Jesús A. López
Emilio Benfenati
Werner Dubitzky (Eds.)

LNAI 3303

Knowledge Exploration in Life Science Informatics

International Symposium, KELSI 2004
Milan, Italy, November 2004
Proceedings

Springer

Jesús A. López Emilio Benfenati
Werner Dubitzky (Eds.)

Knowledge Exploration in Life Science Informatics

International Symposium KELSI 2004
Milan, Italy, November 25-26, 2004
Proceedings

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Jesús A. López
University of Southern Queensland
Department of Mathematics and Computing, Faculty of Sciences
Toowoomba, Queensland, 4350 Australia
E-mail: lopez@usq.edu.au

Emilio Benfenati
Istituto di Ricerche Famacologiche "Mario Negri"
(Laboratory of Environmental Chemistry and Toxicology)
Via Eritrea 62, 20157 Milano, Italy
E-mail: benfenati@marionegri.it

Werner Dubitzky
University of Ulster
School of Biomedial Sciences
Cromore Road, Coleraine BT 52 ISA, Northern Ireland
E-mail: w.dubitzky@ulster.ac.uk

Library of Congress Control Number: 2004115081

CR Subject Classification (1998): I.2, H.2.8, F.1, F.2, E.1

ISSN 0302-9743
ISBN 3-540-23927-8 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11358725 06/3142 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 3303

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Preface

This volume of the Springer Lecture Notes in Computer Science series contains the contributions presented at the International Symposium on Knowledge Exploration in Life Science Informatics (KELSI 2004) held in Milan, Italy, 25–26 November 2004. The two main objectives of the symposium were:

- To explore the symbiosis between information and knowledge technologies and various life science disciplines, such as biochemistry, biology, neuroscience, medical research, social sciences, and so on.
- To investigate the synergy among different life science informatics areas, including cheminformatics, bioinformatics, neuroinformatics, medical informatics, systems biology, socionics, and others.

Modern life sciences investigate phenomena and systems at the level of molecules, cells, tissues, organisms, and populations. Typical areas of interest include natural evolution, development, disease, behavior, cognition, and consciousness. This quest is generating an overwhelming and fast-growing amount of data, information, and knowledge, reflecting living systems at different levels of organization. Future progress of the life sciences will depend on effective and efficient management, sharing, and exploitation of these resources by computational means.

Life science informatics is fast becoming a generic and overarching information technology (IT) discipline for the life sciences. It includes areas such as cheminformatics, bioinformatics, neuroinformatics, medical informatics, socionics, and others. While the precise scientific questions and goals differ within the various life science disciplines, there is a considerable overlap in terms of the required key IT methodologies and infrastructures. Critical technologies include *databases*, *information bases* (i.e., containing aggregated, consolidated, derived data), *executable models* (i.e., knowledge-based and simulation systems), and *emerging grid computing* infrastructures and systems (facilitating seamless sharing and interoperation of widely dispersed computational resources and organizations). These base technologies are complemented by a range of enabling methodologies and systems such as knowledge management and discovery, data and text mining, machine learning, intelligent systems, artificial and computational intelligence, human-computer interaction, computational creativity, knowledge engineering, artificial life, systems science, and others.

This symposium was a first step towards investigating the synergy of these knowledge and information technologies across a wide range of life science disciplines.

Milan, Italy, November 2004

Jesús A. López
Emilio Benfenati
Werner Dubitzky

Acknowledgments

KELSI 2004 was organized and sponsored by the European Science Foundation's COST Action 282. We thank all Action 282 members for their support, inspiration, and contributions. Special thanks goes to the invited speakers – Prof. Michal Linial, Prof. Gustavo Deco, Prof. Stephen Muggleton, and Dr. Martin Kuiper – who reminded us of the mind-boggling breadth and depth of modern life science informatics and the challenges involved. Finally we would like to extend our gratitude to the members of the KELSI 2004 International Program Committee and the reviewers: Kevin Cohen (University of Colorado Health Sciences Center), Joerg Hakenberg (Humboldt-Universitaet zu Berlin), Sanjay Shamaro Kadam and Valadi Krishnamoorthy Jayaraman (both Centre for Development of Advanced Computing, Pune, India), David McSherry (University of Ulster), Alessandra Roncaglioni (Istituto di Ricerche Farmacologiche – "Mario Negri"), and Alexander Seewald (Austrian Research Institute for Artificial Intelligence).

International Program Committee

Jesús A. López (IPC Co-chair), University of Southern Queensland, Australia
Emilio Benfenati (IPC Co-chair and Local Organizer),
  Istituto di Ricerche Farmacologiche – "Mario Negri", Milan, Italy
Werner Dubitzky (IPC Co-chair), University of Ulster, Coleraine, UK
Agnar Aamodt, Norwegian University of Science and Technology (NTNU),
  Trondheim, Norway
Michael R. Berthold, University of Konstanz, Germany
Leon Bobrowski, Institute of Biocybernetics and Biomedical Engineering,
  Warsaw, Poland
Eric G. Bremer, Children's Memorial Hospital, Chicago, IL, USA
Rui Meireles de Brito, University of Coimbra, Coimbra, Portugal
Amilcar Cardoso, CISUC, University of Coimbra, Coimbra, Portugal
Gustavo Deco, Institucion Catalana de Recerca i Estudis Avançats (ICREA),
  Barcelona, Spain
Catherine DeSesa, SPSS Inc., Chicago, IL, USA
Moustafa M. Ghanem, Imperial College, London, UK
Geerd H. Diercksen, Max Planck Institute for Astrophysics, Garching, Germany
Robin Gras, Proteome Informatics Group, Swiss Institute of Bioinformatics,
  Switzerland
Des Higgins, University College Cork, Ireland
Chun-Hsi Huang, University of Connecticut, Storrs, USA
Seiya Imoto, University of Tokyo, Japan
Igor Jurisica, Ontario Cancer Institute, Canada
Astrid Lægreid, Norwegian University of Science and Technology, Trondheim, Norway
Michael Liebman, University of Pennsylvania, Philadelphia, USA
Stephanie McKeown, University of Ulster, UK
Alain Mille, Université Claude Bernard Lyon 1, LIRIS, France
Ray C. Paton, University of Liverpool, UK

Enric Plaza, IIIA-CSIC, Catalonia, Spain
Mathilde Romberg, Research Center Jülich, Jülich, Germany
Vlado Stankovski, University of Ljubljana, Slovenia
Vijayraghavan Sundararajan, Centre for Development of Advanced Computing, Pune, India
Brigitte Trousse, INRIA, AxIS Project Team, Sophia-Antipolis, France
Danny van Welden, University of Gent, Belgium
Gerhard Widmer, Austrian Research Institute for Artificial Intelligence, Vienna, Austria

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 3315: C. Lemaître, C.A. Reyes, J. González (Eds.), Advances in Artificial Intelligence – IBERAMIA 2004. XX, 987 pages. 2004.

Vol. 3303: J.A. López, E. Benfenati, W. Dubitzky (Eds.), Knowledge Exploration in Life Science Informatics. X, 249 pages. 2004.

Vol. 3265: R.E. Frederking, K.B. Taylor (Eds.), Machine Translation: From Real Users to Research. XI, 392 pages. 2004.

Vol. 3264: G. Paliouras, Y. Sakakibara (Eds.), Grammatical Inference: Algorithms and Applications. XI, 291 pages. 2004.

Vol. 3257: E. Motta, N.R. Shadbolt, A. Stutt, N. Gibbins (Eds.), Engineering Knowledge in the Age of the Semantic Web. XVII, 517 pages. 2004.

Vol. 3249: B. Buchberger, J.A. Campbell (Eds.), Artificial Intelligence and Symbolic Computation. X, 285 pages. 2004.

Vol. 3245: E. Suzuki, S. Arikawa (Eds.), Discovery Science. XIV, 430 pages. 2004.

Vol. 3244: S. Ben-David, J. Case, A. Maruoka (Eds.), Algorithmic Learning Theory. XIV, 505 pages. 2004.

Vol. 3238: S. Biundo, T. Frühwirth, G. Palm (Eds.), KI 2004: Advances in Artificial Intelligence. XI, 467 pages. 2004.

Vol. 3230: J.L. Vicedo, P. Martínez-Barco, R. Muñoz, M. Saiz Noeda (Eds.), Advances in Natural Language Processing. XII, 488 pages. 2004.

Vol. 3229: J.J. Alferes, J. Leite (Eds.), Logics in Artificial Intelligence. XIV, 744 pages. 2004.

Vol. 3215: M.G.. Negoita, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. LVII, 906 pages. 2004.

Vol. 3214: M.G.. Negoita, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. LVIII, 1302 pages. 2004.

Vol. 3213: M.G.. Negoita, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LVIII, 1280 pages. 2004.

Vol. 3209: B. Berendt, A. Hotho, D. Mladenic, M. van Someren, M. Spiliopoulou, G. Stumme (Eds.), Web Mining: From Web to Semantic Web. IX, 201 pages. 2004.

Vol. 3206: P. Sojka, I. Kopecek, K. Pala (Eds.), Text, Speech and Dialogue. XIII, 667 pages. 2004.

Vol. 3202: J.-F. Boulicaut, F. Esposito, F. Giannotti, D. Pedreschi (Eds.), Knowledge Discovery in Databases: PKDD 2004. XIX, 560 pages. 2004.

Vol. 3201: J.-F. Boulicaut, F. Esposito, F. Giannotti, D. Pedreschi (Eds.), Machine Learning: ECML 2004. XVIII, 580 pages. 2004.

Vol. 3194: R. Camacho, R. King, A. Srinivasan (Eds.), Inductive Logic Programming. XI, 361 pages. 2004.

Vol. 3192: C. Bussler, D. Fensel (Eds.), Artificial Intelligence: Methodology, Systems, and Applications. XIII, 522 pages. 2004.

Vol. 3191: M. Klusch, S. Ossowski, V. Kashyap, R. Unland (Eds.), Cooperative Information Agents VIII. XI, 303 pages. 2004.

Vol. 3187: G. Lindemann, J. Denzinger, I.J. Timm, R. Unland (Eds.), Multiagent System Technologies. XIII, 341 pages. 2004.

Vol. 3176: O. Bousquet, U. von Luxburg, G. Rätsch (Eds.), Advanced Lectures on Machine Learning. IX, 241 pages. 2004.

Vol. 3171: A.L.C. Bazzan, S. Labidi (Eds.), Advances in Artificial Intelligence – SBIA 2004. XVII, 548 pages. 2004.

Vol. 3159: U. Visser, Intelligent Information Integration for the Semantic Web. XIV, 150 pages. 2004.

Vol. 3157: C. Zhang, H. W. Guesgen, W.K. Yeap (Eds.), PRICAI 2004: Trends in Artificial Intelligence. XX, 1023 pages. 2004.

Vol. 3155: P. Funk, P.A. González Calero (Eds.), Advances in Case-Based Reasoning. XIII, 822 pages. 2004.

Vol. 3139: F. Iida, R. Pfeifer, L. Steels, Y. Kuniyoshi (Eds.), Embodied Artificial Intelligence. IX, 331 pages. 2004.

Vol. 3131: V. Torra, Y. Narukawa (Eds.), Modeling Decisions for Artificial Intelligence. XI, 327 pages. 2004.

Vol. 3127: K.E. Wolff, H.D. Pfeiffer, H.S. Delugach (Eds.), Conceptual Structures at Work. XI, 403 pages. 2004.

Vol. 3123: A. Belz, R. Evans, P. Piwek (Eds.), Natural Language Generation. X, 219 pages. 2004.

Vol. 3120: J. Shawe-Taylor, Y. Singer (Eds.), Learning Theory. X, 648 pages. 2004.

Vol. 3097: D. Basin, M. Rusinowitch (Eds.), Automated Reasoning. XII, 493 pages. 2004.

Vol. 3071: A. Omicini, P. Petta, J. Pitt (Eds.), Engineering Societies in the Agents World. XIII, 409 pages. 2004.

Vol. 3070: L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing - ICAISC 2004. XXV, 1208 pages. 2004.

Vol. 3068: E. André, L. Dybkjær, W. Minker, P. Heisterkamp (Eds.), Affective Dialogue Systems. XII, 324 pages. 2004.

Vol. 3067: M. Dastani, J. Dix, A. El Fallah-Seghrouchni (Eds.), Programming Multi-Agent Systems. X, 221 pages. 2004.

Vol. 3066: S. Tsumoto, R. Słowiński, J. Komorowski, J.W. Grzymała-Busse (Eds.), Rough Sets and Current Trends in Computing. XX, 853 pages. 2004.

Vol. 3065: A. Lomuscio, D. Nute (Eds.), Deontic Logic in Computer Science. X, 275 pages. 2004.

Vol. 3060: A.Y. Tawfik, S.D. Goodwin (Eds.), Advances in Artificial Intelligence. XIII, 582 pages. 2004.

Vol. 3056: H. Dai, R. Srikant, C. Zhang (Eds.), Advances in Knowledge Discovery and Data Mining. XIX, 713 pages. 2004.

Vol. 3055: H. Christiansen, M.-S. Hacid, T. Andreasen, H.L. Larsen (Eds.), Flexible Query Answering Systems. X, 500 pages. 2004.

Vol. 3048: P. Faratin, D.C. Parkes, J.A. Rodríguez-Aguilar, W.E. Walsh (Eds.), Agent-Mediated Electronic Commerce V. XI, 155 pages. 2004.

Vol. 3040: R. Conejo, M. Urretavizcaya, J.-L. Pérez-de-la-Cruz (Eds.), Current Topics in Artificial Intelligence. XIV, 689 pages. 2004.

Vol. 3035: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XII, 326 pages. 2004.

Vol. 3034: J. Favela, E. Menasalvas, E. Chávez (Eds.), Advances in Web Intelligence. XIII, 227 pages. 2004.

Vol. 3030: P. Giorgini, B. Henderson-Sellers, M. Winikoff (Eds.), Agent-Oriented Information Systems. XIV, 207 pages. 2004.

Vol. 3029: B. Orchard, C. Yang, M. Ali (Eds.), Innovations in Applied Artificial Intelligence. XXI, 1272 pages. 2004.

Vol. 3025: G.A. Vouros, T. Panayiotopoulos (Eds.), Methods and Applications of Artificial Intelligence. XV, 546 pages. 2004.

Vol. 3020: D. Polani, B. Browning, A. Bonarini, K. Yoshida (Eds.), RoboCup 2003: Robot Soccer World Cup VII. XVI, 767 pages. 2004.

Vol. 3012: K. Kurumatani, S.-H. Chen, A. Ohuchi (Eds.), Multi-Agents for Mass User Support. X, 217 pages. 2004.

Vol. 3010: K.R. Apt, F. Fages, F. Rossi, P. Szeredi, J. Váncza (Eds.), Recent Advances in Constraints. VIII, 285 pages. 2004.

Vol. 2990: J. Leite, A. Omicini, L. Sterling, P. Torroni (Eds.), Declarative Agent Languages and Technologies. XII, 281 pages. 2004.

Vol. 2980: A. Blackwell, K. Marriott, A. Shimojima (Eds.), Diagrammatic Representation and Inference. XV, 448 pages. 2004.

Vol. 2977: G. Di Marzo Serugendo, A. Karageorgos, O.F. Rana, F. Zambonelli (Eds.), Engineering Self-Organising Systems. X, 299 pages. 2004.

Vol. 2972: R. Monroy, G. Arroyo-Figueroa, L.E. Sucar, H. Sossa (Eds.), MICAI 2004: Advances in Artificial Intelligence. XVII, 923 pages. 2004.

Vol. 2969: M. Nickles, M. Rovatsos, G. Weiss (Eds.), Agents and Computational Autonomy. X, 275 pages. 2004.

Vol. 2961: P. Eklund (Ed.), Concept Lattices. IX, 411 pages. 2004.

Vol. 2953: K. Konrad, Model Generation for Natural Language Interpretation and Analysis. XIII, 166 pages. 2004.

Vol. 2934: G. Lindemann, D. Moldt, M.
Regulated Agent-Based Social Systems.
2004.

Vol. 2930: F. Winkler (Ed.), Automated D
ometry. VII, 231 pages. 2004.

Vol. 2926: L. van Elst, V. Dignum, A. Abe
Agent-Mediated Knowledge Management. XI
2004.

Vol. 2923: V. Lifschitz, I. Niemelä (Eds.), Logi
ming and Nonmonotonic Reasoning. IX, 365 p

Vol. 2915: A. Camurri, G. Volpe (Eds.), Gest
Communication in Human-Computer Interact
558 pages. 2004.

Vol. 2913: T.M. Pinkston, V.K. Prasanna (Eds.),
formance Computing - HiPC 2003. XX, 512 pag

Vol. 2903: T.D. Gedeon, L.C.C. Fung (Eds.), AI 2
vances in Artificial Intelligence. XVI, 1075 pages

Vol. 2902: F.M. Pires, S.P. Abreu (Eds.), Progress
ficial Intelligence. XV, 504 pages. 2003.

Vol. 2892: F. Dau, The Logic System of Concept
with Negation. XI, 213 pages. 2003.

Vol. 2891: J. Lee, M. Barley (Eds.), Intelligent Agen
Multi-Agent Systems. X, 215 pages. 2003.

Vol. 2882: D. Veit, Matchmaking in Electronic Mar
XV, 180 pages. 2003.

Vol. 2871: N. Zhong, Z.W. Raś, S. Tsumoto, E. Su
(Eds.), Foundations of Intelligent Systems. XV, 697 pa
2003.

Vol. 2854: J. Hoffmann, Utilizing Problem Structure Planing. XIII, 251 pages. 2003.

Vol. 2843: G. Grieser, Y. Tanaka, A. Yamamoto (Eds. Discovery Science. XII, 504 pages. 2003.

Vol. 2842: R. Gavaldá, K.P. Jantke, E. Takimoto (Eds.) Algorithmic Learning Theory. XI, 313 pages. 2003.

Vol. 2838: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Knowledge Discovery in Databases: PKDD 2003. XVI, 508 pages. 2003.

Vol. 2837: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Machine Learning: ECML 2003. XVI, 504 pages. 2003.

Vol. 2835: T. Horváth, A. Yamamoto (Eds.), Inductive Logic Programming. X, 401 pages. 2003.

Vol. 2821: A. Günter, R. Kruse, B. Neumann (Eds.), KI 2003: Advances in Artificial Intelligence. XII, 662 pages. 2003.

Vol. 2807: V. Matoušek, P. Mautner (Eds.), Text, Speech and Dialogue. XIII, 426 pages. 2003.

Vol. 2801: W. Banzhaf, J. Ziegler, T. Christaller, P. Dittrich, J.T. Kim (Eds.), Advances in Artificial Life. XVI, 905 pages. 2003.

Vol. 2797: O.R. Zaïane, S.J. Simoff, C. Djeraba (Eds.), Mining Multimedia and Complex Data. XII, 281 pages. 2003.

Vol. 2792: T. Rist, R.S. Aylett, D. Ballin, J. Rickel (Eds.), Intelligent Virtual Agents. XV, 364 pages. 2003.

Vol. 2782: M. Klusch, A. Omicini, S. Ossowski, H. Laamanen (Eds.), Cooperative Information Agents VII. XI, 345 pages. 2003.

Table of Contents

A Pen-and-Paper Notation for Teaching Biosciences

Johannes J. Mandel[1,2] and Niall M. Palfreyman[1]

[1] Dept. of Biotechnology and Bioinformatics,
Weihenstephan University of Applied Sciences, Freising, Germany
`{niall.palfreyman,johannes.mandel}@fh-weihenstephan.de`
[2] School of Biomedical Sciences, University of Ulster, Coleraine, Northern Ireland

Abstract. The authors introduce a graphical notation for representing general dynamical systems and demonstrate its use in three commonly occurring systems in the biosciences. They also indicate how the notation is used to facilitate the acquisition and transfer by students of skills in constructing equations from a verbal description of a system.

1 Modelling in the Biosciences

In her book "Making Sense of Life", Evelyn Fox Keller [1] recounts a confrontation at the 1934 Cold Spring Harbour Symposium on Quantitative Biology between Nicolas Rashevsky and Charles Davenport concerning Rashevsky's [2] mathematical model of division in an idealised spherical cell. Davenport's comment on the model was:

> "I think the biologist might find that whereas the explanation of the division of the spherical cell is very satisfactory, yet it doesn't help as a general solution because a spherical cell isn't the commonest form of cell."

which elicited the following retort from Rashevsky:

> "It would mean a misunderstanding of the spirit and methods of mathematical sciences should we attempt to investigate more complex cases without a preliminary study of the simple ones."

What we observe in this altercation is a deep-set cultural division between biologists and mathematical scientists, and one which must be experienced at some level by any student entering a degree programme in a discipline combining biology with the mathematical or technical sciences. There is a mildly schizophrenic atmosphere about such programmes arising from the diverse approaches of the two groups of scientists: The biologist must learn early in his career that living systems are inherently complex - too complex to hope to understand or explain them in all their gory detail. The engineer on the other hand develops during her training a confidence in her own ability to describe and possibly explain the world in terms of relatively simple equations. Whereas the biologist learns to accept a provisional lack of explanation, the engineer learns to need to explain.

J.A. López et al. (Eds.): KELSI 2004, LNAI 3303, pp. 1–8, 2004.

The result of this division is that the student of the mathematical or technical biosciences is pulled in two conflicting directions: she is required on the one hand to develop a deep appreciation of the complexity of living systems, yet must simultaneously become adept in the technical skill of modelling this complexity mathematically at a level which admits tractable solution. The central skill that this student must learn is therefore to abstract from a given biological system the essential mathematical structure.

Our experience is that bioscience students often have difficulties in learning this skill, and that these difficulties stem from a single question which is repeatedly voiced by our students: "I know how to solve the equations, but I have no idea how to derive these equations from a physical description of the problem!"

In this article we offer three components of a solution to this problem:

1. We propose a pen-and-paper graphical notation (mutuality nets) for describing the dynamical structure of a system. Mutuality nets emphasise the structural similarities between different systems, thus enabling the transfer of knowledge between systems.
2. We define an unambiguous procedure for transcribing mutuality nets into mathematical equations.
3. We illustrate the use of mutuality nets by using them to formulate three design patterns for situations commonly arising in the biosciences. "Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over" (Christopher Alexander, quoted in [3]).

In section 2 we use the *Rain-barrel* pattern to demonstrate how mutuality nets portray the generic dynamical structure in a variety of structurally similar systems, and how this structure can be used to derive a mathematical model. In section 3 we formulate the *Investment* pattern, which describes catalytic processes, and in section 4 we use the *Delayed balancing* pattern to describe the dynamical structure of oscillating systems. Finally, in section 5 we discuss briefly how mutuality nets are woven into a currently running course in bioprocess engineering.

2 Rain-Barrel: Using Feedback to Seek Equilibrium

Mutuality nets arose out of teaching a first course in bioprocess engineering, where almost every equation can be derived in one of two ways - as a **balance equation** for the processes affecting some state variable (**stock**) s:

$$\dot{s} = \text{(sum of input processes) - (sum of output processes)} \qquad (1)$$

or as a **rate equation** for the stocks s_i coordinated by a process p:

$$p = \frac{\dot{s}_1}{a_1} = \frac{\dot{s}_2}{a_2} = \ldots = -\frac{\dot{s}_i}{a_i} = -\frac{\dot{s}_{i+1}}{a_{i+1}} \qquad (2)$$

A mutuality net links these two kinds of equation in a network of interacting stocks and processes. It is a straightforward adaptation of stock and flow diagrams [4] and Petri nets [5], [6], [7] for use in the biosciences; it has been discussed elsewhere [8], [9], and will be described in detail in a forthcoming paper.

To see how mutuality nets are used in teaching, let us use them to represent the very simple system of a leaky rain barrel (fig. 1), into which water runs at a constant rate, but whose contents leak out at a rate which is proportional to the current volume of water in the barrel. This model displays a wide variety of behavioural intricacies which fascinate students - see [10] for an extensive pedagogical discussion of the rain-barrel.

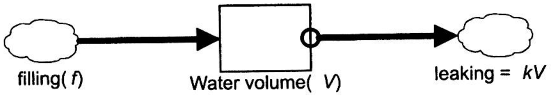

Fig. 1. The rain-barrel model

The first thing to notice here about the rain-barrel model is its wide applicability to biological systems. The following is just a short list of systems whose dynamical structure matches that of the rain-barrel:

- Infusion and subsequent uptake of medication in the blood system.
- mRNA / protein synthesis and degradation.
- Substrate levels in a continuous-feed bioreactor.
- Heating of a body and heat loss to environment.
- Approach to terminal velocity in a falling body.
- Growth of a feeding organism with energy loss through respiration.

Once a student has understood the behaviour of the rain-barrel model, he has little trouble in transferring this knowledge to any of the above situations. In this way mutuality nets facilitate transfer by visually representing the essential dynamical structure common to all of them. In addition this representation facilitates thinking, discussion and the exchange of views by lending itself to simple pen-and-paper constructions.

To obtain the dynamical equation of the rain-barrel system, we transcribe the above diagram into mathematical notation. This is done by treating each box (e.g.: V in the above diagram) as a state variable, and each cloud (e.g. *filling* and *leaking*) as a process which either augments or depletes the value of the state variables to which it is connected by an arrow. The circle notation means in the case of the rain-barrel that V is also an information source for the leaking process, thus making V available to appear in the equation $leaking = kV$. In this way we find the balance equation $\dot{V} = f - kV$, which can easily be solved either analytically or numerically by students to find the typical equilibrium-seeking behaviour of the rain-barrel shown in fig. 2.

Experimenting with the rain-barrel pattern makes clear to students the importance of feedback for system behaviour, since it is precisely the feedback

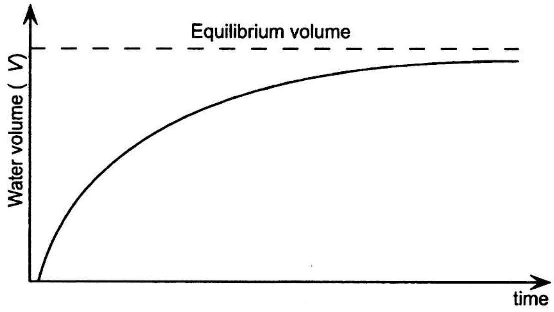

Fig. 2. Equilibrium-seeking behaviour of the rain-barrel

nature of the interaction between V and *leaking* in the rain-barrel which leads to its distinctive behaviour. By experimenting with the feedback constant k, they discover for themselves how it affects the convergence rate of the basic equilibrating behaviour.

3 Investment: Eliminating Idols

In our next example we shall see how the syntax rules of mutuality nets can aid students in the derivation of the Michaelis-Menten (M-M) equation [11] for the enzymatic splitting of a single substrate S.

Our first approximation to the M-M system is a simple adaptation of the rain-barrel model in which we assume that the enzyme E is an **idol** of the system in the sense that it is a state variable which affects the reaction, but without itself being affected by the reaction. Such a system might be denoted as in fig. 3. Here the constant value E conditions the process *splitting* according to the function kES, where k is the rate constant for the reaction.

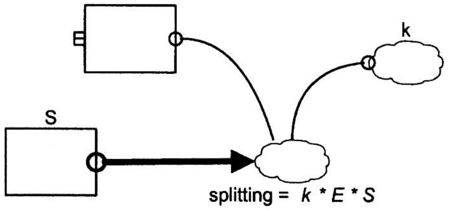

Fig. 3. Syntactically incorrect model of the M-M system

In fig. 3 we distinguish between two kinds of influence: *cause* and *condition*. A **cause** (thick, straight arrow) denotes an incremental flow of quantity between a stock and a process; a **condition** (thin, curved connector) makes available the value of its source (denoted by the small circle) to its target. We say that *splitting causes* changes in the substrate level S, and that E *conditions* this splitting.

Of course, the only problem with this model is that it is physically incorrect! It treats the product kE as the rate constant in an exponential process *splitting*, whereas in reality this is only a part of the complete enzymatic process - even the behavioural curve arising from this model is incorrect.

At this point we offer students as a guide the **mutuality rule** which, although not part of the syntax of mutuality nets, nevertheless constitutes a strong recommendation, particularly when using mutuality nets to model biological systems: *A condition should only connect one process to another process.* The effect of this rule is to discourage the formation of idols such as E in a dynamical model; if we wish E to condition the *splitting* process, then we should usually connect them with a cause, thereby at least admitting the possibility of a mutual interaction between *splitting* and E (hence the name "mutuality rule").

This is, of course, the case in reality, since E actually effects the splitting by physically investing itself in the splitting process. Yet it is also the case that the quantity of E in the system remains unaffected when the reaction is complete. In order to combine these two requirements, we are compelled to introduce a new state variable representing the transitory enzyme-substrate complex ES. This leads us to the physically correct model shown in fig. 4.

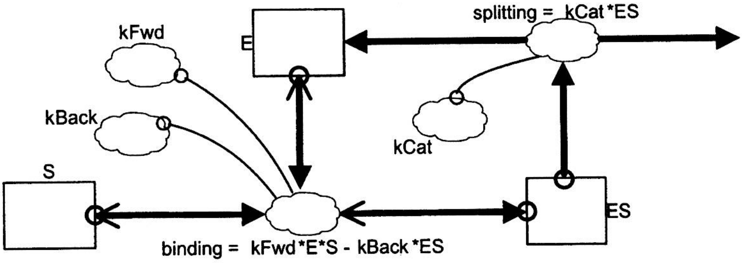

Fig. 4. Syntactically correct model of the M-M system

On the basis of this corrected model it is simple to first transcribe the complete dynamical equations for the M-M system:

$$\frac{dS}{dt} = (kBack)(ES) - (kFwd)(E)(S) \tag{3}$$

$$\frac{dE}{dt} = (kBack + kCat)(ES) - (kFwd)(E)(S) \tag{4}$$

$$\frac{dES}{dt} = (kFwd)(E)(S) - (kBack + kCat)(ES) \tag{5}$$

and then if required deduce the M-M equation by imposing the condition $ES = const$ and defining the M-M constant $K_m \equiv (kBack + kCat)/kFwd$.

From our consideration of the M-M system we have made two discoveries. First, the mutuality rule that no state variable can purely condition a process helps us to formulate a physically realistic mathematical model of the system -

something students often need help with. Second, the cyclical structure of the M-M model in fig. 4 is again a pattern commonly found in the biosciences, which we call "**Investment**". This pattern represents any situation where something is invested in the short term in order to return itself plus a payoff in the long term. Examples of the Investment pattern are:

- Investment of energy by organisms in foraging activities in order to gain energy from food.
- In the cell, phosphorylation of ADP represents an investment which is returned on hydrolysis of ATP, and which transports energy in the process.

4 Delayed Balancing: Creating Oscillations

In this section we shall introduce one final model which illustrates the relationship between oscillatory behaviour and feedback delays [12]. Imagine modifying the basic rain-barrel pattern by introducing a delay in the availability of information regarding the current level of water in the barrel. In this case the leak responds not to the *current* water level, but to some prior level, and the result is that the behaviour becomes no longer a direct convergence to equilibrium, but instead an oscillation about equilibrium as shown in fig. 5.

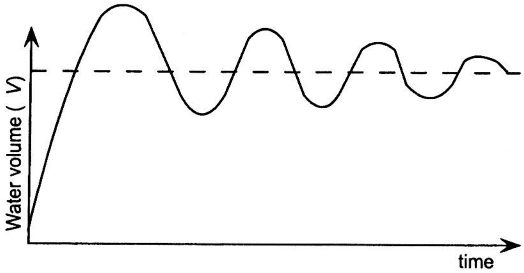

Fig. 5. Oscillations in the delayed-feedback rain-barrel

We can see how such a delay in the equilibrating feedback can lead to oscillatory behaviour, but how do feedback delays arise in the first place? A typical way in which delays can occur is if a process depends not merely upon feedback from its source, but is also modulated by feedback from its effects, as in the Lotka-Volterra model [13] of fig. 6.

The important point in the Lotka-Volterra model is that rabbits are increased by *birthing* and foxes are reduced by *dying*, but both of these effects are countered by the process of *interacting* between the two populations. So where does the delay come in *interacting* is conditioned by two *balancing effects* R and F, which react only sluggishly to changes caused by the *interacting* process. The oscillations of the Lotka-Volterra predator-prey model are well-known, and result directly from the delay thus introduced.