The Knowledge Engineering Review, Vol. 11:2, 1996, 137–160

Automated modelling: a discussion and review

S. XIA[1] and N. SMITH

Department of Computer and Information Sciences, De Montfort University, Milton Keynes, UK

Abstract

Automated modelling is a young research field and is attracting increasingly more attention. It is a cross-disciplinary field involving simulation, modelling, qualitative reasoning, bond graphs and systems dynamics. It is an investigation of the modelling process with the purpose of developing computer tools which will automatically follow modelling principles. In addition, these tools will take into account the details of an application and generate the most appropriate model for the application. Its objective is to develop computer modelling tools which will have perception of model correctness, completeness and appropriateness and can perform modelling automatically. One way to achieve this objective is to introduce well-defined models and automate the process of assembling submodels into models to create well-defined models. This paper reviews the motivation and background behind this new field, its theory and current state of the art, compares existing approaches and discusses the underlying issues. It is hoped that more researchers will become aware of this field and be encouraged to work in it.

1 Introduction

Models are needed to explain how the world works. When we try to understand what is happening with the world around us, we make a model of what we are interested in and use that model in lieu of the real system to understand what is happening through simulation or causal analysis of the processes involved.

Models are the distillation of what is important about any particular system. They allow attention to be given only to those aspects of the systems that are important, while ignoring all irrelevancies. Models are also used to give structure to our thoughts and ideas on why certain phenomena take place (Collins & Gentner, 1983). A model provides guidance to analytical techniques that allows them to be applied to the specific parts of a problem that need solutions (Sussman & Steele, 1980). Good models do not prevent the analysis of other areas, but they do ensure that we know of their relative merits of attacking unimportant problems. Models also allow us to make predictions as to how systems and phenomena will behave in the future or in reaction to certain stimuli, such as the introduction of a fault.

Building a model for a model-based application is traditionally regarded as a manual task and most decision-making and other intelligent functions are performed by hand. This manual approach has many disadvantages. Firstly, the manual process tends to be labour-intensive and time-consuming. Secondly, manually built models are normally *ad hoc* and thus difficult to reuse and validate. Thirdly, it is almost impossible to trace the experience and heuristics embedded in building a model once a project is completed. So it can be a waste of expertise. Finally, because some of the heuristics and assumptions do not appear in a model, it may be inappropriately applied and incorrectly interpreted.

A new research field of automated modelling has developed to address these issues. Automated modelling is an investigation of the modelling process with the purpose of developing computer tools which will automatically follow modelling principles in constructing models. Automated

[1]Email: sxia@dmu.ac.uk and nsmith@dmu.ac.uk

modelling tools will reduce human involvement in building models. Because well-defined models are introduced and the process of assembling submodels into models to create well-defined models automatically follows the generic modelling principles, models generated will be more likely to be conceptually correct. This will facilitate model reuse and validation. Furthermore, experiences, heuristics and assumptions can be organised into domain-dependent principles and practices, and can be reused when a similar case occurs. In addition to addressing these issues, automated modelling relates the purpose of using a model to the process of building it and attempts to build the most appropriate model for the task.

This paper covers only those aspects of models which are relevant in the context of automated modelling. Models, which are not used in their general sense, are analysed in terms of their structural properties and in relation to sub-models of components and connections. Similarly, modelling is studied with a particular emphasis on the intermediate process of systems decomposition and model construction.

This paper is intended both for researchers working on automated modelling and for researchers in other related fields. For researchers working on automated modelling, this paper represents the attempts to formalise the research field and to establish a common vocabulary for the exchange of ideas. For other researchers, this paper serves as an introduction to automated modelling. Researchers who work on modelling, simulation, control, machine learning, model-based reasoning and intelligent tutoring systems will find automated modelling particularly relevant and useful.

This paper should also be useful reading for industrialists who are exploring new artificial intelligence techniques for practical applications. The encouraging progress in practical work, as shown in section 6, demonstrates the potential of automated modelling to industry. Future research advances in solving any major problems in this field will have great effect on the accuracy, reliability and cost of model construction. Hopefully, with the recognition of the great potential in practical engineering, industries and commercial sectors will fully collaborate with the academic research community and actively explore the benefits of this research. In particular, those sectors which employ "whitebox systems", i.e. systems with clear structure, should find the automated modelling work directly applicable.

The following four major issues are discussed in this paper. Firstly, related disciplines, especially the techniques and methods relevant to automated modelling, are covered. Some relevant information on background and studies of modelling is also provided. Secondly, the development of modelling strategies is summarised. Next, the theory and the state of the art are discussed. This is followed by a review of different automated computer modelling systems. Finally, some of the underlying technical issues which need to be addressed to achieve automated modelling are examined.

1.1 Multi-disciplinary study

Automated modelling is a multi-disciplinary study involving modelling, simulation, artificial intelligence and engineering disciplines. It is a good example of the strengths of different disciplines being integrated to offset the deficiencies of the individual ones (Xia et al., 1992). In particular, the representation languages and techniques of systems decomposition and composition developed in engineering disciplines are used to describe and assemble models. Bond graphs provide an example of such a representation language (Rosenberg & Karnopp, 1983). These representation languages are powerful in terms of describing real-life systems but they themselves are inadequate for problem solving and for deriving new information. Indeed they need to be changed in both content and form so that the information required for problem solving is readily available. The capability for problem solving in these representation formalisms is weak and can be improved by the integration of qualitative reasoning and model-based reasoning techniques developed in Artificial Intelligence (Weld & DeKleer, 1990). Among other techniques, bond graphs, systems decomposition and composition and qualitative reasoning play particularly important roles in automated modelling. The roles of these three techniques are discussed in more detail below.

Bond graphs, developed in mechanical and electronic engineering, are used to represent models of dynamic systems. Bond graphs are a concise and uniform language for the description of engineering systems and use a standard set of physical entities to describe physical phenomena in electronic, mechanical, hydraulic and other energy and power conservative domains. Compared with other representations such as block diagrams and signal flow diagrams, bond graphs are more powerful in that they keep structural information and can be transformed into other more detailed representations. As a result, bond graphs can be generated automatically from system structures or system schematic diagram descriptions (Karnopp, 1988; Breedveld, 1988).

System decomposition and co-ordination methods formulated in systems engineering are utilised to support the process of automatic modelling. A list of primitive components are declared in a knowledge base, which determines the kinds of structure which can be modelled. When a structure is provided by users, it is decomposed into substructures connected either in series or in parallel. These substructures are then further decomposed until they can all be modelled by the primitive components.

Qualitative reasoning from artificial intelligence is introduced as a form of problem-solving. It attempts to relate the concepts of function, structure and behaviour of real systems, In qualitative reasoning a structural model is manipulated symbolically to show insight into a real system and identify critical values and important states of the system. The high-level conclusions reached, the causalities captured and many other conceptual issues are essential to the automatic process of modelling and for the purpose of building good models.

1.2 General discussion

It is generally accepted that three stages of work are involved in the model-based approach to analysing a system. Firstly, a model of the system is built. Secondly, a solution is solicited from the model. Finally, a conclusion about the system is reached based on the interpretation of the solution. The importance of using a good model is obvious because building a model is the starting point in the whole process of problem solving. Using a good model can aid problem solving while using a bad model may lead to inefficiency and even invalid solutions.

A model is a distillation of all relevant attributes and principles inherent in a system. It includes information on the perspectives, approximations and abstractions which are used in the building of the model. Formulating the perspective of a system involves collating and refining the set of objects and quantities which are relevant with reference to the behaviour of a physical system. Detailing the approximations of a model involves specifying the required precision and accuracy for relevant objects and quantities. Abstractions are concerned with representing systems and subsystems at appropriate levels of detail. A model is a good model if the perspectives, approximations and abstractions used are suitable for an application.

Simulation, automatic control, qualitative reasoning and fault diagnosis are all model based studies and require the use of good models. Traditional models are constructed by heuristics and are then used in experiments to ensure results are acceptable. Models produced in this manner tend to cover everything, including issues irrelevant to an application, and require large scale computation. They can be very inflexible and are not reusable even for a slightly changed application. They may need more than the required information like unnecessary details and irrelevant variable values to execute a model to answer to query or solve a problem. All these issues indicate the need for new approaches to modelling, based on more rigorously defined modelling processes. These well-defined processes record major intermediate changes together with their underlying conditions explicitly and make them available for examination when necessary. Automated modelling is such an approach, and is built on the advances of techniques in the fields of model representation, symbolic manipulation, artificial intelligence and several other related subjects.

Automated modelling requires systematic studies of underlying modelling issues including modelling methodologies, strategies and management. It needs explicit representation of the modelling knowledge behind all intermediate modelling processes. Several forms of qualitative

reasoning such as conceptual and causal analysis will need to be applied for the determination of the model structure and for parsimony. Automated modelling attempts to generate models which are parsimonious, making needed distinctions apparent and aiding problem solving.

2 Modelling studies

There are still no formal definitions for models although researchers have given many informal descriptions of models for different tasks. Different fields and applications use different models. In some applications models need only have a superficial representation of major features of real systems, while in other areas models need to describe the underlying issues as well as the surface. For some purposes heuristic models are adequate, while for others models need to be formally defined and well formulated.

Despite the lack of formal model definitions, many informal guidelines on the contents and construction of models are agreed across different fields. One example of this informal understanding is that there is no universally best model. Best models are domain-dependent and task-related. They must be associated with applications; applications are the purpose for the models being built in the first place. The best model should be a representation of a real system which is most appropriate for the task and can aid problem solving. Clearly, to construct the best model, the details of an application must be considered.

A model is an appropriate representation of an object for a certain purpose. This representation could be a simple notation of a real system to describe results observed externally, or it could be a very complicated description of a real system from one or several dimensions. For general simulation and prediction purposes, a model could be a collection of simple equations. To explain how a system works, a model needs to describe the system closely and to represent the major causal relationships at an appropriate level. For design, evaluation, fault diagnosis and other applications which involve studies of the underlying principles and need deep understanding, a model must describe closely not only the real system at several levels of detail and from one or more angles, but also any relevant systems sufficiently.

Models may not have direct impact on the methods and techniques which are applied to solve problems. A simple model may require the use of a complicated method, while a complicated model may lead to solutions with easy manipulation techniques. Although manipulation techniques are separate issues from the construction of models, they can be embedded into models. The more manipulation techniques are embedded, the more procedural the models become. On the other hand, declarative models contain no information on how they should be used.

Models can be represented in many forms. These forms include linguistic descriptions, schematic diagrams, mathematical and logical expressions. Mathematical, symbolic and logical models are the most widely used models in engineering applications and academic work because they are more concise, formal and less ambiguous. Furthermore there are more scientific methods available to study and manipulate them.

Models can be classified into many different groups depending on classification criteria (Zeigler et al., 1979). The differences between groups and the details of criteria indicate the breadth of model studies and the various dimensions from which models can be studied. Models can be built manually or automatically. They can be lumped or distributed models. Their structures can be fixed or changeable during the course of execution. Models can be shallow models or deep models. Automated modelling deals with deep models with variable structure. Models used in automated modelling are all composable from base components and can be generated by computer. The choice of an exact modelling mechanism depends on the characteristics of a domain and the nature of application.

The source of modelling knowledge is another important issue in modelling studies (Leitch, 1995). Strictly speaking, the source of modelling knowledge either comes from empirical observation and experience or is derived from information of a domain such as structure knowledge,

although in practice heuristics and experience and domain knowledge tend to be used in combination. Traditionally empirical knowledge was heavily used in model construction, while in current modelling studies especially in artificial intelligence based approaches, models tend to be generated from domain knowledge through the application of the physical principles and laws pertaining to a domain.

3 Development of modelling strategies

The functions of models vary depending on the stages of model development in history. They can be characterised by three types of model—crude models, shallow but accurate models and well-formulated models. Crude models were used in the primitive stages of modelling and simulation work. They only represent the main features of real systems being studied and were used for the purpose of reference and visual aids. In contrast, shallow models describe real systems mathematically and precisely. They tend to be ambitious and intend to cover all possible ranges so they are likely to be large-scale and complicated. For the particular tasks that it is designed for, a shallow model can be very accurate and robust. Shallow models are normally lumped and are predominantly used for simulation. Shallow models can not be decomposed because any individual component such as an equation may not make sense. Similarly, shallow models can not be assembled into larger shallow models because they may contain contradictory assumptions. Composition of shallow models with contradiction may lead to invalid models. The problem with shallow models is that no mechanism is provided for detection of contradiction. In comparison, well-formulated models are built on the basis of principles and laws behind real systems. They contain more background information about real systems than shallow models. The background information includes the environment and assumptions behind models. Well-formulated models tend to be distributed. Any composition or decomposition of well-formulated models will still lead to valid models. Unlike shallow models, well-formulated models have relevant assumptions imbedded, which provides a mechanism for ensuring consistencies and avoiding contradictions.

In its most general sense, a model is a simplified description of a real-world system that is intended to determine how the system functions without experimenting on the system. Simplification involves making assumptions, approximations and abstractions. It is task related because different tasks require models of different types and different degrees of simplification. In order to simplify the description of a real system it is necessary to make assumptions, i.e. what is important and what is negligible, what is relevant and what is irrelevant. Approximation has to be applied to control computational complexity and abstraction has to be made to aid understanding, enhance generalisation and provide knowledge.

At the starting stage of crude modelling, models were made to reflect the major features of real systems like models of new design from original ideas and normally involved only a couple of variables. They could be crude, vague and at a high level are were associated with simple manipulation techniques. Because crude models were often used for reference and simple analytical tasks, accuracy and detail of solution were probably not essential. Information on assumptions, approximations and abstractions made was not stringently applied so its explicit representation was not necessary. Normally people who produced models were the people who applied them. They knew, probably unconsciously, the underlying simplifications and the subsequent limitation on the use of the model and could therefore use them appropriately. Because the applications that models were employed for were reasonably small scale and the tasks not complicated, there was not much need for formal specification of what was simplified. Examples of this type of model are scale models, ICON models, linguistic models and simple mathematical models. The main motivation was that models were less costly and less time-consuming to construct and study than experiments with real systems. Models were mainly used for references and for facilitation of understanding and communication.

As model-based approaches were more widely adopted and applications became more complicated in both scale and complexity, especially in safety related areas, people started to demand

complete, reliable and accurate models. Models were produced largely from experiences and heuristics and strenuous experiments were carried out to validate models to required standards. Models at this stage could be characterised by being shallow but accurate because model-based performances were the main concern in model building. A model was accepted so long as the performance based on it was satisfactory. The underlying knowledge or principles behind models were less emphasised. Often they were studied only when they had great effect on performance. Shallow models were normally very large-scaled and contained large numbers of variables. They tended to be ambitious and covered all conceivable situations. Examples of models in this category are those of nuclear power stations and large economic systems. Models were used to study systems which were impossible to build or with which direct experiments were either dangerous or costly. Shallow models were often intended for design evaluation or performance prediction to assist decision making.

Shallow models work very well for tasks which they have been validated for, but not necessarily so well for other tasks which are not covered in validation. These other tasks can include even the tasks which are similar to but simpler than the tasks shallow models are validated for. Shallow models can normally only be built manually, which can be very time consuming. They are difficult to reuse for different systems. In addition, the model complexity and large coverage often require prohibitive amounts of computation. Finally, shallow models fail to catch the underlying principles of real-world systems and therefore can not be used for tasks which need insight into systems.

Recently, well-formulated models have been proposed to deal with some of the weaknesses associated with shallow models. They are built on many new ideas and problem solving methods put forward in modelling, simulation, model-based reasoning, control, process monitoring, fault diagnosis and other closely related areas. The work in these areas all demands the use of good models, which are appropriate, flexible, reliable, simple but adequate for the tasks under study. Furthermore good models must describe all the principles and laws imbedded in real world systems and explicitly record all background information behind a model such as relevant assumptions. Well-formulated models are widely used by researchers in different fields to achieve good models. They are based on a framework which is well-defined for the formulation of models so they can be automatically constructed from other sources of information as well. A good example of well-formulated models is the deep models used in qualitative reasoning. Models in this category are often used in applications such as to explain how systems work, as well as why systems work.

4 Automated modelling

4.1 Approach

Automated modelling is an investigation of the modelling process with the objective of developing computer tools to generate well-formulated models from structural information automatically. These computer tools will follow generic modelling principles in the construction of models and use the details of an application to produce the simplest but sufficiently adequate model for the application. Automated modelling will address all relevant issues of modelling, including the representation of domain knowledge, formulation of generic modelling principles, model manipulation and model management, representation of structural information and application. The basis of automated modelling is the introduction of well-defined models and well-constructed applications so that models and applications can be related. The approach is to employ a continuous range of models at multiple levels which start from structural information and finish with a computational description. An automated modelling system will automate these processes of model change and transformation and offer appropriate support for users to provide the additional information required for models to be changed from high-level conceptual models to low-level details.

To achieve complete automation in modelling, there are two particularly important requirements, a good library or knowledge base of submodels for elementary and base components and a

set of generic modelling principles (Xia, 1994). They need to be provided in advance of any modelling activity. The knowledge base will provide a source of information on submodels for modelling while generic modelling principles are formulated and represented formally to guide the construction of models (Xia, 1994).

When building a knowledge base for modelling it is important to describe rules generally and completely. Assumptions will have to be made explicit, together with their applicable conditions. This can be summarised as the generality principle and the completeness principle. The generality principle stipulates that components must be described generally in a knowledge base and their descriptions must be applicable to all systems in which the components are used. If the description of components are particular to a certain application, it is impossible to build a general knowledge base for all applications. This renders it difficult to automate the modelling process as different applications may require quite different descriptions of components and hence, different knowledge bases. Therefore a knowledge base which is context-free and independent of certain applications is essential to automatic modelling. The completeness principle means that all possible components must be declared in a knowledge base. If one component is not declared and a system uses the component, the model of the system cannot be generated in an automatic manner. Although this seems to contradict the focused and specific nature of traditional expert systems, our modelling system would not flounder and generate a large search space. This is because rules are declared clearly with their applicable conditions, therefore the number of applicable rules will be limited. Further qualitative reasoning will also help to reduce the search space through using high-level concepts and ignoring irrelevant low-level details.

Building a good knowledge base is important, so is the use of a list of well articulated modelling principles. When a human expert builds a model for a system many standard practices are observed. These standard practices implicitly reflect what should be considered, what should be ignored and what constraints should be met in building a model. In automatic modelling, these practices have to be explicitly declared as generic modelling principles in order to generate sensible models. Two examples of generic modelling principles are the Similar Complexity Principle and Order-of-Magnitude Principle. The Similar Complexity Principle requires that when modelling a system all components in the system should be modelled in similar detail. This principle is violated if one component is modelled at great depth while another, equally important, component is modelled at a shallow level. With the Order-of-Magnitude Principle, variables which play equally important rules in a system are classified into groups and when one variable is considered in a model, all the other variables which belong to the same order-of-magnitude group must also be considered in the model. Similarly, if one variable is neglected, then all the other relevant variables must also be neglected.

4.2 State of the art

It has been generally agreed that the theory of automated modelling will be built based on the integration of all related modelling activities. A framework of this theory will address the underlying relationships among the modelling environment, principles and laws, modeller's expertise, observation and application. A parsimonious model will be an overall consideration and compromise of all these attributes. In this framework, the modelling environment will provide a basis for modelling and offer a boundary for modelling activities. Anything inside this environment is relevant and anything outside is irrelevant. Principles and laws are the knowledge behind the environment and they are the principles to be applied in the process of modelling. A modeller's expertise describes the principles that must be applied. Observation specifies the necessary conditions that a resulting model must satisfy. Application is the purpose for building a model. It can be a query or a task which contains information on the relevant perspectives, approximations, assumptions and even important phenomena, variables and parameters. Although this theory of automated modelling still needs to be instantiated with a mathematical description, researchers have started to use it as an informal guide on further work.

In contrast to the limited theoretical work on automated modelling, encouraging progress has been made in practical systems development and implementation work. Several prototype computer tools have been implemented which perform modelling autonomously or with minimal external intervention. Although they worked differently because they were implemented in different domains and for different applications, these systems all support the process of modelling and perform model quality checks for automatic model improvement. Although they have not addressed completely the issues raised in the theoretical framework, these practical applications have demonstrated that automated modelling is not only beneficial, but also possible. The success of these practical applications will expedite the development of theoretical work.

Automated modelling work is still a new research topic and the progress of theoretical and practical development work is limited and pioneering. The objective is clear, but researchers have not yet agreed on the best way the objective is to be achieved. Previously the emphasis was on developing application-related modelling tools while many theoretical issues were ignored. As tools gradually become more complicated and ambitious, the importance of a theory becomes more obvious. A theory is needed to guide the design and development of large-scale automated modelling tools. As a result, theoretical development is now receiving as much attention as practical work and is expected to increase.

5 Themes in modelling

There are many issues involved in the task of automated modelling, and there are many solutions to each of them. In this section, we describe the more important of these issues and discuss some of the solutions that have been put forward. In section 6 we look at these issues from another perspective. A variety of modelling systems are described in order to show how the different solutions to the problems combine to form working modelling systems.

The structure of an automated modeller depends on many things, including the task the models are for, the strategy and methodology used, the representation of the model, and the criteria used to detect an adequate model. Most modelling systems have been developed to test ideas and theories and as such do not attempt to create models for all types of tasks for all classes of systems. Instead, like the models they create, the automated modelling systems are built for specific functions and uses. This means that there is very little common ground between automated modellers, as will become apparent in these next two sections.

5.1 Modelling tasks

It must be realised that the modelling process is essentially independent of the precise way in which the model is used (Xia & Smith, 1994). Instead, the form of the model depends on the type of task that is to be performed. With a few exceptions, most automated modellers create models for one of two tasks: simulation and explanation. Simulation is a popular task for which models are created because it is the task for which formal models have traditionally been created. As such, there is both a wealth of domain expertise available and simple and objective tasks available for the model, namely performing the simulation. Early modelling systems (e.g. GoM and CM) focused on simulation, often producing quantitative models while still retaining a very qualitative model-building procedure. Later systems also tend to include some form of simulation in the form of a behavioural analysis of the system being modelled.

Explanation, usually in causal terms, is a popular task and is something that qualitative reasoning and physics excels at, and modelling has grown out of that discipline. The models created for explaining how physical systems work tend to take a different form to those produced for simulation, as most explanations are couched in qualitative and, more significantly, causal terms. Other systems do perform other tasks, but generally they contain a component of one of these approaches.

5.2 Strategies

The "holy grail" of modelling is to produce the *parsimonious* model, the simplest model that is capable of performing the required task. There are essentially three approaches to achieving this: to create the parsimonious model directly, to create an over-complex model and then simplify it, or to produce a simple model and then add detail to it. It may seem logical to take the first option, identifying requirements and creating an optimal model. This is similar to the approach taken in the CM system (Falkenhainer & Forbus, 1991) but as de Vries and Breedveld (1992) point out, this is an unachievable goal in practice. Instead, the parsimonious model has to be achieved by an iterative process, gradually refining an initial model until the parsimonious model is reached. With these approaches, the initial model is created to be either too complex or too simple.

Most modellers start from an over-simple model which is augmented until it is sufficiently complex to perform the task. A few, such as that produced by Nayak (1994), and arguably TRIPEL (Rickel, 1994), start with an over-complex model that is simplified until it is no longer adequate. Dealing with the latter case first, these modellers start by producing (or having produced for them) a highly complex model of the system that contains all the detail necessary to perform the required task. However, in most cases much of this is unnecessary and only serves to reduce the efficiency and utility of the model. In order to improve these qualities, the modeller selects parts of the model that are not relevant to the task in hand and either removes them from the model or replaces them by simpler alternatives. This process continues until no more simplifications can be performed.

The other method is to start with an initial simple (perhaps null) model, which the modeller then augments until it is adequate to perform the task specified. The key to this approach lies in detecting how the model fails; and central to this is the role of model satisfaction criteria discussed below. Once the model has been assessed as inadequate, the modeller must ascertain the way in which the model is inadequate and then decide on a correction that can be made that will improve the model with respect to that failing. Often, causal arguments are used to identify which parts of the model to modify.

5.3 Model adequacy checks

Despite its importance in the development of models, the subject of model adequacy checks has received little formal attention in its own right. The major problem faced by both machine and human modellers lies in detecting when a model is adequate for a task. The best test for adequacy is to use the model for its intended task: if the model works, it must have been adequate. This forms the basis of the adequacy test for many automated modellers, particularly the GoM (Addanki et al., 1991) and MM system (Amsterdam, 1993). In this system, the model is used for a simulation of a mechanical system and the results of that simulation are compared to empirical data. Significant differences between the two prompt the GoM system to switch to another model. However, in many cases, this approach is not viable due to cost or sheer impossibility. Instead, other criteria need to be used.

Apparently, the simplest alternative criterion is to check the internal consistency of the model. That is the approach used by CM and relevance reasoning (Iwasaki & Levy, 1994), among others. With this method, the component parts of the model contain details about the restrictions on their use as well as their operational descriptions. Once the model has been completed, these restrictions can be checked to see if they are mutually compatible and, in addition, they can be checked against the assumptions under which the model was built to ensure the model does not violate any of them. As well as behavioural and other restrictions, the model fragments can contain restrictions based on causal arguments. For instance, if an electric motor is to turn a shaft, it must have an electric current imposed on it. These restrictions differ from those simply relating the types of component models' ports in that there is an element of causality involved. Although this technique is powerful, there is a problem with it in that the restrictions on all the model fragments must be defined before they are used, and there has been little work on how this is to be done with the result that most modellers using this method define the restrictions in a fairly *ad hoc* manner.

The most sophisticated method of checking models is that presented by KEMS (Xia et al., 1993). In this system, the criteria the model must fulfil are defined explicitly in terms of a set of modelling principles. The models produced by KEMS should satisfy these principles as well as others specific to the modelling task. This use of general criteria makes the behaviour close to that of a human expert. However, the task-specific criteria must be set up so that any model which satisfies them is adequate for the task. Establishing what these criteria are seems a promising avenue of research.

5.4 Methodologies

There are two stages to creating a model: determining the model boundary and deciding on the best representation of parts within that boundary. The former task is more subtle than it first appears: the model boundary determines not just the physical extent of the model, but also the way in which the various parts of the system are to be treated (Falkenhainer & Forbus, 1991; Amsterdam, 1993). For example, should a pipe be considered something that carries water, a mechanical link, or an electrical conductor? In other words, it is concerned with determining the *scope* and *domain* of the model, using the language of modelling dimensions put forward by Weld (1992). In some automated modellers, the task of determining the physical extent of the model is implicit in the description of the system to be modelled, as it is assumed that the entire system is to be represented in the model. But even in these cases, the modeller still has to determine the correct treatment of all parts and in all automated modellers this forms a significant part of the work being done. The second task, that of finding the best representation of parts within the model, is normally solved by one of two techniques discussed at the end of this section and concerns the selection of the model's *accuracy* (after Weld, 1992).

5.5 Finding the model boundary

In order to determine the model boundary three main approaches have been used. The Graph of Models system (Addanki et al., 1991) essentially avoids the issue by selecting between models that have been created by human experts prior to the GoM being used. Using a human expert to create the models ensures that only sensible model boundaries are available to be chosen by the modeller. The second approach to determining the model boundary is taken by the CM system (Falkenhainer & Forbus, 1991). Here, the system to be modelled is represented in a hierarchical manner, with all components in the system being made up of assemblies of smaller parts. This is used to determine the physical boundary of the model. When several parts are considered as being of interest, CM defines the *minimal covering system* (the physical model boundary) as being the smallest sub-tree of parts that contained all the components of interest. Within this physical boundary, the treatment of the components is determined by causal arguments.

Causality is the most popular method for determining all aspects of the model boundary. Various heuristics are used to determine how components are to be treated as well as which components should be included in the model. The simplest of these states that if a certain effect (e.g. an electric current flowing) occurs at one point then everything that neighbours this point should be modelled so that its response to that effect can be determined. For instance, a metal pipe connected to an electric current should have its electrical properties considered in the model. So too should a wooden pipe under the same conditions—its behaviour might be non-existent, but it must be *shown* to be non-existent. Other heuristics will embody more apparently causal arguments, e.g. electrical effects at one end of a wire imply electrical effects at the other end which will in turn have effects on other components.

5.6 Choosing models of components

Once the boundary of the model has been selected, there remains the task of selecting how each part of the system that is relevant should be modelled. The investigation of the model boundary

identifies the manner in which the system is to be modelled. In most modelling systems, there are several alternatives for the way a particular component can be modelled. These are sometimes referred to as *model fragments* and often differ in the accuracy with which they portray the component. For example a simple model of the electrical properties of a battery is as a perfect voltage source; a more complex model would include the battery's internal resistance and charge decay. The selection of which particular model fragment to use in a particular situation is a thorny problem, addressed by several researchers such as Iwasaki & Levy (1994) and Nayak (1994).

5.7 Model representations

Models are represented in many ways, depending on the type of models generated and the tasks for which they are intended. As with traditional manual modelling techniques, the most popular vehicle for representing a model is a set of mathematical equations, usually without any causal information. Using equations has the advantage of there being a great deal of existing expertise in their solution and applicability. On the other hand, the notation of equations is fairly sparse, and much of the contextual information about how the equations were generated is absent. A significant amount of work is required to recreate this information if it is ever needed for the task the model was created for (Nayak, 1994; Iwasaki & Simon, 1986).

Another popular representation for models is that of qualitative processes theory (QPT) (Forbus, 1984). This has the advantage of retaining the causal information that was generated during the model creation which can be used for giving causal explanations. However, QPT is qualitative in nature and this restriction has limited this representation to modellers that generate explanations (Falkenhainer & Forbus, 1991; Rickel, 1995).

The other main model representation is the bond graph (Rosenberg & Karnopp, 1983), which is becoming more widespread in the mechanical engineering community. This shares some of the advantages of a causal representation with QPT's influences while retaining much of the expressiveness of normal equations. For these reasons it is used to represent models of physical engineering systems (Xia et al., 1993; Wilson & Stein, 1992) though the bond graph representation does impose restrictions on how models are expressed, thus limiting its use to these sorts of systems.

6 Automated modelling systems

In this section we describe some of the automated modelling systems that have been developed and discuss their operation in the light of the issues discussed above.

6.1 Graph of Models

One of the earliest automated modelling systems was the Graph of Models (Addanki et al., 1991). The basic idea behind the GoM approach is to have a collection of models of a system from which the most suitable for a particular task is chosen. Each of the models is only applicable under certain assumptions; each model within the graph will have a different set of assumptions. All of the sensible assumption sets will be reflected as a model. These models are placed in a graph and this graph is traversed to identify the most suitable model. This traversal is accomplished by labelling each edge to indicate which assumptions change when moving between models. When a problem is posed, the simplest model is tried initially to solve the problem. If this model does not adequately solve the problem, the reasons for its failure are analysed and a delta vector is created that indicates which assumptions should be changed to produce a more accurate model. The neighbouring model which most nearly matches the new assumptions is determined, and the problem is then re-evaluated in that model.

The graph is created by hand by an expert in the domain that the graph covers. The first stage in creating the graph is to identify all the relevant simplifying assumptions and combinations of assumptions that an expert would use in analysing problems in that domain. Each of these

assumption sets is noted as a node on the graph of models. Each node is then expanded out to create a full model that will operate under those assumptions. The nodes of the graph are then connected by edges that are labelled with the assumptions that change when moving from one model to the next, and the effects that these changing assumptions will have on certain domain parameters. These are coded as parameter change rules.

In addition to the actual graph, a set of influence nets are created. These store details on how all parameters in the models are related by recording the influences that each parameter has on others. For instance, force has an influence on acceleration and this in turn has an influence on velocity.

Once the GoM has been created, it is used to identify the correct model for a certain situation. This is done by starting with the simplest model (i.e. the one with the most necessary assumptions) and this is used to drive a simulator that generates some results. These are compared with empirical results to see how accurate the model is. Any significant differences are noted with a delta vector, which contains the variable that is to be changed and the direction that it has to change by in order to become in agreement with the empirical results. All variables that are in disagreement are noted. This base list of delta vectors is then expanded using the influence net to include delta vectors for all other variables that are directly dependent on those in the original list. Then, all models that are directly adjacent to the considered model are examined to see if a transition to any of them will result in changes to any or all parameters in the delta list; the most suitable of these is then used to re-analyse the system under consideration.

The GoM approach stresses the iterative nature of model creation by allowing several attempts at the ideal solution to be tried. This means that the modelling of the system can stop as soon as the required accuracy is reached, which results in a parsimonious model. A rich and highly connected graph will ensure that only the complexities that need to be considered are included in the model. However, an expert has to create the initial graph of models by noting all the assumptions that can be made about systems in the relevant domain. In order for the graph of models approach to work successfully, there have to be many nodes in the graph and the level of connectivity needs to be high. If these conditions are not met, the use of this GoM degenerates into a linear progression of models.

The model transitions in the GoM are all based on the comparison of simulated results from the model being compared to empirical results. This can only work for systems which can be run in the real world. Those which would be too difficult to build and too dangerous to run offer little scope for obtaining empirical results and this is a clear limit on the usefulness of the approach. This empirical comparison also introduces the notion of tolerance between the predicted and empirical results. It there was no scope for deviations, the GoM approach would always result in the most complex model possible being used, in order to account for all the fine detail in the real system. This raises the question of how these tolerances are set, for which the paper offers no guidance. In addition, as each model is only appropriate for a particular set of circumstances, the limits on its subsequent domain of applicability have to be identified in order to prevent the model being used in inappropriate circumstances to predict behaviour.

As presented, there is no order of magnitude reasoning about the transitions between models. For instance, if one transition produces small changes in many parameters while another produces large changes in a few parameters, then the latter model should be chosen if the change required is large, but should not even be considered if the required changes are small. However, the model transition mechanism could be easily replaced with a more sophisticated one without altering the fundamental nature of the GoM approach.

The main drawback of the GoM approach is that each graph contains alternative models of a single system. This means that a separate GoM needs to be created for each system that is to be modelled. While this is fine for handling tasks concerned with a single system, it places a great overhead on the ability to model other systems.

In conclusion, the GoM approach is more of a modelling control strategy than a model creation

technique. The graph of models is a mechanism for controlling the transitions from one model to another as opposed to a method of creating models for solving problems. There is no method for creating the models themselves. Once the models have been defined, the data that prompts the transitions is entirely externally generated, and the user decides upon the significance of differences between predicted and experimental results.

6.2 Compositional modelling

The other early and influential automated modelling approach is Compositional Modelling (Falkenhainer & Forbus, 1991), a system developed to generate qualitative and quantitative answers to questions posed about complex physical systems. This modelling approach is geared towards the generation of one-use models in response to the user's task definitions. It does this by analysing the query and creating an *ad hoc* parsimonious model that would be capable of answering that query. CM is significant due to the large number of modelling systems (Nayak, 1994; Iwasaki & Levy, 1994) that have been developed using the model fragment technology that lies at the core of CM.

The basis of CM is the use of a set of model fragments that are combined to form the final model. Each model fragment represents a component under certain conditions. There is a different model fragment for each set of conditions, and each fragment contains all the information needed for its use in a model. For instance, a pipe could have model fragments relating to its ability to carry fluid, its mechanical stiffness, its electrical properties, etc. The process of fluid flow through a pipe will be described by several model fragments: one for inviscid flow, one for laminar flow, turbulent flow and so on. These fluid flow fragments are grouped together into an assumption class, which means that if fluid flow is to be considered for a pipe, then one (and only one) fragment from the fluid flow assumption class must be chosen to model it. Generally, any given assumption class is available to many different component types.

Before CM is able to model a system, there are two independent processes that need to be performed. The first is to analyse the system to be modelled. This analysis will identify the component parts of the system and the possible processes that can occur in that system. This analysis is recursive, proceeding until a suitably fine level of detail has been reached.

The second process is to create (or import) a domain theory that allows these processes to be modelled. The domain theory consists of several domain model fragments, each of which provides the method of modelling a particular process under a certain set of assumptions. These assumptions state the physical environment that the model fragment is applicable to (for instance, a fluid path must exist before fluid flow can occur), as well as the operating assumptions that limit its relevance to a certain set of conditions. Each process will have at least one model fragment associated with it, though most processes will have many to accommodate different operating states.

Once this has been done, CM can start to process the user's queries. CM takes the user's query (expressed in a formal query language) and analyses it in order to determine the components and processes in the modelled system that are of interest. The list of components is then expanded (using the strictly defined system structure) to include all components relevant to the query;CM then attempts to match processes to these components. CM will regard the conditions that define the domain model fragment's applicability and will use the simplest set of fragments to model the components and processes that can answer the query. At this stage, it might be possible that the currently identified model is still insufficient; this is detected by conflicts between model fragments and if this occurs the model is refined until the internal conflicts are resolved. Once this is the case, the query is answered either by using QP theory or quantitative differential equations.

As only a single model is generated at a time, it is important to ensure that this model is a parsimonious one. This starts with the act of using the user's query as the starting point of the model

composition, thereby focusing the model generation process on those parts of the system that are of relevance to the user. The system structure is then used to determine which other parts of the system are relevant to the parts and processes included in the query.

It is interesting to note that with the introduction of multiple model fragments for each component, the notion of class-wide assumptions (de Kleer & Brown, 1984) has been significantly weakened. No longer is it the case that each instance of a component will be modelled in the same way, as advocated by de Kleer and Brown, instead, this principle is carefully broken by explicitly stating in general terms the environment in which a specific model fragment can operate, a more reasonable assumption (Keuneke & Allemang, 1989). This is carried further by allowing each section of the model to have different assumptions and hence different model fragments being applied. For instance, one fluid flow might be treated as inviscid while another flow of the same fluid in the same model is treated as viscid and turbulent.

The domain theories are created independently of the structural decomposition of the system. This allows the same domain theory, in the shape of a set of conditional model fragments, to be used and reused in the analysis of several different systems. However, the CM approach relies on the system being modelled being fully analysed and decomposed before any modelling begins. The way the system is broken down strictly limits the perspectives on the system that can be used in the answering of queries and also limits the levels of abstraction and detail that can be reasoned about by the CM system. In addition, there exists no methodology behind the decomposition of the system; this means that every new system must be decomposed by hand, with all the attendant possibilities for error. There is also no methodology for the creation of the domain theory, in the shape of the set of model fragments used. This is a significant failing, especially considering the central role that model compatibility plays in determining model adequacy.

In summary, CM shows the way towards how automated modellers should develop. It has a clearly defined separation between domain knowledge and system structure and this should be maintained in all future automated modelling systems. However, CM's major failing is the lack of any definite and explicit tests to determine when a parsimonious model has been generated. The test used, identifying conflicts between model fragments, is too reliant on the formulation of model fragments to detect all conflicts.

6.3 Causal approximations

The research into causal approximations (Nayak, 1994) has extended and generalised the use of model fragments. This was done in two main ways: by defining an ordering on the different model fragments that can represent a component or process, which in turn allows the space of all models to be ordered; and by defining a method for efficiently solving the equations that make up a model. The models generated are intended for producing causal explanations of mechatronic systems.

The model is constructed from model fragments. Each fragment consists of a set of equations that relate certain parameters to each other. As in CM, each fragment is a partial description of a component's behaviour. The model fragments are organised into assumption classes, which intuitively are different partial descriptions of the same behaviour at different levels of complexity. For instance, three models of a wire (perfect conductor, perfect insulator, and resistive conductor) would be organised into a single assumption class. Fragments within an assumption class are mutually contradictory, i.e. they cannot both be present in the same model with the same parameters. Fragments are also partially ordered within a class by the approximation relation (more of which below); the ordering defines a single most complex fragment within an assumption class, but not necessarily a most simple fragment.

The ordering of model fragments by simplicity (and hence the ordering of models), as well as the ability to efficiently solve the equation sets generated, comes from the process of causal ordering. Each model fragment identifies a set of parameters that can be causally determined by equations in that fragment. This identification is domain dependent and is *a priori* knowledge. This is defined for a complex set of equations (a complete set is one where there are as many equations as there are

parameters and there are no subsets with more equations than parameters) by determining the single equation that fixes the value of each parameter. This can then be used to find a causal ordering that shows which parameters go to causally define which others: the goal of the modelling process. The causal ordering is used to define the order in which the equations are to be solved, which changes the equations satisfaction problem from being in NP to P. This form of causality was first proposed by Iwasaki and Simon (1986), and is rather similar to the causality found on block diagrams, where again sets of equations have a definite order to their processing.

In addition, it is shown that selecting the simplifications to make an initial, complex model to produce a parsimonious one is also in general intractable because the space of possible models is unordered by overall simplicity. Causal approximations are introduced to relieve this problem. A causal approximation is a relation between two fragments within an assumption class. One fragment is a causal approximation of another if, when the simpler fragment is replaced by the more complex one, all the variables in the simpler fragment are retained and there is no change to the causal ordering of any variable present anywhere in the simpler model. In other words, making a causal approximation means simply removing part of the causal ordering graph and making some variables exogenous. Most real-world approximations are found to fit this criterion. Due to the fact that a simpler model consists of a strictly smaller causal ordering, there is now an ordering on models (known as the upward failure property). This shifts the emphasis of determining simplicity from being a global property of the model as a whole to being a local property that can be determined by the examination of each model fragment individually.

An initial model is created by identifying all the assumption classes that could be present in a model (generated by using modelling heuristics) and selecting the most complex model fragment from each assumption class. From this starting point, an algorithm for producing parsimonious models is derived. Firstly, all the model fragments in the initial model are simplified by using causal approximations while ensuring that all the assumption classes remain in the model. This procedure finishes when no adequate (i.e. complete and coherent) model exists which is simpler than the final model that contains any simplification of any of the fragments in the model. The causal approximations ensure that the effects of any simplification are local to the fragment being simplified. The second and final stage is to remove all the unnecessary assumption classes (i.e. phenomena) from the model. The result of this is the parsimonious model.

This modelling algorithm is adequate as far as it goes. The concept of causal approximations yields two important results in making tractable the generally intractable problems of selecting which simplifications to make in a model and which order to solve equations. However, it is heavily based on a mathematical formulation of the modelling program and exclusively uses mathematical techniques to derive a parsimonious model. While this serves the purpose, it is somewhat distant from the ideal of qualitative physics, which is to be able to use common-sense assumptions in the derivation of models and behaviours. Once physical components and processes have been converted to sets of equations for insertion into model fragments, there is no subsequent reference back to the physics that prompted the equations. In addition, the causal ordering expounded by Nayak solely determines the order in which equations are to be solved; it has little bearing on the notion of causality understood by most people (Bunge, 1979; de Kleer & Brown, 1984).

6.4 MM

The MM modelling system (Amsterdam, 1993) is a behaviour-based model generator. MM is fed with a "structural" description of the system to be modelled and the required behaviour that the model must exhibit. The structural description can be either geometrical, component-based, or a combination of the two. The behaviour definition is qualitative in nature (referring to a modified QSIM) and specifies the input and output variables and the time-varying behaviour of interest. The model is represented by a bond graph and is used by the modified QSIM program to generate a behaviour trace.

Modelling starts with all the inputs modelled as sources and all the outputs modelled as either bond graph junctions or as the state variables of C or I elements. The physical description of the system is then analysed and broken down into spatial sections so that there is a uniform behaviour within each section. Identifying these sections is called the *lumping* problem. Then, all possible behaviours of these sections are examined, and all links between the inputs and outputs are noted. Once this has been done, the resultant bond graph is analysed and its shortcomings are identified. These come in four varieties: *uninterpreted* (not all outputs are defined in the model), *resistor needed* (there are no dissipative elements in the model), *order too low* (there are too few state variables in the model to account for the stated behaviour) and *wrong behaviour* (all other problems).

The first three of these failures can be detected by direct comparisons between the model generated and the behaviour stated (for instance, the *resistor needed* rule will fire every time a model is generated that does not have a resistance; *uninterpreted* will fire when outputs' values are inadequately defined). If the model passes the first three tests, the bond graph will be used to derive some QDEs that will be fed into a modified QSIM; the resultant behaviour, when compared to the given behaviour, will identify any further failings of the proposed model. Due to the ambiguity of qualitative simulation, this comparison is performed by creating a total envisionment of the bond graph model, and seeing if the given behaviour can be deduced from it (the envisionment is pruned during creation and the behaviour only has to match "qualitatively").

These failings are corrected through the use of a set of modelling rules. These have conditions that relate to the type of the problem identified and the situation in which they are to be applied. For instance, any constriction in a fluid pipe is a candidate for adding an R element to the corresponding part of the bond graph. There are similar rules for adding other bond graph elements, all created as needed to model systems. There is a separate rule stated for each correction that can be added and each situation in which it could be considered.

MM is a task-driven modelling program, which means that parsimonious models are generated. It also starts with a simple model that is augmented as needed to produce this parsimonious one. The main contribution of MM is that there is no definite model for any given part of the system: basic models are defined of different components and then augmented as needed to yield the desired behaviour. However, the main drawback with this approach is the same as with the published GoM system: the criterion the model must fulfil is to reproduce a given behaviour. In MM, as with GoM, if the desired behaviour is known, there seems to be little point in generating the model, since it will only reproduce that behaviour; it cannot be guaranteed to be valid for any other conditions or behaviours. For instance, an over-damped oscillator could be modelled as a zero-order system and so a zero-order bond graph will be generated. But if the damping is removed, it turns in to a second-order system, but this behaviour will not be exhibited by the bond graph. This problem of applying models to other conditions is an open research problem.

6.5 TRIPEL

TRIPEL (Rickel, 1995) is an automated modelling system developed to take advantage of the simplifications and abstractions that are possible through consideration of the differing speeds of processes. A system is set up in the form of an influence graph which contains all the variables in the system and connects them by influence paths to reflect all the effects variables can have on each other. These influences are stated in terms of Qualitative Processes influences (Forbus, 1984): functional relationships are termed indirect influences and differential relations are termed direct influences. Every direct influence is further annotated with the time scale on which its effects first become significant, e.g. seconds or hours.

Like CM, TRIPEL is used to generate causal explanations. A query is defined in terms of the affected variables and the affecting (driving) variables. TRIPEL firstly scans the influence graph to determine the time scale of interest; this is the smallest time scale on which it is possible to link the

designated variables using only direct (i.e. instantaneous) influences and direct influences that act at least as quickly as the given time scale. Once the time scale has been determined, some simplifications can be made, such as assuming the faster direct influences to be instantaneous, and the graph is traversed again to determine the final model. This search is constrained by the time scale chosen and takes the form of a best first search of the influence graph, moving from the affected variables and eventually reaching the driving variables. The boundary of the model is determined by deciding which variables can be deemed exogenous; this is the case when they are found to be unaffected by any other variable in the model. Eventually a complete model, i.e. a set of influences, is found.

The emphasis of TRIPEL is more akin to GoM than AIM: TRIPEL takes an existing set of relations between variables and determines the smallest set of them that can be used to generate a parsimonious model.

6.6 OMLECO

The team working on the OMLECO project (Top et al., 1995) have taken a different approach to the modelling of systems. Instead of creating a modeller that can automatically create a model of a given device or system, they have opted to create a "modeller's assistant" type package, which allows the user to create a model by using a library of standard parts. To model a system, the user defines the structure of the system in terms of the connections between components. The user then has to identify the correct way to model each of these components. This is done by presenting the user with a hierarchy of components that become more specific and detailed as the user moves down the hierarchy. The user is invited to select the most appropriate component for the library. This component description is then represented at a conceptual level by a bond graph fragment. This level is used to understand the physical processes occurring in a component, as opposed to decomposing a component into other physical units as was done at the previous level. The bond graphs can be used to define a set of equations and hence allow simulation to take place.

While the OMLECO work is not an automated modeller in the sense of CM, this approach again highlights the decisions that need to be made to produce a parsimonious model. However, due to the reliance of OMLECO on human interaction to make modelling decisions, the resultant models suffer from the problems of insufficiently defined constraints on the use of the model, in a manner similar to models created entirely manually.

6.7 KEMS

KEMS (Knowledge-based Environment for Modelling and Simulation) has been developed by Xia et al. (1993) as a general-purpose automated modelling system. It takes as input a structural description of a system to be modelled and a set of conditions that the model must obey. The structural description is given in terms of the components that go to make up the system and how they are connected. The modelling conditions are given in terms of the behaviours the model should exhibit in certain circumstances and includes steady state behaviour, transient behaviours and energy domain changes in the model. These are chosen to cover several different conditions, thus ensuring the model has a reasonable scope of applicability.

Each component has models stored in a library. There are many models for each component, each a more complex representation of the simplest one. For instance, an electric motor is initially represented as a GY element. Alternative models of the motor are the same GY element with, say, friction or coil inductance added. There is a fixed range of models available for each component.

An initial model is created by finding the simplest model for each component (expressed as a bond graph) and then combining them to form an initial model. From this, a set of qualitative equations are created that are used for simulating the system under the conditions specified in the problem definition. If the model does not meet the conditions specified, a component is chosen to

be altered and it is replaced with the next most complex model for that component. The model is then reassessed and the cycle repeats until all the model criteria are met.

The main feature that KEMS brings to automated modelling is the explicit description of the criteria that the model must fulfil. This ensures that the model created, if it satisfies those criteria, is guaranteed to be fit for the task in question. As the modelling conditions cover a range of behaviours, the model is guaranteed to be useful in a range of situations. However, there does not seem to be any formal way of defining these conditions; instead, they seem to result from an *ad hoc* process by a human expert.

As it stands, KEMS does not allow for the fact that a component can be used in more than one way—the model library only contains a single type of model for each component. Different conditions on use (i.e. using the motor as a counterweight) are not addressed in KEMS and this is equivalent to making decisions about how each part of the system is to be modelled before starting on the automated phase.

6.8 Relevance reasoning

The concept of modelling using relevance reasoning (Iwasaki & Levy, 1994) is a further extension to CM and Nayak's causal approximations. They use the concept of relevance reasoning (Levy 1994; Subramanian & Genesereth, 1987) to determine what is and what is not relevant to a query. This is implemented in the context of their modelling system by deciding which phenomena (i.e. assumption classes) should be included in the model and what level of abstraction (i.e. model fragments within assumption classes) should be used to represent those phenomena. The modelling process starts with the modelling of the subject of the query by representing it with the simplest applicable model fragment. Any additional assumptions that have to be made to allow this model fragment to be used are added to the model's assumption list. The model fragment is then examined to discover which non-exogenous variables are present in the fragment, and these are processed in turn by including additional model fragments to determine them. This may introduce more variables to the model which will need determining by other fragments. The process repeats until all the variables in the model are either determined within the model or are defined as exogenous. As new model fragments are added, their usage conditions are examined to see if they violate any of the previously postulated assumptions. If that is the case, the offending fragment is exchanged in the model for a more complex model fragment which does not require the assumption in question to be made. The implications of this change are then propagated throughout the system, including noting any new variables that have been introduced.

In this modelling system, the main criterion for including any specific effect in the model is that of relevance or significance. Decisions about which parts of the system to include in the model are made on the strength of what is important in determining the behaviour of a variable of interest stated in the task definition. However, this algorithm can only cope with a single direction of causal interaction from a single variable, since the model is created by backward chaining through the causal interactions from the variable of interest to exogenous variables. Other systems, such as KEMS, allow both that technique and forward chaining from the variable of interest in order to discover what a certain variable affects as well as what it is affected by.

6.9 MODA

Some work on automated modelling has grown out of traditional quantitative engineering disciplines. A representative example of this is the Model Order Detection Algorithm (MODA) for modelling dynamic systems, by Wilson and Stein (1992). This is a quantitative algorithm for model refinement that is used to determine the minimum model order for representing the dynamic behaviour of a mechanical system for a Frequency Range Of Interest (FROI). The system that MODA is embedded in is given a geometric description of a mechanical system, as well as the physical parameters for all the components in the system. The model builder then creates a bond-

Table 1 Summaries

	Type of task performed	*Different models of components?*	*Component/ process/equation centred*	*Model satisfaction criteria*
GoM	Simulation	No—defined models	Model	Behavioural
CM	Explanation	Yes	Process	Model structure
MM	Simulation	No—holistic model	Process	Behavioural
KEMS	Simulation	Yes	Component	Behavioural
Relevance	Explanation	Yes	Component	Model structure
MODA	Simulation	No—holistic model	Equation	Behavioural
OMLECO	General analysis	Yes	Component	Human decision
TRIPEL	Explanation	No	Process	Time scale and model structure
Causal Approximations	Explanation	Yes	Equation	Model structure

	Form of representation	*Modelling strategy*	*Example or case study*
GoM	Equations	Model selection	Mechanical gears
CM	Equations	Bottom up composition	Naval propulsion system
MM	Bond graphs	Bottom up composition	Hydraulics
KEMS	Bond graphs	Bottom up composition	Control systems
Relevance	Equations	Bottom up composition	Electrical system
MODA	Bond graphs	Bottom up composition	Mechanical shafts
OMLECO	Bond graphs	Model classification	Electric motors
TRIPEL	Influence diagrams	Bottom up composition	Plants
Causal Approximations	Equations	Top down composition	Mechatronic systems

graph model of the system, assuming all the components are rigid. MODA then takes each component in turn and considers the compliance of that component and the effects that this new compliance has on the entire system dynamics. The new compliance that has an effect at the lowest frequency of input is then added to the bond graph model and the process is repeated. If a compliance has no effect on the system behaviour at or below the FROI, then that compliance is ignored. As soon as there are no more compliances that can be added that have an effect below the FROI, the model modification stops.

This algorithm is a highly quantitative one. It relies on a strict definition of the FROI as well as precise parameters for all the components. It also applies compliances in strict order of frequency effected. The algorithm makes no distinction between those compliances that will have a major effect and those that will have a negligible effect. The easiest way of converting this rather useful algorithm for qualitative use would be to use some sort of order-of-magnitude test on all the compliances and to add big effects first. The problem then is still to include enough quantitative information so that the frequencies and orders of magnitude can be calculated accurately enough to allow meaningful choices to be made.

7 Underlying issues

Although the existing modelling systems work very differently from each other and support different aspects of modelling activities, as shown in the above table, they share the same principles of automated modelling and face the same technical challenges. The underlying technical issues of automated modelling can be grouped into three areas: model representation, reasoning with models and criteria of model acceptance. Model representation deals with the representation of knowledge required for automated modelling. Reasoning with models is concerned with the

techniques of manipulating models and deriving new information from models. Criteria of model acceptance refer to the conditions for terminating the process of automated modelling.

Model representation is implemented in the form of a knowledge base, which lays a foundation for automated modelling. The completeness, soundness and usability of a knowledge base will directly affect the whole process of automated modelling. A complete and general knowledge base will improve the generality of the automated modelling. Equally important is the formulation of a list of generic modelling principles, which will be used to guide the construction of models from submodels. Other issues of model representation include explicit representation of assumptions, boundary and applicability conditions, representation of function and behaviour in structure, multiple model representation at different levels of detail, abstraction and approximation.

In reasoning with models, the major issues include boundary confinement, model functionality, transformations between multiple models and manipulation on model form and content. Boundary confinement involves causal studies of models to determine what is relevant and what is important. Obviously anything which is considered in a model is more important than anything which is not. Model functionality deals with the background information such as assumptions and simplifications behind a model and outlines when a model is valid and when an analysis is beyond the scope of a model. Model form transformation is a content-preservation transformation where only the forms of model representation change for reasons of higher efficiency and usability. Unlike model form transformation, model content transformation is a content-modifying transformation where the content of a model is changed, i.e. certain attributes are added or removed. In model content transformation, the model form may or may not change.

On the criteria of model acceptance, the main issue is to decide when the process of automated modelling terminates. There are two main approaches. The first approach is a task-independent one in which the process of modelling is governed by a set of modelling rules. So any model generated at the end will be acceptable automatically. The second approach is a task-dependent one in which a list of acceptance criteria relating to a task are provided. Models are generated and tested against the criteria. If a model satisfies all criteria it is accepted. These two approaches work very differently: one is a straightforward process and the other is a generate-and-test method. In practice they are combined and work in cooperation. An example of this mixed approach is that models are generated under modelling principles and are experimented against specified requirements. If all requirements are met then a model is accepted. Otherwise the model will be changed or expanded until the specified experiment performance is achieved.

To summarise, the major technical issues in automated modelling can be grouped into the following three categories. The knowledge representation category includes issues of formulation of generic modelling principles, explicit representation of assumptions, a family of models at different levels of detail, abstraction and approximation, representation of functional information into structure and consistency maintenance of assumptions in different models of the same phenomena. The model based reasoning category involves issues of deciding the boundary, functionality and conducting content preservation and content modifying transformations among a family of models. The model acceptance criteria category includes the issues of requirements and model completeness, which decides when the process of automated modelling terminates.

8 Conclusions and research directions

Automated modelling is a new research field and its objectives are to establish a formal theory of automated modelling and to develop computer tools which perform modelling tasks automatically. Researchers in modelling, simulation, artificial intelligence and various engineering fields are actively conducting research in the field and have implemented several computer systems in different fields and for different tasks. These systems can all support modelling activities and some of them can even perform complicated decision-making in composing and selecting models. This encouraging progress in practice has demonstrated the great potential that automated modelling will have on practical engineering and on the design and development of intelligent engineering

systems. The benefits of automated modelling will become even clearer when more researchers start to work on it and some of the major technical issues are resolved.

In terms of research directions, more effort needs to be made both on theoretical aspects of automated modelling and on the design of automated modelling systems. Theoretical work includes mathematical and logical formulation of formal models, model semantics and possible computational mechanisms. Although a general framework of automated modelling has been laid, the details still need to be completed and elaborated. Although theoretical progress is important to guide practical work, any progress on practical work will have implications on theoretical development. In particular, the current level of efforts on implementing and completing existing automated modelling tools should be maintained. The functionality of these tools can be gradually expanded so that tasks will be performed with a higher degree of automation and a lesser degree of human intervention. Future practical research work on automated modelling can be pursued along two lines. The first line of work applies to applications where tasks are well-defined and complete automation is achievable. Then all the underlying technical issues presented in Section Six above should be addressed. The second line of work relates to applications where modelling activities are supporting activities to other major tasks and complete automation is optional. Then model building, selection and management will be the main research areas.

References

Addanki, S, Cremonini, R and Penberthy, JS, 1991. "Graphs of models" *Artificial Intelligence* **51** 145–177.

Amsterdam, J, 1993. "Automated qualitative modelling of dynamic physical systems" *PhD Thesis*, MIT AI Lab, January.

Bunge, M, 1979. *Causality and Modern Science* Dover (Third edition).

Collins, A and Gentner, D, 1983. "Multiple models of evaporation processes" In: *Proceedings Fifth Meeting of Cognitive Science Society-83* Lawrence Erlbaum. (Also in Weld & de Kleer, 1990, pp 508–512.)

De Kleer, J and Brown, JS, 1984. "A qualitative physics based on confluences" *Artificial Intelligence* **24** 7–83.

de Vries, TJA and Breedveld, PC, 1992. "A model of the modelling process" In Breedveld & Dauphin-Tanguy (eds) *Bond Graphs for Engineers* 301–312.

Falkenhainer, B and Forbus, K, 1992. "Compositional modelling: finding the right model for the job" *Artificial Intelligence* **51** 95–143.

Forbus, KD, 1984. "Qualitative process theory" *Artificial Intelligence* **24** 85–168. (Also in: Weld & de Kleer, 1990, pp 178–219.)

Iwasaki, Y, and Simon, HA, 1986. "Causality in device behaviour" *Artificial Intelligence* **29** 3–32. (Also in: Weld & de Kleer, 1990, pp 631–645.)

Iwasaki, Y and Levy, AY, 1994. "Automated model selection for simulation" In: *Proceedings AAAI-94* 1183–1190.

Keuneke, A and Allemang, D, 1989. "Exploring the no-function-in-structure principle" *J Experimental and Theoretical Artificial Intelligence* **1** 79–89.

Leitch, R, 1995. "Model properties and classifications" In: *Workshop on Taxonomies of Models for Simulation and Diagnosis*, 27–28 March 1995, Manchester, UK.

Levy, AY, 1994. "Creating abstractions using relevance reasoning" In: *Proceedings AAAI-94* 588–594.

Nayak PP, 1994. "Causal approximations" *Artificial Intelligence* **70** 277–334.

Rickel, J and Porter, B, 1994. "Automated modelling for answering prediction questions: selecting the time scale and system boundary" In: *Proceedings AAAI-94* 1191–1198.

Rickel, J, 1995. "Automated modelling of complex systems to answer prediction questions" *PhD Thesis*, AI Laboratory, University of Texas at Austin.

Rosenberg, RC and Karnopp, DC, 1983. *Introduction to Physical System Dynamics* McGraw-Hill.

Subramanian, D and Genesereth, MR, 1987. "The relevance of irrelevance" In: *Proceedings IJCAE-87* 416–422.

Sussman, GJ and Steele, GL Jr., 1980. "CONSTRAINTS—A language for expressing almost-hierarchical descriptions" *Artificial Intelligence* **14** 1–39.

Top, J, Breunese, APJ, Broenink, JF and Akkermans, JM, 1995. "Structure and use of a library for physical systems models" In: *Proceedings ICBGM95* 97–102.

Weld DS, 1992. "Reasoning about model accuracy" *Artificial Intelligence* **56** 225–300.

Wilson, BH and Stein, JL, 1992. "An algorithm for obtaining minimum-order models of distributed and discrete systems" In Falkenhainer, B and Stein, JL (eds) *Automated Modelling, DCS* **41** American Society of Mechanical Engineers, 37–46.

Xia, S, Linkens, DA and Bennett, S, 1992. "Integration of qualitative reasoning and bond graphs: an engineering approach" *J IMACS Trans.: Special Issue on Bond Graphs for Engineers* March.

Xia, S, Linkens, DA and Bennett, S, 1993. "Automatic modelling and analysis of dynamic physical systems using qualitative reasoning and bond graphs" *Intelligent Systems Engineering* **3**(2) 201–212.

Xia, S, 1994. "Formulation of generic principles of modelling industrial systems" *Int J Systems Analysis, Modelling and Simulation* **15** 283–291.

Xia, S and Smith, N, 1994. "Mixed quantitative and qualitative simulation of dynamic systems" In: *Proceedings of the European Simulation Symposium (ESS-94)* 230–234.

Zeigler, B, Elzas, M, Klir, G and Oren, T, 1979. *Methodology in Systems Modelling and Simulation* North-Holland.

Further reading

Amsterdam, J, 1992. "Automated modelling of physical systems" In Falkenhainer, B and Stein, JL (eds) *Automated Modelling, DCS* **41** American Society of Mechanical Engineers, 21–30.

Bobrow, DG, 1984. "Qualitative reasoning about physical systems: an introduction" *Artificial Intelligence* **24** 1–5.

Bobrow, DG, 1993. "Artificial intelligence in perspective: a retrospective on fifty volumes of the *Artificial Intelligence Journal*" *Artificial Intelligence* **59** 5–20.

Chandrasekaran, S, 1994. "Models versus rules, deep versus compiled, content versus form" *IEEE Expert* April, 75–79.

Collins, JW and Forbus, KD, 1987. "Reasoning about fluids via molecular collections" In: *Proceedings AAAI-87* Morgan Kaufmann, 590–595. (Also in: Weld and de Kleer, 1990, pp 503–507.)

Cornet, A and Lorenz, F, 1988. "Equation ordering using bond graph causality analysis" In: *Proceedings 12th World Conference on Scientific Computation (IMACS-88)* **1** 43–46.

Dague, P, Raiman, O and Devès, P, 1987. "Troubleshooting: when modeling is the trouble" In: *Proceedings of AAAI-87* Morgan Kaufmann, 600–605. (Also in: Weld and De Kleer, 1990, pp 435–440.)

Davis, R, 1984. "Diagnostic reasoning based on structure and behaviour" *Artificial Intelligence* **24** 347–410.

Davis, R, 1990. "Diagnosis via causal reasoning: paths of interaction and the locality principle" In: *Readings in Qualitative Reasoning about Physical Systems,* Weld, D and DeKleer, J (eds) Morgan Kaufmann, 535–541.

DeKleer, J, 1977. "Multiple representation of knowledge in a mechanics problem-solver" In *Proceeding IJCAI-77* Cambridge, MA, 299–304.

DeKleer, J, 1984. "How circuits work" *Artificial Intelligence* **24** 205–280.

Dormy, JL and Raiman, O, 1988. "Assembling a device" *Proceedings AAAI-88* 330–335. (Also in: Weld and de Kleer, 1990, pp 306–311.)

Eagar, JW and Musen, MA, 1994. "Graph-grammer assistance for automated generation of influence diagrams" *IEEE Trans Systems and Cybernetics* **24**(11) 1625–1642.

Falkenhainer, B, 1993. "Ideal physical systems" In: *Proceedings AAAI-93* 600–605.

Falkenhainer, B and Forbus, K, 1988. "Setting up large scale qualitative models" In: *Proceedings AAAI-88* Morgan Kaufmann, 301–308. (Also in: Weld and De Kleer, 1990, pp. 553–558.)

Falkenhainer, B and Forbus, KD, 1990. "Setting up large scale qualitative models" In *Readings in Qualitative Reasoning about Physical Systems* Weld, D and DeKleer, J (eds) Morgan Kaufmann, 553–558.

Falkenhainer, B and Stein, JL (eds) 1992. *Automated Modelling, DCS* **41** American Society of Mechanical Engineers.

Ferris, J, Stein, JL and Bernitsas, M, 1994. "Development of proper models of hybrid systems" In: *1994 ASME WAM Symposium on Automated Modeling* Chicago, IL.

Ferris, JB and Stein, JL, 1994. "Development of proper system models for hybrid systems: a bond graph approach" In: *Proceedings of the 1994 IMACS World Congress* Atlanta, GA (extended Abstract).

Ferris, JB and Stein, JL, 1995. "Development of proper models of hybrid systems: a bond graph formulation" In: *Proceedings of the 1995 International Conference on Bond Graph Modelling* Las Vegas, NV, pp 43–48.

Fishwick, PA, 1995. *Simulation Model Design and Execution: building digital worlds* Prentice Hall.

Fishwick, PA and Luker, PA (eds) 1991. *Qualitative Simulation Modelling and Analysis* Springer-Verlag.

Franke, DW and Dvorak, DL, 1989. "Component connection models" In: *Workshop on Model-Based Reasoning, IJCAI-89* Detroit, MI, 97–101.

Gero, JS, Tham, KW and Lee, HS, 1992. "Behaviour: a link between function and structure in design" *IFIP Intelligent CAD* 193–225.

Giunchiglia, F and Walsh, T, 1989. "Abstract theorem proving" In: *Proceedings IJCAI-89* 372–377.

Hobbs, JR, 1990. "Granularity" In *Readings in Qualitative Reasoning about Physical Systems* Weld, D and DeKleer, J, (eds), Morgan Kaufmann, 542–545.

Hunt, JE, Pugh, DR and Price, CJ, 1995. "Failure mode effects analysis: a practical application of functional modelling" *Applied Artificial Intelligence* **9** 33–44.

Iwasaki, Y. 1988. "Causal ordering in a mixed structure" In: *Proceedings AAAI-88* **1** 313–318.

Keller, RM, 1991. "Applying knowledge compilation techniques to model-based reasoning" *IEEE Expert* April, 82–87.

Kraan, I, Richards, B and Kuipers, B, 1991. "Automatic abduction of qualitative models" In: *QR-91* Austin, Texas, 295–301.

Kuipers B, 1990. "Abstraction by time-scale in qualitative simulation" In *Readings in Qualitative Reasoning about Physical Systems* Weld, D and DeKleer, J (eds), Morgan Kaufmann, 530–534.

Leitch, R and Shen, Q, 1992. "Being committed to qualitative simulation" In: *QR-92* Edinburgh, UK, August, 281–293.

Linkens, DA, Xia, S and Bennett, S, 1992. "Systematic model transformation for analyzing dynamic physical systems at multiple levels" In: *Proceedings of 1992 European Simulation Multiconference* York, UK, June.

Linkens, DA, Wang, H, Bennett, S and Xia, S, 1992. "Using qualitative bond graph reasoning to derive lookup tables for fuzzy logic controllers" In: *Proceeding of the First International Conference on Intelligent Systems Engineering* Edinburgh, UK, August.

Liu, ZY and Farley, AM, 1991. "Tasks, models, perspectives, dimensions" In: *QR-91* Austin, Texas, May, 1–9.

Lrich, KT and Seering, WP, 1988. "Function sharing in mechanical design" In: *Proceedings AAAI-88* 342–346.

Mavrovouniotis ML and Stephanopoulos, G, 1988. "Formal order-of-magnitude reasoning in process engineering" *Computer Chemical Engineering* **12** 867–880. (Also in: Weld and de Kleer, 1990, pp 323–336.)

Minsky, M, 1975. "A framework for representing knowledge" In Winston, PH (ed) *The Psychology of Computer Vision* McGraw-Hill, 211–277.

Murthy, SS, 1988. "Qualitative reasoning at multiple resolutions" In: *Proceedings AAAI-88* 296–300.

Nayak, PP, 1992. "Automated modelling of physical systems" *PhD Thesis,* Stanford University, Department of Computer Science, Stanford, CA.

Nayak, PP, 1992a. "Selecting models for causal explanation" In Falkenhainer, B and Stein, JL (eds) *Automated Modelling, DCS* **41** American Society of Mechanical Engineers, 11–20.

Raiman, O, 1988. "Order of magnitude reasoning" In: *Proceedings AAAI-87* Morgan Kaufmann, 100–104. (Also in: Weld and de Kleer, 1990, pp 318–322.)

Raiman, O, 1991. "Order of magnitude reasoning" *Artificial Intelligence* **51** 11–38.

Rosenberg, RC and Redfield R (eds) 1988. *Automated Modelling for Design* ASME Bound Volume, No. G00460, ASME.

Rosenberg, RC and Wang, Y-Y, 1992. "Some structuring issues in modelling" In Falkenhainer, B and Stein, JL (eds) *Automated Modelling, DCS* **41** American Society of Mechanical Engineers.

Schmidt, D, Haddock, J, Wallace, WA and Wright, R, 1992. *Visual Modelling—A Knowledge Acquisition Method for Intelligent Process-Control Systems. Lecture Notes in Artificial Intelligence* **604** 294–303.

Smith, N, 1994. "Automated modelling: a review and proposal" *Research Note RN94-2,* School of Computing and Maths, De Montfort University, Milton Keynes, UK.

Stein, JL (ed.) 1991. *Automated Modeling* ASME Bound Volume, No. H00709, ASME.

Stein, JL (ed.) 1993. *Automated Modeling for Design* ASME Bound Volume, No. H00843, ASME.

Stein, JL (ed.) 1994. *Proceedings of the Symposium, The Automated Modeling: Model Synthesis Algorithms 1994 International Mechanical Engineering Congress and Exposition-Proceedings of the ASME Dynamic Systems and Control Division* **2**, Radcliffe, CJ (ed.) ASME Bound Volume, No. G0909B, ASME, 617–661.

Stein, JL (ed.) 1995. *Proceedings of the Symposium, Automated Modeling, 1995 International Mechanical Engineering Congress and Exposition-Proceedings of the ASME Dynamic Systems and Control Division,* T Alberts (ed.), ASME Bound Volume, No. H01025.

Stein, JL and Falkenhainer, B (eds) 1992. *Automated Modeling* ASME Bound Volume, No. G00747, ASME.

Top, J, Breunese, APJ, Broenink, JF and Akkermans, JM, 1995. "Structure and use of a library for physical systems models" In: *Proceedings 1995 International Conference on Bond Graph Modelling and Simulation (ICBGM95)* 97–102.

van Dijk, J and Breedveld, PC, 1991. "Simulation of systems models containing zero-order causal paths—I: classification of zero-order causal paths" *Franklin Institute* **328** (5/6) 959–979.

van Dijk, J, de Vries, TJA, Breunese, AJP and Breedveld, PC, 1992. "Automated mechatronic systems modelling using MAX" In Breedveld & Dauphin-Tanguy (eds) *Bond Graphs for Engineers* 279–290.

Vaidhyanathan, R and Venkatasubramanian, V, 1995. "Digraph-based models for automated hazop analysis" *Reliability Engineering and System Safety* **50**(1) 33–49.

Weld, D, 1990. "The use of aggregation in causal simulation" In *Readings in Qualitative Reasoning about Physical Systems*, Weld, D and DeKleer, J (eds), Morgan Kaufmann, 513–529.

Weld, D, 1990. "Approximation reformulation" In: *Proceedings AAAI-90* Boston, MA.

Weld, DS, 1988. "Comparative analysis" *Artificial Intelligence* **36** 333–373.

Weld, DS, 1992. "Generating simplified models with confidence" In Falkenhainer, B and Stein, JL (eds) *Automated Modelling, DCS* **41** American Society of Mechanical Engineers, 21–30.

Weld, DS and de Kleer, J, 1990b. "Multiple ontologies and automated modelling: introduction" In Weld, DS and de Kleer, J (eds) *Readings in Qualitative Reasoning About Physical Systems* Morgan Kaufmann, 481–483.

Wiener, N, 1950. *The Human Use of Human Beings: Cybernetics and Society* Houghton, Miflin.

Williams, BC, 1991. "A theory of interactions: unifying qualitative and quantitative algebraic reasoning" *Artificial Intelligence* **51** 39–94.

Williams, BC and de Kleer, J, 1991. "Qualitative reasoning about physical systems: a return to roots" *Artificial Intelligence* **51** 1–9.

Wilson, BH, 1992. "Model-building assistant: an automated modelling tool for machine-tool drive-trains" *PhD Thesis*, The University of Michigan.

Wilson, BH and Stein, JL, 1992. "An algorithm for obtaining minimum-order models of distributed and discrete systems" In Falkenhainer, B and Stein, JL (eds) *Automated Modelling DCS* **41** American Society of Mechanical Engineers, 37–46.

Wilson, BH and Stein, JL, 1993. "Model building assistant: an automated modeling tool for machine tool drive systems" In: *1993 Proceedings of the International Conference on Bond Graph Modeling and Simulation* San Diego, CA.

Winsor, J and Maccallum, K, 1994. "A review of functionality modelling in design" *The Knowledge Engineering Review* **9**(2) 163–199.

Xia, S, 1995. "Conceptual, causal and numerical modelling and analysis of engineering systems" In: *Proceedings 8th Intl. Conf. on Industrial and Engineering Applications of AI & ES* Invited and additional papers, Melbourne, Australia, June, 81–90.

Xia, S, 1995. "Deriving conceptual, causal and numerical models for simulation from structural information of physical systems" In: *Proceedings of 1995 Summer Computer Simulation Conference* July, Ontario, Canada, 251–256.