrix} 0.52 & 0.39 \\ 0.48 & 0.61 \end{bmatrix} \diamond \begin{bmatrix} 0.54 & 0.24 \\ 0.46 & 0.76 \end{bmatrix} \diamond \begin{bmatrix} 0.75 & 0.31 \\ 0.25 & 0.69 \end{bmatrix} \diamond \begin{bmatrix} 0.79 & 0.13 \\ 0.21 & 0.87 \end{bmatrix}.
 \end{aligned}$$

To illustrate, we can describe some results of comparisons between female (N^1) and male (N^2) students' patterns for the rightmost matrices as follows:

- In $G_{\sigma_{N^1}}$ and $G_{\sigma_{N^2}}$, 80% of female and 51% of male students have low score for inequalities and low total score, respectively ($\sigma_{N^1}(I_L, T_L) = 0.80$, $\sigma_{N^2}(I_L, T_L) = 0.51$).
- In $G_{cer_{N^1}}$ and $G_{cer_{N^2}}$, 84% of female and 85% of male students who have low score for inequalities will also have low total score, respectively ($cer_{N^1}(I_L, T_L) = 0.84$, $cer_{N^2}(I_L, T_L) = 0.85$).
- In $G_{cov_{N^1}}$ and $G_{cov_{N^2}}$, all female and 87% of male students who have low total score are students who have low score for inequalities ($cov_{N^1}(I_L, T_L) = 1.00$, $cov_{N^2}(I_L, T_L) = 0.87$).

The rest of the matrices can be interpreted by the same approach, e.g., the leftmost matrix in $G_{\sigma_{N^1}}$ indicates that there are only 5% of female students who have high score for geometry and high score for number theory ($\sigma_{N^1}(G_H, N_H) = 0.05$), 30% of female students have high score for geometry and low score for number theory ($\sigma_{N^1}(G_H, N_L) = 0.30$), 65% of female students have both low score for geometry and low score for number theory ($\sigma_{N^1}(G_L, N_L) = 0.65$) but there is no female student who has low score for geometry and high score for number theory ($\sigma_{N^1}(G_L, N_H) = 0$) and so on.

Next, we use Theorem 6 to calculate the strength, certainty and coverage of the composed flow graph: $N^1 \cup N^2$. The results in matrix form are

$$G_{\sigma_{N^1 \cup N^2}} = \begin{bmatrix} 0.25 & 0.16 \\ 0.11 & 0.48 \end{bmatrix} \diamond \begin{bmatrix} 0.17 & 0.19 \\ 0.24 & 0.40 \end{bmatrix} \diamond \begin{bmatrix} 0.27 & 0.14 \\ 0.20 & 0.39 \end{bmatrix} \diamond \begin{bmatrix} 0.25 & 0.22 \\ 0.08 & 0.45 \end{bmatrix} \diamond \begin{bmatrix} 0.27 & 0.06 \\ 0.10 & 0.57 \end{bmatrix},$$

Table 2
Models for N^1 .

	Classification models for N^1				
	Model 1: G and N	Model 2: N and A	Model 3: A and C	Model 4: C and I	Model 5: I and T
	IF G is high, THEN N is low. IF G is low, THEN N is low.	IF N is high, THEN A is high. IF N is low, THEN A is low.	IF A is high, THEN C is high. IF A is low, THEN C is low.	IF C is high, THEN I is low. IF C is low, THEN I is low.	IF I is high, THEN T is high. IF I is low, THEN T is low.
Sensitivity (%)	0	11.00	75.00	0	100
Specificity (%)	100	100	67.00	100	84.21
Accuracy (%)	95.00	60.00	70.00	95.00	85.00

Table 3
Models for N^2 .

	Classification models of N^2				
	Model 1: G and N	Model 2: N and A	Model 3: A and C	Model 4: C and I	Model 5: I and T
	IF G is high, THEN N is high. IF G is low, THEN N is low.	IF N is high, THEN A is low. IF N is low, THEN A is low.	IF A is high, THEN C is high. IF A is low, THEN C is low.	IF C is high, THEN I is high. IF C is low, THEN I is low.	IF I is high, THEN T is high. IF I is low, THEN T is low.
Sensitivity (%)	68.57	0	52.63	75.00	76.67
Specificity (%)	77.78	68.08	73.81	67.75	89.13
Accuracy (%)	73.75	45.57	63.75	71.25	83.75

$$G_{\sigma_{N^1 \cup N^2}} = \begin{bmatrix} 0.61 & 0.39 \\ 0.19 & 0.81 \end{bmatrix} \diamond \begin{bmatrix} 0.47 & 0.53 \\ 0.38 & 0.62 \end{bmatrix} \diamond \begin{bmatrix} 0.66 & 0.34 \\ 0.34 & 0.66 \end{bmatrix} \diamond \begin{bmatrix} 0.53 & 0.47 \\ 0.15 & 0.85 \end{bmatrix} \diamond \begin{bmatrix} 0.82 & 0.18 \\ 0.15 & 0.85 \end{bmatrix},$$
$$G_{\text{cov}_{N^1 \cup N^2}} = \begin{bmatrix} 0.69 & 0.25 \\ 0.31 & 0.75 \end{bmatrix} \diamond \begin{bmatrix} 0.41 & 0.32 \\ 0.59 & 0.68 \end{bmatrix} \diamond \begin{bmatrix} 0.57 & 0.26 \\ 0.43 & 0.74 \end{bmatrix} \diamond \begin{bmatrix} 0.76 & 0.33 \\ 0.24 & 0.67 \end{bmatrix} \diamond \begin{bmatrix} 0.73 & 0.10 \\ 0.27 & 0.90 \end{bmatrix}.$$

From the above matrices, we can interpret some results for students' patterns ($N^1 \cup N^2$) of the rightmost matrices as follows:

- In $G_{\sigma_{N^1 \cup N^2}}$, 57% of students have low score for inequalities and low total score ($\sigma_{N^1 \cup N^2}(I_L, T_L) = 0.57$).
- In $G_{\text{cer}_{N^1 \cup N^2}}$, 85% of students who have low score for inequalities will also have low total score ($\text{cer}_{N^1 \cup N^2}(I_L, T_L) = 0.85$).
- In $G_{\text{cov}_{N^1 \cup N^2}}$, 90% of students who have low total score are students who have low score for inequalities ($\text{cov}_{N^1 \cup N^2}(I_L, T_L) = 0.90$).

Here we focus on geometry and total scores. We can construct the combined normalized flow graphs for female (N^1) and male (N^2) students and apply Theorem 4 to compute all coefficients of the corresponding connections:

$$G_{\sigma_{N^1}}(G_i, T_j) = \begin{bmatrix} 0.07 & 0.28 \\ 0.13 & 0.52 \end{bmatrix}, \quad G_{\text{cer}_{N^1}}(G_i, T_j) = \begin{bmatrix} 0.21 & 0.79 \\ 0.20 & 0.80 \end{bmatrix}, \quad G_{\text{cov}_{N^1}}(G_i, T_j) = \begin{bmatrix} 0.36 & 0.35 \\ 0.64 & 0.65 \end{bmatrix},$$
$$G_{\sigma_{N^2}}(G_i, T_j) = \begin{bmatrix} 0.18 & 0.24 \\ 0.24 & 0.34 \end{bmatrix}, \quad G_{\text{cer}_{N^2}}(G_i, T_j) = \begin{bmatrix} 0.42 & 0.58 \\ 0.41 & 0.59 \end{bmatrix}, \quad G_{\text{cov}_{N^2}}(G_i, T_j) = \begin{bmatrix} 0.43 & 0.42 \\ 0.57 & 0.58 \end{bmatrix}.$$

Note that we calculate $G_{\text{cov}_{N^1}}(G_i, T_j)$ and $G_{\text{cer}_{N^1}}(G_i, T_j)$ by using Theorem 4 with 32 multiplications and 16 additions whereas traditional formulas require 130 multiplications and 22 additions. Similarly, we compute $G_{\text{cov}_{N^2}}(G_i, T_j)$ and $G_{\text{cer}_{N^2}}(G_i, T_j)$ with 32 multiplications and 16 additions whereas traditional formulas require 320 multiplications and 60 additions.

Finally, we can use Theorem 6 to directly calculate three coefficients of connections for the composed normalized flow graph $N^1 \cup N^2$ as $G_{\sigma_{N^1 \cup N^2}}(G_i, T_j) = \begin{bmatrix} 0.16 & 0.25 \\ 0.22 & 0.38 \end{bmatrix}$, $G_{\text{cer}_{N^1 \cup N^2}}(G_i, T_j) = \begin{bmatrix} 0.39 & 0.61 \\ 0.37 & 0.63 \end{bmatrix}$ and $G_{\text{cov}_{N^1 \cup N^2}}(G_i, T_j) = \begin{bmatrix} 0.42 & 0.40 \\ 0.58 & 0.60 \end{bmatrix}$.

Sensitivity, specificity and accuracy analysis and model selection

From $G_{\text{cer}_{N^1}}$, $G_{\text{cer}_{N^2}}$ and $G_{\text{cer}_{N^1 \cup N^2}}$ above, we extract association rules and decision rules (or models) from the row of each matrix by selecting rules which give maximum certainty. We consider five models as illustrated in Tables 2–4. We compare the sensitivity, specificity and accuracy of each model.

In Tables 2–4, “high” is considered as a positive case and “low” is considered as a negative case. We can interpret as follows:

- Model 5 for N^1 has maximum sensitivity that is 100%. This means that we can use this model to certainly recognize all actual positives.

Table 4
Models for $N^1 \cup N^2$.

	Classification models for $N^1 \cup N^2$				
	Model 1: G and N	Model 2: N and A	Model 3: A and C	Model 4: C and I	Model 5: I and T
	IF G is high, THEN N is high. IF G is low, THEN N is low.	IF N is low, THEN A is low. IF N is high, THEN A is low.	IF A is high, THEN C is high. IF A is low, THEN C is low.	IF C is high, THEN I is high. IF C is low, THEN I is low.	IF I is high, THEN T is high. IF I is low, THEN T is low.
Sensitivity (%)	69.44	0	57.44	75.76	72.97
Specificity (%)	75.00	100	73.59	67.16	90.48
Accuracy (%)	73.00	59.00	66.00	70.00	84.00

- Models 1, 2 and 4 for N^1 and 2 for $N^1 \cup N^2$ have maximum specificity. For example, model 2 for N^1 tells us that no female student who has low score for algebra is identified as having high score for algebra.
- Models 1 and 4 for N^1 have maximum accuracy equal to 95%. That is, by using these models we can classify female students' score for number theory and score for Inequalities with 95% accuracy.

In conclusion, we see that some models have high sensitivity but low accuracy or have both high sensitivity and specificity but low accuracy. By the meaning of sensitivity, specificity and accuracy, the most desirable model for classification should have high sensitivity, specificity and high accuracy. For this data set, models 5 for N^1 , N^2 and $N^1 \cup N^2$ are appropriate for classification.

5. Conclusion

In this paper, we represent Pawlak's graphical flow graphs in matrix forms which allow more flexible computation. We present formulas for calculating all coefficients by using properties of matrices. Moreover, we introduce a composition flow graph from heterogeneous data sources and formulas for calculating its coefficients. We also performed a real-world experiment on the POSN data set for demonstration. Our future work is to propose new methods to allow extensive computation for large data sets.

Acknowledgements

Thanks to G.M. Zaverucha, J. Tadrat and the Promotion of Academic Olympiads and Development of Science Education Foundation, Thailand. The authors are grateful to the anonymous referees for their valuable and constructive comments.

References

[1] Z. Pawlak, Rough sets, *Int. J. Inf. Comput. Sci.* 11 (5) (1982) 341–356.
[2] Z. Pawlak, Rough sets, decision algorithms and Bayes' theorem, *European J. Oper. Res.* 136 (2002) 181–189.
[3] Z. Pawlak, Rough set and flow graphs, in: D. Slezak, et al. (Eds.), *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, in: *Lect. Notes. Artif. Int.*, vol. 3641, Springer-Verlag, Berlin, 2005, pp. 1–11.
[4] L.R. Ford, D.R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, NJ, 1962.
[5] Z. Pawlak, Probability truth and flow graphs, in: *Proceedings of Rough Sets in Knowledge Discovery and Soft Computing*, Warsaw, Poland, 2003, pp. 1–9.
[6] Z. Pawlak, Decision algorithms and flow graphs: a rough set approach, *J. Telecom. Inform. Tech.* 3 (2003) 98–101.
[7] Z. Pawlak, Flow graphs and data mining, in: J.F. Peters, A. Skowron (Eds.), *T. Rough Sets III*, in: *Lect. Notes Comp. Sci.*, vol. 3400, Springer, Heidelberg, 2005, pp. 1–36.
[8] Z. Pawlak, Decision trees and flow graphs, in: S. Greco (Ed.), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.*, vol. 4259, Springer-Verlag, Berlin, 2006, pp. 1–11.
[9] S. Greco, Z. Pawlak, R. Slowinski, Generalized decision algorithms, rough inference rules, and flow graphs, in: J.J. Alpigini, et al. (Eds.), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.*, vol. 2475, Springer-Verlag, Berlin, 2002, pp. 93–104.
[10] A. Czyzewski, B. Kostek, Musical metadata retrieval with flow graphs, in: S. Tsumoto, et al. (Eds.), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.*, vol. 3066, Springer-Verlag, Berlin, 2004, pp. 691–698.
[11] C.J. Butz, W. Yan, B. Yang, An efficient algorithm for inference in rough set flow graphs, in: *T. Rough Sets V*, in: *Lect. Notes. Artif. Int.*, vol. 4100, Springer-Verlag, Berlin, 2006, pp. 102–122.
[12] 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.
[13] J. Sun, H. Liu, H. Zhang, An extension of Pawlak's flow graphs, in: *Proceedings of the 1st International Conference on Rough Sets and Knowledge Technology*, Chongqing, 2006, pp. 191–199.
[14] A. Mieszkowicz-Rolka, L. Rolka, Flow graph and decision tables with fuzzy attributes, in: *Proceedings of the 8th International Conference Artificial Intelligence and Soft Computing*, Zakopane, Poland, in: *Lect. Notes. Artif. Int.*, vol. 4029, Springer-Verlag, Berlin, 2006, pp. 268–277.
[15] Z. Suraj, K. Pancierz, Flow graphs as a tool for mining prediction rule of changes of components, in: *Temporal Information Systems, Rough Sets and Knowledge Technology*, in: *Lect. Notes. Artif. Int.*, vol. 2475, Springer-Verlag, Berlin, 2007, pp. 468–475.
[16] C.C. Chien, S. Tsumoto, On learning decision rules from flow graphs, in: *Proceedings of North American Fuzzy Information Processing Society, Annual Meeting of the North American*, 2007, pp. 655–658.
[17] J. Sun, H. Liu, C. Qi, H. Zhang, Interpretation of extended Pawlak's flow graphs using granular computing, in: J.F. Peters, A. Skowron (Eds.), *T. Rough Sets VIII*, in: *Lect. Notes. Comp. Sci.*, vol. 5084, Springer, Heidelberg, 2008, pp. 93–115.
[18] Z. Matusiewicz, K. Pancierz, Rough set flow graphs and max- $\rightarrow$ fuzzy relation equations in state prediction problems, in: C.C. Chan, et al. (Eds.), *Rough Sets and Current Trends in Computing*, in: *Lect. Notes. Artif. Int.*, vol. 5306, Springer-Verlag, Berlin, 2008, pp. 359–368.

- [19] D. Chitcharoen, P. Pattaraintakorn, Knowledge discovery by rough sets mathematical flow graphs and its extension, in: Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria, 2008, pp. 340–345.
- [20] D. Chitcharoen, P. Pattaraintakorn, Towards theories of fuzzy set and rough set to flow graphs, in: Proceedings of the 2008 IEEE World Congress on Computational Intelligence, Hong Kong, 2008, pp. 1675–1682.
- [21] M. Lenzerini, Data integration: a theoretical perspective, in: 21th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2002, Madison, Wisconsin, USA, June 2002.
- [22] D. Poole, Linear Algebra: A Model Introduction, second ed., Thomson, 2006.
- [23] T. Corman, C. Leiserson, R. Rivest, C. Stein, Introduction to Algorithms, second ed., MIT Press, McGraw-Hill, 2001.



AUTHOR BIOGRAPHY

Name: Miss Doungnat Chitcharoen

Date of Birth: 13 March 1978

Place of Birth: Bangkok

Educations: Bachelor of Science in Mathematics, Chiang Mai University, Thailand, 2000

Master of Science in Mathematics, Ramkhamhaeng University, Thailand, 2005

