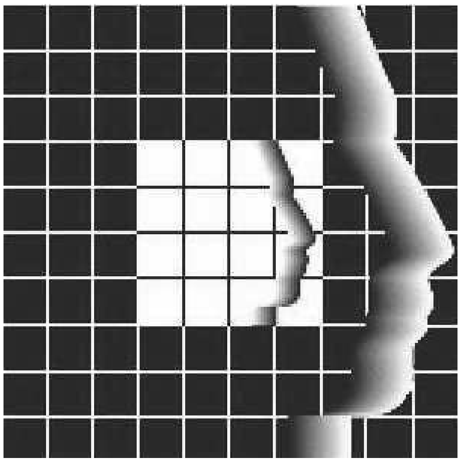
Model-based and Qualitative Reasoning in Biomedicine

Working notes of the workshop held during
The 9th *European Conference on Artificial Intelligence in Medicine, AIME'03*
Protaras, Cyprus, 19th October, 2003



Peter Lucas

(Editor)

Preface

These are the working notes of the workshop on MODEL-BASED AND QUALITATIVE REASONING IN BIOMEDICINE, which was held during the *European Conference on Artificial Intelligence in Medicine, AIME'03*, on 19th October, 2003, in Protaras, Cyprus. The workshop brought together various researchers involved in the development and use of model-based and qualitative reasoning methods in tackling biomedical problems.

Much of the biomedical knowledge is essentially *model-based*, as it is the understanding of the structure and function of biomedical systems that researchers wish to achieve, and this is best done by developing models of these systems. In situations where it may not be appropriate or possible to use quantitative methods, researchers use *qualitative* approaches. Depending on the biomedical problem concerned, such descriptions may involve causal, temporal and spatial knowledge, possibly of an uncertain nature. Also in the medical management of disorders in patients, qualitative and model-based approaches are being used. For example, systems used for diagnosing disease rely on explicit models of normal or abnormal structure and behaviour (often referred to as 'first-principles models') of the underlying disease process. Qualitative knowledge plays a role in the modelling of disease and treatment processes, including the handling of the uncertainty involved in these processes.

Hence, there is little doubt that model-based and qualitative methods fit the biomedical domain really well. However, one of the problems with research in the biomedical field is that researchers applying model-based and qualitative-reasoning methods are often closely linked to their application field, such as, for example, cell biology or clinical medicine, and find it difficult to keep in contact with colleagues doing similar research, but working in a different biomedical application field. This is even more difficult if the techniques used are also different. For example, researchers involved in Bayesian network research and researchers using qualitative simulation methods hardly exchange views and ideas, despite the fact that their methods have in common that they emphasise representing qualitative biomedical knowledge. It was the aim of this workshop to bring together researchers along the entire spectrum of the biomedical field, from health-care research and clinical medicine to human biology, using a variety of methods and techniques, from (qualitative) Bayesian networks and symbolic machine learning, to qualitative simulation. By looking at the table of content of these working notes, the reader may observe that these aim have indeed been achieved.

I am grateful to my colleagues who served on the programme committee of the workshop (Klaus-Peter Adlassnig, Pedro Barahona, Ivan Bratko, George Coghill, Enrico Coiera, Carlo Combi, Marie-Odile Cordier, Marek Druzdzel, John Fox, Raffaella Guglielmann, Werner Horn, Liliana Ironi, Hidde de Jong, Elpida Keravnou, Rudibert King, Benjamin Kuipers, Casimir Kulikowski, Peter Lucas, Simon Parsons, René Quiniou, Silja Renooij, Steffen Schulze-Kremer, Yuval Shahar, Robert Trelease, and Stefania Tentoni). They carefully read and reviewed each submission. Thanks are further due to Ivan Bratko and Linda van der Gaag for accepting my invitation to give an invited talk at the workshop.

The workshop was organised by the Biomedical Task Group of MONET[1], the Network of Excellence in Model-based Systems and Qualitative Reasoning. Without the commitment of the task-group members and the generous financial support offered by MONET, it would not have been possible to reach the same level of coverage of relevant topics as was achieved. I am grateful to the task-group members who were helpful in making the workshop a success,

[1] http://monet.aber.ac.uk

in particular: Liliana Ironi, Raffaella Guglielmann, Stefania Tentoni, Marie-Odile Cordier, René Quiniou, and Philip Bang. Last but not least, the participants of the workshop made the effort of organising the workshop worthwhile. Also to them I would like to express my gratitude.

Peter Lucas, University Nijmegen, The Netherlands

28th September, 2003

Contents

Invited Talks

Automated Modelling with Qualitative Representations

Ivan Bratko

Faculty of Computer and Information Sc.,
University of Ljubljana
Trzaska 25, Ljubljana, Slovenia

Abstract

One way of automating the modelling task is by means of machine learning. Observed behaviours of a modelled system are used as examples for a learning algorithm which constructs a model that is consistent with the data. In this paper we review some approaches to the learning of qualitative models, either from numerical data or qualitative observations: learning QDE models, learning qualitative trees, and Q^2 learning.

Introduction

Model construction is usually the most demanding aspect of modelling and simulation. In this paper we look at research that aims at automating this task whereby making use of qualitative representations.

One idea is to use observations of the modelled system, obtained through measurements, and try to find a model that would reproduce behaviours similar to the observed ones. This task is known as *system identification*, and is just the opposite of system simulation. Of course, to be of interest, the model induced from observations should be more general than the observations themselves. The induced model should be capable of making predictions also in situations other than those literally included among the observations. This task can also be viewed as machine learning from examples. The observed system behaviours are taken as examples for a learning algorithm, and the result of learning, usually called a *theory*, or a *hypothesis* induced from the examples, represents a model of the system.

In this paper we consider approaches to system identification that involve qualitative descriptions. One possibility is that the result of learning is a qualitative model. Another possibility is that the induced model is quantitative, as in traditional numerical system identification, but the construction of the quantitative model relies on using an intermediate qualitative model or qualitative background knowledge.

These possibilities are illustrated in Figure 1. Typical scenarios are:

1. Transform observed data into a qualitative system's behaviour, apply symbolic learning to this qualitative description to obtain a qualitative model, typically in the form of a qualitative differential equation (QDE).

2. Induce a qualitative model from the observed data, and then transform this qualitative model into a quantitative model (Q2Q transformation) so that the so obtained final model respects the constraints in the qualitative model and fits the observed data numerically.

3. Perform the Q2Q transformation using the observed data and qualitative background knowledge provided by a domain expert.

It should be noted that, in comparison with traditional (quantitative) system identification, in "qualitative system identification" there is much more emphasis on obtaining comprehensible models, models that intuitively explain how the system works.

In the rest of this paper we review some realisations of the components involved in the above scenarios, and discuss some specific points of interest.

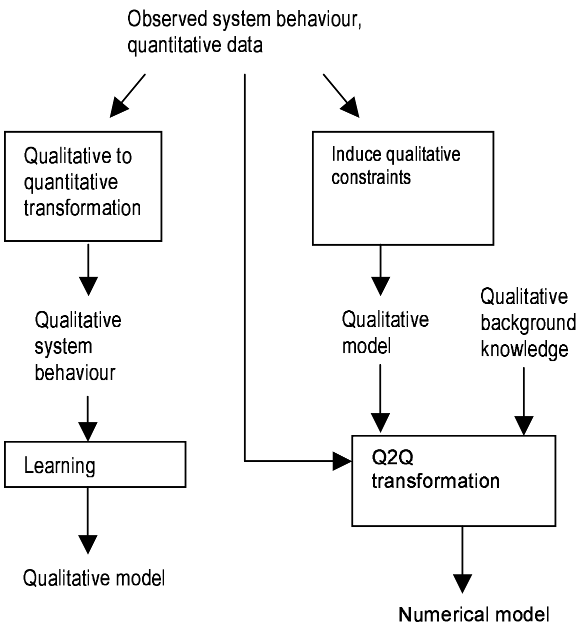

Observed system behaviour,
quantitative data
Qualitative to quantitative transformation
Induce qualitative constraints
Qualitative system behaviour
Qualitative model
Qualitative background knowledge
Learning
Q2Q transformation
Qualitative model
Numerical model

Fig. 1: Qualitative modelling scenarios

2 Learning QDE models

In this section we will look into learning models expressed as Qualitative Differential Equations (QDEs). QDEs are an abstraction of ordinary differential equations. Qualitative simulation based on QDE models is usually based on the assumption that variables in the QDE model behave continuously and smoothly in time. The best-known example is the QSIM qualitative simulation algorithm (Kuipers 1986; 1994).

A QDE model is defined by a set of variables, their possible qualitative values, and a set of constraints among these variables. Typical examples of such constraints are:

> $Y = M^{+}(X)$ (Y is a monotonically increasing function of X);
> $Y = M^{-}(X)$ (Y is a monotonically decreasing function of X);
> add(X, Y, Z) $(Z = X + Y)$;
> deriv(X, Y) $(Y = dX/dt)$.

All these constraints are applied to "qualitative states" of variables rather than on variables' numerical values. For example, such a qualitative state of variable X can be: positive and increasing in time, written as: X = (pos,inc). The add(X,Y,Z) constraint is satisfied, for example, by the following qualitative states: X = (pos,inc), Y = (pos,inc), Z= (pos,inc).

The problem of learning QDE models from example system behaviours is: given qualitative behaviours of a set of observed variables, find a set of qualitative constraints that are consistent with the given behaviours. There have been a number of attempts at automatically constructing QDE models from examples of system's behaviour. In the following we review some of these systems, their basic ideas and mention some points of discussion.

Basic QDE learning algorithm

The basic QDE learning algorithm, introduced by Coiera (1989) in his program GENMODEL, constructs a model from examples as follows:

1. Construct all the syntactically possible constraints, using all the observed variables (that is those appearing in the example behaviours) and the given repertoire of types of qualitative constraints.

2. Evaluate all the constraints constructed in step 1 on all the qualitative system's states in the given example behaviours. Retain those constraints that are satisfied by all the states, and discard all other constraints. The set of retained constraints constitute the induced qualitative model.

It is possible to induce, using this simple method, correct models for some simple systems, such as the U-tube or the spring-mass oscillator. For example, consider the learning of a qualitative model of the U-tube with GENMODEL. The U-tube system consists of two containers A and B, connected at the bottom by a thin pipe. So the three components form a U-shape. Suppose that initially there is some water in container A while container B is empty. The difference between the two water levels, LevA and LevB, causes a positive flow from A to B. Thus LevA will be decreasing and LevB increasing, until both levels become equal, and the flow becomes zero. This behaviour can be stated qualitatively as a sequence of three qualitative states that are given to GENMODEL as three examples of system's qualitative behaviour. GENMODEL will generate the possible qualitative constraints among the three observed variables, and find that three of these constraints are satisfied in all three observed qualitative states. These three constraints in conjunction constitute the model induced by GENMODEL:

> LevB = M^{-}(LevA), add(LevB,Flow,LevA),
> deriv(LevB, Flow)

This is actually a correct model of the U-tube. However, in general the GENMODEL algorithm is very limited because it relies on some strong assumptions:

(1) That *all* the variables in the target model are observed, so they explicitly appear in the example behaviours. The problem is, what to do if not all the variables in the target model were observed. In such a case we say that a variable that should appear in the model is "hidden" (that is, it is not mentioned in the example behaviours). GENMODEL does not handle hidden variables.

(2) The approach is biased toward learning the most *specific models* in the sense that these models contain *all* the possible constraints that are satisfied by the example data. There is of course no guarantee that all these constraints should actually be part of the target model.

(3) The resulting model is assumed to apply to the *complete* state space of the dynamic system. This is not appropriate for the cases when the system can be in more than one "operating region". For example, consider water level increasing in a container. When the level reaches the top of the container, the level can no longer keep increasing, and the system starts behaving according to a different law.

Hidden variables and generality of induced models

The difficulty with hidden variables can be illustrated by the U-tube example when the two levels LevA and

LevB are observed only. The GENMODEL algorithm finds that the only constraint satisfied by all the states in the example behaviour is:

LevB = M^{-}(LevA)

This model is under constrained. It allows for example an obviously impossible behaviour when LevA becomes (zero,std) and at the same time LevB is (pos,std). The GENMODEL algorithm cannot find a more specific model (that is one with more constraints) because such a model requires the introduction of new variables. Therefore a more general algorithm has to reconstruct also the "hidden" variables, for example the flow in our case. Say and Kuru's (1996) QSI algorithm introduces new variables as follows. It hypothesises the existence of a new variable, and constructs a possible qualitative constraint between this variable and existing variables. Such a constraint qualitatively defines the new variable. So QSI can in this U-tube example introduce a new variable, X, by constructing the constraint deriv(LevB,X). QSI executes the GENMODEL algorithm iteratively. In the second iteration, when X has been introduced, QSI will find three satisfied constraints:

LevB = M-(LevA), add(LevB, X, LevA),
deriv(LevB, X)

In this way QSI discovers the hidden variable X that precisely corresponds to the flow of water. This model only allows the given observed behaviour, so QSI stops here and outputs this as the final result.

Generally, QSI iteratively introduces new, "deep" variables, which enables the addition of further constraints to the model. Each successive model is therefore more specific, that is, it allows only a subset of behaviours of the more general models. In successive iterations, the "depth" of model also increases. A new variable is introduced with a constraint in which the new variable appears together with at least one old variable. These iterations stop when the model is "sufficiently specific". A model is accepted as sufficiently specific when it only allows an acceptable number of qualitative behaviours. The user of QSI has to specify an acceptable degree of behaviour branching allowed by a model, which is related to the generality of the model. In this way QSI rather nicely determines an appropriate number of new variables, and thus achieves an appropriate generality of the model.

Learning from positive-only examples

As nature can only provide positive examples and no negative examples (that is behaviours that in nature cannot happen), it is often considered that a model should be learned from positive-only examples. This is the case in both GENMODEL and QSI, as well as in several other systems including MISQ (Richards et al. 1992) and QOPH (Coghill et al. 2002). This is a non-typical situation for general machine learning. The problem of defining just the "right degree of generality" of a model is particularly critical when models are learned from positive only examples. Since there are no negative examples given, the completely unconstrained model (empty model, with no constraints) is consistent with the learning data. Such a model, although consistent with the observations, is of course useless. Therefore such models should be avoided by an appropriate learning bias, which should prevent useless, although consistent hypotheses. All the above mentioned systems are biased toward selecting a most specific model. GENMODEL simply selects the most specific model constructed from the given types of constraints and the observed variables. This is where GENMODEL stops, because it does not introduce new variables. However, it is not always possible to construct a sufficiently specific model just using the observed variables. MISQ (Richards et al. 1992) is similar to GENMODEL, but it introduces new variables aiming at a connected model; that is a model in which all the observed variables are connected by chains of constraints in which new variables may appear. The connectedness requirement is of course merely a heuristic that may not result in the intended model. The QSI system controls the introduction of new variables in a more general way that results in a more sophisticated learning bias: QSI constructs the most specific model using the observed variables and all the new variables up to the maximal depth. The depth is determined by the acceptable branching of the model. The model has to be sufficiently deep to prevent excessive branching. In this respect QSI makes a kind of an implicit closed world assumption, although this assumption is only enforced "softly". Namely, the QSI algorithm treats the states in the given example behaviours as positive examples, and the siblings of these states as potential negative examples.

Is "positive-only" really useful?

The learning from positive only examples in this context is often considered as an advantage of a learning system because the observations are supposed to come from nature which only provides positive examples. However, this advantage of learning from positive only examples is not so clear. As mentioned above, QSI makes a kind of closed world assumption by which it considers some things that were not observed, effectively as negative examples. Also, to compensate for the lack of negative examples, these algorithms adopt the bias toward the most specific models, which may also be debatable. On the other hand, the user (e.g. an expert) may well be able to specify negative examples on the basis of the background knowledge and the general understanding of the problem domain. So the restriction to the learning from positive *only* examples does not seem to be really necessary in practice.

It is sometimes considered that it is not realistic to expect that the domain expert be capable of providing negative examples, unless the expert already knows the target model completely. But then, if the model is already known, there would be no point in learning a model from data. I believe that such a view is mistaken, because it assumes that the expert either knows the right model *completely*, or has *no idea at all*. However, model building in practice is usually between these two extremes. The expert normally does have some ideas about the domain (often referred to as "background knowledge"), but these are insufficient to immediately put together a completely correct model. The incomplete expert's knowledge can often be expressed in terms of negative examples: the expert just states what he or she believes can surely not happen. For example, in the case of modelling a U-tube, the modeller may not be able to define a complete and correct model. Still, he may be easily capable of stating, by means of negative examples, that the amount of water in a container cannot be negative, and that the total amount of water in the whole system is constant.

Background knowledge

In addition to negative examples, the expert may also be able to specify some specific background knowledge that may be useful in the learning of qualitative models in the particular domain. Inductive Logic Programming (ILP) is the machine learning framework that ideally suits this situation. The following is the ILP problem formulation that applies to the learning of qualitative models from background knowledge BK, QDE constraints QC, and sets POS and NEG of positive and negative examples respectively. Given BK, QC, POS and NEG, find a model M such that:

```
  For each example P in POS:
      BK & QC & M   |--  P
  and for each example N in NEG:
      BK & QC & M   |-\-  N
```

Once the problem is so formulated in logic, a general purpose ILP learning program can be applied. Bratko et al. (1991) used such an ILP system GOLEM in an experiment to induce a model of a U-tube from positive and negative examples. Although this exercise was impeded by some technical limitations of GOLEM, it showed the advantages of using ILP: (a) it was not necessary to develop a special purpose learning program for QDE learning, and (b) since GOLEM (like most ILP systems) introduces new variables itself, there was no special care needed in respect of hidden variables. Another advantage that comes automatically with ILP is the learning of models with multiple operating regions. General purpose ILP programs generate multiple clause logic programs, so each clause may cover one operating region. The QOPH system (Coghill et al. 2002) also relies on using such a general ILP program called Aleph (Srinivasan 2000). QOPH does remarkably well with introducing new variables, although this relies on a number of heuristics that need further study.

Transforming numerical data to qualitative behaviours

The programs mentioned above learn qualitative models from given examples of qualitative behaviours. In an actual application, it is more likely, however, that the observed data are numerical. Most of these programs require a transformation of such numerical data into qualitative behaviours. Some of the above discussed approaches (Say and Kuru, 1996; Coghill et al. 2002) also include such a transformation. It is not easy to do this well, specially when there is noise in the numerical data. Džeroski's and Todorovski's (1995) QMN program is interesting in that it builds QDE models directly from numerical data, avoiding such a numerical-to-qualitative transformation. Program SQUID (Kay et al. 2000) is also relevant in this respect. It learns "semi-quantitative" models (a combination of QDEs with numerical elements) from numerical data. Another idea to handle numerical data is to apply QUIN in combination with QDE learning.

Experimental applications

The QDE learning programs reviewed above were usually tested on small experimental domains, and rarely on problems of realistic complexity. Probably the most impressive application on real-life data is described by Nau and Coiera (1997). Their system transforms signals measured in time into qualitative behaviours which are input into GENMODEL. They applied this system to actually measured cardio vascular signals from a number of patients and induced, from these data, qualitative models characterising individual patients. Another ambitious application oriented work is Mozetič's program QuMAS (Bratko et al., chapter 5) which learned models of the electrical system of the heart capable of explaining many types of cardiac arrhythmias. QuMAS did not use QDE constraints as modelling primitives, but a set of problem-specific logical descriptions used by what would now be recognised as an ILP learning system.

Assessment and future work

A number of systems exist for learning of QDE models. Their development demonstrates improvements in respect of several rather intricate problems involved in the learning of QDE models. In general, however, the impact of QDE learning techniques on the practice of qualitative modelling has been rather slow. This is particularly surprising in the view of enormous increase in the past decade of machine learning applications in other areas (Michalski et al. 1998). It seems that further progress is still required in several respects. These include: better methods for transforming numerical data into qualitative data, targeted explicitly towards the particular qualitative modelling language; deeper study of principles or heuristics associated with the discovery of hidden variables, the generality and the size of models; more effective use of general ILP techniques.

2 Learning qualitative trees

QUIN (Qualitative Induction) is a learning program that looks for qualitative patterns in numerical data (Šuc 2001; Šuc and Bratko 2001). Induction of the so-called qualitative trees is similar to the well-known induction of decision trees (e.g. CART, Breiman et al. 1984; C4.5, Quinlan 1993). The difference is that in decision trees the leaves are labelled with class values, whereas in qualitative trees the leaves are labelled with what we call *qualitatively constrained functions* (QCFs for short). These are a kind of monotonicity constraints generalised to functions of several variables. For example, Z = $M^{+,-}$(X,Y) says that Z monotonically increases in X and decreases in Y. If both X and Y increase, then according to this constraint, Z may increase, decrease or stay unchanged. In such a case, a QCF cannot make an unambiguous prediction of the qualitative change in Z.

QUIN takes as input a set of numerical examples and looks for regions in the data space where monotonicity constraints hold. A set of such regions are represented as a qualitative tree. As in decision trees, the internal nodes in a qualitative tree specify conditions that split the attribute space into subspaces. In a qualitative tree, however, each leaf specifies a QCF that holds among the input data that fall into that leaf.

As an example from biological modelling, consider the growth or decay of zooplankton, depending also on phytoplankton, in an aquatic system. QUIN may induce from measured concentrations Z and P of zoo and phytoplankton the following qualitative tree about the change in time DZ of zooplankton. This tree is here written as an if-then-else expression:

if P is low **then** DZ = M^{-}(Z) **else** DZ = $M^{+,+}$(P,Z)

This can be interpreted as: phytoplankton being the food for zooplankton, low P means shortage of food and the main effect is zooplankton dying; high P means plenty of food and the main effect is zooplankton growing, where the growth is qualitatively proportional to both P and Z.

3 Q^2 learning

Q^2 learning (qualitatively faithful quantitative prediction; Šuc et al. 2003) is based on an idea of numerical learning while respecting qualitative constraints. This corresponds to the right hand branch in Fig. 1. Q^2 learning consists of two stages:

1. Eliciting qualitative constraints that hold in the domain of modelling. Such constraints can have the form of qualitative trees, and can be provided by a domain expert, or obtained automatically from the observed data e.g. by QUIN.

2. Q2Q transformation (qualitative to quantitative transformation), whose goal is to transform qualitative monotonicity constraints (QCFs in the leaves of a qualitative tree) into numerical regression functions that respect these constraints and fit the observed data numerically.

The so obtained quantitative models have guaranteed "qualitative correctness" which is important with respect to their interpretability. While an expert may tolerate numerical errors in model's predictions, qualitative inconsistencies are particularly disturbing because they obscure the interpretation.

Q2Q transformation can be done in various ways. Šuc et al. (2003) used LWR (locally weighted regression), a well-known numerical regression technique to construct a numerical model from observed data. LWR's parameter was tuned so that the obtained numerical model satisfied the qualitative constraints. Another approach, called Qfilter (Šuc and Bratko 2003), also starts with LWR predictions, but then computes minimal changes in these predictions so that the modified predictions respect the qualitatuve constraints. More precisely, the changes are minimised with respect to the sum of their squares. This optimisation is formulated as a quadratic programming problem.

Experimental results show that Q^2 learning, in addition of ensuring the qualitatuve correctness, also significantly reduces numerical predictive error in comparison with standard techniques of numerical machine learning or numerical regression. In one surprizing application of Q^2 learning, it was possible to significantly simplify complex car simulation models, developed by the INTEC simulation company, thus drastically improving the time efficiency of the simulator.

4 Conclusions

Several approaches to automated modelling using various forms of qualitative representations were reviewed in the paper, including the learning of QDE models, learning of qualitative trees and Q2 learning (qualitatively faithful quantitative learning). The learning of QDE models has been explored by a number of researchers, but it seems it still requires significant improvements before it will make considerable impact to the practice of modelling. Qualitative trees and Q2 learning seem to be more ready in this respect. Some interesting applications of qualitative trees, including qualitative reverse engineering and skill reconstruction are reviewed in (Bratko and Šuc 2003).

References

Bratko, I., Šuc, D. (2003) Learning of qualitative models. *AAAI Magazine*, in press.

Bratko, I., Mozetič, I., Lavrač, N.. (1989) *KARDIO: A Study in Deep and Qualitative Knowledge for Expert Systems*. Cambridge, MA: MIT Press.

Bratko, I., Muggleton, S., Varšek, A. (1991) Learning qualitative models of dynamic systems. *Proc. Inductive Logic Programming ILP-91*, Viana do Castelo, Portugal, pp. 207-224. Also in: *Inductive Logic Programming* (ed. S. Muggleton) Academic Press 1992, pp.. 437-452.

Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J. (1984) *Classification and Regression Trees.* Monterey, CA: Wadsford.

Coiera, E. (1989) Generating qualitative models from example behaviours. DCS Report 8901, Dept. of Computer Sc., Univ. of New South Wales, Sydney, Australia.

G. M. Coghill, S. M. Garrett and R. D. King (2002) Learning Qualitative Models in the Presence of Noise, *QR'02 Workshop on Qualitative Reasoning*, Sitges, Spain 2002.

Džeroski, S., Todorovski, L. (1995) Discovering dynamics: from inductive logic programming to machine discovery. *J. Intell. Information Syst.*, 4:89-108.

Kuipers, B. (1986) Qualitative simulation. *Artificial Intelligence*, 29: 289-338.

Kuipers, B. (1994) *Qualitative Reasoning*. Cambridge, MA: MIT Press.

Kay, H., Rinner, B., Kuipers, B. (2000) Semi-quantitative system identification. *Artificial Intelligence*, 119: 103-140.

Michalski, R.S., Bratko, I., Kubat, M. (1998, eds.) *Machine Learning and Data Mining: Methods and Applications*. Wiley.

Nau, D.T., Coiera, E.W. (1997) Learning qualitative models of dynamic systems. *Machine Learning*, 26: 177-211.

Quinlan, J.R. (1992) Learning with continuous classes. *Proc. Fifth Australian Joint Conf. on Artificial Intelligence*, Hobart, Tasmania. World Scientific, pp. 343-348.

Quinlan, J.R. (1993) *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

Richards, B.I., Kraan, I., and Kuipers, B.J. (1992) Automatic abduction of qualitative models. *Proc. Tenth National Conf. Artificial Intelligence*, 723-728.

Say, A.C.C, Kuru, S. (1996) Qualitative system identification: deriving structure from behavior. *Artificial Intelligence*, 83: 75-141.

Srinivasan, A. (2000) Aleph web site: web.comlab.ox.ac.uk/oucl/research/areas/mach-learn/Aleph/aleph_toc.html

Šuc, D. (2001) *Machine Reconstruction of Human Control Strategies*. Ph. D. Thesis, Univ. of Ljubljana, Faculty of Computer and Information Sc.

Šuc, D., Bratko, I. (2001) Induction of qualitative trees. *Proc. ECML'01 (European Conf. Machine Learning)*, Freiburg, Germany. Springer-Verlag, LNAI Series.

Šuc, D., Bratko, I. (2003) Improving numerical predictions with qualitative constraints. *Proc. ECML'03 (European Conf. Machine Learning)*, Dubrovnik, Croatia.

Šuc, D., Vladušič, D., Bratko, I. (2003) Qualitatively faithful quantitative prediction. *Proc. IJCAI'03 (Int. Joint Conf. Artificial Intelligence)*, Acapulco, Mexico.

Model-based Reasoning with Qualitative Probabilistic Networks

Linda C. van der Gaag
Institute of Information and Computing Sciences, Utrecht University
PO Box 80.089, 3508 TB Utrecht, The Netherlands
linda@cs.uu.nl

Abstract

Qualitative probabilistic networks are graphical models of the probabilistic influences among a set of statistical variables. Each influence is associated with a qualitative sign, that indicates the direction of shift that it induces in the probability distribution for the influenced variable. Probabilistic reasoning with a qualitative network then amounts to computing the sign of the net influence of a number of observations on a variable of interest. In this paper, we review the basic concepts of qualitative networks and exemplify qualitative probabilistic reasoning with a real-life network.

1 Introduction

Qualitative probabilistic networks were introduced in the early 1990s for probabilistic reasoning in a qualitative way [1]. A qualitative probabilistic network is a qualitative model of a joint probability distribution over a set of statistical variables. It comprises a directed acyclic graph that encodes the statistical variables involved as nodes and the probabilistic influences between them as arcs. An arc $A \rightarrow B$ between two variables A and B expresses that observing a value for A occasions a shift in the probability distribution for B. The direction of this shift can be positive, negative or ambiguous, and is indicated by a qualitative sign.

Probabilistic reasoning with a qualitative network amounts to computing the net influence of one or more observations on a variable of interest. In a qualitative network for a medical diagnostic application, for example, reasoning provides for establishing whether or not the findings for a patient support the hypothesis that a particular disease is present. Reasoning is shaped by propagating and combining signs throughout the network [2]. The basic algorithm for this purpose is polynomial in the network's size. Probabilistic reasoning with a qualitative network thus is less demanding from a computational point of view than numerical probabilistic inference which is known to be NP-hard [3].

Real-life applications of qualitative probabilistic networks do not yet abound. Since qualitative networks capture the influences among their variables at a relatively coarse level of representation detail, they do not provide for modelling the intricacies involved in weighing conflicting influences. As a consequence, they do not provide for resolving trade-offs, that is, for establishing the net result of opposite shifts in the probability distribution for a variable of interest. Probabilistic reasoning will then typically result in ambiguous net influences. It is this tendency to yield ambiguous and, hence, uninformative results upon reasoning that has restricted the use of qualitative networks for real-life applications. In this paper, we demonstrate by means of a real-life network in the field of oesophageal cancer that qualitative networks can in fact be used to derive insightful results, even if they model complex trade-offs.

The paper is organised as follows. In Section 2, we briefly review Bayesian networks. In Section 3, we introduce the formalism of qualitative probabilistic networks. In Section 4, we present the basic algorithm for reasoning with a qualitative probabilistic network. In Section 5, we exemplify model-based probabilistic reasoning with a real-life qualitative network. In Section 6, we indicate recent advances in the field of qualitative probabilistic networks. The paper ends with some concluding observations in Section 7.

2 Preliminaries

Qualitative probabilistic networks bear a strong resemblance to Bayesian networks and, in fact, are generally looked upon as qualitative abstractions of these numerical models of uncertainty. Before introducing qualitative networks, therefore, we briefly review the formalism of Bayesian networks.

A *Bayesian network* is a model of a joint probability distribution over a set of statistical variables [4]. It consists of a qualitative part and an associated quantitative part. The qualitative part takes the form of a directed acyclic graph. Each node A in this digraph represents a statistical variable that takes one of a finite set of values. In this paper, we assume all variables to be binary, taking one of the values *true* and *false*; for abbreviation, we use a to denote $A = \textit{true}$ and $\bar{a}$ to denote $A = \textit{false}$. We further assume that a variable's values are ordered, where $\textit{true} > \textit{false}$.

The arcs in the digraph model the probabilistic influences between the represented variables. Informally speaking, an arc $A \rightarrow B$ between the nodes A and B indicates that there is an influence between the associated variables A and B. Absence of an arc between A and B means that the corresponding variables do not influence each other directly. The variables may influence each other indirectly, however, through an unblocked chain. We say that a chain between A and B is *blocked* by the available evidence if it includes either an observed variable with at least one outgoing arc, or an unobserved variable with two incoming arcs and no observed descendants. If all chains between the two variables are blocked, then there is no influence between them and they are considered conditionally independent given the available evidence [4].

Associated with the qualitative part of a Bayesian network are numerical quantities from the encoded probability distribution. With each variable A in the digraph is associated a set of *conditional probability distributions* $\Pr(A \mid \pi(A))$; each of these distributions describes the joint effect of a specific combination of values for the (immediate) predecessors $\pi(A)$ of A, on the probability distribution over A's values. The sets of probability distributions with each other constitute the quantitative part of the network.

Example. As an example, we consider the *Lymphatic Metastases* network shown in Figure 1. The network represents a small fragment of knowledge in oncology, pertaining to lymphatic metastases of an oesophageal tumour. The variable L represents the location of the primary tumour in the patient's oesophagus. The value *true* of the variable represents the information that the tumour resides in the lower two-third of the oesophagus; $\bar{l}$ expresses that the tumour is located in the oesophagus' upper one-third. As the cancer progresses, it typically results in lymphatic metastases, that is, in secondary tumours in lymph nodes. The variable M represents the extent of these metastases. The value *false* of M indicates that just the local and regional lymph nodes are affected; m denotes that the distant lymph nodes are affected by cancer cells. Which lymph nodes are local or regional and which are distant, depends on the location of the primary tumour in the oesophagus. The lymph nodes in the neck, or cervix, for example, are regional for a primary tumour in the upper one-third of the oesophagus, and distant otherwise. The lymph nodes near the truncus coeliacus in the upper abdomen, on the other hand, are always distant, irrespective of the location of the primary tumour. In the network, the variable C represents the presence or absence of metastases in the cervical lymph nodes; the variable T models the presence or absence of metastases near the truncus coeliacus. The presence of metastases in the cervical lymph nodes can be established by means of a sonography of the neck; the result of the sonography is captured by the variable S. □

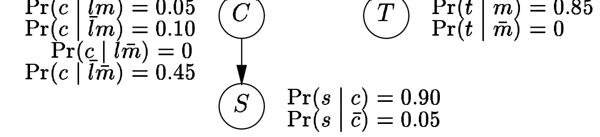

$\Pr(l) = 0.90$
L
M
$\Pr(m) = 0.15$
$\Pr(c \mid lm) = 0.05$
$\Pr(c \mid \bar{l}m) = 0.10$
$\Pr(c \mid l\bar{m}) = 0$
$\Pr(c \mid \bar{l}\bar{m}) = 0.45$
C
T
$\Pr(t \mid m) = 0.85$
$\Pr(t \mid \bar{m}) = 0$
S
$\Pr(s \mid c) = 0.90$
$\Pr(s \mid \bar{c}) = 0.05$

Figure 1: The *Lymphatic Metastases* network.

A Bayesian network defines a joint probability distribution over its variables. In its initial state, the network captures the prior distribution. As observations are entered, the network converts to another state and represents the posterior distribution given the evidence. For computing prior and posterior probabilities of interest from a Bayesian network, various algorithms are available [4; 5]. These algorithms have an exponential computational complexity in general.

3 Qualitative Networks

A *qualitative probabilistic network*, like a Bayesian network, is a model of a joint probability distribution over a set of statistical variables. It equally comprises a directed acyclic graph that represents the probabilistic influences between its variables. Instead of conditional probability distributions, however, a qualitative network associates with its digraph *qualitative influences* and *qualitative synergies* that capture qualitative features of the modelled distribution [1].

3.1 Qualitative influences

A *qualitative influence* between two variables expresses how observing a value for the one variable affects the probability distribution for the other variable. For example, a *positive qualitative influence* of a variable A on a variable B along an arc $A \rightarrow B$ expresses that observing a higher value for A makes the higher value for B more likely, regardless of any other direct influences on B, that is,

$$\Pr(b \mid ax) - \Pr(b \mid \bar{a}x) \geq 0$$

for any combination of values x for the set $\pi(B) \setminus \{A\}$ of (immediate) predecessors of B other than A. The influence is denoted $S^+(A, B)$, where the '+' is termed the *sign* of the influence. A *negative qualitative influence*, denoted by S^-, and a *zero qualitative influence*, denoted by S^0, are defined analogously, replacing $\geq$ in the above formula by $\leq$ and $=$, respectively.

For a positive, negative or zero qualitative influence of A on B, the difference $\Pr(b \mid ax) - \Pr(b \mid \bar{a}x)$ has the same sign for *all* combinations of values x for the set of variables $\pi(B) \setminus \{A\}$. These influences thus describe a *monotonic* effect of a shift in A's

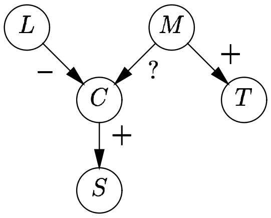


Figure 2: The *Lymphatic Metastases* network with qualitative influences.

probability distribution on the distribution for B. If the influence of A on B is positive in one state of the network and negative in another state, however, the influence is *non-monotonic.* Non-monotonic influences are associated with the sign '?', indicating that their effect is ambiguous.

Example. We consider again the *Lymphatic Metastases* network from Figure 1 and address the influences in its qualitative abstraction. From the conditional probability distributions specified for the network, the signs of the qualitative influences are readily computed. For example, from the conditional distributions specified for the variable C, we have that

$$\begin{aligned}\Pr(c \mid lm) - \Pr(c \mid \bar{l}m) &= 0.05 - 0.10 \leq 0\\ \Pr(c \mid l\bar{m}) - \Pr(c \mid \bar{l}\bar{m}) &= 0 - 0.45 \leq 0\end{aligned}$$

from which we conclude that the variable L exerts a negative qualitative influence on C: the lower a patient's primary tumour is located in the oesophagus, the less likely is the presence of metastases in the cervical lymph nodes. The influence of the variable M on C is non-monotonic since

$$\begin{aligned}\Pr(c \mid ml) - \Pr(c \mid \bar{m}l) &= 0.05 - 0 \geq 0\\ \Pr(c \mid m\bar{l}) - \Pr(c \mid \bar{m}\bar{l}) &= 0.10 - 0.45 \leq 0\end{aligned}$$

The influence is therefore associated with the ambiguous sign '?'. Figure 2 shows the digraph of the qualitative probabilistic network that is obtained from the *Lymphatic Metastases* network, with the signs of the qualitative influences indicated over the arcs. □

Note that, although in the previous example the signs of the qualitative influences between the variables have been computed from the conditional probability distributions of the corresponding numerical network, in real-life applications these signs are elicited directly from domain experts.

The set of all influences of a qualitative probabilistic network exhibits various important properties [1]. The property of *symmetry* states that, if the network includes the influence $S^{\delta}(A,B)$, then it also includes $S^{\delta}(B,A)$, $\delta \in \{+,-,0,?\}$. The *transitivity* property asserts that the qualitative influences along a trail that specifies at most one incoming arc for each variable, combine into a net influence whose sign is defined by the $\otimes$-operator from Table 1. The property of *composition* asserts that multiple influences between two variables along parallel trails combine into a net influence whose sign is defined by the $\oplus$-operator. The three properties with each other provide for establishing the sign of the net influence between any two variables in a qualitative network. We will return to this observation in Section 4.

$\otimes$	$+$	$-$	0	$?$
$+$	$+$	$-$	0	$?$
$-$	$-$	$+$	0	$?$
0	0	0	0	0
$?$	$?$	$?$	0	$?$

$\oplus$	$+$	$-$	0	$?$
$+$	$+$	$?$	$+$	$?$
$-$	$?$	$-$	$-$	$?$
0	$+$	$-$	0	$?$
$?$	$?$	$?$	$?$	$?$

Table 1: The $\otimes$- and $\oplus$-operators for combining signs.

From the definition of the $\oplus$-operator in Table 1, we observe that the composition of two influences along parallel trails may give rise to an ambiguity; the sign '?' is then used to reflect that the net influence is *unknown.* Such an ambiguity results whenever two influences with opposite signs are combined. The $\otimes$-operator cannot introduce new ambiguities.

3.2 Qualitative synergies

In addition to influences, a qualitative network includes *synergies* to model the joint influences among its variables. An *additive synergy* between three variables expresses how the values of two variables combine to yield a joint effect on the probability distribution for the third variable [1]. For example, a *positive additive synergy* of the variables A and B on their common successor C, denoted $Y^{+}(\{A,B\},C)$, expresses that the joint influence of A and B on C is greater than the sum of their separate influences, regardless of any other influences on C, that is,

$$\Pr(c \mid abx) + \Pr(c \mid \bar{a}\bar{b}x) - \Pr(c \mid a\bar{b}x) - \Pr(c \mid \bar{a}bx) \geq 0$$

for any combination of values x for the set $\pi(C) \setminus \{A,B\}$ of (immediate) predecessors of C other than A and B. A negative additive synergy, denoted Y^{-}, and a zero additive synergy, denoted Y^{0}, are defined analogously, replacing $\geq$ in the above formula by $\leq$ and $=$, respectively. An ambiguous additive synergy of the variables A and B on C is denoted by $Y^{?}(\{A,B\},C)$. A qualitative network specifies an additive synergy for each pair of variables and their common successor.

Product synergies express how observing a value for a specific variable affects the probability distribution for another variable in view of a value for a third variable [6]. A *negative product synergy* of variable A on variable B (and vice versa) given the value c for their common successor C, denoted $X^{-}(\{A,B\},c)$, expresses that, given c, observing a higher value for A renders the higher value for B less likely, that is,

$$\Pr(c \mid abx) \cdot \Pr(c \mid \bar{a}\bar{b}x) - \Pr(c \mid a\bar{b}x) \cdot \Pr(c \mid \bar{a}bx) \leq 0$$

for any combination of values x for the set of predecessors of C other than A and B. *Positive*, *zero* and *ambiguous product synergies* again are defined analogously. For each pair of variables with a common successor, a qualitative probabilistic network specifies *two* product synergies, one for each possible value of the successor.

Example. We consider again the *Lymphatic Metastases* network from Figure 1 and address the synergies in its qualitative abstraction. From the conditional distributions specified for the network, we find

$$\Pr(c \mid lm) + \Pr(c \mid \bar{l}\bar{m}) - \Pr(c \mid l\bar{m}) - \Pr(c \mid \bar{l}m) \geq 0$$

from which we have that the joint influence of L and M on C is larger than the sum of their separate influences. We conclude that L and M exhibit a positive additive synergy on C. We further have that

$$\Pr(c \mid lm) \cdot \Pr(c \mid \bar{l}\bar{m}) - \Pr(c \mid l\bar{m}) \cdot \Pr(c \mid \bar{l}m) \geq 0$$

from which we conclude that L and M exhibit a positive product synergy for the value *true* of the variable C. For the value *false* of C, we find a negative product synergy. □

The product synergies of a qualitative network serve to prescribe the signs of *induced* influences. We observe that in a network's initial state where no observations have been entered, any chain $A \rightarrow C \leftarrow B$ between A and B is blocked. If a specific value for the variable C is observed, however, the chain becomes unblocked, thereby inducing a new influence between A and B. The qualitative influence that is thus induced is termed an *intercausal influence*. The sign of this intercausal influence now equals the sign of the product synergy associated with the observed value.

4 Qualitative Probabilistic Reasoning

For reasoning with a qualitative probabilistic network, an efficient algorithm is available [2]. This algorithm provides for computing the effect of an observation that is entered into the network, upon the probability distributions for the other variables. It is based on the idea of propagating and combining signs, building upon the properties of symmetry, transitivity and composition of qualitative influences. The algorithm is summarised in pseudocode in Figure 3.

The sign-propagation algorithm traces the effect of an observation by message passing between neighbouring variables in a network. For each variable V, it determines a *node sign* 'sign[V]' that indicates the direction of the shift in its probability distribution that is occasioned by the observation; initially, all node signs equal '0'. Now, for the newly observed variable, an appropriate sign is entered into the network, that is, either a '+' for the observed value *true*, or a '−' for the value *false*. The observed variable updates its sign using the $\oplus$-operator, and subsequently sends appropriate messages to its neighbours. The sign of such a message is the $\otimes$-product of the variable's (new) sign and the sign of the influence it traverses. Any variable that thus receives a message, updates its sign with the sign-sum of its original sign and the sign of the message. If its sign has changed, the variable in turn sends appropriate messages to its (possibly induced) neighbours. This process is repeated throughout the network. Since a variable can change sign at most twice, once from '0' to '+' or '−', and then only to '?', the process visits each variable at most twice and is therefore guaranteed to halt.

```
procedure Process-Observation($Q$,$O$,$sign$):
  for all  $V_i \in V(G)$ in $Q$
  do sign[$V_i$] ←'0';
  Propagate-Sign($Q$,$\varnothing$,$O$,$sign$).

procedure Propagate-Sign($Q$,$trail$,$to$,$message$):
  sign[$to$] ← sign[$to$] $\oplus$ $message$;
  $trail$ ← $trail \cup \{to\}$;
  for each (possibly induced) neighbour $V_i$ of $to$ in $Q$
  do  $linksign$ ← sign of influence between $to$ and $V_i$;
      $message$ ← sign[$to$] $\otimes$ $linksign$;
      if $V_i \notin trail$ and sign[$V_i$] $\neq$ sign[$V_i$] $\oplus$ $message$
      then Propagate-Sign($Q$,$trail$,$V_i$,$message$).
```

Figure 3: The sign-propagation algorithm.

Example. We consider again the qualitative *Lymphatic Metastases* network from Figure 2. Suppose that, in a specific patient, a sonography of the neck has revealed enlarged lymph nodes. Further suppose that we are interested in the effect of this finding on the probability distributions for the variables L and M. Reasoning is started by entering a '+' for the variable S into the network. S updates its node sign to $0 \oplus + = +$ and subsequently sends the message $+ \otimes + = +$ to its neighbour C. Upon receiving this message, variable C updates its node sign to $0 \oplus + = +$; this node sign expresses that, given the finding from the sonography, the presence of metastases in the cervical lymph nodes has become more likely. Variable C subsequently computes the messages to be sent to its neighbours L and M. To the variable L, it sends the message $+ \otimes - = -$. L updates its sign to '−'; the observation of enlarged lymph nodes in the neck thus has rendered a primary tumour in the upper one-third of the oesophagus more likely. To M, variable C sends the message $+ \otimes ? = ?$, causing it to update its node sign to '?'. The ambiguous sign shows that the influence of the observation on the probability distribution for M is inconclusive and depends on the true, yet unknown location of the primary tumour. □

The sign-propagation algorithm reviewed above in essence serves to compute the effect of a *single* observation on the probability distributions for *all* other variables in a network. In real-life applications, however, often the simultaneous, joint effect of multiple observations on a single variable is of interest. The sign-propagation algorithm can be applied for this purpose by entering the available observations into the

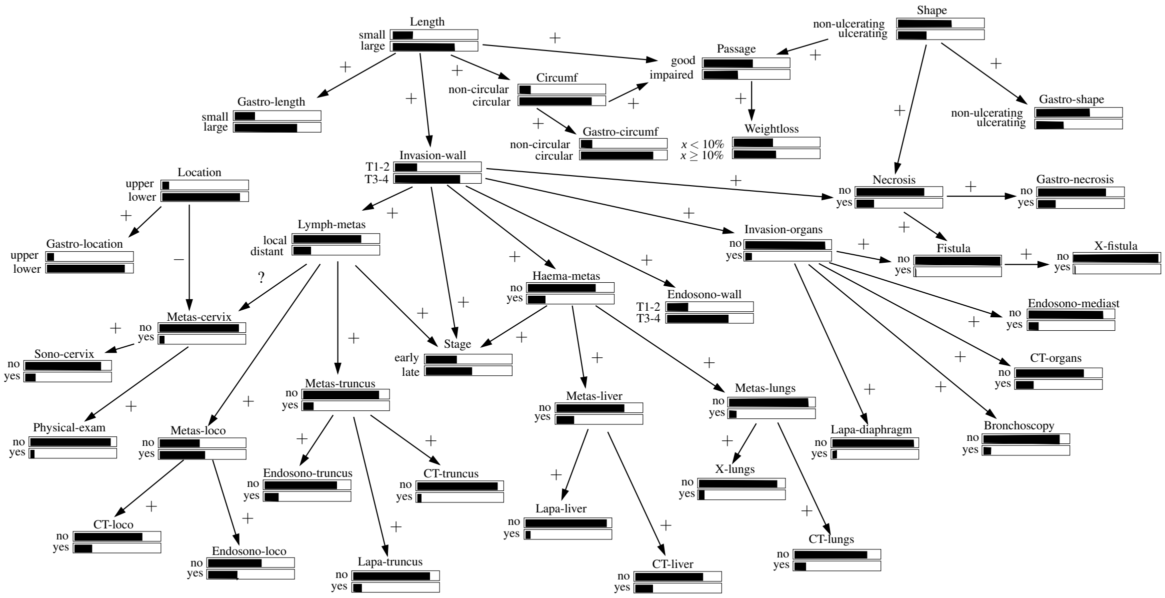


Figure 4: The qualitative *Oesophageal Cancer* network.

network simultaneously and propagating them to the variable of interest along unblocked chains [7].

5 A Real-life Qualitative Network

We exemplify qualitative probabilistic reasoning with a real-life network in the field of oesophageal cancer.

A chronic lesion of the oesophageal wall, for example associated with smoking and drinking habits, may develop into a malignant tumour. Such a tumour has various presentation characteristics that influence its prospective growth; these characteristics include the length and the macroscopic shape of the tumour. The tumour typically invades the oesophageal wall and upon further growth may invade adjacent organs. In time, the tumour may give rise to metastases in lymph nodes and to haematogenous metastases in the lungs and the liver. The depth of invasion and extent of metastasis are summarised in the cancer's *stage*. To establish these factors in a patient, various diagnostic tests are performed, ranging from a gastroscopy of the oesophagus to CT scans of the thorax and abdomen.

The state-of-the-art knowledge about oesophageal cancer has been captured in a Bayesian network [8]. From this numerical network, we have constructed the qualitative *Oesophageal Cancer* network, part of which has been used for the examples in the previous sections. In constructing the *Oesophageal Cancer* network, we have translated all variables into binary variables. We have further computed the signs for the qualitative influences and synergies from the conditional probability distributions specified for the numerical network. Figure 4 shows the digraph of the resulting qualitative network; the signs of the influences are shown over the digraph's arcs. The figure in addition shows the prior probability distributions for the included variables.

Suppose that a patient presents with oesophageal cancer. He suffers from an impaired passage of food through the oesophagus, yet shows little weight loss. A gastroscopic examination of the oesophagus reveals a large, circular tumour. A CT scan of the abdomen shows enlarged lymph nodes near the truncus coeliacus, but no evidence of secondary tumours in the liver are found. A CT scan of the thorax reveals a tumour-like mass in the lungs. The radiograph, on the other hand, does not show any evidence of metastases in the lungs. We would like to establish the effect of these findings on the probability distribution for the variable *Stage*, which is the main diagnostic variable.

Straightforward application of the sign-propagation algorithm for qualitative probabilistic reasoning as outlined in Section 4, results in numerous ambiguities throughout the network. More specifically, the effect of the various findings for the patient on the probability distribution for the variable *Stage*, is inconclusive. Close examination of the signs that are propagated over the separate chains in the network, however, serves to yield more insightful results.

The findings from the gastroscopic examination, that is, the observations for the variables *Gastro-length* and *Gastro-circumference*, with each other exert a positive influence on the variable *Stage*: the length and circumference of the primary tumour both are indicative of a later stage of the caner. The impaired passage of food also points to a later stage. The absence of weight loss, although conflicting with the finding of an impaired passage of food, does not affect the probability distribution for the cancer's stage, as it is blocked from the variable of interest. The observations for the variables *Gastro-length*, *Gastro-circumference* and *Passage* have a positive qualitative net influence also on the variable *Lymph-metas*. The finding of enlarged lymph nodes near the truncus coeliacus, that is, the observation for the variable *CT-truncus*, has a positive qualitative influence on the variable *Lymph-metas* as well, thereby confirming the hypothesis that distant lymph nodes are affected. The various influences on the variable *Haema-metas*, in contrast, are conflicting. The finding, from the CT scan, of a tumour-like mass in the patient's lungs has a positive influence on the variable *Haema-metas*, thereby suggesting the presence of haematogenous metastases. The lack of evidence of secondary tumours in the liver and the negative finding from the radiograph, on the other hand, result in a negative net influence on the variable *Haema-metas*. We conclude that the findings for the patient point to metastases in distant lymph nodes, yet are inconclusive with respect to the presence of haematogenous metastases.

By studying the signs that are propagated over the separate chains in a qualitative network, the observations that exert a joint positive influence on a variable of interest can be distinguished from the observations that exert a joint negative influence. Moreover, the variables upon which truely conflicting influences are exerted, are readily identified.

6 Recent Advances

Various researchers have studied the tendency of qualitative probabilistic networks to yield ambiguous, uninformative results upon reasoning, and have proposed extensions to the basic formalism. We briefly review some of the recent advances in the field.

Qualitative probabilistic networks capture the influences between their variables at a relatively coarse level of detail. As a consequence, influences may hide context-specific information. For example, if the influence of a variable A on a variable B is positive in one context, that is, for one specific combination of values for some other variables, and zero in all other contexts, then the influence is captured by a positive sign. This positive sign then effectively hides the zero influences. *Context-specific signs* allow for making such hidden information explicit [9]. These signs are exploited upon reasoning by propagating the most specific sign that is available for the current context. Closely linked to the coarse level of representation detail is the issue of non-monotonicity. A non-monotonic influence cannot be associated with an unambiguous sign of general validity. The influence, however, is unambiguous in every particular state of the network. *Situational signs* now capture the current sign of a non-monotonic influence given the available evidence [10]. Upon reasoning, these situational signs are used rather than the generally valid ambiguous signs. A method is available for updating situational signs as further evidence is entered and the network converts to another state.

Qualitative probabilistic reasoning in essence does not allow for resolving trade-offs between conflicting influences. The basic sign-propagation algorithm has been extended to provide for trade-off resolution by various methods that are based upon the idea of reverting to numerical probabilities whenever necessary [11]. Another method for trade-off resolution introduces a notion of strength into the basic formalism of qualitative networks, by distinguishing between strong and weak qualitative influences [12]. Resolving trade-offs then is based on the idea that strong influences dominate over conflicting weak ones. Yet another approach to handling the trade-offs in a qualitative network is taken in an algorithm that identifies the information that would serve to resolve them [13]. The algorithm builds upon the idea of zooming in on the part of the network where the actual trade-offs reside and constructing an informative result for the variable of interest in terms of the variables involved.

7 Concluding Observations

Qualitative probabilistic networks are qualitative models of uncertainty that have a firm basis in probability theory. They include an intuitively appealing graphical structure that has associated simple signs that allow for ready interpretation. Experience shows that, as a result, constructing a qualitative probabilistic network with domain experts is less demanding by far, than building a numerical network requiring thousands of probabilities. A qualitative probabilistic network, moreover, allows for efficient probabilistic reasoning in a qualitative way. Although qualitative probabilistic reasoning may not always yield conclusive results, studying the separate influences throughout a network will yield detailed insight in the domain under study. We hope to have demonstrated in this paper that qualitative probabilistic networks deserve a prominent place in real-life applications of artificial intelligence in medicine.

Acknowledgment

This research was supported by the Netherlands Organisation for Scientific Research. The paper is based to a large extent on joint work with Silja Renooij and Janneke Bolt. I am most grateful for all their help.

References

[1] M.P. Wellman (1990). Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence*, vol. 44, pp. 257 – 303.

[2] M.J. Druzdzel and M. Henrion (1993). Efficient reasoning in qualitative probabilistic networks. *Proceedings of the Eleventh National Conference on Artificial Intelligence*, AAAI Press, Menlo Park, pp. 548 – 553.

[3] G.F. Cooper (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, vol. 42, pp. 393 – 405.

[4] J. Pearl (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann Publishers, Palo Alto.

[5] S.L. Lauritzen and D.J. Spiegelhalter (1988). Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Series B*, vol. 50, pp. 157 – 224.

[6] M. Henrion and M.J. Druzdzel (1991). Qualitative propagation and scenario-based approaches to explanation in probabilistic reasoning. In: P.P. Bonissone, M. Henrion, L.N. Kanal, and J.F. Lemmer (editors). *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence*, Elsevier Science, North-Holland, pp. 17 – 32.

[7] S. Renooij, L.C. van der Gaag, and S. Parsons (2002). Propagation of multiple observations in QPNs revisited. In: F. van Harmelen (editor). *Proceedings of the 15th European Conference on Artificial Intelligence*, IOS Press, Amsterdam, pp. 665 – 669.

[8] L.C. van der Gaag, S. Renooij, C.L.M. Witteman, B.M.P. Aleman, and B.G. Taal (2002). Probabilities for a probabilistic network: a case study in oesophageal cancer. *Artificial Intelligence in Medicine*, vol. 25, pp. 123 – 148.

[9] S. Renooij, L.C. van der Gaag, and S. Parsons (2002). Context-specific sign-propagation in qualitative probabilistic networks. *Artificial Intelligence*, vol. 140, pp. 207 – 230.

[10] J.H. Bolt, L.C. van der Gaag, and S. Renooij (2003). Introducing situational influences in QPNs. In: T.D. Nielsen and N.L. Zhang (editors). *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, Lecture Notes in Artificial Intelligence, vol. 2711, Springer-Verlag, Berlin, pp. 113 – 124.

[11] C.-L. Liu and M.P. Wellman (1998). Incremental tradeoff resolution in qualitative probabilistic networks. In: G.F. Cooper and S. Moral (editors). *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, Palo Alto, pp. 338 – 345.

[12] S. Renooij and L.C. van der Gaag (1999). Enhancing QPNs for trade-off resolution. In: K.B. Laskey and H. Prade (editors). *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, Palo Alto, pp. 559 – 566.

[13] S. Renooij, L.C. van der Gaag, S.D. Parsons, and S.D. Green (2000). Pivotal pruning of trade-offs in QPNs. In: G. Boutilier, M. Goldszmidt (editors). *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, Palo Alto, pp. 515 – 522.

Papers

Two-Stage Bayesian Networks for Metabolic Network Prediction

Jung-Wook Bang[1]* Raphael Chaleil[2]* and Jon Williamson[3]*

[1]Computational Bioinformatics Group, Imperial College London, UK
[2]Structural Bioinformatics Group, Imperial College London, UK
[3]Dept. of Philosophy, King's College London, UK

Abstract

Metabolism is a set of chemical reactions, used by living organisms to process chemical compounds in order to take energy and eliminate toxic compounds, for example. Its processes are referred as metabolic pathways. Understanding metabolism is imperative to biology, toxicology and medicine, but the number and complexity of metabolic pathways makes this a difficult task. In our paper, we investigate the use of causal Bayesian networks to model the pathways of yeast *saccharomyces cerevisiae* metabolism: such a network can be used to draw predictions about the levels of metabolites and enzymes in a particular specimen. We, propose a *two-stage methodology* for causal networks, as follows. First construct a causal network from the network of metabolic pathways. The viability of this causal network depends on the validity of the causal Markov condition. If this condition fails, however, the principle of the common cause motivates the addition of a new causal arrow or a new `hidden' common cause to the network (stage 2 of the model formation process). Algorithms for adding arrows or hidden nodes have been developed separately in a number of papers, and in this paper we combine them, showing how the resulting procedure can be applied to the metabolic pathway problem. Our general approach was tested on neural cell morphology data and demonstrated noticeable improvements in both prediction and network accuracy.

1 Introduction

Functional genomics is the search for understanding of the functionality of specific genes, their relations to diseases, their associated proteins and their roles in biological processes. In functional genomics, a cell can be seen as a biochemical machine that consumes simple molecules to generate more complex ones by chaining together biochemical reactions into long sequences; its processes are referred to as *metabolic pathways*[4]. Genes play an essential role in these networks by providing the information to synthesize the enzymes that catalyze biochemical reactions. Understanding metabolism is an important problem for biology, pharmacology (in particular toxicology) and medicine but the size, complexity and uncertainty of the network of pathways has made this task difficult.

Lately, Friedman *et al.* used Bayesian nets to model gene expression data [9,10] and justified their use by their rich graphical and probabilistic representation of gene expression data and their ability to explain relations among gene variables. They reported many biologically plausible conclusions from real expression data of Spellman *et al.* by deploying heuristic search algorithms and statistical confidence measurements [17]. They proposed adopting continuous variables to capture precise local probabilities and improving the heuristic search algorithm as topics for further study. Imoto *et al.* applied a non-parametric regression model in Bayesian networks for constructing genetic networks from gene expression data [12]. They claimed to have success in microarray gene expression data and generalized method to deal with more general cases in the future.

In this paper, we demonstrate how causal networks can be used to model and predict yeast metabolism whose pathways essentially form a causal graph, one component of a causal net. Causal nets depend on the causal Markov condition as a primitive assumption and we propose a *two-stage methodology* to deal with any failure of the condition. First construct a causal net from the net of metabolic pathways; second alter that net to ensure the causal Markov condition is satisfied by adding new causal arrows or new `hidden' common causes.

In section 2, we illustrate issues in metabolic pathway in yeast described in KEGG. In section 3, we review causal Bayesian networks and present a causal Bayesian network modeling an aromatic amino acid pathway of yeast *saccharomyces cerevisiae*. In section 4 and 5, we discuss the case where the causal Markov condition fails and propose our two-stage method to deal with the problem. In section 6, we illustrate the effectiveness of adding hidden nodes in a real biological domain. In section 7 and 8, we discuss our approach and issues to be studied in the near future.

* Co-authors in alphabetical order

2 Metabolic Networks

Metabolism is a set of chemical reactions, used by living organisms to process chemical compounds in order to take energy, extract building blocks and eliminate toxic compounds. Most of these reactions would not be executed without specialized proteins called enzymes, whose function is to catalyze these chemical reactions.

Metabolism was previously seen as a combination of distinct pathways, such as glycolysis, citrate cycle, urea cycle, amino acid biosynthesis and many others. All these pathways are connected to each other. In recent years, metabolism has started to be studied in a network approach. Information about the structure of metabolic networks can now be partially extracted and represented in graphical form using KEGG, WIT and MetaCyc. For example, the data of the Kyoto Encyclopaedia of gene and Genomes (KEGG) consists of information on interacting molecular and gene pathways. Related to KEGG are the Biochemical Pathways (BP) index of Boehringer Mannheim and the Encyclopaedia of E. Coli Genes and Metabolism (EcoCyc).

Enzymes are proteins encoded by genes and these genes can be expressed at will. Therefore some subgraphs of the network or pathways can be activated or inactivated. In prokaryotic cells, a whole subgraph can be activated or inactivated at once, this is the notion of operons; and in eukaryotic cells which do not have operons, genes can be controlled individually allowing an even thinner regulation. The whole network is very dynamic, responding to the environment and the cell's needs, and some parts of the network can be activated in mutually exclusive or inclusive way. If we look at the level of a single biochemical reaction in the network, its activation depends on the presence of the reaction substrates, therefore depends on the previous step. It also depends on whether the gene coding for the enzyme is activated, and of course what activates the gene, which can be one of the substrates or some external stimuli.

3 Constructing Causal Networks

A Bayesian network is a tool for representing a probability function. It is defined over a finite domain V of variables, each of which may be discrete or continuous - for ease of exposition we will restrict attention to discrete variables which take a finite number of values. A Bayesian network consists of a DAG *G* whose nodes are the variables in *V*; a probability specification *S* which contains the probability distribution $p(V_i|Par_i)$ of each variable $V_i \in V$ conditional on its parents Par_i in *G*; and an assumption, called the *Markov condition*, which states that each variable $V_i \in V$ is probabilistically independent of its non-descendants, ND_i, conditional on its parents, written $V_i \,\_\, ND_i \mid Par_i$.

A *causally interpreted Bayesian network*, or *causal network*, is a Bayesian network in which the graph *G* represents the causal relations amongst the variables in *V*, with an arrow from V_i to V_j if V_i is a direct cause of V_j [14]. There are arguments to the effect that, if a Bayesian network is causally interpreted then the Markov condition - now called the *causal Markov condition* - is a valid assumption: *i.e.* that any variable is probabilistically independent of its non-effects, conditional on its direct causes [15, 20]. Thus in cases where causal relations are known, the graphical structure *G* in a Bayesian network can just be taken to be the causal graph. This reduces the problem of constructing a Bayesian network to that of determining the probability specification *S*, and this is normally done by taking the corresponding sample frequencies from a database of past case data, or by eliciting degrees of belief from experts.

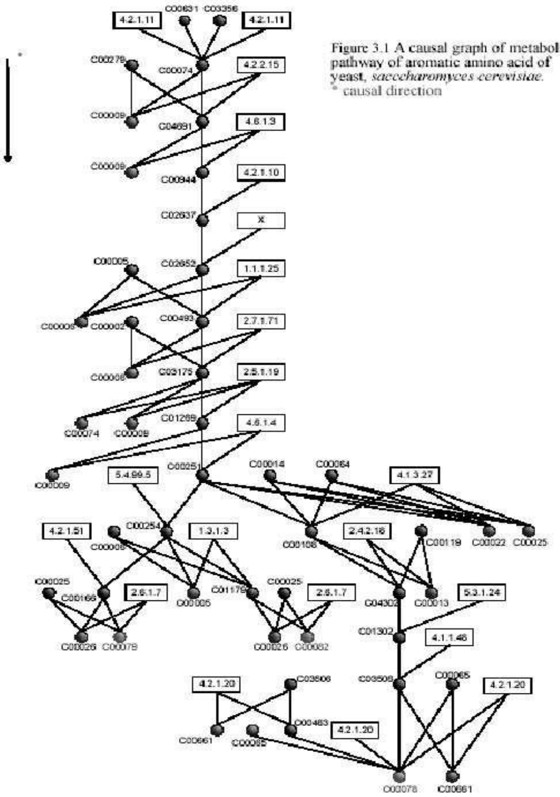


Figure 3.1 A causal graph of metabolic pathway of aromatic amino acid of yeast, *saccharomyces cerevisiae.*
* causal direction

In the case of yeast *saccharomyces cerevisiae* metabolism the network of aromatic amino acid pathways takes the form of a causal graph (Figure 3.1). The direction of causation (*i.e.* the direction of the reaction) is from the top to the bottom of the diagram. The rectangular nodes are enzymes and the circular nodes are metabolites. The red circular nodes are the aromatic amino acids - phenylalanine (C00079), tyrosine (C00082) and tryptophan (C00078) whose values we are interested in predicting. The values that the variables take are their concentrations (by mass). C00079 is produced by C00166 and C00025 under enzyme 2.6.1.7. There is a period of flux before each reaction settles down to an equilibrium during which the reaction often takes place in both directions. There is inevitably a single overall direction to the reaction however, which is determined reaction however, which is determined after equilibrium is reached. To complete the causal network all that remains is to add probability specifiers, i.e. the probability distribution of each node conditional on its direct causes. In KEGG, the metabolic pathways represent all known pathways in a given organism. However, when, in some case, enzymes could not be located, a simple deductive rule was used to uncover alternative reaction paths from an initial substrate and a final product [11]. However, Goto describes difficulties in path computation from a given list of enzymes - there still exists a number of unknown pathways for secondary metabolisms and metabolisms that are revealed under stressful conditions.

4 Two-Stage Methodology

As mentioned above, taking the graph of a Bayesian network to be the causal graph can simplify the problem of network construction. However there is a potential difficulty with this strategy: the causal Markov condition, which is required to hold if the causal network is to coincide with physical probability (frequency, propensity, chance), may in fact fail. There are a number of ways in which the causal Markov condition may fail [18]: 1. Causal information may be missing: some causal relationships amongst the variables may simply not be known; some common causes of variables in V may be omitted from V. 2. Probability specifiers may be poor estimates of physical probabilities: if specifiers were determined from a database of past case data there may be too little data to determine the required probabilities accurately, or the database may represent a biased sample from the population at large; if specifiers be elicited from experts, the experts' degrees of belief may poorly reflect physical probabilities. 3. Probabilistic dependencies which contradict the causal Markov condition may be induced by non-causal relationships amongst the variables: variables may have overlapping meaning, they may be logically or mathematically related, they may be related by non-causal physical laws or by problem constraints, or they may be subject to accidental correlations. Although the causal Markov condition may fail, it remains a good default assumption, in the following sense. If an agent's background knowledge consists just of the two components of a causal network, a causal graph and the associated probability specifiers, then the agent's personal probabilities (her degrees of belief) ought to satisfy the causal Markov condition [20, 21]. Thus the causal network is the best model available given just causal knowledge and knowledge of the conditional probability distribution of each variable conditional on its parents.

This suggests a *two-stage methodology* for employing Bayesian networks: Stage One: Construct a causal network from causal knowledge and corresponding probability specifiers. This is a good default model. Stage Two: If the network fails to perform well (this is indicative of failure of the causal Markov condition), modify the network so that it better approximates physical probability.

5 Network Modification

If a Bayesian network is to be modified to better represent physical probability, one can either change its graphical structure, or its probability specifiers or both. Changing probability specifiers to better approximate the corresponding physical probabilities is a statistical problem. We shall assume here that this problem is solvable - i.e. a mechanism is available for determining physical probabilities - and focus our attention on the graphical problem. Two routes are available for changing the graph in a Bayesian network: one can change the nodes in the graph (add, delete or combine nodes) or change the arrows in the graph (add, delete or re-orient arrows) or both. We shall present an example of each strategy: adding arrows and adding hidden nodes.

5.1 Adding Arrows

The adding-arrows approach is conceptually very simple. If one adds an arrow from V_i to V_j in a Bayesian network (and change the corresponding probability specifiers accordingly) then the new network will be no worse an approximation to physical probability than the old network, and will be a closer approximation if and only if V_i and V_j are probabilistically dependent conditional on the other direct causes of V_j[18]. So a simple strategy for changing a network to better approximate physical probability is to add arrows corresponding to conditional dependencies. If at each stage one adds the arrow corresponding to strongest conditional dependence then one achieves the closest approximation at each stage. Moreover this simple greedy algorithm finds networks that are close to the global best approximation [18,21,22].

5.2 Symmetric Hidden Node Method

The hidden node approach was originally proposed by Pearl and Verma [13]. Whenever two nodes B & C with no arrow between them are probabilistically dependent conditional on a common parent A (a violation of the causal Markov condition), then a 'hidden node' H is added as a new parent of B and C, with the arrows from A to B and C redirected through H. Then probability specifiers for H, B and C will be learned from data using the symmetric propagation algorithm called *Symmetric Hidden Node Method* (SHNM) [1]. In neural cell morphology, SHNM improved the prediction accuracy in Bayesian networks up to 42% (from 59% to 84%) [2,3]. Comparative analysis on other machine learning techniques also showed the strength of SHNM. These included neural networks and C4.5. The neural networks had one hidden layer (with up to 5 hidden nodes). The number of learning cycles was in the range 10,000 to 500,000, compared to 800 cycles for learning neural networks and Bayesian networks, respectively. The C4.5 weights were set in the range 2 to 4. It showed that the C4.5 method gave a comparable performance to the naive Bayesian network, but neural networks were considerably worse in this case. This paper also details how to systematically identify the place to add a hidden node, using a conditional dependency measure to test for violations of the Markov condition.

5.3 A Combined Approach

In this paper we advocate a combination of these two strategies for network modification. According to the principle of the common cause a probabilistic dependency which violates the causal Markov condition indicates that either a direct causal relation between the dependent nodes, or a common cause of the two nodes, is missing from the causal graph [18]. Thus to generate a causal network that satisfies the causal Markov condition we need the flexibility to add either a new arrow or a new common cause (a hidden node). The new graph can be treated as a new causal hypothesis, and can motivate closer scrutiny to verify the new posited causal connections [20].

In deciding whether to add an arrow or add a hidden node to modify a network, there are two key considerations to take into account. First the new network should be plausible when construed as a hypothesis about causal relations. Thus if it is implausible that two dependent variables are directly causally related, one ought not add an arrow between them - one ought to add a hidden node (interpreted as a common cause) to account for their dependency. Second, (Occam's razor) one ought to pursue the option that, other things being equal, increases the complexity of the network least. The complexity of a network can be measured in terms of the number of probability specifiers required in the network. In most situations adding an arrow will increase complexity least, but in cases where two or more nodes share a large number of parents, adding a hidden node can even decrease complexity.

6 Application To Yeast Metabolism

Our methodology for network modification has two objectives: Firstly, to significantly improve the prediction accuracy of the causal network. Secondly, to suggest new common causes (chemical reactions) and causal relations (reaction pathways between substrates and products) in the network. Our algorithm is as follows

1. For each pair of variables (substrates and products) *B* and *C* in the network, check their probabilistic dependence conditional on the parents *A* of *C*, via the mutual information formula

$$\mathrm{MI}_{\mathrm{c}}(\mathrm{B},\mathrm{C} \mid \mathrm{A}) = \sum_{\mathrm{B},\mathrm{C}} \left[\mathrm{P}(\mathrm{b},\mathrm{c} \mid \mathrm{a}) \log \frac{\mathrm{P}(\mathrm{b},\mathrm{c} \mid \mathrm{a})}{\mathrm{P}(\mathrm{b} \mid \mathrm{a})\mathrm{P}(\mathrm{c} \mid \mathrm{a})} \right]$$

2. If there are any such dependencies then the causal Markov condition has failed. Choose the maximal dependency and generate two new causal hypotheses: one by adding a hidden node and the other by adding an arrow from *B* to *C*.

3. The corresponding probability specifiers need to be learned. In the case of the model with added arrow these can be equated with the corresponding frequencies in the database. In the case of the model with hidden node, execute a learning process in that particular local structure [1].

4. Try to find the referent of a hidden node *H* as follows. Regenerate a data set for *H* relevant to the original data set. Using a pattern matching techniques with correlation calculation, try to locate a variable with unknown functionality that scores the highest mark.

5. Try to verify a new arrow by checking that it corresponds to probability raising of the effect by the cause, conditional on the effects other direct causes. If so, check that intervening to fix the value of the cause fixes the values of the effect, controlling for the effect's other direct causes.

6. Eliminate a hypothesis if it has no plausible causal interpretation. If both hypotheses remain, then eliminate the most complex hypothesis. Proceed to step 1.

Consider the following example. Suppose enzyme *e* catalyzes a chemical reaction with substrate m_i and product m_j where i = 3 and j = 3. Figure 6.1a shows an example of a highly connected causal graph in a single chemical reaction. We start by examining conditional dependency between the substrates and products to identify the location by deploying systematic search (step 1). If a dependency is found, we spawn new hypotheses, test their causal interpretation and eliminate one (steps 2-6). We see how adding a hidden node (Figure 6.1c) can in some cases offer a hypothesis of lower complexity than that generated by adding arrows (Figure 6.1b). A mixed approach (Figure 6.1d) is likely to result however.

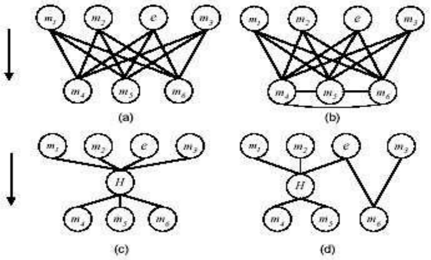


Figure 6.1 A possible network scheme between substrates, enzyme and products.
(a) Original network (b) Adding arrows (c) Adding a hidden node
(d) Adding a limited hidden node * causal direction (top to button)

7 Conclusion

In this paper we have shown how causal networks can be used to model and predict yeast metabolism. A network of metabolic pathways is essentially a causal graph. By augmenting this causal graph with probability specifiers (the probability distribution of each variable conditional on its direct causes) we construct a causal network. The viability of a causal network depends on the validity of the causal Markov condition. If this condition fails, the principle of the common cause motivates the addition of a new causal arrow or a new `hidden' common cause to the network. Algorithms for adding hidden nodes and adding arrows have been developed separately in a number of papers, and in this paper we combine them, showing how the resulting procedure can be applied to the metabolic pathway problem.

The next step in this line of research is clearly to test the resulting methodology on real yeast metabolism data. We plan to use data from Biochemistry group (Prof. Jeremy K. Nicholson) at Imperial College. Having developed one or more causal networks which model the data well, we intend to examine their plausibility as causal hypotheses. i.e. we intend to see whether new common causes and causal connections that have been posited in these models really do correspond to causes and causal connections. This will be done by collecting further observational and experimental data, to see whether each new posited cause raises the probability of its effects and whether intervening to change the value of causes changes the values of their effects.

In sum, our two-stage methodology for causal networks has a double goal: to model probabilistic relations amongst the variables involved in the metabolic pathways and to model the causal connections amongst these variables. In this paper we present the biological problem and modeling methodology; in future papers we intend to assess our proposed solution.

Acknowledgments

The authors are grateful to Stephen Mugleton, Mike Sternberg and Jeremy K. Nicholson at Imperial College. This paper was partially supported by DTI project (Metalog Project), UK.

References

[1] J-W Bang and D. Gillies. Estimating Hidden nodes in Bayesian Networks. *Proceedings of Int'l Conference on Machine Learning and Applications*, Las Vegas, USA, 2002.

[2] J-W Bang and D. Gillies. Using Bayesian Networks with Hidden Nodes to Recognize Neural cell Morphology. *In Proceedings of the Seventh Pacific Rim Int'l Conference in Artificial Intelligence*, LNAI, Springer-Verlag, Tokyo, Japan, 2002.

[3] J-W Bang, Alexandros Pappas and Duncan Gillies, "Interpretation of Hidden Node Methodology with Network Accuracy." Technical Report: ISSN 1469-4166, Dept. of Computing, Imperial College, London, UK, 2003.

[4] C. Bryant, S. Muggleton, S.G. Oliver, D.B. Kell, P. Reiser, and R.D. King. *ElectronicTransactions on Artificial Intelligence*, 5(B1):1-36, 2001.

[5] D. Chickering, D. Geiger & D. Heckerman: 'Learning Bayesian networks is NP-hard', Technical Report MSR-TR-94-17, Microsoft Research, November 1994.

[6] G.F. Cooper: 'The computational complexity of probabilistic inference using Bayesian belief networks', Artificial Intelligence 42, pages 393-405, 1990.

[7] David Corfield & Jon Williamson(eds.): 'Foundations of Bayesianism', Kluwer Applied Logic Series, Dordrecht: Kluwer Academic Publishers, 2001.

[8] A. Flook. The use of Dilation Logic on the Quantimet to Achieve Fractal Dimension Characterization of Textured and Structured Profiles. Powder Technology, 21, 195-198, 1978.

[9] N. Friedman, K. Murphy and S. Russell. Learning the structure of dynamic probabilistic networks. In Proc. Fourteenth Conference on Uncertainty in Artificial Intelligence, 1998.

[10] N. Friedman, M. Linial, I. Nachman, D. Pe'er, 'Using Bayesian networks to analyze expression data, *Journal of Computational Biology*, 7:601—620, 2000.

[11] S. Goto, H. Bono, H. Ogata, W. Fujibuchi, T. Nishioka, & K. Sato: 'Organizing and Computing Metabolic Pathway Data in Terms of Binary Relations', Pacific Symposium Biocomputing, 2, 175-186, 1996.

[12] S. Imoto, T. Goto and S. Miyano. Estimation of genetic networks and functional structures between genes by using Bayesian network and nonparametric regression', Proc. Pacific Symposium on Biocomputing. 7, 175-186, 2002.

[13] J. Pearl and T. Verma. 'Statistical Semantics for Causation, *Statistics and Computing*, 2, Chapman and Hall, 1993.

[14] J. Pearl: 'Probabilistic reasoning in intelligent systems: networks of plausible inference', San Mateo, CA: Morgan Kaufmann, 1988.

[15] J. Pearl: 'Causality: models, reasoning, and inference', Cambridge University Press, 2000.

[16] D. Sholl. Dendritic Organization in the Neurons of the Visual and Motor cortices of the Cat. *Journal of Anatomy*, 87, 387-406, 1953.

[17] Spellman, P., Sherlock, G., Zhang, M., Iyer, V., Anders, K., Eisen, M., Brown, P., Botstein, D. & Futcher, B. `Comprehensive identification of cell cycle-regulated genes of the yeast *saccharomyces cerevisiae* by microarray hybridization', *Molecular Biology of the Cell,* 9, 3273–3297, 1998.

[18] Jon Williamson: 'Foundations for Bayesian networks', in [7], pages 75-115, 2001.

[19] Jon Williamson: 'Maximizing Entropy Efficiently', Electronic Transactions in Artificial Intelligence 6, www.etaij.org, 2002.

[20] Jon Williamson: 'Learning causal relationships', Technical Report 02/02, LSE Centre for Natural and Social Sciences, www.lse.ac.uk/Depts/cpnss/proj_causality.htm, 2002.

[21] Jon Williamson: `Approximating discrete probability distributions with Bayesian networks', in Proceedings of the International Conference on Artificial Intelligence in Science and Technology, Hobart Tasmania, 16-20 December 2000, pp. 106-114.

[22] Jon Williamson: `A probabilistic approach to diagnosis', Proceedings of the Eleventh International Workshop on Principles of Diagnosis (DX-00), Morelia, Michoacen, Mexico, June 8-11 2000.

Qualitative Reasoning for modeling metabolic systems: a case–study

Riccardo Bellazzi
Dip. di Informatica e Sistemistica
Università di Pavia,
Via Ferrata 1, Pavia, Italy

Raffaella Guglielmann
Dip. di Matematica
Università di Pavia,
Via Ferrata 1, Pavia, Italy

Liliana Ironi
Istituto di Matematica Applicata
e Tecnologie Informatiche, CNR
Via Ferrata 1, Pavia, Italy

Abstract

Modeling the dynamics of a great deal of metabolic systems may be problematic due to the incompleteness of the available knowledge about the underlying mechanisms and to the lack of an adequate observational data set. Qualitative modeling techniques represent a valid approach to cope with structural knowledge incompleteness as they allow us to formulate a structural model even if a priori information is incomplete. However in many application domains, such as the biomedical context, quantitative predictions are necessary. The integration of qualitative models and input–output schemes, which learn a nonlinear input–output relation directly from experimental data, can be a valuable way to cope with the mentioned problems. More precisely, the method we propose uses the outcomes of the simulation of a qualitative structural model to build a good initialization of a fuzzy system identifier. It has been successfully exploited to derive an input-output model of the intracellular thiamine kinetics in the intestine tissue. As the structural assumptions are relaxed, we obtain a model a little bit less informative than a purely structural one, but robust enough to be used as a simulator.

1 Introduction

The prediction of the evolution over time of the patient's state plays a crucial role both in a diagnostic and therapeutic medical context. A traditional way to approach such a problem deals with both the formulation of mathematical models of the dynamics of patho-physiological systems and the simulation of their behaviour [4]. Such models, which are generally described by ordinary differential equations, are computationally tractable with classical methods which allow us to derive, either analytically or numerically, meaningful predictions of the behaviour of the considered system. But, for the medical domain as for many other physical domains, quantitative model formulation may be not successfully applicable due to the incompleteness of the available knowledge about either the functional relationships between variables or the numerical values of model parameters, which could be non identifiable both for the lack of adequate experimental settings and for the impossibility of measuring 'in vivo' the values of a few variables. Qualitative modeling methods are capable to cope with difficulties in model building in presence of incomplete knowledge [6]. But, whereas qualitative predictions of a patho-physiological system behaviour may be properly exploited in the testing phase of diagnostic reasoning [5], they are almost always inadequate to be used in a therapy planning context as the effects of different therapies are requested to be deeply investigated at a quantitative level.

As alternative to conventional mathematical modeling frameworks, the so-called input-output approaches, that are able to describe the dynamics of a real system from input-output data, have been proposed in the literature. Neural networks, multi-variate splines and fuzzy logic systems are the best known approximation schemes used for learning an input-output relation from data. Although these approaches are successfully applied to a variety of domains, they are affected by two main drawbacks that are particularly serious in medicine: first, the identification result, a nonlinear function, does not capture any structural knowledge; second, the model identification procedure usually requires a large amount of data and is often extremely inefficient. In practice, such methods fail when the experimental data set is poor either in size or in quality. Such a situation is not rare in the case of metabolic systems as they are very often characterized by an intrinsic difficulty in performing experiments and in measuring the variables of interest.

Qualitative Reasoning (QR) may effectively be integrated with classical input-output approaches to solve the problems above in a great deal of situations: patho-physiological knowledge is very often available even if insufficient to formulate a quantitative model, and it could be conveniently embedded into a fuzzy identifier. FS–QM [1] is a hybrid approach which is half way between the structural and input-output approach. It uses the outcomes of the simulation of a qualitative model to build a good initialization of a fuzzy system identifier. Such an initialization allows us to efficiently cope with both the incompleteness of knowledge and the inadequacy of the available data set, and to derive an accurate input-output model of a great deal of metabolic systems also in data poor contexts where conventional methods fail. The resulting model which embeds patho-physiological knowledge provide an interpretative key of the underlying mechanisms. The range of applicability of FS–QM to study metabolic systems is quite large as shown by its application to study a dynamic pathological system in response to exogenous perturbations, namely the blood glucose level in insulin-dependent diabetes mellitus patients in response to insulin therapy and meal ingestion [3], and to successfully identify the nonlinear dynamics of intracellular thiamine in the intestine tissue [2]. Let us observe that the classical compartmental approach to metabolic system modeling revealed to be inapplicable to model the latter system for the incompleteness of the available knowledge and for the difficulty of gathering an experimental data set rich enough to guarantee the well-posedness of the parameter estimation procedure. The poor data set has been also responsible of the failure of input-output nonlinear regression approaches.

The quantitative assessment of different patho-physiological processes is often a necessary step for an insightful interpretation of metabolic systems and for therapy planning. The integration of QR techniques with conventional modeling approaches plays a key role towards the development of robust and efficient quantitative modeling methods capable to deal with the complexity of the biomedical systems.

2 The method

We deal with the general problem of approximating discrete nonlinear systems, described by the following input-output equation:

$$y_{k+1} = f(\underline{x}_k, \underline{\theta}) + \epsilon_{k+1} \qquad (1)$$

where $\underline{x}_k \in \Re^n$ is the regressor vector at discrete time instant k and $y \in \Re$ is the measurable output, $\underline{\theta}$ is the vector of parameters, and the terms ϵ_k's account for measurements errors. The function $f(\cdot)$ is in general unknown: our goal is therefore to find a continuous function approximator of $f(\cdot)$. In our work we use a Fuzzy System (FS) to build such a function: by exploiting FS's with the *singleton fuzzifier*, the *product inference rule*, the *center average defuzzifier*, and under the assumption of *Gaussian membership functions*, we obtain an approximator which is known to possess some desirable properties for several classes of membership functions, like the capability of approximating any continuous function with an arbitrary degree of accuracy [10]. The proposed FS can be trained by using a back–propagation scheme [10]. The crucial issue in fuzzy system identification deals with the definition of a fuzzy rule base capable to describe properly the system dynamics. In order to cope with this problem, we propose a new method for FS initialization based on qualitative simulation.

Qualitative simulation (QSIM, [6]) derives qualitative descriptions of the possible behaviors of a dynamical system from a qualitative representation both of its physical structure and of an initial state $QS(t_0)$. The system is represented through a Qualitative Differential Equation (QDE). The set of qualitative behaviors generated by QSIM includes all possible behaviors of the system described by the QDE and the initial state.

The qualitative values are represented through landmark values: a landmark value is a symbolic name for a particular real number, whose value may be unknown, and defines regions where qualitative system properties hold. Fuzzy sets may be used to represent qualitative regions, too: the underlying range of a real variable can be discretized into a finite ordered set, whose elements are fuzzy sets. In this representation, the qualitative regions are expressed through membership functions which may be viewed as a measure of the suitability of applying a given qualitative description to a state variable. We can establish a one–to–one correspondence between landmark values and fuzzy sets: in such a way a real value can be represented in both frameworks. Then, on the basis of this mapping, we can translate into the fuzzy formalism the tree of behaviors, i.e. we build the fuzzy rules from the qualitative behaviors generated by QSIM: the resulting fuzzy rules can be seen as a measure of the possible transition from qualitative regions, or equivalently from states, to the next ones. As a consequence, the fuzzy rule base, which includes the rules generated from all the behaviors, captures the entire range of possible system dynamics. As far as the values of the parameters, they are initialized on the basis of the domain knowledge as well: the resulting FS provides for a good initialization of the fuzzy approximator searched for, and it can be tuned on the experimental data.

3 A case study: thiamine kinetics identification

We have applied our methodology to solve identification problems arising from metabolic systems, in particular to approximate blood glucose level dynamics in patients affected by Insulin–Dependent Diabetes Mellitus in response to exogenous insulin and to meal ingestion, and Thiamine kinetics in the intestine tissue.

In this work, we will focus on the second system. Within cells thiamine (Th), also known as vitamin B_1, participates in the carbohydrate metabolism, in the central and peripheral nerve cell function and in the myocardial function. After its absorption from plasma into the intestinal mucosa, Th is released into plasma

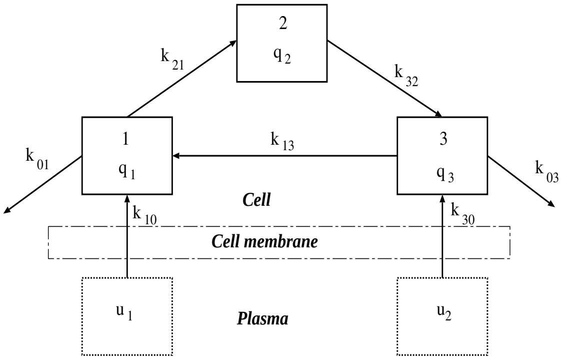


Figure 1: The compartmental structure of the system within cells. The inputs of the system are the plasma specific activity of Th, and ThMP (u_1, u_2, respectively). The state variables are the specific activity of Th, ThPP, and ThMP (q_1, q_2 and q_3, respectively). The flows between the compartments reflect the physiological chemical pathway.

for distribution to other tissues, either in its original chemical form (Th) or in a mono-phosphorilated one (ThMP); Th is then directly transformed into a higher energy compound, Thiamine Piro-Phosphate (ThPP), which is dephosphorylated into ThMP. Figure 1 shows the compartmental structure of the system. The chemical reactions are enzyme-mediated, and, therefore, their products saturate as the input quantities increase. Moreover, the transport through the cellular membrane is a nonlinear saturable process, and the underlying mechanisms are unknown.

The considered class of experimental studies is related to the quantitative assessment, around the steady state condition, of the normal and pathological conditions underlying Th chemical transformations, and cellular uptake and release. In particular, from an experimental viewpoint, we have considered animal tracer experiments, in which a small amount of labeled Th is injected into the peritoneum of a group of rats; the specific activity of labeled Th is subsequently measured in plasma and in the cells of different tissues. We have applied our methodology in both normal and pathological conditions. More precisely, we have studied Th intestine tissue metabolism in normal subjects and in subjects suffering from Diabetes, both insulin treated and untreated [2], with the final goal to quantitatively evaluate the differences in Th metabolism between the three different classes of subjects; on the basis of this evaluation it has been possible to understand whether insulin treatment is effective in re-establishing the Th metabolism to quasi-normal conditions.

To obtain a robust mathematical description of Th metabolism in the intestine tissue, we investigated a collection of different modeling approaches, which range from compartmental models to nonlinear regressions [2]. Many studies based on the compartmental modeling approach have been carried out to quantitatively assess the intracellular Thiamine kinetics in several tissues, both in normal and pathological conditions [7; 8; 9]. From the modeling viewpoint these studies exploit linear differential equations: unfortunately the linearity hypothesis revealed physiologically unacceptable to model Th metabolism in the intestine tissue. On the other hand, the lack of structural knowledge prevented from formulating a nonlinear compartmental model. Moreover, the small number of the available samples has also prevented from a successful representation of the system dynamics through classical input-output approaches. For such reasons, FS–QM seems the right tool to build a robust nonlinear model of this system.

As experimental data are available for all the three chemical forms of thiamine, the whole system can be modeled by three decoupled subsystems, each of them associated with a Th chemical form and represented by a nonlinear regression model of the type given in equation (1). We have identified independently three fuzzy systems f_1, f_2, f_3, each of them tuned on data related to normal subjects. For each subsystem a qualitative model has been formulated, and then simulated. Then the fuzzy rule base for each f_i has been derived according to the procedure described in [2].

As our final goal was to investigate the effects of diabetes on the overall system dynamics, we have applied the three identifiers as follows:

1. we assumed that the nonlinear regression functions f_1, f_2, f_3 express the mathematical model of thiamine kinetics in normal subjects.

2. We run such models with the inputs u_1 and u_2 as measured in the untreated and treated subjects.

3. We compared the obtained profiles with the actual data to obtain a better comprehension of the discrepancies in Th metabolism between different classes of subjects. From a physiological viewpoint , it is of great interest to quantitatively assess both the impact of diabetes mellitus and the effect of the insulin treatment on the thiamine intracellular metabolism in the intestine tissue. As a matter of fact, diabetes mellitus is a major disturbance of carbohydrate metabolism, where thiamine plays a crucial role.

All the results are reported in [2]: they show both the efficiency and the robustness of the method. The high performance of the method are due to the embodiment of the physical knowledge into each f_i brought in by qualitative models. Moreover FS–QM clearly outperforms other methods, i.e. the compartmental approach, and the input–output schemes, such as neural networks and fuzzy systems, built directly from data.

As far as the simulation results are concerned, we can observe that the model provides acceptable predictions of the data related to the treated subjects [2], so we can conclude that insulin normalizes the Th absorption from plasma, but seems able to partially restore the intracellular transformations. On the other hand, a complete restoration could not occur since both the dosage and the delivery of insulin are not related to physiological stimuli, and, even more important, the therapeutical treatment is supplied when the disease has already heavily altered the physiological mechanisms. On the contrary, the simulated profiles related to untreated subjects do not match actual data: this could depend on the fact that the overall model does not capture the dynamics of Th in diabetes because the underlying qualitative models do not describe the pathological states at all.

The main contribution of this work is the construction of a nonlinear model of the intracellular thiamine in the intestine tissue. Let us observe that FS–QM, although it is an input–output predictive tool, preserves the diagnostic capability offered by structural models. For instance its potential in hypothesis testing could be efficiently exploited. In the case of thiamine metabolism, several models based on different pathological assumptions are necessary to take into account the complexity of the alterations caused by diabetes. Such FS–QM models are obtainable by different QSIM models derived either by introducing structural variations in the underlying physiological QSIM model or by perturbing the normal state of the system, in accordance with the strategy proposed in [5]. The validation of the different models would allow us to derive the most plausible one, and then the most plausible causal explanation.

References

[1] R. Bellazzi, R. Guglielmann, and L. Ironi. A qualitative-fuzzy framework for nonlinear black-box system identification. In F. Zhao and K. Yip, editors, *Qualitative Reasoning - The Twelfth Int. Workshop*, pages 10–20, Cape Cod, MA, 1998. AAAI Press.

[2] R. Bellazzi, R. Guglielmann, L. Ironi, and C. Patrini. A hybrid input-output approach to model metabolic systems: An application to intracellular thiamine kinetics. *Journal of Biomedical Informatics*, 34:221–248, 2001.

[3] R. Bellazzi, L. Ironi, R. Guglielmann, and M. Stefanelli. Qualitative models and fuzzy systems: an integrated approach for learning from data. *Artificial Intelligence in Medicine*, 14:5–28, 1998.

[4] E.R. Carson, C. Cobelli, and L. Finkenstein. *The Mathematical Modeling of Metabolic and Endocrine Systems.* Wiley, New York, 1983.

[5] L. Ironi, M. Stefanelli, and G. Lanzola. Qualitative models in medical diagnosis. *Artificial Intelligence in Medicine*, 2:85–101, 1990.

[6] B. J. Kuipers. *Qualitative Reasoning: modeling and simulation with incomplete knowledge.* MIT Press, Cambridge MA, 1994.

[7] A. Nauti, C. Patrini, C. Reggiani, A. Merighi, R. Bellazzi, G. Rindi, In vivo study of the kinetics of thiamine and its phosphates in the deafferented rat cerebellum, *Brain Metabolic Disease*, 12 (1997) 145-160.

[8] G. Rindi, C. Patrini, V. Comincioli and C. Reggiani, Thiamine content and turnover rates of some rats nervous regions, using labeled Thiamine as tracer, *Brain Research*, 181 (1980) 369-380

[9] G. Rindi, C. Reggiani, C. Patrini, G. Gastaldi and U. Laforenza, Effect on ethanol on the in vivo kinetics of thiamine phosphorilation and dephosphorilation in different organs-II, *Acute effects Alcohol and Alcoholism*, 27 (1992) 505-522.

[10] L.X. Wang, *Adaptive Fuzzy Systems and Control*, (Prentice hall, Engelwood Cliffs, N.J., 1994).

Kardio and Calicot: a comparison of two cardiac arrhythmia classifiers

E.Fromont, M.-O. Cordier and **R.Quiniou**
IRISA/INRIA, Campus de Beaulieu,
35042 Rennes Cedex FRANCE
{efromont,cordier,quiniou}@irisa.fr

A.I.Hernandez
LTSI, Campus de Beaulieu,
35042 Rennes Cedex FRANCE
Alfredo.Hernandez@univ-rennes1.fr

Abstract

This paper gives a comparison of two different systems that induce cardiac arrhythmia rules by symbolic learning: Kardio and Calicot. In particular, it proposes a detailed methodology to compare them and gives some results of this comparison.

Introduction

Coronary Care Units (CCU) were introduced in the 60's in order to monitor the vital functions of patients suffering a cardiac attack and, especially, to prevent, detect and control lethal arrhythmias by therapeutic actions. Cardiac arrhythmia detection and recognition have been studied in order to assist physicians and trigger alarms when necessary. In this article, we compare two systems that focus on this subject: Kardio [1] and Calicot [2]. Both systems can induce cardiac arrhythmia identification rules by symbolic learning. The aim of this paper is to give a methodology to compare these two different systems by a specific evaluation method that handle their differences while preserving a valid, quantitative comparison. The first part sketches the architectures and principles of the two systems. The second part presents the comparison methodology and the obtained results. The last part concludes on the positive features of each system and their possible future.

1 Compared architectures

1.1 Presentation

This section does not give a detailed description of each system architecture but points out their differences and similarities as shown in Figure 1. Further details can be found in [1] for Kardio and in [2] for Calicot.

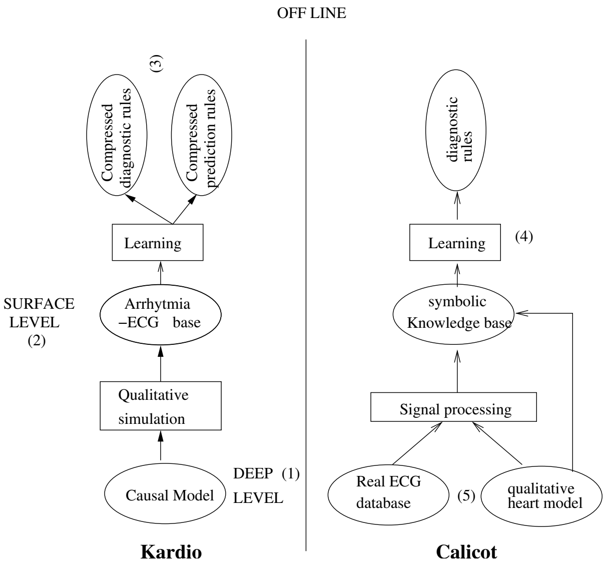


Figure 1: Architectures

The aim of Kardio is to diagnose cardiac arrhythmias from ECG descriptions. To do so, it looks for rules that describe all possible cardiac arrhythmias (single or multiple) corresponding to a given symbolic description of an ECG. Rule learning relies on a qualitative model of the heart that simulates the cardiac electrical activity: over 2,400 heart disorders can be related to over 140,000 ECG descriptions (see number 1, Fig.1). Then, using deductive and inductive inference techniques, simulations produced by the qualitative model are automatically transformed into a set of compressed prediction and diagnostic rules (number 3, Fig.1). The diagnostic rules can answer the question "which heart disorders could be the source of a given ECG feature?". They can be used for a precise diagnosis only using a diagnoser (ie an abductive method) and are difficult to compare with the expert system-like rules of Calicot. Thus, we have decided to focus on the prediction rules. They are causal rules of the form :
$P \Rightarrow (S_1 \vee S_2 \vee \ldots \vee S_n)$ where P is an arrhythmia and $S_1, \ldots S_n$ are selected ECG features of the form $S_i \equiv (S_{i_1} \wedge S_{i_2} \ldots S_{i_n})$.

These rules are used to filter out non possible diagnosis: if a given ECG does not match the ECG description, the arrhythmia P is eliminated as a possible diagnoses. Kardio has been able to induce rules for 943 heart conduction defects from 5,240 ECG descriptions.

The architecture of Calicot can be described in two steps : the first one, on which we focus, is done off-line (see Figure 1) and its aim is to build a set of high-level symbolic characterizations of cardiac arrhythmias, directly from real ECGs [2]. The learning step (number 4, Fig.1) relies on inductive logic programming (ILP) techniques. It makes use of learning examples (number 5, Fig.1) which are either real signals (like the labelled ECG signals from the MIT-BIH database [6]) or signals obtained by simulating arrhythmias on the Carmen cardiac model [5]. The second step is an on-line step which is in charge of analyzing the signal and identifying arrhythmias by matching the symbolic representation of the signal to prestored characterizations.

1.2 Analysis

In Calicot, the qualitative description of the ECG is computed from signal analysis methods. In contrast, the qualitative description of the signal in Kardio is directly given by the heart model (there is no signal processing step). In Calicot, the qualitative language is bounded by signal processing technologies because a very precise description of each heart wave is very difficult to obtain directly from a real signal. The language used in Kardio could be, thanks to the model, as rich and powerful as needed. For example, ambiguities remain in distinguishing between a Left Bundle Branch Block (LBBB) and a Right Bundle Branch Block (RBBB) on a real signal (both arrhythmias come from an intraventricular conduction disturbance but the first one comes from the left bundle branch and the second one from the right bundle branch). Kardio avoids this problem by using attribute values like *wide-lbbb* or *wide-rbbb* to describe the signal, even if this refinement level is difficult to reach by signal processing algorithms. We can then expect that the discrimination power of Kardio is better than the discrimination power of Calicot since Calicot cannot distinguish some arrhythmias.

Nevertheless, as explained in section 1.1, the final aim of Calicot is to analyze the signal on-line in order to identify arrhythmias by matching the symbolic representation of the signal to prestored patterns. Consequently, the qualitative language needs to fit what signal processing algorithms can currently achieve. Using Kardio rules for on-line analysis would meant to use signal processing algorithms able to produce on-line very detailed descriptions of the signal adapted to Kardio language. However, even if signal processing technologies are evolving very quickly, so precise descriptions are unconceivable in the next few years (especially under the noisy conditions which are often associated with CCU or ambulatory recording).

In contrast, for the same reasons, the validation of the Calicot rules can be done on real signals. Indeed, it is easy to translate the Prolog rules induced by the ILP module into chronicles [4] and, to compare the on-line diagnosis results with labels provided by experts annotations. For Kardio, the only way to validate results is to ask an expert. We can then wonder if this is reliable. Indeed, a human expert judges the rules according to his own criteria and could not be able to evaluate information coming from an unknown sensor.

Finally, the semantics of the rules of the two systems is different. To give a precise diagnosis, it is not possible to use Kardio diagnostic rules without a diagnoser. However, the authors suggest to use their prediction rules by modus tollens, and then, to filter out the possible diagnoses by identifying that some symptoms are absent in the ECG description. On the contrary, Calicot induces associative rules which link symptom patterns to disorders and can be directly used for diagnosis by modus ponens.

2 Proposed comparison methodology

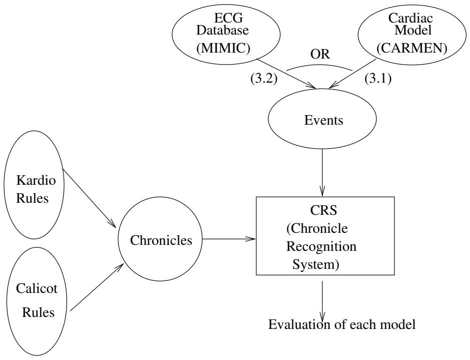


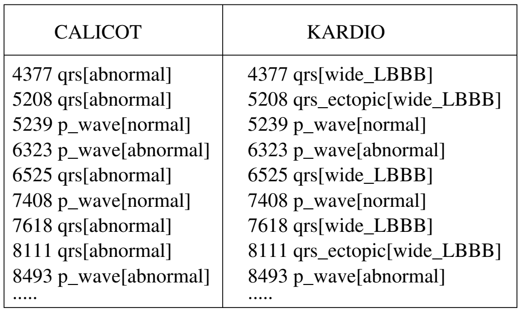
Figure 2: Comparison methodology

Figure 2 shows the principles of the methodology. To compare the two systems, we have selected a few rules for common arrhythmias from Kardio and Calicot : Sinus rhythm, LBBB and Mobitz rules. Each rule is then transformed into a CRS chronicle[4].

Chronicle recognition consists in skimming the flow of events coming from an observed process and detecting the specific events that belong to a chronicle. This process is similar to pattern-matching associated with temporal constraint satisfaction. In this article, chronicles are used to compare the detection capacity of each system from a set of events produced directly from real ECGs or from a cardiac model.

The next step is to translate Kardio prediction rules into CRS chronicles. Since we are dealing with a subset of Kardio knowledge, the closed world assumption is not valid and we cannot use the predicate completion [3] to transform the prediction rules into rules concluding positively on disorders. We have then decided to transform Kardio prediction rules by taking a weak version of the contrapositive. The resulting rule $(S_1 \vee S_2 \vee \ldots \vee S_n) \Rightarrow \diamond P$ can be interpreted as follows: "if $(S_1 \vee S_2 \vee \ldots \vee S_n)$ describes the ECG then P is a possible disorder". This implies that if an arrhythmia is recognized, it does not mean that this arrhythmia is the only possible diagnosis corresponding to a given ECG. We can just assume that a non-recognized arrhythmia is not a possible diagnosis.

The cardiac model used for the experiments is Carmen. Carmen is a macroscopic-level semi-quantitative cardiac model that is able to synthesize ECG signals and generate a physiological interpretation by means of ladder diagrams [5]. Different cardiac rhythm disorders can be simulated by manually defining an appropriate set of model parameters or by direct identification of the model parameters from real ECG signals. During a simulation, the model can also generate different symbolic representations of each synthesized ECG wave, describing its instant of occurrence, its morphology and its relation with the preceding wave(s). These symbolic representations have been constructed so as to be compatible with Kardio and Calicot description languages. Figure 3 shows a generated symbolic ECG in the Kardio and Calicot description language. These events are the input of CRS.



CALICOT	KARDIO
4377 qrs[abnormal]	4377 qrs[wide_LBBB]
5208 qrs[abnormal]	5208 qrs_ectopic[wide_LBBB]
5239 p_wave[normal]	5239 p_wave[normal]
6323 p_wave[abnormal]	6323 p_wave[abnormal]
6525 qrs[abnormal]	6525 qrs[wide_LBBB]
7408 p_wave[normal]	7408 p_wave[normal]
7618 qrs[abnormal]	7618 qrs[wide_LBBB]
8111 qrs[abnormal]	8111 qrs_ectopic[wide_LBBB]
8493 p_wave[abnormal]	8493 p_wave[abnormal]
.....	.....

Figure 3: Example of an ECG description for Calicot and Kardio

The chronicle recognition results are then used to evaluate the recognition performance of each system. To cope with the differences between the semantics of the rules of the two systems, we have decided to count as a true positive recognition (TP) an arrhythmia which is in the set of possible diagnoses and should be recognized, as false negative (FN) an arrhythmia which is not in the set whereas it should have been, and as false positive (FP) an arrhythmia which is in the set whereas it should not be. In the FP case, if the arrhythmia which should be recognized is not one of the three studied arrhythmias, it is exceptionally count as a TP if the detected arrhythmia is the *sinus rhythm*. Indeed, in Kardio rules, the *sinus rhythm* can often be combined with other disorders and in this case, it is still possible that the right arrhythmia would have also been recognized. In practice, the true negative recognitions (TN) are computed as $TN = Tot - TP + FP + FN$ where Tot is the total amount of recognitions.

The criteria used for the comparison are the *sensitivity* which gives the probability of correct classification of a given observed rhythm, and the *specificity*, that reflects the ability of the system to not propose a particular rhythm class if the observed rhythm does not belong to that class. They are respectively computed by :
$SENS = \frac{TP}{TP+FN}$, $SPEC = \frac{TN}{FP+TN}$.

3 Results

Two experimentations were achieved and the results are given in confusion matrices in Tables 1,2,3 and 4. The first experiment compares directly Kardio and Calicot and, the second one compares Calicot with a weakened Kardio. In each experiment, the matrix rows represent the detection and the columns represent the annotation given by Carmen in the first experiment and by the MIT database for the second one. The word UK is used for "Unknown" arrhythmias ie arrhythmias which are not considered in this paper (for example *rbbb* or *pvc*). The word NR is used for "non recognized" arrhythmias.

3.1 Kardio vs Calicot

In the first experiment, it was decided to leave the Kardio language unchanged and to compare it directly to Calicot. To do so, we need to use the same ECG but with two different symbolic representations to fit the two systems. Since we had no real ECG described in Kardio language, we used Carmen (see section 2) to produce different symbolic descriptions for the same ECG (see Fig. 3). The synthesis is performed in three steps:
Firstly, before starting the simulation, the cardiac model is initialized. A set of model parameters, which has been previously identified from real ECG signals and represents a given cardiac pathology (LBBB or a Mobitz rhythm), is loaded into Carmen.
Secondly, in order to generate different scenarii associated with the same cardiac disorder during the simulation, a model driver algorithm modifies randomly the following physiological model properties, every 4 seconds:

- Heart rate: from 40 to 190 beats per minute, using a uniformly distributed random variable.
- Atrio-ventricular conduction delay: a normally distributed random variable is used to define a conduction delay between 80 and 320 ms. This delay is distributed throughout the different atrio-ventricular structures of the model.
- Bundle branch conduction delay: the altered bundle branch (left or right) is chosen randomly and its conduction delay is defined with a normal distribution between 11 and 50 ms.
- Ectopic focus activation: an ectopic focus with a uniform random discharge period (defined between 1200 and 2100 ms) and a randomly chosen ventricular location is activated with probability 0.2.

Thirdly, at the end of the simulation, the internal symbolic representation of each wave generated by the model during the simulation, is translated into Calicot or Kardio language.

The rules learned by the two systems are then transformed into CRS chronicles. Examples of CRS chronicles for Kardio and Calicot are given in Figures 4 and 5. A comment is given after each event to give a brief description of the meaning of the predicate or, the Prolog rule from which the chronicle is generated. We can notice that only one chronicle is needed to recognize an LBBB with Calicot whereas the chronicle shown in Figure 5 is the first one of thirteen chronicles that describe the LBBB arrhythmia in Kardio. Indeed, Kardio language is a lot more precise than that of Calicot. For example, in Figure 5, we can see that the dominant QRS should be *wide_lbbb* and the ectopic QRS should be whether *wide_lbbb* or *wide_other* whereas in the Calicot rule shown in Figure 4, there is no difference between a dominant and an ectopic QRS, we only know that the shape of the QRS wave should be *abnormal*.

```
chronicle lbbb[]() {
occurs(0,0,p_wave[*],(start+1,R0-1))//no p_wave in [START,R0-1]
occurs(0,0,qrs[*], (start+1,R0-1))//no qrs in [START, R0-1]
event(qrs[?w0], R0)  //(qrs( R0 ,abnormal, _ ),
?w0 in {abnormal}

occurs(0,0,p_wave[*],(R0+1,P1-1))//no p_wave in [R0+1, P1-1]
occurs(0,0,qrs[*], (R0+1,P1-1))//no qrs  in [R0+1, P1-1]
event(p_wave[?w1], P1)  //p_wav( P1 ,normal, R0 ),
?w1 in {normal}
R0 < P1

occurs(0,0,p_wave[*],(P1+1, R1-1))//no p_wave in [P1+1, R1-1]
occurs(0,0,qrs[*], (P1+1, R1-1))//no qrs in [P1+1, R1-1]
event(qrs[?w2], R1)  //qrs( R1,abnormal, P1),
?w2 in {abnormal}

P1 < R1
R1 - P1 in  normalpr1 //pr1( P1 , R1 ,normal)

 end - start in nb_cycles1}
```

Figure 4: A CRS chronicle for Calicot corresponding to the LBBB arrhythmia

```
chronicle lbbb[]() {
occurs(0, 0,qrs[*], (start+1, R0-1)) //no qrs in [START, R0-1]
event(qrs[?w0], R0) //qrs(R0,_,wide_LBBB),
?w0 in {wide_LBBB}

occurs(0, 0,qrs[*], (R0+1, R01-1)) //no qrs  in [R0+1, R01-1]
event(qrs_ectopic[X], R01) //qrs_ectopic(R01,R0,X),
R0 < R01

X in {wide_LBBB,wide_other}
occurs(0, 0,qrs[*], (R01+1, R1-1)) //no qrs in [R01+1, R1-1]
event(qrs[?w3], R1) //qrs(R1,_,wide_LBBB),
?w3 in {wide_LBBB}

R0 < R1
R1 - R0 in shortrr1 //rr1(R0, R1, short)

end - start in nb_cycles1}
```

Figure 5: A CRS chronicle for one of the Kardio rule for LBBB

	mobitz	lbbb	normal	UK	Total
mobitz	114	0	0	32	146
lbbb	0	20	0	0	20
normal	14	6	2765	0	2785
NR	909	3	0	0	912
Total	1037	29	2765	32	3863
Sensit	0.11	0.69	1	0	
Specif	0.99	1	0.98	1	

Table 1: The confusion matrix for Kardio rules with Carmen signal

	mobitz	lbbb	normal	UK	Total
mobitz	30	0	0	20	50
lbbb	0	22	0	705	727
normal	1	0	1816	0	1817
NR	1328	7	0	0	1335
Total	1358	29	1816	725	3928
Sensit	0.02	0.76	1	0	
Specif	0.99	0.66	1	1	

Table 2: The confusion matrix for Calicot rules with Carmen signal

The experiment results are given in Table 1 for Kardio and Table 2 for Calicot. First, we can notice that there are a lot of non recognized *mobitz* for both systems. This comes from the arrhythmia annotations provided by Carmen. Indeed, Carmen generates events randomly. It could generate some rare event patterns labeled as a *mobitz* (for example, four consecutive p_waves). However, since those patterns are not very common in medicine, the corresponding rules have not been induced by both systems and then, these patterns are not recognized. This brings a lot of false negative for the *mobitz* class. Moreover, Calicot rules for *mobitz* are more precise than Kardio. Indeed, it specifies that the p_wave occurring in a *mobitz* should be *normal* whereas Kardio does not specify anything on the shape of the p_wave so the latter has more recognitions for *mobitz* and his sensitivity is better. Besides, we can notice that the results for *lbbb* and more particularly the number of false positive is a lot better for Kardio (0) than for Calicot (705). This comes from the fact that Calicot never makes the difference between an *rbbb* and an *lbbb* as explain in section 1.2. Finally, there are a lot more TP in the *normal* class for Kardio (2785) than for Calicot (1816). This comes from the choice we made about the detection of unknown arrhythmias as explained in section 2. Indeed, when Kardio detects a normal rhythm instead of an unknown one (for example *rbbb*) we have considered that it was a correct detection for the *normal* class because Kardio has eliminated the *mobitz* and the *lbbb*.

	mobitz	lbbb	normal	UK	Total
mobitz	427	0	0	0	427
lbbb	0	2006	8	1106	3120
normal	0	0	2292	0	2292
NR	0	0	291	0	291
Total	427	2006	2591	1106	6130
Sensit	1	1	0.88	0	
Specif	1	0.73	1	1	

Table 3: The confusion matrix for Calicot rules

	mobitz	lbbb	normal	UK	Total
mobitz	427	0	0	0	427
lbbb	0	2006	0	1432	3438
normal	0	0	7406	0	7406
NR	0	0	0	0	0
Total	427	2006	7406	1432	11271
Sensit	1	1	1	0	
Specif	1	0.85	1	1	

Table 4: The confusion matrix for weakened Kardio rules

3.2 Weakened Kardio vs Calicot

In a second step, we have weakened Kardio language to fit current signal processing algorithm possibilities. Every shape that was not described as *normal* was assumed *abnormal* and every ectopic QRS was considered as a dominant QRS. Indeed, nowadays, it is still difficult to differentiate an ectopic QRS from the dominant one or to feature precisely a wave shape just by analyzing the signal.

In this experiment, the signal comes from a real ECG and the symbolic description is the same for the two systems. An example of a CRS chronicle for the weakened Kardio is given in Figure 6. This chronicle corresponds to the same rule that was used to create the chronicle of Figure 5.

```
chronicle lbbb[]() {
occurs(0, 0,qrs[?],(start+1, R0-1)) //no qrs in [START, R0-1]
event(qrs[?w0], R0) //qrs(R0,_,abnormal),
?w0 in {abnormal}

occurs(0, 0,qrs[?],(R0+1, R1-1))  //no qrs  in [R0+1, R1-1]
event(qrs[?w1], R1) //qrs(R1,_,abnormal),
?w1 in {abnormal}

occurs(0, 0,qrs[?],(R1+1, R2-1)) //no qrs in [R1+1, R2-1]
event(qrs[?w2], R2) //qrs(R2,_,abnormal),
?w2 in {abnormal}

R0 < R2
R2 - R0 in shortrr1 //rr1(R0,R2,short)

end - start in nb_cycles2 }
```

Figure 6: A CRS chronicle for one of the weakened Kardio rule for LBBB

Results are given in Tables 3 and 4. We can notice that the results are good and quite the same for Kardio and Calicot except that Kardio has twice as much detections than Calicot. Indeed, the choices made to classify Kardio detections are all in Kardio advantage because if there is a multiple detection in Kardio, it is counted as a true positive for *normal* if the right solution is in the set of detected arrhythmias. In particular, for each *normal* rhythm, the weakened Kardio has detected the *normal* rhythm and a lot of other arrhythmias (*lbbb* or *mobitz*). We can also notice that there is a lot of FP for *lbbb* for both systems. This comes from the fact that Kardio can not distinguish between a *rbbb* and a *lbbb* with its new weakened language. These results are similar to those of Calicot presented in Table 2 for *lbbb*.

Conclusion

This paper has given a comparison of two cardiac arrhythmia classifiers with very different nature. It has proven to be very difficult due to the different semantics attributed to the rules of the two systems. We also have presented a qualitative evaluation methodology of Kardio which has never been done before. The comparison has shown that Kardio with its powerful language is more precise than Calicot whereas the latter is more adapted to current signal processing technologies and so, to an on-line application. As the signal processing techniques evolve, we plan to improve the knowledge base for the ILP module of Calicot to make Calicot language more powerful. An another interesting experiment will then be compare the new rules induced by Calicot with the rules of Kardio.

References

[1] I. Bratko, I. Mozetič, and N. Lavrač. *Kardio: A Study in Deep and Qualitative Knowledge for Expert Systems.* MIT Press, Cambridge, MA, 1989.

[2] G. Carrault, MO. Cordier, R. Quiniou, and F.Wang. Temporal abstraction and inductive logic programming for arrhythmia recognition from ecg. *Artificial Intelligence in Medicine*, 2003. In press.

[3] K. Clark. Negation as failure. In H. Gallaire and J. Minker, editors, *Logic and Data Bases.* Plenum Press, New York, NY, 1978.

[4] C. Dousson, P. Gaborit, and M. Ghallab. Situation recognition: representation and algorithms. In *Proceedings of IJCAI-93*, pages 166–172, Chambéry, France, 1993. Morgan Kaufman.

[5] A. I. Hernández, G. Carrault, F. Mora, and A. Bardou. Model-based interpretation of cardiac beats by evolutionary algorithms: signal and model interaction. *Artificial Intelligence in Medicine*, 23,3(211-235), 2002.

[6] G. B. Moody. Ecg database programmers guide. Harvard-MIT Division of Health Sciences and Technology Biomedical Engineering Center, Ninth Edition, 1997. http://www.physionet.org/physiobank/database/mitdb/.

Challenging QR research issues in Electrocardiology

Liliana Ironi, Stefania Tentoni
IMATI - CNR
via Ferrata 1,
27100 Pavia, Italy
E-mail: {ironi,tentoni}imati.cnr.it

Electrocardiology is a stimulating and promising application domain for Qualitative and Model-based Reasoning research. Much of the research effort has been so far aimed at the automated interpretation of electrocardiograms (ECG's) [4; 14; 11; 13], for which the interpretative rationale is well established. However, a major drawback of ECG's, which record electrical signals from nine sites only on the body surface, is their poor spatial covering. Some arrythmias, but also conduction anomalies caused by infarcts and ischemia, can be missed by ECG's.

Thanks to the latest advances in technology, a far more informative technique, namely body surface mapping (BSM), is becoming more widely available. In BSM, the electrical potential is measured over the entire chest surface, and more spatial information about the heart electrical activity is available, though the ability to relate visual features to the underlying complex phenomena still belongs to very few experts [12]. Besides body surface maps, also epicardial or endocardial maps are becoming available, either measured by means of endocavitary probes or obtained non invasively from body surface data through mathematical inverse procedures. These kinds of maps are the most precise in locating the anomalous conduction sites [5].

While a rationale for the interpretation of electrocardiographic maps is progressively being defined, the goal of bringing such techniques to the clinical practice gets closer. In this context, an important role can be played by QR methodologies for spatial/temporal reasoning, that could 1) support the expert electrocardiologist in identifying salient features in the maps and help him in the definition of an interpretative rationale, and 2) achieve the long-term goal of completely automating map interpretation to be used in a clinical context. The delivery of such a tool would be of great impact on health care as cardiac map interpretation is essential for the localization of the ectopic sites associated with ventricular arrythmia.

At the current status of research, map interpretation mainly deals with activation isochrone maps and sequences of isopotential maps. The former ones deliver a lot of information about the wavefront structure and propagation: each contour line aggregates all and none but the points that share the same excitation state, and subsequent increasing isocurves are nothing but subsequent snapshots of the travelling wavefront, as it is illustrated by Figure 1; whereas the latter ones give temporal information about electrical phenomena, such as current inflows and outflows.

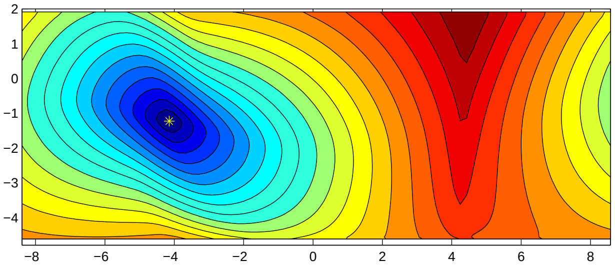


Figure 1: Example of activation isochrone map (cylindrical projection of contour lines of the activation time generated on a model ventricular surface).

The interpretation of activation isochrone maps currently grounds on features such as contour shapes, the directions along which the front velocity is higher, minimum and maximum regions. The contour shapes and highest velocity directions reflect the conduction anisotropy, and the fiber architecture. Minimum and maximum correspond to the points where the front emerges and extinguishes, respectively. Such features can be correlated to the expected activation sequence, and used in a diagnostic context: deviations from the expected patterns would highlight ectopic sites associated with ventricular arrythmias, and regions with altered conductivity.

Conventional contouring tools would allow us to efficiently visualize patterns of electrical potential distribution but they do not facilitate the automated extraction of general rules to infer the correlation between pathopysiological patterns and the wavefront structure and propagation.

Within the QR research framework, the Spatial Aggregation (SA) approach [3; 6], which is aimed at the interpretation of numeric input fields that are spatially represented, is potentially capable to capture structural information about the underlying physical phenomenon, and to identify its global patterns and the causal relations between events. These characteristics are due to its hierarchical strategy in aggregating and abstracting geometrical objects at higher and higher levels, capturing spatial adjacencies at multiple scales. Figure 2 illustrates the basic steps performed within the SA abstraction processes.

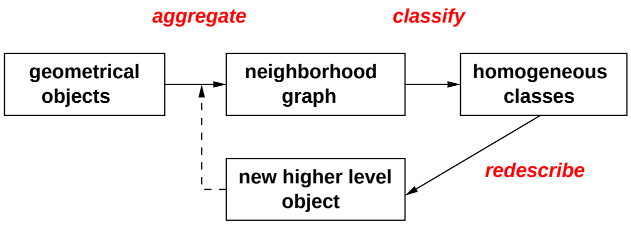


Figure 2: Basic steps performed within the SA abstraction processes.

The SA approach has been successfully applied to different domains and tasks, [10; 15; 2; 1; 16], among those we mention the weather data analysis problem [7] as it presents some similarities with the cardiac map interpretation: global patterns and structures are identified by looking at the isocurves and reasoning about their spatial relations. However, in the case of contouring, which is often the first level for spatial reasoning, the currently available SA methods have some major drawbacks that limit their use in many applications. A first limitation deals with the way topological contiguity of isopoints is taken into account: the current methods use a kind of neighborhood graph that fails to adequately represent such property.

Secondly, the strong adjacency relation, upon which isocurve abstraction from the set of given data points is performed, is based on a metric criterion [7]. When dealing with complex domain geometry and/or with non uniform data meshes, as it is often in application contexts, the original version of SA may unsoundly perform contouring: isocurve entanglement and/or segmentation phenomena may occur due to metric-based adjacency relations and a scarse density of isopoints. Figure 3 illustrates such problems as they arise in abstracting an isochrone map from simulated activation time data relative to a horizontal section of a model heart ventricle.

As robust contouring is a must in view of the identification and extraction of salient features and structural information about the underlying electrical phenomenon, a thorough revision and extension of SA is necessary to deal with the 3D geometries of heart and chest, where data are given on the nodes of non uniform meshes: either planar 2D elements when transversal or longitudinal sections of the myocardium are considered, or 3D surface elements when the epicardial surface is explored.

Some work has been done to improve the robustness of SA as to the costruction of isocurves from a non uniform numeric input field [8; 9] by providing new algorithms and definitions for soundly building neighborood relations upon 2D complex domain geometry. This is a meaningful result from our application point of view as 3D processes are suitably studied also by transversal/longitudinal sections of the 3D domain.

Future work will deal with the identification and definition of spatial relations between isocurves, as well as between possible higher-level aggregated objects. Such relations should reveal adjacencies, correspondences, and other kinds of interactions between the spatial objects, and, then, underlie the feature extraction process. To this end, a crucial issue deals with the specification of the neighborhood graphs that properly represent the spatial relations between geometrical objects at different level of abstraction.

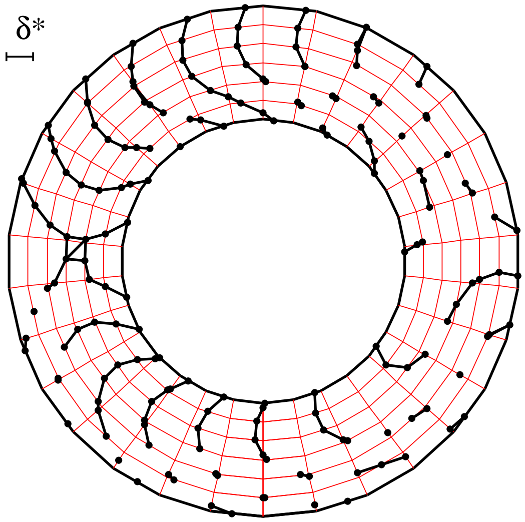


Figure 3: Artifactual segmentation and/or entanglement of isocurves.

References

[1] C. Bailey-Kellog and F. Zhao. Influence-based model decomposition for reasoning about spatially distributed physical systems. *Artificial Intelligence*, 130 (2): pp. 125–166, 2001.

[2] C. Bailey-Kellogg and F. Zhao. Spatial aggregation: Modelling and controlling physical fields. In L. Ironi, editor, *QR97*, pp. 13–22. IAN-CNR, 1997.

[3] C. Bailey-Kellogg, F. Zhao, and K. Yip. Spatial aggregation: Language and applications. In *Proc. AAAI-96*, pp. 517–522, Los Altos, 1996. Morgan Kaufmann.

[4] I. Bratko, I. Mozetic, and N. Lavrac. *Kardio: A Study in Deep and Qualitative Knowledge for Expert Systems.* MIT Press, Cambridge, MA, 1989.

[5] P. Colli Franzone, L. Guerri, and M. Pennacchio. Spreading of excitation in 3-d models of the anisotropic cardiac tissue. ii. effect of geometry and fiber architecture of the ventricular wall. *Mathematical Biosciences*, 147: pp. 131–171, 1998.

[6] X. Huang and F. Zhao. Relation-based aggregation: finding objects in large spatial datasets. *Intelligent Data Analysis*, 4: pp. 129–147, 2000.

[7] X. Huang. *Automatic analysis of spatial data sets using visual reasoning techniques with an application to weather data analysis.* PhD thesis, The Ohio State University, 2000.

[8] L. Ironi and S. Tentoni, Towards automated electrocardiac map interpretation: an intelligent contouring tool based on Spatial Aggregation, in *Lecture Notes in Computer Science* 2810, pp. 397–408, Proc. of 5th International Symposium on Intelligent Data Analysis, Berlino, 28-30 Agosto, 2003. Springer.

[9] L. Ironi and S. Tentoni, On the problem of adjacency relations in the Spatial Aggregation approach, in *Proc. 17th International Workshop on Qualitative Reasoning*, Brasilia, 20-22 August 2003, pp. 111–118.

[10] L. Joskowicz and E. Sacks. Computational kinematics. *Artificial Intelligence*, (51): pp. 381–416, 1991.

[11] M. Kundu, M. Nasipuri, and D.K. Basu. A knowledge based approach to ecg interpretation using fuzzy logic. *IEEE Trans. Systems, Man, and Cybernetics*, 28 (2): pp. 237–243, 1998.

[12] B. Taccardi, B.B. Punske, R.L. Lux, R.S. MacLeod, P.R. Ershler, T.J. Dustman, and Y. Vyhmeister. Useful lessons from body surface mapping. *Journal of Cardiovascular Electrophysiology*, 9 (7): pp. 773–786, 1998.

[13] R.L. Watrous. A patient-adaptive neural network ecg patient monitoring algorithm. *Computers in Cardiology*, pp. 229–232, 1995.

[14] F. Weng, R. Quiniou, G. Carrault, and M.-O. Cordier. Learning structural knowledge from the ecg. In *ISMDA-2001*, volume 2199, pp. 288–294, Berlin, 2001. Springer.

[15] K. Yip. Structural inferences from massive datasets. In L. Ironi, editor, *QR97*, pp. 215–220. IAN-CNR, 1997.

[16] F. Zhao. Intelligent simulation in designing complex dynamical control systems. In Zfafestas and Verbruggen, editors, *Artificial Intelligence in Industrial Decision Making, Control, and Automation*. Kluwer, 1995.

Implementing Clinical Guidelines in an Organizational Setup

Anand Kumar[a], Barry Smith[b], Mario Stefanelli[a], Silvana Quaglini[a], Matteo Piazza[a]

[a]Laboratory of Medical Medical Informatics, Department of Computer Science, University of Pavia, Italy
[b]Institute for Formal Ontology and Medical Information Science, Faculty of Medicine, University of Leipzig, Germany

Outcomes research in healthcare has been a topic much addressed in recent years. Efforts in this direction have been supplemented by work in the areas of guidelines for clinical practice and computer-interpretable workflow and careflow models.

Clinical Practice Guidelines are defined by the Institute of Medicine as "systematically developed statements to assist practitioner and patient decisions about appropriate healthcare for specific clinical circumstances." In what follows we present the outlines of a framework for understanding the relations between organizations, guidelines, individual patients and patient-related functions. The derived framework provides a means to extract the knowledge contained in the guideline text at different granularities that can help us to assign tasks within the healthcare organization and to assess clinical performance in realizing the guideline. It does this in a way that preserves the flexibility of the organization in the adoption of the guidelines.

TEAM-ENABLED WORKFLOW

Workflow systems typically employ a tripartite categorization of cases, work-items and resources. A *case* is a specific situation in which the workflow system is applied; a *work-item* is a task to be performed in relation to this case; *resources* are the persons and facilities needed to execute given work-items, their ability to do this being represented as a *role* [1,2].

Classically, work-items have been assigned by available applications to specific workers. An application may in some way recognize that teams exist, but it is pre-selected individual members of teams who are called upon at specific times for the execution of specific work-items. In reality, however, teams within healthcare organizations have collective functions, they have certain degrees of freedom in delegating tasks within the team on a progressive basis and they may be collectively responsible for the execution of these tasks in the management of patients. To put it simply: doctors, nurses, technicians, and assistants work in tandem, and current workflow models do not do justice to this fact. We have used this team concept for a framework for the implementation of the task recommendations in clinical guidelines [3].

PARTITIONS AND APPROXIMATION

Different partitions are needed to throw light on different aspects of the organization and of the workflow process at different levels of granularity. At the same time a framework is needed within which these different partitions can be manipulated simultaneouly, for which we employ the Theory of Granular Partitions (TGP) [4,5]. A partition, from the perspective of TGP, consists of a network of cells and subcells, the latter being nested within the former; the cells, in turn, are projected onto objects. The hierarchy of available human resources, the functions they perform as well as the physical facilities at the disposal of the organization – all of these determine partitions which we need for the complete representation of team-based workflow. When a particular human resource, for example nurse A, is entitled to carry out a particular function F in a particular location within the physical structure of a hospital, then this means that the cell labelled nurse A in the partition of responsibilities is projected onto function F. When we assess how A exercises this function, then we have a new partition where the cell labelled nurse A is projected onto processes. Functions do not have temporal annotations, while the processes which realize functions, exist in specific intervals of time. In this paper, we consider the question of implementation only at the level of the former.

Compare the way we use the reference partition defined by the borders of the fifty states in the USA, in describing, say, an area of high pressure in a weather forecast. The boundaries of the separate states are well defined and we can use this fact to specify the location of the area of higher pressure even though we do not know exactly where it is; for example, by asserting that it overlaps with Texas and Arizona but not with any other state.

The same idea can be used in giving an account of the way the responsibilities are assigned to the members of a team within a healthcare organization. Here, the *ex ante* boundaries of the functions associated with any given member of the team are vague – that is one of the characteristics of genuine teamwork. The *ex ante* boundaries of the actual healthcare processes which will become associated with the functions mentioned in a clinical practice guideline are also vague. But the complete list of responsibilities in the organization, or the complete list of the functions determined by the guideline text, are crisp and so can be used as reference partitions. The boundaries of the units of the physical structure of the hospital or other places of healthcare delivery are equally crisp, and so is the hierarchy of the human resources in the organization. These can be used to specify the functions of the team of human resources in the organization in a formal way as follows.

Let x be the responsibilities of nurse A within her team, which are projected by partition P onto the collective functions to be exercised by the team as a whole. P does not project crisply onto any single function. Rather it is projected onto functions with full overlap (*fo*) to some cells and of partial overlap (*po*) or no overlap (*no*) to others. Some functions are indeed meant to be performed by nurse A only, which implies for those functions a case of full overlap. Some are collective functions executed by nurse A and/or other team members.

IMPLEMENTATION FORMALISM

The above-mentioned approach can now be applied to the creation, implementation and application of computer-interpretable guideline models. This requires not only that one is able to interpret the guideline text and models based thereon, but also that one has to find a way to understand the actual workflow within a given organization implementing the guideline.

The functions are divided into recommended and actualized functions; into homogenized and dehomogenized functions and into functions before and after approximation.

Recommended vs. Actualized Functions. The former are the idealized functions, mentioned in the guidelines, providing the best-practice approach in a specific clinical situation. The latter are the functions actually implemented within a particular organization.

Homogenized vs. Dehomogenized Functions.

Homogenized functions provide a complete list of functions mentioned in either the recommended or actualized functions, while dehomogenized functions are the functions which have a built-in ontology, which reflects a hierarchical organization among the functions. Dehomogenization could occur in the idealized condition depicted in guidelines in the form of task-hierarchy, that is, by identification of tasks and their subtasks; or it could occur in the actualized functions present within a organization by making a task-hierarchy based on the existent hierarchy of human resources in the organization.

Unapproximated vs. Approximated Functions.

Approximation describes the overlaps which can occur between different functions, it could be partial, full or no overlap; again either between the idealized functions in the guidelines or between the functions which can be carried out by the human resources in the organization. The functions which are present before the approximation are unapproximated functions, while the functions existing after the approximation are approximated functions.

Unapproximated (U) Recommended (R)

The partition of recommended homogenized functions f(H,R,U) contains the complete list of functions which are related to the management of a given pathology mentioned in the guideline without any relationship between the functions, which are presented as a list. While the dehomogenized functions record which functions are a part of which other functions. If function C is a part of function D, then the implementation of function C will be within the context of function D, thus, determining the context within which a particular function is to be executed.

Unapproximated (U) Actualized (Ac)

The partition of actualized homogenized function f(H,Ac,U) of a healthcare organization is the list of all the functions which can be carried out within the organization, irrespective of the agent executing the function.

The partition of dehomogenized functions f(D,Ac,U) describes the different functions in terms of the agent which will carry it, for example, echocardiography is a function of a cardiologist; while blood pressure determination can be done by a cardiologist or a staff nurse or a general practitioner. But this partition does not do justice to the existence of team-work, where the allocation of functions can be joint or vague.

Approximated(A) Recommended (R)

Within the partition of homogenized functions f(H,R,A), the boundaries of the functions recommended for a pathology do not change after applying the concept of approximation. This is in line with the assumption that the knowledge mentioned in the guideline is considered to be the standard for further steps. However, the functions present can stand to each other in complete, partial or no overlap relations. If there is no overlap within the functions, then their representations are not modified at this stage. The functions with complete overlap are considered as the same function. Functions with partial overlap are depicted with a connection.

Within the partition of homogenized functions f(D,R,A), the connection is strengthened if the overlapping functions are present within the same context. The context and task hierarchy are related, because if a subfunction is a part-of another funtion, then the subfunction is carried out within the context of the first.

Approximated (A) Actualized (Ac)

Within the partition of homogenized functions f(H,Ac,A), the functions with no overlap are designated to be carried by single team members, while the functions with partial overlap are carried out by team members jointly.

Within the partition of homogenized functions f(D,Ac,A), the functions carried out in the organization can involve different human agents jointly in a team or the allocation of the functions to the members of such a team is vague, that is, the overlap between the functions is projected onto the team in the organization.

Using these definitions, we can designate the functional (F) level at which the organization stands in implementing a guideline or further steps it might take towards a more complete implementation of the guideline.

PARTITIONS CONSIDERED

The WHO Hypertension Guideline deals with the different tasks performed in diagnosis, classification, investigation and management of hypertensive patients [6]. We take the examples of the tasks mentioned in this guideline to demonstrate the working of the framework sketched above.

Considering all the many definitions within the context of a guideline such as that of hypertension, we can deal here only with some sample partitions, which will help to create an overall picture. The different levels are created within this setup by connecting the different partitions and using approximation.

A) **Partition of the Physical Structure of a Hospital**: This has cells corresponding to the Departments of Internal Medicine, of Surgery, of Cardiology and so on; the Department of Internal Medicine will itself have parts: Inpatient Unit, Procedure Room, Outpatient Unit, and so on.

B) **Partition of Healthcare Professionals**: This has cells corresponding to: Physicians, Nurses, Technicians and so on; the cell Physicians has subcells: Physician internists, cardiologists etc.; the cell Nurses has subcells: nursing staff internal medicine, specialized nurse emergency, junior nurse staff etc.

C) **Partition of Recommended Functions:** This gives a complete list of functions mentioned in guidelines, for example, “Measurement of Blood Pressure”, “Advice: Change Dietary Intake” etc.

EXAMPLES OF THE IMPLEMENTATION FORMALISM

F1. The guideline text has been used to manually extract the recommended functions as two distinct partitions, in two steps – f(H,R,U) and f(D,R,U). Fulfilment on this level with respect to a particular guideline would mean that all the functions mentioned in the guideline are presented as a list in the first partition and then a hierarchy is established among them as by identifying the tasks and their subtasks from the list.

Example of f(H,R,U) – The WHO guideline mentions the tasks which should be performed as a part of therapeutic or prevention procedures for management of hypertensive patients. At the level of f(H,R,U), this consists of a list of functions without a hierarchy: (Therapeutic or Preventive Procedures; Advice: Lifestyle Changes; Advice: Change Dietary Intake; Advice: Change Fiber Intake; Advice: Change Fat Intake; Advice: Change Potassium Intake; Advice: Increased Magnesium Intake; Advice: Salt Intake Reduction; Advice: Encourage Intake of Calcium)

Example of f(D,R,U) – This consists of a hierarchical representation based on the parthood relationships. This level points out which potential tasks are carried out within which context, for example, the advice of changing the fiber intake is carried out within the context of advice regarding changing dietary intake, which in turn is carried out within the context of advice regarding changing one’s lifestyle. (Advice: Change Fiber Intake; Advice: Change Fat Intake; Advice: Change Potassium Intake; Advice: Increased Magnesium Intake; Advice: Salt Intake Reduction; Advice: Encourage Intake of Calcium) *is part-of* (Advice: Change Dietary Intake) *is part-of* (Advice:

Lifestyle Changes) *is part-of* (Therapeutic or Preventive Procedures)

F2. The functions carried out in a healthcare organization are modelled as two distinct partitions, in two steps – f(H,Ac,U) and f(D,Ac,U). Fulfilment of this level would mean that the functions performed by the organization as a whole or by a particular department are mentioned as a list, and then the different tasks are assigned to the different human agents in the organization. At this level, the concept of approximation has not been applied and thus the existence of teams has not been acknowledged.

Example of f(H,Ac,U) – (Measurement of Blood Pressure; *Echocardiography*; *Risk assessment of Hypertensive Patients*; Advice: Change Fiber Intake; Advice: Change Fat Intake; Advice: Change Potassium Intake; Advice: Increased Magnesium Intake; Advice: Salt Intake Reduction; Advice: Encourage Intake of Calcium)

Example of (D,Ac,U) for an Internist – (Measurement of Blood Pressue; Advice: Change Fiber Intake; Advice: Change Fat Intake; Advice: Change Potassium Intake; Advice: Increased Magnesium Intake; Advice: Salt Intake Reduction; Advice: Encourage Intake of Calcium)

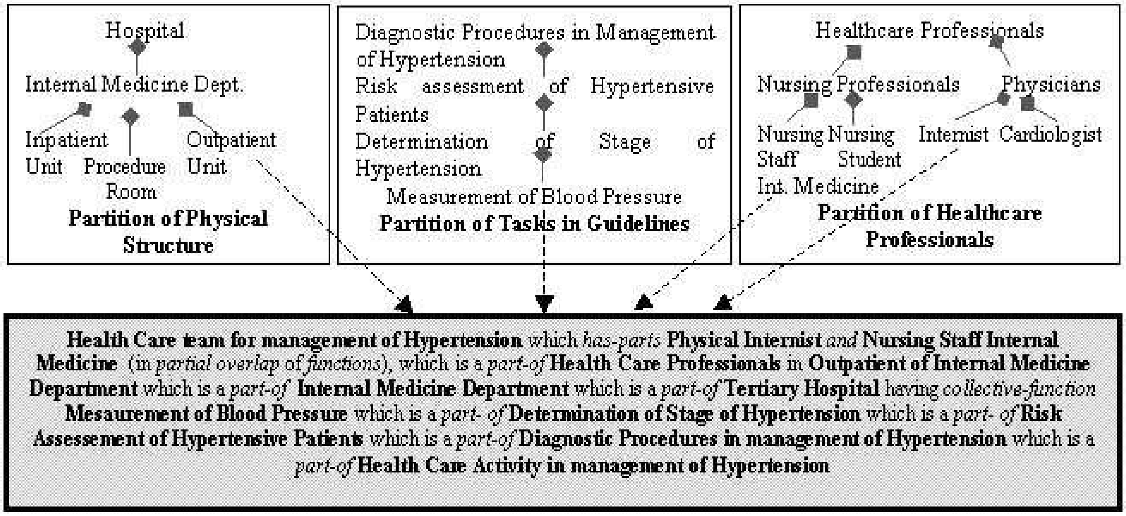
Example of f(H,Ac,U) for a junior nurse staff – (*Measurement of Blood Pressure*)

F3. Approximated functions are created. Fulfilment on this level would mean that the vagueness between the recommended functions has been taken into account in the guideline text based functions; and the team-based functions have been accounted for in the organizational set-up. For example, DASH or Dietary Approaches to Stop Hypertension, as promoted by the National Institute of Health [7]. The WHO guideline text employs the functions mentioned in DASH but does not mention DASH explicitly. However, one DASH diet does *not* include all the recommendations mentioned under "Advice: Change Dietary Intake" in the WHO guideline. For example, the DASH recommendation of "Have a 1/2 cup serving of lowfat frozen yogurt instead of a 11/2-ounce milk chocolate bar. You'll save about 110 calories" does not include a recommendation to decrease sodium intake though it has overlaps with reduction in cholesterol. On the other hand, implementation of *all* the DASH recommendations will include *all* the functions in the hypertension guideline regarding Advice to Change Dietary Intake. Thus, if we consider all the DASH recomendations together, the boundary of collective DASH-related functions is within the boundary of "Advice: Lifestyle Changes" and fully overlaps with "Advice: Change Dietary Intake". However, individual DASH-related functions are in partial overlap with the individual functions mentioned under "Advice: Change Dietary Intake".

F4. f(H,R,A) is compared with f(H,Ac,A). In order for all the guideline-recommended functions to be realizable within the healthcare organizational setup, all the functions in f(H,R,A) must be a part of f(H,Ac,A); that is, the boundary of the former must be within the boundary of the latter.

F5. f(D,R,A) is compared to f(D,Ac,A). If F4 is satisfied and the functions within the same context in the guideline are carried out by the team as partially overlapping functions, then the team functions *with interaction* and is *compliant* with the guideline at the functional level. If F4 is satisfied and the functions within the same context in the guideline are carried out by the team members as fully overlapping or non-overlapping functions, then the team functions *without interaction* but is *compliant* with the guideline at the functional level. For example, the physician P with dietician D and nurse A could be involved in implementing the function of "Advice to Change Dietary Intake". Thus, the referral to D by P, the giving of advice to A by D and the supervision of A by P within the context of this function describes an interactive team design which preserves the context mentioned in the guideline, and is thus compliant with it. That is, if 'Measurement of blood pressure' is carried out as a part-of the task 'Risk assessment of hypertensive patients', then, the parthood relationship also implies that the former is carried out within the context of carrying out the latter. If all the functions mentioned within a particular context are perfomed by the same agent, for example if only D is involved in the implementation of "Advice to Change Dietary Intake" without any role for P or A, then the team is overall compliant with the guideline recommendations but it is not interactive.

Thus these levels of implementation of clinical practice guidelines within the organizational set-up using the team-based approach formally specify the steps one would need to make the guideline based recommendations fully integrated with the existing practices within the healthcare organization.


Hospital
Internal Medicine Dept.
Inpatient Unit
Procedure Room
Outpatient Unit
Partition of Physical Structure
Diagnostic Procedures in Management of Hypertension
Risk assessment of Hypertensive Patients
Determination of Stage of Hypertension
Measurement of Blood Pressure
Partition of Tasks in Guidelines
Healthcare Professionals
Nursing Professionals
Physicians
Nursing Staff Int. Medicine
Nursing Student
Internist
Cardiologist
Partition of Healthcare Professionals

Health Care team for management of Hypertension which *has-parts* **Physical Internist** *and* **Nursing Staff Internal Medicine** (in *partial overlap* of *functions*), which is a *part-of* **Health Care Professionals** in **Outpatient of Internal Medicine Department** which is a *part-of* **Internal Medicine Department** which is a *part-of* **Tertiary Hospital** having *collective-function* **Mesaurement of Blood Pressure** which is a *part- of* **Determination of Stage of Hypertension** which is a *part- of* **Risk Assessement of Hypertensive Patients** which is a *part-of* **Diagnostic Procedures in management of Hypertension** which is a *part-of* **Health Care Activity in management of Hypertension**

Fig 1 Sample of different partitions used to describe the implementation of tasks recommended in guidelines in a healthcare organization (——◆ = is-part-of)

CONCLUSION

This paper describes the different levels of guideline implementation, considering the tasks as functions and defining successive levels of granularity of functional assignment or representation. These functions are executed as processes or clinical activities in real-time, for which we will describe corresponding granularity levels in further work, where we will take into consideration also the sequence of execution of the workflow tasks.

References

1. Panzarasa S, Madde S, Quaglini S, Pistarini C, Stefanelli M. Evidence-based careflow management systems: the case of post-stroke rehabilitation. J Biomed Inform. 2002 Apr; 35(2): 123-39.
2. Aalst WMP & Kumar A. Team-Enabled Workflow Management System. Data and Knowledge Engineering, 38(3):335-363, 2001.
3. Peleg M, Tu S, Bury J, Ciccarese P, Fox J, Greenes RA, Hall R, Johnson PD, Jones N, Kumar A, Miksch S, Quaglini S, Seyfang A, Shortliffe EH, Stefanelli M. Comparing computer-interpretable guideline models: a case-study approach. J Am Med Inform Assoc. 2003 Jan-Feb;10(1):52-68.
4. Bittner T & Smith, B. Vagueness and granular partitions. Proceedings of the Conference on Formal Ontology in Information Systems (FOIS 2001), Sheridan Press
5. Bittner T & Smith B. A Theory of Granular Partitions, Foundations of Geographic Information Science, M Duckham, MF. Goodchild and MF. Worboys, eds., London: Taylor & Francis, 2003, 117-151.
6. WHO Hypertension Guideline. http://www.bpcr.-com/diet/who
7. Appel LJ. Lifestyle modification as a means to prevent and treat high blood pressure. J Am Soc Nephrol. 2003 Jul; 14(7 Suppl 2): S99-S102.

Acknowledgement

Work on this paper was supported by the Wolfgang Paul Program of the Alexander von Humboldt Foundation.

Contact Information: Anand Kumar, Laboratory of Medical Informatics, Department of Computer Science, University of Pavia, Via Ferrata 1, Pavia, Italy. PV 27100. email: anand@aim.unipv.it

Effect of Imprecision in Probabilities on Bayesian Network Models: An Empirical Study

Agnieszka Oniśko
Faculty of Computer Science
Białystok Technical University
ul. Wiejska 45A
15–351 Białystok, Poland
aonisko@ii.pb.bialystok.pl

Marek J. Druzdzel
Decision Systems Laboratory
School of Information Sciences
and Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260, USA
marek@sis.pitt.edu

Abstract

While most knowledge engineers believe that the quality of results obtained from Bayesian networks is not too sensitive to imprecision in probabilities, this remains a conjecture with only modest empirical support. Our work on a Bayesian network model for diagnosis of liver disorders, Hepar II, presented us with an opportunity to test this conjecture in a practical setting. We present the results of an empirical study in which we systematically introduce noise in Hepar II's probabilities and test the diagnostic accuracy of the resulting model. We replicate an experiment conducted by Pradhan et al. [13] and show that Hepar II is more sensitive to noise in parameters than the CPCS network that they examined. Our data show that the diagnostic accuracy of the model deteriorates almost linearly with noise. While our result is merely a single data point that sheds light on the hypothesis in question, we suggest that Bayesian networks are more sensitive to the quality of their numerical parameters than popularly believed.

1 Introduction

Decision-analytic methods provide an orderly and coherent framework for modeling and solving decision problems in intelligent systems [4]. A popular modeling tool for complex uncertain domains is a Bayesian network [12], an acyclic directed graph quantified by numerical parameters and modeling the structure of a domain and the joint probability distribution over its variables. There exist algorithms for reasoning in Bayesian networks that typically compute the posterior probability distribution over some variables of interest given a set of observations. Because these algorithms are mathematically correct, they essentially solve the underlying model. Hence, the ultimate quality of reasoning depends directly on the quality of this model and its parameters. These parameters are rarely precise, as they are often based on subjective estimates. Even if they are based on statistics, these may not be directly applicable to the decision model at hand and not fully trusted.

Search for those parameters whose values are critical for the overall quality of decisions is known as sensitivity analysis. Sensitivity analysis studies how much a model output changes as various model parameters vary through the range of their plausible values. It allows to get insight into the nature of the problem and its formalization, helps in refining the model so that it is simple and elegant (containing only the factors that matter), and checks the need for precision in refining the numbers [16]. Several researchers proposed efficient algorithms for performing sensitivity analysis in Bayesian networks (e.g., [1; 2; 3; 5]).

There is no doubt that it is theoretically possible that small variations in a numerical parameter cause large variations in the posterior probability of interest. Van der Gaag and Renooij [15] found that networks may indeed contain such parameters. Because practical networks are often constructed with only rough estimates of probabilities, a question of practical importance is whether overall imprecision in network parameters is important. If not, the effort that goes into polishing network parameters might not be justified, perhaps with the exception of some small number of critical parameters. Furthermore, qualitative schemes might perform well without the need for precise, numerical estimates. Conversely if network results are sensitive to the precise values of probabilities, it is unlikely that qualitative schemes will match the performance of quantitative problem specification. There is a popular belief, supported by some anecdotal evidence, that Bayesian network models are overall quite tolerant to imprecision in their numerical parameters. Pradhan et al. [13] tested this on a large medical diagnostic model, the CPCS network [6; 14]. Their key experiment focused on systematic introduction of noise in the original parameters (assumed to be the gold standard) and measuring the influence of the magnitude of this noise on the average posterior probability of the true diagnosis. They observed that this average was fairly insensitive to even very large noise. This experiment, while thought provoking, had two weaknesses. The first of these, pointed out by Coupé and van der Gaag [3], is that the experiment focused on the average posterior rather than individual posterior in each diagnostic case and how it varies with noise, which is of most interest. The second weakness is that the posterior of the correct diagnosis is by itself not a sufficient measure of model robustness. Practical model performance will depend on how these posteriors are used. In order to make a rational diagnostic decision, for example, one needs to know at least the probabilities of rival hypotheses (and typically the joint probability distribution over all disorders). Only this allows for weighting the utility of correct against the dis-utility of incorrect diagnosis. If the focus of reasoning is differential diagnosis, it is of importance to observe how the posterior in question compares to the posteriors of competing disorders. Effectively, the question whether actual performance of a Bayesian network model is robust to imprecision in its numerical parameters remains open.

Our earlier work on a Bayesian network model for diagnosis of liver disorders, Hepar II [9; 10], presented us with an excellent opportunity to shed some light on this question in a practical setting. In this paper, we present the results of an empirical study in which we systematically introduce noise in Hepar II's probabilities and test the diagnostic accuracy of the resulting model. Similarly to Pradhan et al. [13], we assume that the original set of parameters and the model's performance are ideal. Noise in the original parameters leads to deterioration in performance. The main result of our analysis is that noise in numerical parameters starts taking its toll from the very beginning and not, as suggested by Pradhan et al., only when it is very large. Because Hepar II is a medical diagnostic model, we also study the influence of noise in each of the three major classes of variables: (1) medical history, (2) physical examination, (3) laboratory tests, and (4) diseases, on the diagnostic performance. Although the differences here were rather small, it seemed that noise in the results of laboratory tests was most influential for the diagnostic performance of our model. While our result is merely a single data point that sheds light on the hypothesis in question, we suggest that Bayesian networks may be more sensitive to the quality of their numerical parameters than popularly believed.

The remainder of this paper is structured as follows. Section 2 provides a brief overview of the Hepar II model. Section 3 describes our experimental setup and the results of our experiments. Finally, Section 4 discusses our results in light of previous work and also offers some insight into the problem of sensitivity of Bayesian networks to imprecision in their numerical parameters.

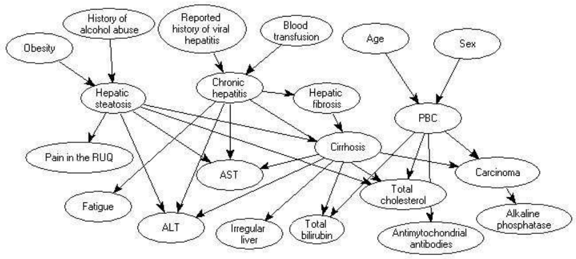


Figure 1: A simplified fragment of the Hepar II network

2 The Hepar II model

The Hepar II project [9; 10] aims at applying decision-theoretic techniques to the problem of diagnosis of liver disorders. Its main component is a Bayesian network model involving over 70 variables. The model covers 11 different liver diseases and 61 medical findings, such as patient self-reported data, signs, symptoms, and laboratory tests results. The structure of the model, (i.e., the nodes of the graph along with arcs among them) was built based on medical literature and conversations with our domain expert, a hepatologist Dr. Hanna Wasyluk and two American experts, a pathologist, Dr. Daniel Schwartz, and a specialist in infectious diseases, Dr. John N. Dowling. The elicitation of the structure took approximately 50 hours of interviews with the experts, of which roughly 40 hours were spent with Dr. Wasyluk and roughly 10 hours spent with Drs. Schwartz and Dowling. This includes model refinement sessions, where previously elicited structure was reevaluated in a group setting. The structure of the model consists of 121 arcs and the average number of parents per node is equal to 1.73. There are on average 2.24 states per variable. In the version used in our experiments, none of the gates was a canonical gate, such as Noisy-OR or Noisy-MAX (although experiments on Hepar II conducted elsewhere [17] showed that as many as 50% of the gates with the parents could be approximated reasonably well by Noisy-MAX gates). Figure 1 shows a simplified fragment of the Hepar II network.

The numerical parameters of the model (there are 1,488 of these in the most current version), i.e., the prior and conditional probability distributions, were learned from the Hepar database. The Hepar database, was created in 1990 and thoroughly maintained since then at the Gastroentorogical Clinic of the Institute of Food and Feeding in Warsaw. The current database contains over 800 patient records and its size is steadily growing. Each hepatological case is described by over 160 different medical findings, such as patient self-reported data, results of physical examination, laboratory tests, and finally a histopathologically verified diagnosis. The version of the Hepar data set, available to us, consisted of 699 patient records.

As the current paper focuses on the model performance as a function of noise in its numerical parameters, we owe the reader an explanation of the metric that we used to test the model performance. We focused on diagnostic accuracy, which we defined as the percentage of correct diagnoses on real patient cases in the HEPAR database. Because we used the same database to learn the model parameters, we applied the method of "leave-one-out" [7], which involved repeated learning from 698 records out of the 699 records available and subsequently testing it on the remaining 699th record. When testing the diagnostic accuracy of HEPAR II, we were interested in both (1) whether the most probable diagnosis indicated by the model is indeed the correct diagnosis, and (2) whether the set of w most probable diagnoses contains the correct diagnosis for small values of w (we chose a "window" of w=1, 2, 3, and 4). The latter focus is of interest in diagnostic settings, where a decision support system only suggest possible diagnoses to a physician. The physician, who is the ultimate decision maker, may want to see several alternative diagnoses before focusing on one.

With diagnostic accuracy defined as above, the most recent version of the HEPAR II model reached the diagnostic accuracy of 57%, 69%, 75%, and 79% for window sizes of 1, 2, 3, and 4 respectively [11]. The model compared very favorably against medical practitioners on a randomly selected 10 patient cases [8]. More details about the performance of HEPAR II model can be found in [9; 10].

For the purpose of our experiments, we assumed that the model parameters were perfectly accurate and effectively, the diagnostic performance achieved was the best possible. In the experiments we study how this baseline performance degrades under the condition of noise. Of course, in reality the parameters of the model may not be accurate and the performance of the model can be improved upon.

3 Experimental results

We have performed several experiments to investigate how noise introduced into network parameters affects the diagnostic accuracy of HEPAR II. To that effect, we have successively created various versions of the model with different levels of noise and tested the performance of these models. The following sections describe the noise generation and the observed results.

3.1 Replication of the experiments of Pradhan et al.

As a starting point, we replicated the experiment performed by Pradhan et al. [13] on the HEPAR II model. The original experiment investigated the robustness of a very large medical diagnostic network, CPCS [6; 14], to noise in numerical parameters. The noise was introduced by transforming each original probability into log-odds function, adding normal noise with a standard deviation σ, and transforming it back to probability, i.e.,

$$p' = Lo^{-1}[Lo(p) + \text{Normal}(0, \sigma)] \quad (1)$$

where

$$Lo(p) = \log_{10}[p/(1-p)] \ . \quad (2)$$

The original experiment involved only binary variables and the transformation yielded a valid probability. In our case, many model variables were not binary. We added a normalization step — after transforming all probabilities within a distribution, we made sure that they add up to 1.0. Similarly to Pradhan et al., we derived the posterior probability assigned by the HEPAR II network to the true diagnosis, averaged over the set of test cases for $\sigma \in < 0.0, 3.0 >$ with 0.1 increments.

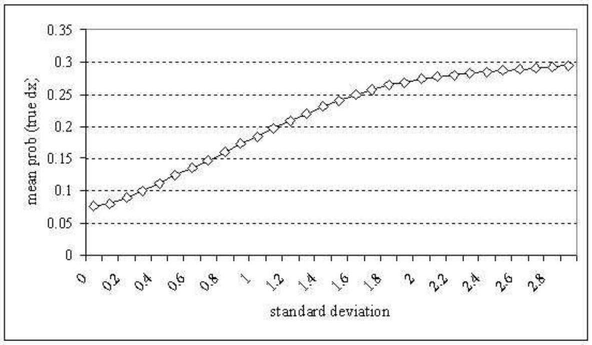


Figure 2: The average posteriors for the true diagnoses as a function of σ

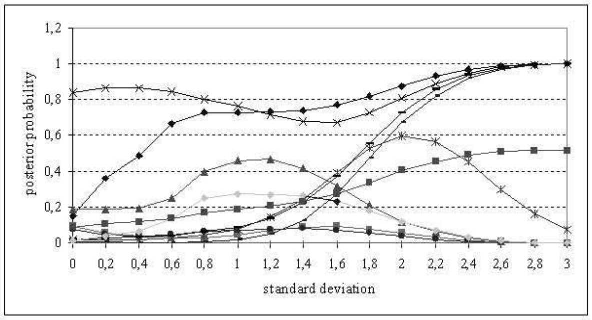


Figure 3: The posterior probabilities of HEPAR II disorders as a function of σ on a single patient case

The results, shown in Figure 2, indicate, similarly to Pradhan et al., that the average posteriors are not sensitive to accuracy in probabilities. These posteriors actually increased with the increase in σ. We believe that this is due to an overall increase in a priori probabilities of all diseases. The prevalence of each of the disease is rather small and noise typically increases it (please note that HEPAR II is a multiple-disorder model).

We have subsequently studied how individual posteriors of all disorders included in the model change

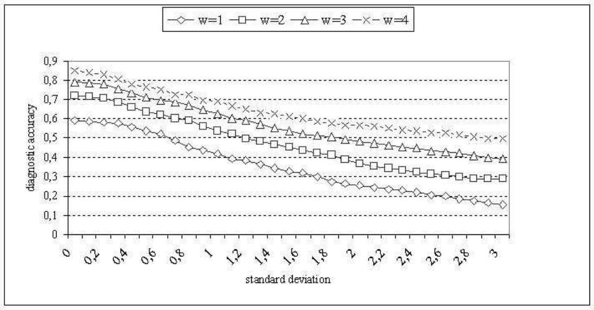


Figure 4: The diagnostic accuracy as a function of σ

as a function of noise. Here we observed that not only the probabilities, but also the order between them changes, demonstrating that average over all runs does not reflect sensitivity well. Figure 3 shows a plot of posterior probabilities of the 11 disorders as a function of noise on a single patient case. This case is quite representative for the cases we examined. We can see that the posterior probabilities change with the noise to the point of changing the order of the most probable diagnoses.

3.2 Effect of noise on Hepar II's diagnostic performance

Our next experiment studied Hepar II performance under the noise conditions described in the previous section. We tested each of the 30 versions of the network (each for a different standard deviation of the noise $\sigma \in < 0.0, 3.0 >$ with 0.1 increments) on the set of test cases and computed its diagnostic accuracy, plotted in Figure 4 for different values of window size as a function of σ.

It is clear that the diagnostic performance deteriorates for even smallest values of noise. This result is quite different from that reported in the previous section (Figure 2). It shows that the measure adopted by Pradhan et al. may not reflect well the practical performance of the model.

3.3 Partial noise

Given a medical diagnostic model, it is of interest to know which of the semantically distinct parts of the model (i.e., medical history, physical examination, disease prevalence, and laboratory results) are most crucial for the network performance. We addressed this question by introducing noise in these parts only and studying the resulting performance.

Figure 5 shows the diagnostic accuracy of the model for noise in each part of the network as a function of σ. We can observe that noise in the results of laboratory tests impacts the diagnostic performance of our model most.

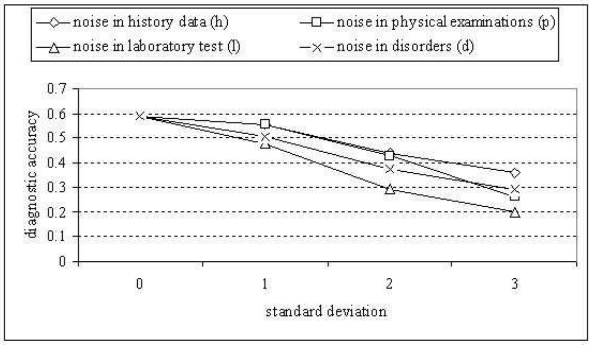


Figure 5: The diagnostic accuracy of the model as a function of σ (w=1).

4 Discussion

This paper has studied the influence of precision in parameters on model performance in the context of a practical medical diagnostic model, Hepar II. Our study has shown that the performance of Hepar II is sensitive to noise in numerical parameters, i.e., the diagnostic accuracy of the model decreases after introducing noise into numerical parameters of the model. The main result of our analysis is that noise in numerical parameters starts taking its toll from the very beginning and not, as suggested by Pradhan et al., only when it is very large. We believe that there are two possible explanations of this difference. The first and foremost is that Pradhan et al. used a different criterium for model performance — the average posterior probability of the correct diagnosis. We focused on the diagnostic performance of the model. Another, although perhaps a less influential factor, may be differences between our models. The CPCS network used by Pradhan et al. consisted of only Noisy-OR gates, which may behave differently than general nodes. In Hepar II only roughly 50% of all nodes could be approximated by Noisy-MAX.

We have also studied the influence of noise in each of the three major classes of variables: (1) medical history, (2) physical examination, (3) laboratory tests, and (4) diseases, on the diagnostic performance. It seemed that noise in the results of laboratory tests was most influential for the diagnostic performance of our model. This can be explained by the high diagnostic value of laboratory tests. This value decreases with the introduction of noise.

The results of our experiment touch the foundations of qualitative modeling techniques. As qualitative schemes base their results on approximate or abstracted measures, one might ask whether their performance will match that of quantitative schemes, either in terms of their strength or the correctness of their results.

While our result is merely a single data point that sheds light on the hypothesis in question, we suggest that Bayesian networks may be more sensitive to the quality of their numerical parameters than popularly believed. We argue that further empirical studies of this topic should use hard context-dependent performance measures (such as the quality or correctness of system's recommendation). Alternatively, one might use measures such as admissible deviation (a change in probability that does not impact the order of most likely diagnoses) proposed by van der Gaag and Renooij [15].

Acknowledgments

This work was supported by the Air Force Office of Scientific Research grants F49620–00–1–0112 and F49620–03–1–0187, by the Polish Committee for Scientific Research grant 4T11E05522 and by the Białystok Technical University grant W/WI/1/02.

References

[1] Hei Chan and Adnan Darwiche. When do numbers really matter? *Journal of Artificial Intelligence Research*, 17:265–287, 2002.

[2] Veerle H. M. Coupé and Linda van der Gaag. Practicable sensitivity analysis of Bayesian belief networks. In *Prague Stochastics '98 — Proceedings of the Joint Session of the 6th Prague Symposium of Asymptotic Statistics and the 13th Prague Conference on Information Theory, Statistical Decision Functions and Random Processes*, pages 81–86, Union of Czech Mathematicians and Physicists, 1998.

[3] Veerle H. M. Coupé and Linda C. van der Gaag. Properties of sensitivity analysis of Bayesian belief networks. *Annals of Mathematics and Artificial Intelligence*, 36:323–356, 2002.

[4] Max Henrion, John S. Breese, and Eric J. Horvitz. Decision Analysis and Expert Systems. *AI Magazine*, 12(4):64–91, Winter 1991.

[5] Uffe Kjaerulff and Linda C. van der Gaag. Making sensitivity analysis computationally efficient. In *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference (UAI-2000)*, pages 317–325, San Francisco, CA, 2000. Morgan Kaufmann Publishers.

[6] B. Middleton, M.A. Shwe, D.E. Heckerman, M. Henrion, E.J. Horvitz, H.P. Lehmann, and G.F. Cooper. Probabilistic diagnosis using a reformulation of the INTERNIST–1/QMR knowledge base: II. Evaluation of diagnostic performance. *Methods of Information in Medicine*, 30(4):256–267, 1991.

[7] A.W. Moore and M.S. Lee. Efficient algorithms for minimizing cross validation error. In *Proceedings of the 11th International Conference on Machine Learning*, San Francisco, 1994. Morgan Kaufmann.

[8] Agnieszka Oniśko. Evaluation of the Hepar II system for diagnosis of liver disorders. In *Working notes of the European Conference on Artificial Intelligence in Medicine (AIME-01): Workshop Bayesian Models in Medicine*, Cascais, Portugal, July 2001.

[9] Agnieszka Oniśko, Marek J. Druzdzel, and Hanna Wasyluk. Extension of the Hepar II model to multiple-disorder diagnosis. In S.T. Wierzchoń M. Kłopotek, M. Michalewicz, editor, *Intelligent Information Systems,* Advances in Soft Computing *Series*, pages 303–313, Heidelberg, 2000. Physica-Verlag (A Springer-Verlag Company).

[10] Agnieszka Oniśko, Marek J. Druzdzel, and Hanna Wasyluk. Learning Bayesian network parameters from small data sets: Application of Noisy-OR gates. *International Journal of Approximate Reasoning*, 27(2):165–182, 2001.

[11] Agnieszka Oniśko, Marek J. Druzdzel, and Hanna Wasyluk. An experimental comparison of methods for handling incomplete data in learning parameters of Bayesian networks. In S.T. Wierzchoń M. Kłopotek, M. Michalewicz, editor, *Intelligent Information Systems,* Advances in Soft Computing *Series*, Heidelberg, 2002. Physica-Verlag (A Springer-Verlag Company). 351–360.

[12] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1988.

[13] Malcolm Pradhan, Max Henrion, Gregory Provan, Brendan del Favero, and Kurt Huang. The sensitivity of belief networks to imprecise probabilities: An experimental investigation. *Artificial Intelligence*, 85(1–2):363–397, August 1996.

[14] M.A. Shwe, B. Middleton, D.E. Heckerman, M. Henrion, E.J. Horvitz, H.P. Lehmann, and G.F. Cooper. Probabilistic diagnosis using a reformulation of the INTERNIST–1/QMR knowledge base: I. The probabilistic model and inference algorithms. *Methods of Information in Medicine*, 30(4):241–255, 1991.

[15] Linda C. van der Gaag and Silja Renooij. Analysing sensitivity data from probabilistic networks. In *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference (UAI-2001)*, pages 530–537, San Francisco, CA, 2001. Morgan Kaufmann Publishers.

[16] Detlof von Winterfeldt and Ward Edwards. *Decision Analysis and Behavioral Research.* Cambridge University Press, Cambridge, 1988.

[17] Adam Zagorecki and Marek J. Druzdzel. How common are Noisy–OR distributions in practice? In preparation, 2003.

Qualitative Reasoning in Integrative Biology: Evolving a Network Experiment Modeling System for Collaborative Multilevel Systems Research

Robert B. Trelease
Dept. of Pathology and Lab. Med.
Geffen School of Medicine, UCLA
Los Angeles, California, 90095 USA
E-mail: trelease@ucla.edu

Jack Park
Thinkalong Software
Box 61257
Palo Alto, California, 94306 USA
E-mail: jackpark@thinkalong.com

Abstract

We address important design considerations in effecting a network server-based experiment modeling system intended for integrative biology and biomedical research. We proceed to that end by following the progressive development of application features for our experimental qualitative process (QP) modeling system. Early work on a PC-based system for experimental biology focused on multiple-level visualizations of cellular immune responses, as well as receptor-mediated processes regulating protooncogene expression. Other practical applications focused on lymphocyte circulation and organopedesis, hyperbaric oxygenation effects on angiogenesis, macrophage migration, and fibroblast elaboration of ground substance. With the introduction of stabile Java technology in the mid-90s, work began on migrating the original PC-based modeling system environment (TSC) to a framework that would eventually support parallel execution of inference processes in a server-based environment. As the project has evolved, we have had to consider incorporating output functions of cooperative applications, issues for database and ontology importing, and technical challenges in representing level-unifying knowledge and experimental process rules.

1 Introduction

In integrative biology and medical research, new attention [2] has been focused on the need for integrating data, information and knowledge from multiple frames of reference and disciplines, in order to provide more comprehensive scientific models of how dynamical processes interact at molecular, cellular, organismal, behavioral, and population levels of representation. In genomics, for example, recent emphasis has been put on the need to accumulate usable ontologies of specific gene functions, in order to provide a foundation for understanding the dynamics of various biological processes[3].

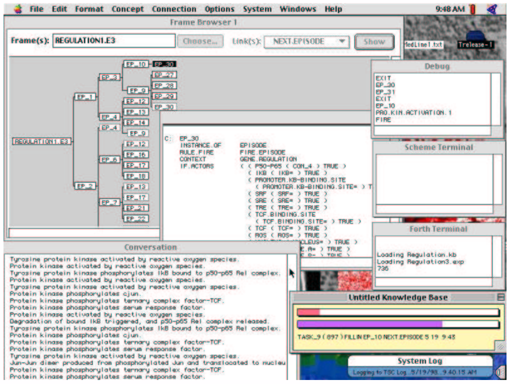
Practically applying quantitative methods to implementing process models at a few levels has lead to the development of useful 'in silico biology' programs such as the Virtual Cell, In Silico Cell, and Physiome. However, in this context, it has been openly stated that so much data, information and knowledge are being collected, that they far outstrip the ability of scientists and programmers to make practical integrative use of them. As one genomics pioneer noted recently [12], strictly quantitative approaches to in-silico biology immediately face an intractable combinatorial explosion of multiple levels of parallel equations exceeding the capabilities of current computer resources. More significantly, many biological process descriptions and relationships are largely qualitative and are quantitatively undefined or deterministically undefinable. Under such circumstances, it is worth considering qualitative reasoning (QR) for an alternative approaches for managing complexity in multilevel biological modeling.

During the early 1990s, we began the development of a qualitative process modeling system for running simulated experiments on biological processes, cells, and organisms[15][13]. As the early years passed, practical use of the World Wide Web burgeoned, and great new resources became available for 'doing science better' via Internet connectivity. We have thus seen that our qualitative modeling environment could evolve into a more powerful multidisciplinary collaborative environment for research ultimately supported by today's distributed computing networks, online databases, ontologies, and powerful, openly available information tools.

2 Methods and Design Issues: Early efforts with workstation-based qualitative modeling

Initial work focused on creating a broad-based hierarchical knowledge base system incorporating features of biological taxonomy as well as details of eukaryotic cell biology. In seeking to pioneer some applications of qualitative reasoning in biology, the senior author was particularly influenced by the concepts of QP theory (QPT, from Kenneth Forbus [4]) and qualitative simulation (QSIM, from Benjamin Kuipers [9][10]) as applied to physics problems. Lessons were also learned from Peter Karp's development of EcoCYC, a pioneering modeling-oriented ontology for microbial genetics [8].

For our biological modeling system, concepts, substances, cells, organisms, and multilevel processes were symbolically defined in an object-oriented environment known as TSC (The Scholar's Companion). The initial programming environment include underlying Pascal, Forth, and Scheme components, with knowledge bases coded in Scheme frames. Quantitative data were handled by numerical decoding process rules. This system was used successfully to model primary immune responses to bacterial and viral pathogens [16], as well as regulatory mechanisms for protooncogenes underlying specific hormonal, free-radical, and immune system signal transduction [14].



```
C:  EP_30
    INSTANCE.OF      EPISODE
    RULE.FIRE        FIRE.EPISODE
    CONTEXT          GENE.REGULATION
    IF.ACTORS        ( ( P50-P65 ( CON_4 ) TRUE )
                       ( IKB ( IKB= ) TRUE )
                       ( PROMOTER.KB-BINDING.SITE
                         ( PROMOTER.KB-BINDING.SITE= ) T
                       ( SRF ( SRF= ) TRUE )
                       ( SRE ( SRE= ) TRUE )
                       ( TRE ( TRE= ) TRUE )
                       ( TCF.BINDING.SITE
                         ( TCF.BINDING.SITE= ) TRUE )
                       ( TCF ( TCF= ) TRUE )
                       ( ROS ( ROS= ) TRUE )
                                   CLEUS= ) TRUE )
                                 E.A= ) TRUE )
```

```
EXIT
EP_30
EP_31
EXIT
EP_10
PRO.KIN.ACTIVATION.1
FIRE
```

```
Loading Regulation.kb
Loading Regulation3.exp
736
```

```
Tyrosine protein kinase activated by reactive oxygen species.
Protein kinase activated by reactive oxygen species.
Tyrosine protein kinase phosphorylates IkB bound to p50-p65 Rel complex.
Protein kinase activated by reactive oxygen species.
Tyrosine protein kinase activated by reactive oxygen species.
Protein kinase phosphorylates cjun.
Protein kinase phosphorylates ternary complex factor-TCF.
Protein kinase phosphorylates serum response factor.
Protein kinase activated by reactive oxygen species.
Degradation of bound IkB triggered, and p50-p65 Rel complex released.
Tyrosine protein kinase phosphorylates IkB bound to p50-p65 Rel complex.
Protein kinase phosphorylates cjun.
Protein kinase phosphorylates ternary complex factor-TCF.
Protein kinase phosphorylates serum response factor.
Tyrosine protein kinase activated by reactive oxygen species.
Jun-Jun dimer produced from phosphorylated Jun and translocated to nucleu
Protein kinase phosphorylates ternary complex factor-TCF.
Protein kinase phosphorylates serum response factor.
```

```
TASK_9 ( 897 ) FILLIN EP_10 NEXT.EPISODE 5 19 9 43
```

```
Logging to TSC Log...5/19/98...9.40.15 AM
```

Figure 1: TSC running a gene regulation experiment.

Other (unpublished) process models created with the initial TSC system simulated diapedesis and systemic circulation of lymphocytes (with organ-specific binding mediated by cell adhesion molecules), as well as macrophage migration, angiogenesis and wound healing under conditions of hyperbaric oxygenation. Beyond the research environment, the modeling system was also employed in trials in secondary school instruction for simulating scientific experiments and representing theories as diverse as those of immunity, ecology, and extinction. In one original instance, operating outside of the biological domain with a different knowledge base, TSC provided conjectures which supported the discovery of new process control rules in a polymer curing environment [1].

With the introduction of stabile, fast-executing Java versions in the mid-90s, the senior author began experimenting with a simple applet version of a QP modeling system. Using a transliteration of earlier TSC knowledge base content into the CLIPS-like Scheme dialect of JESS (Ernest Friedman-Hill's Java Expert System shell), basic published proto-oncogene experiments were repeated. In this simplified example of a Web-based simulation resource, only textual envisionments were generated, but they indicated the feasibility of building a multiplatform QP modeling system with modifications of a forward-chaining inference environment.

3 The evolution of the TSC QR environment

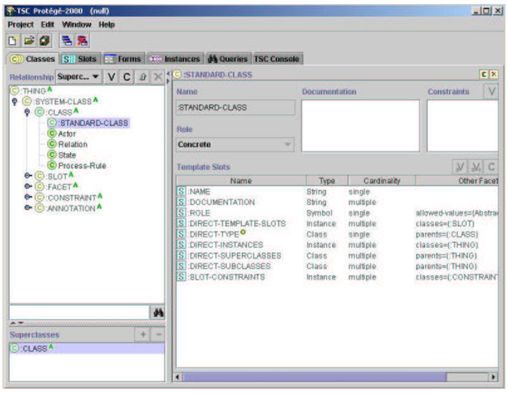
At the same time, TSC developer Jack Park began the process of translating TSC resources into Java. Java libraries for manipulation of list structures, coupled with built in automatic garbage collection of the Java working memory, made the translation relatively straight-forward. TSC used a supervisory agenda-based inference engine (envisionment builder), one that posted tasks to an agenda for software agents to perform. The move to Java made it possible to integrate the agenda functionality of TSC into a tuplespace Web portal being developed to support collaborative research and learning projects.

Tuplespace is the name given a kind of public repository blackboard architecture created by David Gelernter [6] , wherein tuple data or parameters (corresponding to non-relational database records) are used for communication between multiple active programs distributed over different machines. Tuplespace is thus well suited for agent coordination, and it has been implemented in a variety of ways in Linda, JavaSpaces [5], IBM's TSpaces, and TupleSpace4J. TupleSpace4J is the implementation being used for TopicSpaces, a Collaboratory being developed by Jack Park for research and learning projects.

The move to Java and coupling to a tuplespace implementation of the agenda management meant that the original inference engine itself was no longer needed for organizing agent functions. In Java-based TSC (called TSC4J), as tasks come into the agenda, they are distributed in tuplespace to be handled by specific agents which are, at once, available and programmed for the chosen task. In this context, tuplespace can also be viewed as associative memory for the QR system.

In essence, TSC4J is now a collection (collective) of agents intended for building models, studying those models, studying data flows in relation to models, noticing expectation failures (where data and model predictions fail to agree), and forming conjectures regarding the nature of expectation failures. Consistent with the original purpose of TSC and continuing its legacy in process discovery, TSC4J continues to serve a role as a modeling and analytical assistant.

TSC4J operates on its own internal information and knowledge structures, but includes import agents capable of translating and loading knowledge in many known ontology structures, e.g., RDF (Resource Description Framework), XTM (XML Topic Maps [11]), and OKBC (Open Knowledge Base Connectivity). At the same time, available export agents allow the results of TSC4J sessions to be written to any of those ontology representation frameworks. TSC4J is also being functionally integrated (”tabbed”) with Stanford’s Protégé 2000, allowing its use as a knowledge engineering tool for preparing new model ontologies.



:STANDARD-CLASS

Name: :STANDARD-CLASS

Role: Concrete

Name	Type	Cardinality	Other Facet
:NAME	String	single	
:DOCUMENTATION	String	multiple	
:ROLE	Symbol	single	allowed-values={Abstra
:DIRECT-TEMPLATE-SLOTS	Instance	multiple	classes={:SLOT}
:DIRECT-TYPE	Class	single	parents={:CLASS}
:DIRECT-INSTANCES	Instance	multiple	classes={:THING}
:DIRECT-SUPERCLASSES	Class	multiple	parents={:THING}
:DIRECT-SUBCLASSES	Class	multiple	parents={:THING}
:SLOT-CONSTRAINTS	Instance	multiple	classes={:CONSTRAIN

Figure 2: TSC running as a ”TAB” with Protégé.

The proposed TSC4J Biology Network Experiment Modeling System (BioNEMS) would support not only collaborative modeling in multi-disciplinary professional research groups, but networked collaboration in science education and public enthusiast sectors. TSC4J development efforts are being managed within the larger TopicSpaces/TopicMaps Web resources.

4 Knowledge engineering issues in collaboration space

In seeking to incorporate existing online databases and ontologies, TSC4J BioNEMS faces several related challenges. At the outset, there is the functionality needed to match export and import agents to the task of translating knowledge contained in any of the currently available RDF-based (e.g., DAML/OIL, OWL) or OKBC-based (e.g., Ontolingua) formats, as well as to others which may be composed of tab- or comma-delimited text. Some of the necessary translation agents already exist within the TSC4J and TopicSpaces projects. The TopicSpaces Collaboratory engine combines XML topic maps for navigation with tuplespace for agent coordination, providing a powerful schema-neutral entity-attribute representation scheme. Tuples are capable of representing any object which can be decomposed into data fields or elements. Each input field gets a name, a value, and a value type, to which the tuple appends metadata on authorship, dates, security/privacy codes and identity of approved viewers in the case of private fields. The value in this simplified, canonical representation is that it can be easily mapped to/from other representation or serialization schemes, particularly tagged-language (XML, SGML, RDF, etc) serializations.

Federating BioNEMS research collaboratories by means of a unified XML topic map environment provides three important enhancements to traditional knowledge engineering tasks: 1) Rapid and precise navigation of the joint information resources created; 2) collaborative filtering by way of annotations and extensive linking and cross linking; 3) the ability to extend the topic map (knowledge base) by way of additional knowledge engineering processes and by way of linking to related materials found by mining the knowledge engineering work product.

Consider, for example, a BioNEMS model of a particular immune function. The subject of discussion is that immune function, and the model will contain representations of many episodes or sets of experimental conditions and state transitions for processes occurring during the course of a simulated experiment. Each episode is linked into the topic map.

The topic map provides immediate indexing of all actors, relations, states, and processes involved in the model, and those entities are all related, through typed topic map associations, the types of which are also topics which are indexed and available for discussion. When the system is used to its fullest capabilities by researchers, all references (such as journal articles, dissertation references, and other related technical information) will be integrated into the topic map. Reference information is thus linked directly to each and every simulation element for which there is an association.

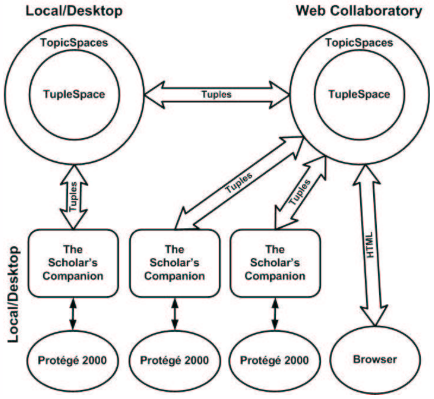


Figure 3: Conceptual block diagram of TSC4J components and BioNEMS.

TSC4J (with BioNEMS) is thus integrated into a Web-based collaboration space where all objects in models being built are also objects available for collaborative filtering, for discussion, and as content for learning experiences. Tuplespace provides the mechanism coordinating all agents, including those which provide TSC4J’s modeling functionality. Thus, the topic map is itself continuously maintained by agents not directly related to TSC4J, but which are privy to

TSC4J's results as they appear in tuplespace. With this level of coordination of agents, it becomes possible, for example, for users to subscribe to different kinds of results and receive email reports of modeling progress.

5 Infrastructure issues for BioNEMS

Given the cross-platform Java servlet, XML, SQL and Web environments hosting BioNEMS, two initial configurations are anticipated. A simple local datamining approach would have BioNEMS installed on a small laboratory computing cluster, with hosting of imported, enhanced, mission-specific databases and ontologies locally as private resources. The larger extra-institutional collaboratory environment would be a publicly accessible resource supporting wide-scale scientific discovery efforts with publicly accessible ontologies and databases.

Although BioNEMS and its associated environment are essentially platform-nonspecific, we recognize that hardware design is an import issue for scientific collaboratory infrastructure intended to support intensive biological modeling. As part of this effort, we are investigating the performance of BioNEMS in a small, localized Unix-based supercomputing cluster configuration (gigabit interconnected multiprocessor servers).

6 Future directions

TSC4J and BioNEMS are very ambitious projects still in the midst of development. Given success of ongoing grant applications and more programming effort, we hope to be testing a functional network modeling system within the next year. Further development of the TSC4J BioNEMS system is expected to involve the continuous evolution of the knowledge engineering functionality. New agents will be created which treat ontologies much like collections of database entries and mine those entries for purposes of modeling and inference. Large and heterogeneous ontologies represent significant challenges to our ability to federate them in service of diverse bio-informatics communities of practice. Some ontologies will not easily submit to automated agent-based federation, and the need for humans in the knowledge engineering loop clearly will drive future enhancements in the system. The inclusion of Web-based communication via Topic Space enables multiple human experts to work synchronously or asynchronouslly on the same model or simulation application (human cluster processing).

At the same time in the infrastructure, Moore's Law should remain in effect: We see opportunities for continuous evolution of the code to take advantage of improved process threading on the hardward platforms hosting BioNEMS, of improved database systems, and of other advances in software technology. Since BioNEMS is an open source project which will include an SDK (systems development kit) for creating new agents, it will be possible for users in various bio-informatics communities to add new functionality to BioNEMS as their capabilities and needs evolve. The future of TSC4J BioNEMS hopefully involves continued expansion of system functions and modules which facilitate its continuing evolution with and by connected systems user communities.

References

[1] F.L. Abrams, *Process Discovery: Automated Process Development for the Control of Polymer Curing*, Doctoral Dissertation, University of Dayton, 1995.

[2] R. Altman and S. Raychauduri, Whole genome expression analysis: challenges beyond clustering, *Current Opinions on Structural Biology*, Vol. 11, 2001, pp. 340-347.

[3] M. Ashburner, and S. Lewis, On ontologies for biologists: the Gene Ontology–untangling the Web, In *'In Silico' Simulation of Biological Processes, Novartis Foundation Symposium 247*, pp. 66-83, John Wiley and Sons, New York, 2002.

[4] K.D. Forbus, Qualitative process theory, *Artificial Intelligence*, Vol. 24, 1984 pp. 85-168.

[5] E. Freeman, S. Hupfer, and K. Arnold, *JavaSpaces Principles, Patterns, and Practices*, Addison-Wesley, New York, 1999.

[6] D. Gelernter, *Mirror Worlds: Or the Day Software Puts the Universe in a Shoebox - How It Will Happen and What It Will Mean*, Oxford University Press, New York, 1992.

[7] P.D. Karp, Hypothesis formation as design, In J. Shrager and P. Langley, Eds., *Computational Models of Scientific Discovery and Theory Formation*, pp. 276-317, Morgan Kaufman, San Mateo, 1990.

[8] P.D. Karp, Frame representation and relational databases: Alternative information-management technologies for systematic biology, In R. Fortuner, Ed., *Advances in Computer Methods for Systematic Biology*, pp. 275-285, The Johns Hopkins University Press, Baltimore, 1993.

[9] B. Kuipers, Qualitative Simulation, *Artificial Intelligence*, Vol. 29, 1986, pp. 289-388.

[10] B. Kuipers, *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge*, The MIT Press, Cambridge, 1994.

[11] J. Park and S. Hunting, *XML Topic Maps: Creating and Using Topic Maps for the Web*, Addison-Wesley, New York, 2002.

[12] B. Stewart, *An Interview with Jim Kent.* O'Reilly Bioinformatics Web site, http://www.oreillynet.com/pub/a/network/2002/12/10/kent.html, 2002.

[13] R.B. Trelease, Development of a feature-based knowledge base for biologically-based materials and processes. *USAF Contributive Research and Development*, Vol. 145, 1994, pp. 1-10.

[14] R.B. Trelease, R.A. Henderson, and J. Park, Experiment-based qualitative process modeling system for regulation of NF-kB and AP-1 protooncogenes, *Artificial Intelligence in Medicine*, Vol. 17, 1999, pp. 303-321.

[15] R.B. Trelease, and J. Park, Use of a heuristic discovery system in computer-based qualitative process modeling of biological systems, *Anatomical Record Supplement*, Vol. 1, 1993, pp. 115.

[16] R.B. Trelease, and J. Park, Qualitative process modeling of cell-cell-pathogen interactions in the immune system, *Computer Methods and Programs in Biomedicine*, Vol. 51, 1996, pp. 171-181.