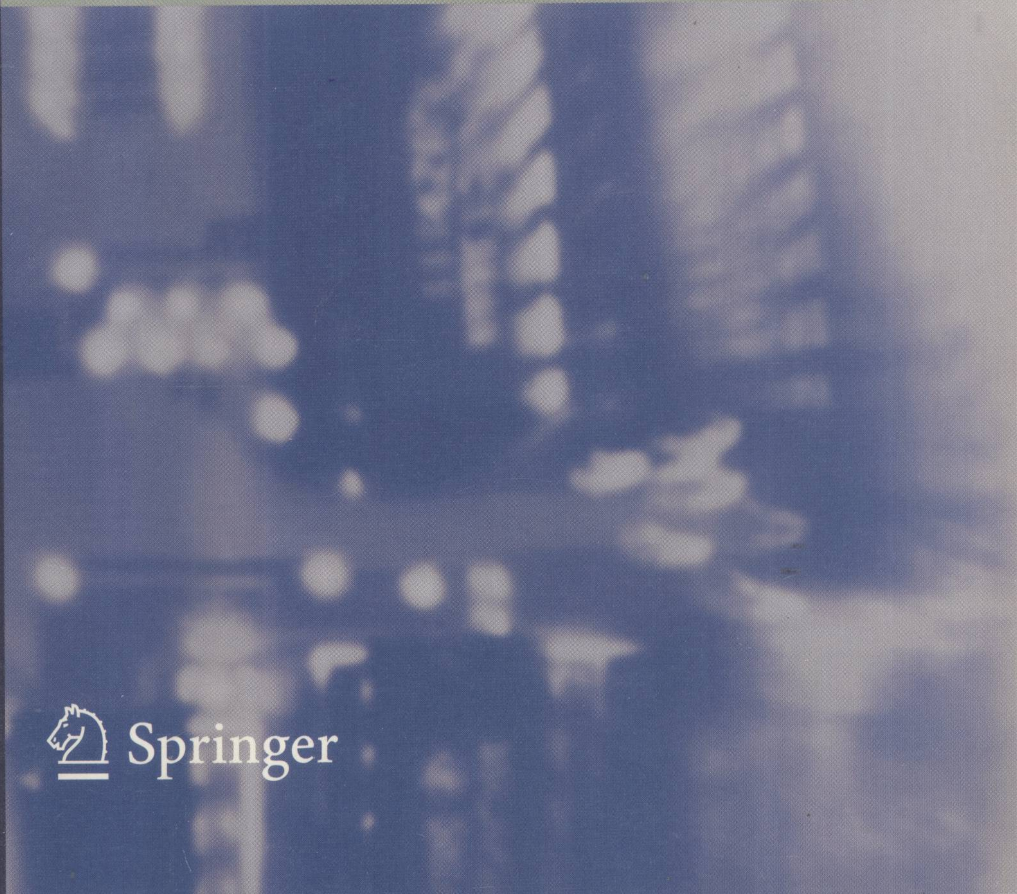
LNCS 3400 Journal Subline

Transactions on
Rough Sets III

James F. Peters · Andrzej Skowron

Editors-in-Chief



Springer

James F. Peters Andrzej Skowron (Eds.)

Transactions on Rough Sets III

Springer

Volume Editors

James F. Peters
University of Manitoba
Department of Electrical and Computer Engineering
Winnipeg, Manitoba R3T 5V6, Canada
E-mail: jfpeters@ee.umanitoba.ca

Andrzej Skowron
Warsaw University
Institute of Mathematics
Banacha 2, 02-097 Warsaw, Poland
E-mail: skowron@mimuw.edu.pl

Library of Congress Control Number: 2005925573

CR Subject Classification (1998): F.4.1, F.1, I.2, H.2.8, I.5.1, I.4

ISSN 0302-9743
ISBN-10 3-540-25998-8 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-25998-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11427834 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3400

Preface

Volume III of the Transactions on Rough Sets (TRS) introduces advances in the theory and application of rough sets. These advances have far-reaching implications in a number of research areas such as approximate reasoning, bioinformatics, computer science, data mining, engineering (especially, computer engineering and signal analysis), intelligent systems, knowledge discovery, pattern recognition, machine intelligence, and various forms of learning. This volume reveals the vigor, breadth and depth in research either directly or indirectly related to the rough sets theory introduced by Prof. Zdzisław Pawlak more than three decades ago. Evidence of this can be found in the seminal paper on data mining by Prof. Pawlak included in this volume. In addition, there are eight papers on the theory and application of rough sets as well as a presentation of a new version of the Rough Set Exploration System (RSES) tool set and an introduction to the Rough Set Database System (RSDS).

Prof. Pawlak has contributed a pioneering paper on data mining to this volume. In this paper, it is shown that information flow in a flow graph is governed by Bayes' rule with a deterministic rather than a probabilistic interpretation. A cardinal feature of this paper is that it is self-contained inasmuch as it not only introduces a new view of information flow but also provides an introduction to the basic concepts of flow graphs. The representation of information flow introduced in this paper makes it possible to study different relationships in data and establishes a basis for a new mathematical tool for data mining.

In addition to the paper by Prof. Pawlak, new developments in rough set theory are represented by five papers that investigate the validity, confidence and coverage of rules in approximation spaces (Anna Gomolińska), decision trees considered in the context of rough sets (Mikhail Ju. Moshkov), study of approximation spaces and information granulation (Andrzej Skowron, Roman Świniarski and Piotr Synak), a new interpretation of rough sets based on inverse probabilities and the foundations for a rough Bayesian model (Dominik Ślęzak), and formal concept analysis and rough set theory considered relative to topological approximations (Marcin Wolski). The theory papers in this volume are accompanied by four papers on applications of rough sets: knowledge extraction from electronic devices for power system substation event analysis and decision support (Ching-Lai Hor and Peter Crossley), processing of musical data using rough set methods, RSES and neural computing (Bożena Kostek, Piotr Szczuko, Paweł Żwan and Piotr Dalka), computational intelligence in bioinformatics (Sushmita Mitra), and an introduction to rough ethology, which is based on a biologically inspired study of collective behavior and reinforcement learning in intelligent systems using approximation spaces (James Peters).

This volume also celebrates two landmark events: a new version of RSES and the availability of a Rough Set Database System (RSDS). The introduction of a new version of the Rough Set Exploration System (RSES 2.2) is given in a paper by Jan G. Bazan and Marcin Szczuka. This paper gives an overview of the basic features of the new version of RSES: improved graphical user interface as well as production of decomposition trees and rules based on training samples. The decomposition tree and rules resulting from training can be used to classify unseen cases. The paper by Zbigniew Suraj and Piotr Grochowalski gives an overview of RSDS, which now includes over 1900 entries and over 800 authors. RSDS includes a number of useful utilities that make it possible for authors to update the database via the web, namely, append, search, download, statistics and help. In addition, RSDS provides access to biographies of researchers in the rough set community.

This issue of the TRS has been made possible thanks to the efforts of a great many generous persons and organizations. We express our thanks to the many anonymous reviewers for their heroic efforts in providing detailed reviews of the articles in this issue of the TRS. The editors and authors of this volume also extend an expression of gratitude to Alfred Hofmann, Ursula Barth, Christine Günther and the other LNCS staff members at Springer for their support in making this volume of the TRS possible. The Editors of this volume have been supported by the Ministry of Scientific Research and Information Technology of the Republic of Poland, Research Grant No. 3T11C00226, and the Natural Sciences and Engineering Research Council of Canada (NSERC), Research Grant No. 185986.

January 2005

James F. Peters
Andrzej Skowron

LNCS Transactions on Rough Sets

This journal subline has as its principal aim the fostering of professional exchanges between scientists and practitioners who are interested in the foundations and applications of rough sets. Topics include foundations and applications of rough sets as well as foundations and applications of hybrid methods combining rough sets with other approaches important for the development of intelligent systems.

The journal includes high-quality research articles accepted for publication on the basis of thorough peer reviews. Dissertations and monographs up to 250 pages that include new research results can also be considered as regular papers. Extended and revised versions of selected papers from conferences can also be included in regular or special issues of the journal.

Editorial Board

Lecture Notes in Computer Science

For information about Vols. 1–3382

please contact your bookseller or Springer

Vol. 3525: A.E. Abdallah, C.B. Jones, J.W. Sanders (Eds.), Communicating Sequential Processes. XIV, 321 pages. 2005.

Vol. 3517: H.S. Baird, D.P. Lopresti (Eds.), Human Interactive Proofs. IX, 143 pages. 2005.

Vol. 3510: T. Braun, G. Carle, Y. Koucheryavy, V. Tsaoussidis (Eds.), Wired/Wireless Internet Communications. XIV, 366 pages. 2005.

Vol. 3508: P. Bresciani, P. Giorgini, B. Henderson-Sellers, G. Low, M. Winikoff (Eds.), Agent-Oriented Information Systems II. X, 227 pages. 2005. (Subseries LNAI).

Vol. 3503: S.E. Nikoletseas (Ed.), Experimental and Efficient Algorithms. XV, 624 pages. 2005.

Vol. 3501: B. Kégl, G. Lapalme (Eds.), Advances in Artificial Intelligence. XV, 458 pages. 2005. (Subseries LNAI).

Vol. 3500: S. Miyano, J. Mesirov, S. Kasif, S. Istrail, P. Pevzner, M. Waterman (Eds.), Research in Computational Molecular Biology. XVII, 632 pages. 2005. (Subseries LNBI).

Vol. 3498: J. Wang, X. Liao, Z. Yi (Eds.), Advances in Neural Networks – ISNN 2005, Part III. L, 1077 pages. 2005.

Vol. 3497: J. Wang, X. Liao, Z. Yi (Eds.), Advances in Neural Networks – ISNN 2005, Part II. L, 947 pages. 2005.

Vol. 3496: J. Wang, X. Liao, Z. Yi (Eds.), Advances in Neural Networks – ISNN 2005, Part II. L, 1055 pages. 2005.

Vol. 3495: P. Kantor, G. Muresan, F. Roberts, D.D. Zeng, F.-Y. Wang, H. Chen, R.C. Merkle (Eds.), Intelligence and Security Informatics. XVIII, 674 pages. 2005.

Vol. 3494: R. Cramer (Ed.), Advances in Cryptology – EUROCRYPT 2005. XIV, 576 pages. 2005.

Vol. 3492: P. Blache, E. Stabler, J. Busquets, R. Moot (Eds.), Logical Aspects of Computational Linguistics. X, 363 pages. 2005. (Subseries LNAI).

Vol. 3489: G.T. Heineman, J.A. Stafford, H.W. Schmidt, K. Wallnau, C. Szyperski, I. Crnkovic (Eds.), Component-Based Software Engineering. XI, 358 pages. 2005.

Vol. 3488: M.-S. Hacid, N.V. Murray, Z.W. Raś, S. Tsumoto (Eds.), Foundations of Intelligent Systems. XIII, 700 pages. 2005. (Subseries LNAI).

Vol. 3475: N. Guelfi (Ed.), Rapid Integration of Software Engineering Techniques. X, 145 pages. 2005.

Vol. 3467: J. Giesl (Ed.), Term Rewriting and Applications. XIII, 517 pages. 2005.

Vol. 3465: M. Bernardo, A. Bogliolo (Eds.), Formal Methods for Mobile Computing. VII, 271 pages. 2005.

Vol. 3463: M. Dal Cin, M. Kaâniche, A. Pataricza (Eds.), Dependable Computing - EDCC 2005. XVI, 472 pages. 2005.

Vol. 3462: R. Boutaba, K. Almeroth, R. Puigjaner, S. Shen, J.P. Black (Eds.), NETWORKING 2005. XXX, 1483 pages. 2005.

Vol. 3461: P. Urzyczyn (Ed.), Typed Lambda Calculi and Applications. XI, 433 pages. 2005.

Vol. 3459: R. Kimmel, N.A. Sochen, J. Weickert (Eds.), Scale Space and PDE Methods in Computer Vision. XI, 634 pages. 2005.

Vol. 3456: H. Rust, Operational Semantics for Timed Systems. XII, 223 pages. 2005.

Vol. 3455: H. Treharne, S. King, M. Henson, S. Schneider (Eds.), ZB 2005: Formal Specification and Development in Z and B. XV, 493 pages. 2005.

Vol. 3454: J.-M. Jacquet, G.P. Picco (Eds.), Coordination Models and Languages. X, 299 pages. 2005.

Vol. 3453: L. Zhou, B.C. Ooi, X. Meng (Eds.), Database Systems for Advanced Applications. XXVII, 929 pages. 2005.

Vol. 3452: F. Baader, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XI, 562 pages. 2005. (Subseries LNAI).

Vol. 3450: D. Hutter, M. Ullmann (Eds.), Security in Pervasive Computing. XI, 239 pages. 2005.

Vol. 3449: F. Rothlauf, J. Branke, S. Cagnoni, D.W. Corne, R. Drechsler, Y. Jin, P. Machado, E. Marchiori, J. Romero, G.D. Smith, G. Squillero (Eds.), Applications of Evolutionary Computing. XX, 631 pages. 2005.

Vol. 3448: G.R. Raidl, J. Gottlieb (Eds.), Evolutionary Computation in Combinatorial Optimization. XI, 271 pages. 2005.

Vol. 3447: M. Keijzer, A. Tettamanzi, P. Collet, J.v. Hemert, M. Tomassini (Eds.), Genetic Programming. XIII, 382 pages. 2005.

Vol. 3444: M. Sagiv (Ed.), Programming Languages and Systems. XIII, 439 pages. 2005.

Vol. 3443: R. Bodik (Ed.), Compiler Construction. XI, 305 pages. 2005.

Vol. 3442: M. Cerioli (Ed.), Fundamental Approaches to Software Engineering. XIII, 373 pages. 2005.

Vol. 3441: V. Sassone (Ed.), Foundations of Software Science and Computational Structures. XVIII, 521 pages. 2005.

Vol. 3440: N. Halbwachs, L.D. Zuck (Eds.), Tools and Algorithms for the Construction and Analysis of Systems. XVII, 588 pages. 2005.

Vol. 3439: R.H. Deng, F. Bao, H. Pang, J. Zhou (Eds.), Information Security Practice and Experience. XII, 424 pages. 2005.

Vol. 3437: T. Gschwind, C. Mascolo (Eds.), Software Engineering and Middleware. X, 245 pages. 2005.

Vol. 3436: B. Bouyssounouse, J. Sifakis (Eds.), Embedded Systems Design. XV, 492 pages. 2005.

Vol. 3434: L. Brun, M. Vento (Eds.), Graph-Based Representations in Pattern Recognition. XII, 384 pages. 2005.

Vol. 3433: S. Bhalla (Ed.), Databases in Networked Information Systems. VII, 319 pages. 2005.

Vol. 3432: M. Beigl, P. Lukowicz (Eds.), Systems Aspects in Organic and Pervasive Computing - ARCS 2005. X, 265 pages. 2005.

Vol. 3431: C. Dovrolis (Ed.), Passive and Active Network Measurement. XII, 374 pages. 2005.

Vol. 3429: E. Andres, G. Damiand, P. Lienhardt (Eds.), Discrete Geometry for Computer Imagery. X, 428 pages. 2005.

Vol. 3427: G. Kotsis, O. Spaniol (Eds.), Wireless Systems and Mobility in Next Generation Internet. VIII, 249 pages. 2005.

Vol. 3423: J.L. Fiadeiro, P.D. Mosses, F. Orejas (Eds.), Recent Trends in Algebraic Development Techniques. VIII, 271 pages. 2005.

Vol. 3422: R.T. Mittermeir (Ed.), From Computer Literacy to Informatics Fundamentals. X, 203 pages. 2005.

Vol. 3421: P. Lorenz, P. Dini (Eds.), Networking - ICN 2005, Part II. XXXV, 1153 pages. 2005.

Vol. 3420: P. Lorenz, P. Dini (Eds.), Networking - ICN 2005, Part I. XXXV, 933 pages. 2005.

Vol. 3419: B. Faltings, A. Petcu, F. Fages, F. Rossi (Eds.), Constraint Satisfaction and Constraint Logic Programming. X, 217 pages. 2005. (Subseries LNAI).

Vol. 3418: U. Brandes, T. Erlebach (Eds.), Network Analysis. XII, 471 pages. 2005.

Vol. 3416: M. Böhlen, J. Gamper, W. Polasek, M.A. Wimmer (Eds.), E-Government: Towards Electronic Democracy. XIII, 311 pages. 2005. (Subseries LNAI).

Vol. 3415: P. Davidsson, B. Logan, K. Takadama (Eds.), Multi-Agent and Multi-Agent-Based Simulation. X, 265 pages. 2005. (Subseries LNAI).

Vol. 3414: M. Morari, L. Thiele (Eds.), Hybrid Systems: Computation and Control. XII, 684 pages. 2005.

Vol. 3412: X. Franch, D. Port (Eds.), COTS-Based Software Systems. XVI, 312 pages. 2005.

Vol. 3411: S.H. Myaeng, M. Zhou, K.-F. Wong, H.-J. Zhang (Eds.), Information Retrieval Technology. XIII, 337 pages. 2005.

Vol. 3410: C.A. Coello Coello, A. Hernández Aguirre, E. Zitzler (Eds.), Evolutionary Multi-Criterion Optimization. XVI, 912 pages. 2005.

Vol. 3409: N. Guelfi, G. Reggio, A. Romanovsky (Eds.), Scientific Engineering of Distributed Java Applications. X, 127 pages. 2005.

Vol. 3408: D.E. Losada, J.M. Fernández-Luna (Eds.), Advances in Information Retrieval. XVII, 572 pages. 2005.

Vol. 3407: Z. Liu, K. Araki (Eds.), Theoretical Aspects of Computing - ICTAC 2004. XIV, 562 pages. 2005.

Vol. 3406: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing. XVII, 829 pages. 2005.

Vol. 3404: V. Diekert, B. Durand (Eds.), STACS 2005. XVI, 706 pages. 2005.

Vol. 3403: B. Ganter, R. Godin (Eds.), Formal Concept Analysis. XI, 419 pages. 2005. (Subseries LNAI).

Vol. 3402: M. Daydé, J.J. Dongarra, V. Hernández, J.M.L.M. Palma (Eds.), High Performance Computing for Computational Science - VECPAR 2004. XI, 732 pages. 2005.

Vol. 3401: Z. Li, L.G. Vulkov, J. Waśniewski (Eds.), Numerical Analysis and Its Applications. XIII, 630 pages. 2005.

Vol. 3400: J.F. Peters, A. Skowron (Eds.), Transactions on Rough Sets III. IX, 461 pages. 2005.

Vol. 3399: Y. Zhang, K. Tanaka, J.X. Yu, S. Wang, M. Li (Eds.), Web Technologies Research and Development - APWeb 2005. XXII, 1082 pages. 2005.

Vol. 3398: D.-K. Baik (Ed.), Systems Modeling and Simulation: Theory and Applications. XIV, 733 pages. 2005. (Subseries LNAI).

Vol. 3397: T.G. Kim (Ed.), Artificial Intelligence and Simulation. XV, 711 pages. 2005. (Subseries LNAI).

Vol. 3396: R.M. van Eijk, M.-P. Huget, F. Dignum (Eds.), Agent Communication. X, 261 pages. 2005. (Subseries LNAI).

Vol. 3395: J. Grabowski, B. Nielsen (Eds.), Formal Approaches to Software Testing. X, 225 pages. 2005.

Vol. 3394: D. Kudenko, D. Kazakov, E. Alonso (Eds.), Adaptive Agents and Multi-Agent Systems II. VIII, 313 pages. 2005. (Subseries LNAI).

Vol. 3393: H.-J. Kreowski, U. Montanari, F. Orejas, G. Rozenberg, G. Taentzer (Eds.), Formal Methods in Software and Systems Modeling. XXVII, 413 pages. 2005.

Vol. 3392: D. Seipel, M. Hanus, U. Geske, O. Bartenstein (Eds.), Applications of Declarative Programming and Knowledge Management. X, 309 pages. 2005. (Subseries LNAI).

Vol. 3391: C. Kim (Ed.), Information Networking. XVII, 936 pages. 2005.

Vol. 3390: R. Choren, A. Garcia, C. Lucena, A. Romanovsky (Eds.), Software Engineering for Multi-Agent Systems III. XII, 291 pages. 2005.

Vol. 3389: P. Van Roy (Ed.), Multiparadigm Programming in Mozart/Oz. XV, 329 pages. 2005.

Vol. 3388: J. Lagergren (Ed.), Comparative Genomics. VII, 133 pages. 2005. (Subseries LNBI).

Vol. 3387: J. Cardoso, A. Sheth (Eds.), Semantic Web Services and Web Process Composition. VIII, 147 pages. 2005.

Vol. 3386: S. Vaudenay (Ed.), Public Key Cryptography - PKC 2005. IX, 436 pages. 2005.

Vol. 3385: R. Cousot (Ed.), Verification, Model Checking, and Abstract Interpretation. XII, 483 pages. 2005.

Vol. 3383: J. Pach (Ed.), Graph Drawing. XII, 536 pages. 2005.

¥679.68元

Table of Contents

Regular Papers

Dissertations and Monographs

Flow Graphs and Data Mining

Zdzisław Pawlak[1,2]

[1] Institute for Theoretical and Applied Informatics,
Polish Academy of Sciences,
ul. Bałtycka 5, 44-100 Gliwice, Poland
[2] Warsaw School of Information Technology,
ul. Newelska 6, 01-447 Warsaw, Poland
zpw@ii.pw.edu.pl

Abstract. In this paper we propose a new approach to data mining and knowledge discovery based on information flow distribution in a flow graph. Flow graphs introduced in this paper are different from those proposed by Ford and Fulkerson for optimal flow analysis and they model flow distribution in a network rather than the optimal flow which is used for information flow examination in decision algorithms. It is revealed that flow in a flow graph is governed by Bayes' rule, but the rule has an entirely deterministic interpretation without referring to its probabilistic roots. Besides, a decision algorithm induced by a flow graph and dependency between conditions and decisions of decision rules is introduced and studied, which is used next to simplify decision algorithms.

Keywords: flow graph, data mining, knowledge discovery, decision algorithms.

Introduction

In this paper we propose a new approach to data analysis (mining) based on information flow distribution study in a flow graph.

Flow graphs introduced in this paper are different from those proposed by Ford and Fulkerson [4] for optimal flow analysis and they model rather flow *distribution* in a network, than the optimal flow.

The flow graphs considered in this paper are not meant to model physical media (e.g., water) flow analysis, but to model information flow examination in decision algorithms. To this end branches of a flow graph can be interpreted as decision rules. With every decision rule (i.e., branch) three coefficients are associated: the *strength, certainty* and *coverage factors.*

These coefficients have been used under different names in data mining (see, e.g., [14, 15]) but they were used first by Łukasiewicz [8] in his study of logic and probability.

This interpretation, in particular, leads to a new look at Bayes' theorem. Let us also observe that despite Bayes' rule fundamental role in statistical inference it has led to many philosophical discussions concerning its validity and meaning, and has caused much criticism [1, 3, 13].

J.F. Peters and A. Skowron (Eds.): Transactions on Rough Sets III, LNCS 3400, pp. 1–36, 2005.

This paper is a continuation of some of the authors' ideas presented in [10, 11], where the relationship between Bayes' rule and flow graphs has been introduced and studied (see also [6, 7]).

This paper consists of two parts. Part one introduces basic concepts of the proposed approach, i.e., flow graph and its fundamental properties. It is revealed that flow in a flow graph is governed by Bayes' rule, but the rule has an entirely deterministic interpretation that does not refer to its probabilistic roots. In addition, dependency of flow is defined and studied. This idea is based on the statistical concept of dependency but in our setting it has a deterministic meaning.

In part two many tutorial examples are given to illustrate how the introduced ideas work in data mining. These examples clearly show the difference between classical Bayesian inference methodology and the proposed one.

The presented ideas can be used, among others, as a new tool for data mining, and knowledge representation. Besides, the proposed approach throws new light on the concept of probability.

1 Flow Graphs

1.1 Overview

In this part the fundamental concepts of the proposed approach are defined and discussed. In particular flow graphs, certainty and coverage factors of branches of the flow graph are defined and studied. Next these coefficients are extended to paths and some classes of sub-graphs called connections. Further a notion of fusion of a flow graph is defined.

Further dependences of flow are introduced and examined. Finally, dependency factor (correlation coefficient) is defined.

Observe that in many cases the data flow order, represented in flow graphs, explicitly follows from the problem specification. However, in other cases the relevant order should be discovered from data. This latter issue will be discussed elsewhere.

1.2 Basic Concepts

A flow graph is a *directed, acyclic, finite* graph $G = (N, \mathcal{B}, \varphi)$, where N is a set of *nodes*, $\mathcal{B} \subseteq N \times N$ is a set of *directed branches*, $\varphi : \mathcal{B} \to R^+$ is a *flow function* and R^+ is the set of non-negative reals.

Input of a node $x \in N$ *is the set* $I(x) = \{y \in N : (y, x) \in \mathcal{B}\}$; *output* of a node $x \in N$ is defined by $O(x) = \{y \in N : (x, y) \in \mathcal{B}\}$.

We will also need the concept of *input* and *output* of a graph G, defined, respectively, as follows: $I(G) = \{x \in N : I(x) = \emptyset\}$, $O(G) = \{x \in N : O(x) = \emptyset\}$.

Inputs and outputs of G are *external nodes* of G; other nodes are *internal nodes* of G.

If $(x, y) \in \mathcal{B}$, then $\varphi(x, y)$ is a *throughflow* from x to y.

With every node x of a flow graph G we associate its *inflow*

$$\varphi_+(x) = \sum_{y \in I(x)} \varphi(y, x), \tag{1}$$

and *outflow*

$$\varphi_-(x) = \sum_{y \in O(x)} \varphi(x, y). \tag{2}$$

Similarly, we define an inflow and an outflow for the whole flow graph, which are defined by

$$\varphi_+(G) = \sum_{x \in I(G)} \varphi_-(x), \tag{3}$$

$$\varphi_-(G) = \sum_{x \in O(G)} \varphi_+(x). \tag{4}$$

We assume that for any internal node x we have $\varphi_+(x) = \varphi_-(x) = \varphi(x)$, where $\varphi(x)$ is a *throughflow* of node x.

Then, obviously, $\varphi_+(G) = \varphi_-(G) = \varphi(G)$, where $\varphi(G)$ is a *throughflow* of graph G.

The above formulas can be considered as *flow conservation equations* [4].

Example

We will illustrate the basic concepts of flow graphs by an example of a group of 1000 patients put to the test for certain drug effectiveness.

Assume that patients are grouped according to presence of the disease, age and test results, as shown in Fig. 1.

For example, $\varphi(x_1) = 600$ means that these are 600 patients suffering from the disease, $\varphi(y_1) = 570$ means that there are 570 old patients $\varphi(z_1) = 471$ means that 471 patients have a positive test result; $\varphi(x_1, y_1) = 450$ means that there are 450 old patients which suffer from disease etc.

Thus the flow graph gives clear insight into the relationship between different groups of patients.

Let us now explain the flow graph in more detail.

Nodes of the flow graph are depicted by circles, labeled by $x_1, x_2, y_1, y_2, y_3, z_1, z_2$. A branch (x, y) is denoted by an arrow from node x to y. For example, branch (x_1, z_1) is represented by an arrow from x_1 to z_1.

For example, inputs of node y_1 are nodes x_1 and x_2, outputs of node x_1 are nodes y_1, y_2 and y_3.

Inputs of the flow graph are nodes x_1 and x_2, whereas the outputs of the flow graph are nodes z_1 and z_2.

Nodes y_1, y_2 and y_3 are internal nodes of the flow graph. The throughflow of the branch (x_1, y_1) is $\varphi(x_1, y_1) = 450$. Inflow of node y_1 is $\varphi_+(y_1) = 450 + 120 = 570$. Outflow of node y_1 is $\varphi_-(y_1) = 399 + 171 = 570$. Inflow of the flow graph is $\varphi(x_1) + \varphi(x_2) = 600 + 400 = 1000$, and outflow of the flow graph is $\varphi(z_1) + \varphi(z_2) = 471 + 529 = 1000$.

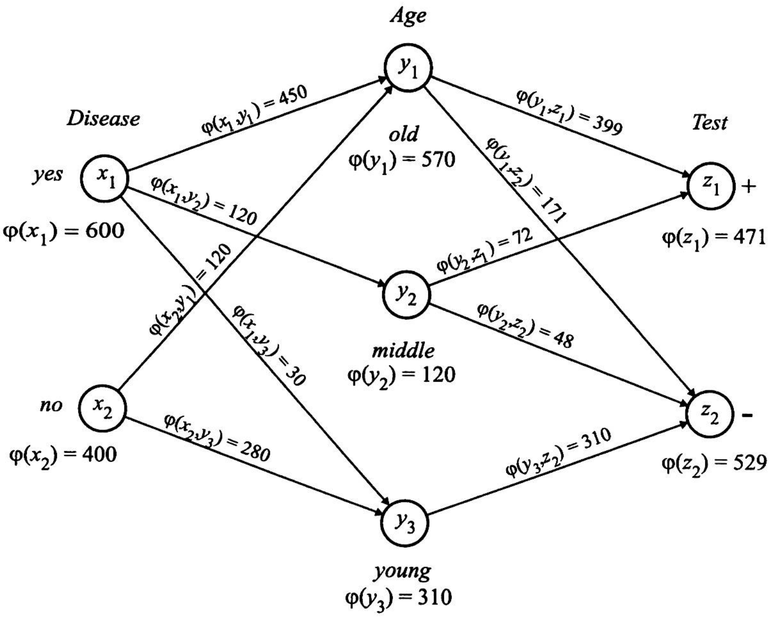


Fig. 1. Flow graph.

Throughflow of node y_1 is equal to $\varphi(y_1) = \varphi(x_1, y_1) + \varphi(x_2, y_1) = \varphi(y_1, z_1) + \varphi(y_2, z_2) = 570$. □

We will now define a *normalized flow graph*.

A normalized flow graph is a *directed, acyclic, finite* graph $G = (N, \mathcal{B}, \sigma)$, where N is a set of *nodes*, $\mathcal{B} \subseteq N \times N$ is a set of *directed branches* and $\sigma : \mathcal{B} \rightarrow \; < 0, 1 >$ is a *normalized flow* of (x, y) and

$$\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)} \tag{5}$$

is a *strength* of (x, y). Obviously, $0 \leq \sigma(x, y) \leq 1$. The strength of the branch (multiplied by 100) expresses simply the percentage of a total flow through the branch.

In what follows we will use normalized flow graphs only, therefore by flow graphs we will understand normalized flow graphs, unless stated otherwise.

With every node x of a flow graph G we associate its *inflow* and *outflow* defined by

$$\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(y, x), \tag{6}$$

$$\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x, y). \tag{7}$$

Obviously for any internal node x, we have $\sigma_+(x) = \sigma_-(x) = \sigma(x)$, where $\sigma(x)$ is a *normalized throughflow* of x.

Moreover, let

$$\sigma_+(G) = \frac{\varphi_+(G)}{\varphi(G)} = \sum_{x \in I(G)} \sigma_-(x), \tag{8}$$

$$\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(G)} \sigma_+(x). \tag{9}$$

Obviously, $\sigma_+(G) = \sigma_-(G) = \sigma(G) = 1$.

Example (cont.) The normalized flow graph of the flow graph presented in Fig. 1 is given in Fig. 2.

In the flow graph, e.g., $\sigma(x_1) = 0.60$, that means that 60% of total inflow is associated with input x_1. The strength $\sigma(x_1, y_1) = 0.45$ means that 45% of total flow of x_1 flows through the branch (x_1, y_1) etc. □

Let $G = (N, \mathcal{B}, \sigma)$ be a flow graph. If we invert direction of all branches in G, then the resulting graph $G = (N, \mathcal{B}', \sigma')$ will be called an *inverted* graph of G. Of course, the inverted graph G' is also a flow graph and all inputs and outputs of G become inputs and outputs of G', respectively.

Example (cont.) The inverted flow graph of the flow graph from Fig. 2 is shown in Fig. 3. □

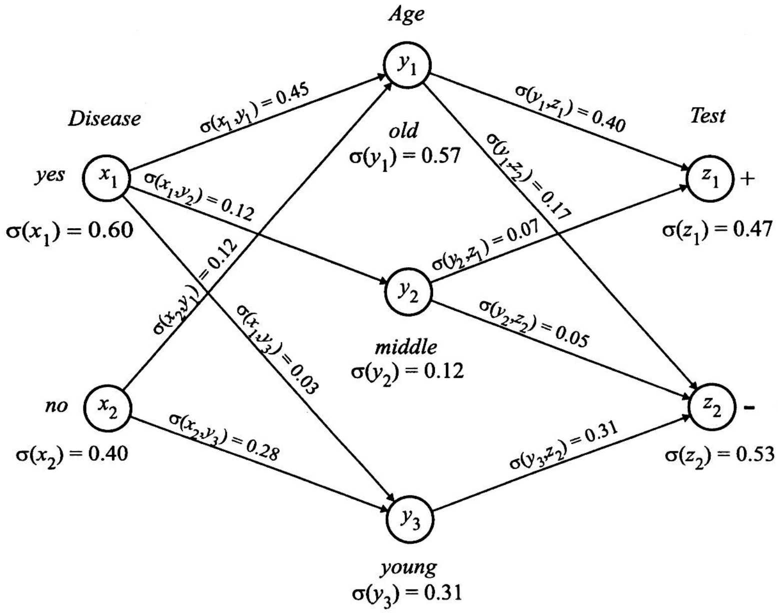


Fig. 2. Normalized flow graph.

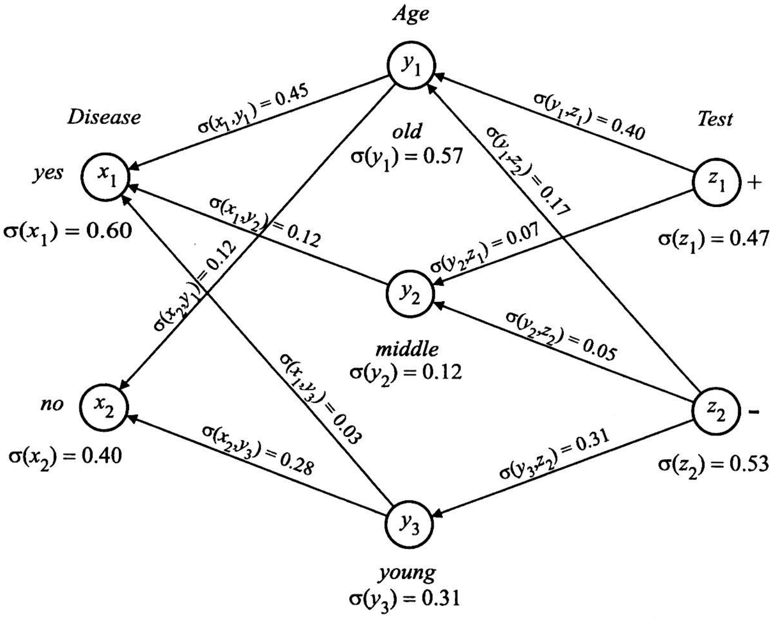


Fig. 3. Inverted flow graph.

1.3 Certainty and Coverage Factors

With every branch (x, y) of a flow graph G we associate the *certainty* and the *coverage factors.*

The *certainty* and the *coverage* of (x, y) are defined by

$$cer(x, y) = \frac{\sigma(x, y)}{\sigma(x)}, \tag{10}$$

and

$$cov(x, y) = \frac{\sigma(x, y)}{\sigma(y)}. \tag{11}$$

respectively.

Evidently, $cer(x, y) = cov(y, x)$, where $(x, y) \in \mathcal{B}$ and $(y, x) \in \mathcal{B}'$.

Example (cont.) The certainty and the coverage factors for the flow graph presented in Fig. 2 are shown in Fig. 4.

For example, $cer(x_1, y_1) = \frac{\sigma(x_1,y_1)}{\sigma(x_1)} = \frac{0.45}{0.60} = 0.75$, and $cov(x_1, y_1) = \frac{\sigma(x_1,y_1)}{\sigma(y_1)}$ $= \frac{0.45}{0.57} \approx 0.79$. □

Below some properties of certainty and coverage factors, which are immediate consequences of definitions given above, are presented:

$$\sum_{y \in O(x)} cer(x, y) = 1, \tag{12}$$

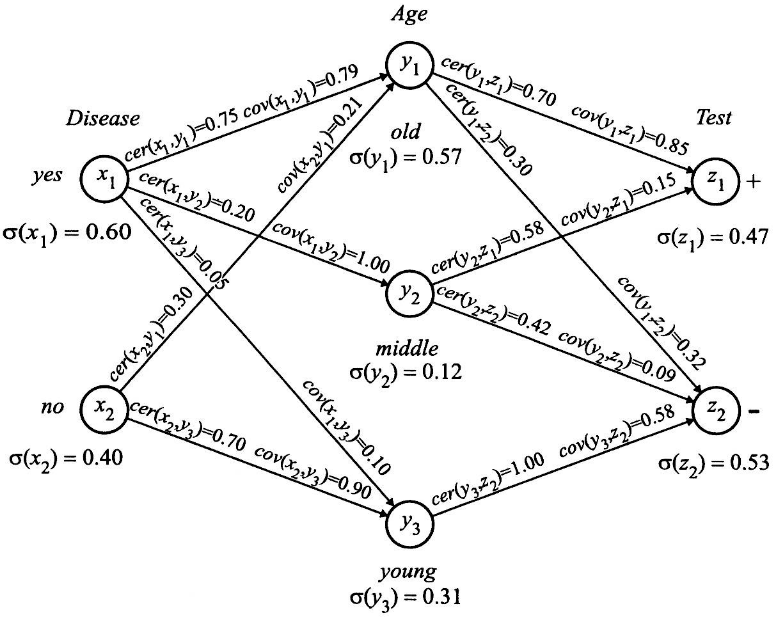


Fig. 4. Certainty and coverage.

$$\sum_{x \in I(y)} cov(x, y) = 1, \tag{13}$$

$$\sigma(x) = \sum_{y \in O(x)} cer(x, y)\sigma(x) = \sum_{y \in O(x)} \sigma(x, y), \tag{14}$$

$$\sigma(y) = \sum_{x \in I(y)} cov(x, y)\sigma(y) = \sum_{x \in I(y)} \sigma(x, y), \tag{15}$$

$$cer(x, y) = \frac{cov(x, y)\sigma(y)}{\sigma(x)}, \tag{16}$$

$$cov(x, y) = \frac{cer(x, y)\sigma(x)}{\sigma(y)}. \tag{17}$$

Obviously the above properties have a probabilistic flavor, e.g., equations (14) and (15) have a form of total probability theorem, whereas formulas (16) and (17) are Bayes' rules. However, these properties in our approach are interpreted in a deterministic way and they describe flow distribution among branches in the network.

1.4 Paths, Connections and Fusion

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