KNOWLEDGE-BASED MODELLING

Andreas M. Kleinhans

Betriebswirtschaftliches Institut
University of Stuttgart
West Germany

Abstract

Knowledge-based modelling consists of qualitative models which are implemented in a hybrid, rule- and frame-oriented programming language. Qualitative models achieve flexible and detailed description of both the simulation entities and relationships. This paper presents a simulation expert system which is based on a multi-level representation scheme of causal diagrams. It offers a qualitative "presimulation", that is conclusions about the sensitivity of the elements, the restrictions and the possible behaviour of the model. It enables explanation of the various implicit structural and dynamic relationships and the user to be guided to efficient quantitative simulation.

1. Introduction

In traditional simulation the kind and quality of effects are usually perceived only by several appropriate simulation runs /1/. Both the model structure and dynamic behaviour require the interpretation of the (human) model user. Complex models claim high model knowledge both on the model builder and user and there is need for detailed descriptions and explanations on every level. For that reason the acceptance of traditional simulation systems is still very low /2/.

Many authers express the urgent demand to simplifiy the model use and evaluation, to support the validation process and to guide the user to efficient simulation runs /3/. The user should be able to concentrate on the model and to ask the system any question about the structure and dynamic of the model. This requires a user friendly dialogue module, helping procedures and explanation utilities during every modelling amd simulation stages /4/.

The advantages of Artificial Intelligence (AI) techniques for simulation have been widely discussed /5/6/ and several useful combinations have been presented /7/8/9/. This paper shows how knowledge-based, rule and frame oriented techniques support qualitative structural descriptions of models and how they enrich traditional modelling by representation of application oriented knowledge.

2. BAMBOO II

As a basis for the discussion the hybrid prototype system BAMBOO II is introduced. It has been developed at our institute as a general simulation expert system demonstrating the usefulness of object oriented techniques for simulation /9/. We got most experience with models representing flexible assembling systems, however the aspects of qualitative modelling are fundamental.

BAMBOO II offers useful and detailed model knowledge to the user in order to guide him to more efficient simulation. That is the system behaves like an experienced model expert. Its knowledge consists of diverse structural and dynamic peculiarities such as statements about critical paths, the sensitivity of elements, relations and loops, effects of restrictions and the possible behaviours resulting from different initial states.

The main objects are:

1. Representation and processing of contextual knowledge of the simulation entities
2. Qualitative "presimulation" of relationships and restrictions
3. Disclosure of application oriented backgrounds
4. Representation of generic structures
5. Handling diverse models simultaneously

In order to improve the transparency of models the system must be able to answer all questions about the structure and dynamic interactions. That requires a capability of detailed element, causal path and loop analysis including the determination of the sensitivity of all entitities. So-called "presimulation" is based on the model structure and provides inference about the structural pecularities and the possible behaviour of the model /10/. Direct decision support is offered by processing qualitative, application oriented information and knowledge, giving semantical descriptions and explanations and offering feasable polices.

Often there are several versions of the same model, i.e. the same models with different constant values and initial states. Or, in a more complex situation, there are different, but close models. In order to handle diverse models appropriately, the system must build generic models, which keep the knowledge of what they share in common.

3. Datastructures of the model entities

Like in System Dynamics the basis for the model structure are causal diagrams. Though in System Dynamics the model structure is appropriated by transforming it to the flow diagram, in BAMBOO II the causal diagram itself is processed and enriched by contextual knowledge. Furthermore the model may be viewed in a hierarchy of different levels. They either consist of a structural order of element-, path and loop objects or of application oriented strategy and policy objects. Every objects holds its own particular attributes.

Frames are very useful datastructures for representing objects. In general they consist of a frame name, references to superclasses and slots (attributes) which store the properties of the object. In theory there are several statements and functions ("flavours") for every slot, like the specification of value-sets, default values, constraints and "demons", which are automatically activated in reading, changing or removing the slot-values /11/12/.

The most important structural frames are the element, relation, path and loop frames (see fig. 1). Elements represent the real objects, relations their causal connections, paths provide chains of elements and relations and loops represent circular paths. Every frame owns different attributes, which however are often based on each other.

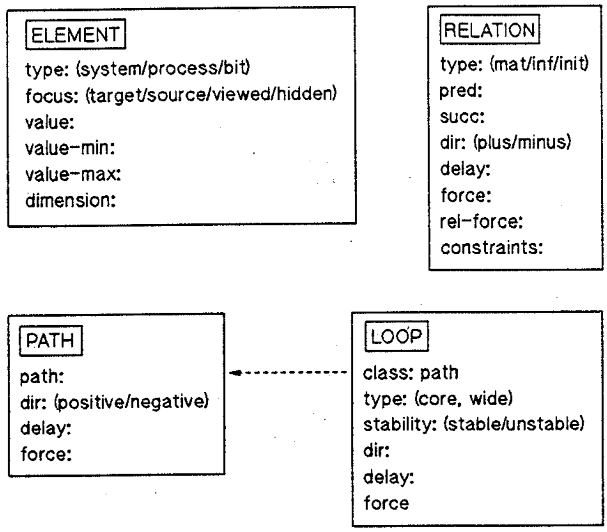

Fig. 1 Basic frames of the causal diagram

The element type indicates the nature of the element: it may be a system, a process or a bit (of information). They roughly correspond to the System Dynamics concepts of level, rate and auxiliary respectively. The nature of a system is usually (but not necessarily) material, and that of a bit is always immaterial. The focus attribute determines how the elements are used respecting the users interest. Target elements are those which are after all important for the user, but which he cannot manipulate directly. Source elements are those which can be changed and which act like operators, i.e adjustable parameters. The attribute value "viewed" makes an element transparent to the user during the session, i.e. only viewed elements are mentioned in a system dialogue, others are omitted. Targets and sources are always viewed.

A relation of the type information indicates an information flow, that of the type material, a material flow. All types are analogical to the elementtypes. Usually an information flow leads to an information bit, a material flow to a system and an activation starts a process. But different combinations are allowed and there are as many interpretation rules as combinations possible. The pres/succ attributes store the predecessor and successor of a relation. These statements hold the basic structural information of the model. The dir (direction) property indicates whether the relation influence is positive (i.e. in same direction) or negative (i.e. in opposite direction). The delay attribute stores the time lag, the force attribute its strength and the rel-force attribute, its influence relatively to other relations with the same successor. The force may be strengthening (+), steady (=), weakening (-) or a qualitative change (#). Finally constraints store the description of all restrictions which limit the relation.

"Qualitative change" signifies a non-linear, possibly discontinuous and only vague formalizable relationship.

Note, that type attributes of the both the elements and the relations refer to corresponding superclasses. In these classes general information is stored, e.g. the fact that a information-relation always succeeds without any delay, that the strength of a material-relation must not be strengthening or that the minimum value of a system must be always equal to or greater than zero.

Paths and Loops are based on elements and relations. In general they describe the property sum of their components. They especially refer to the attributes "dir", "delay" und "force", however the interpretations are different. While a "negative" direction of a path indicates an opposite effect between the first and last element, a "negative" direction of a loop shows the loop to be goal-seeking. Furthermore it proved to be useful to distinguish between relevant ("core") and less relevant ("wide") loops. Finally they consist of an attribute which describes their stability.

The main advantage of the object-oriented representation is the easily modification and expansion of the model entities. The modification can take place at every time even dynamically during the simulation run. Note that the same attributes in different objects may get a different interpretation, too. Furthermore the knowledge based simulation procedure has several advantages. The relations need not be implemented by numerical equations but can also consist of qualitative rules. Moreover the simulation process may be triggered by forward rules which are implemented as "demons" and are activated automatically.

4. Structural Analyses

According to the model objects there exist several structural analysis procedures of the causal diagram. They consist both of elementary anaylses of elements and relations and of more complex evaluations of paths, path sets, loops and the total model. The main features are:

- -- determination of the peculiarities of the elements and relations
- -- description of the model dynamic
- -- generation and evaluation of any path/loop
- -- determination of the most/least sensitive element/relation in the path/loop respecting dir/delay/force/rel-force
- -- determination of the most/least sensitive path/loop of the model respecting dir/delay/force
- -- evaluation of critical restrictions in paths/loops
- -- description of the elements/relations of nested paths/loops

Element and relation analyses primarily deal with evaluations of predessors and successors. Path and loop analyses offer an evaluation of every detail in the chain including the disclosure of all restrictions. Fig. 2 shows example protocols of a given path. The first example demonstrates an ex-ante-analysis ("presimulation") of the path, the other example an evaluation at a certain time during the actual simulation. For instance the causal step between production and supply is indicated as in the same direction (+) with a delay between 4 and 6 periods and a steady force. At time 22 the force counts 60% relatively to other influences on supply.

```
"presimulation" mode:

     dir  delay force
                        production
      +    4/6    =     supply
      +     0     =     diff
      +     1     #     marketing
      +     2     #     demand
----------------------------------
      +    7/9    =     total

actual path analyis at time 22:

     dir  delay force %force
                              `production
      +     6     =     60    supply
      +     0     =     40    diff
      +     1     #    100    marketing
      +     2     #     15    demand
----------------------------------------
      +     9     =      3    total
```

Fig. 2 Path analyses examples

Loop analyses are comparable to path analyses. They consist of some additional analysis procedures which deal with the different types of loops. E.g. there are local adjustment loops, which only act as a delay and which often may be simply replaced by a constant. On the other side the importance (force) of a loop relative to others can generally be only determined by appropriate simulation runs.

Because of their complex combination possibilites path and loop analyses proved to be very unwieldy. In particular the compact and user friendly representation and explanation of the results are very difficult. However, dealing with high levelled applications, like strategic management problems, the most efficient information for the user was given by disclosing only elementary relationships and inferences. That is because strategic problems consist rather of little but contextually compact data.

5. Application oriented objects

Causal diagrams are a very convenient for modelling real structures and dynamic interrelations. Unfortunately elements and their relations are often too complex to be appropriately represented by simple causal objects. Especially when the entities of material and information flows are not homogeneous, or when different attributes of the flow objects are addressed, a well-suited representation is essential. In general a causal diagram like in fig. 3 is intuitively well understood, however the relations are too inexact and looking closer may lead to some severe misunderstandings. While the supply-sale and demand-sale relation addresses the same object, that is the product in total, the sale-profit relation only points to one aspect, that is the product price. The connection sale-satisfaction is even more complex, it refers to the product quality, which consists of all product attributes itself.

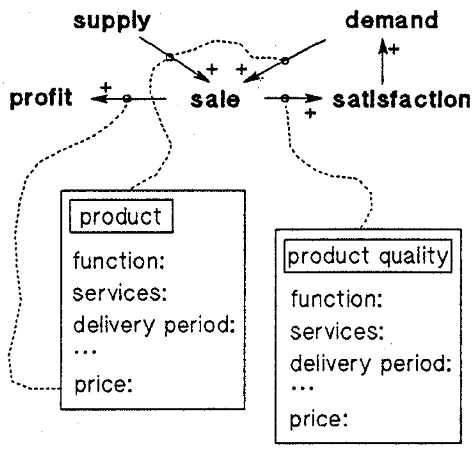

Fig. 3 Product and quality objects

While traditional simulations have to use diverse tricks to represent such problems, the object-oriented techniques allow an easy and application-oriented way by selecting the relevant attributes directly from the object frames. By this means all model entities are appropriately representable.

6. Application oriented policies

Policies are formed during an analysis of the entire model and are based on the elementary analyses. The questions are global and require distinguished knowledge about all model pecularities:

- Which external parameters are how sensitive respecting which target variables?
- Which restrictions must be changed in which way to pass which bottle-necks?

According to the overall scanning of the diagram the structural analyses usually offer a huge number of policies. However not the possibilities of the model on principle but rather the realizable strategies and their appropriate formalization are decisive for the user.

Like the differentiation of the flow objects, a suitable refinement of policies and strategical parameters is necessary. E.g. usually the goal hierarchy of a company consists of multiple levels and strategic meanings. The formalization into frames can be directly performed. Fig. 4 shows an example hierarchy of an existing company which is operating in three different markets.

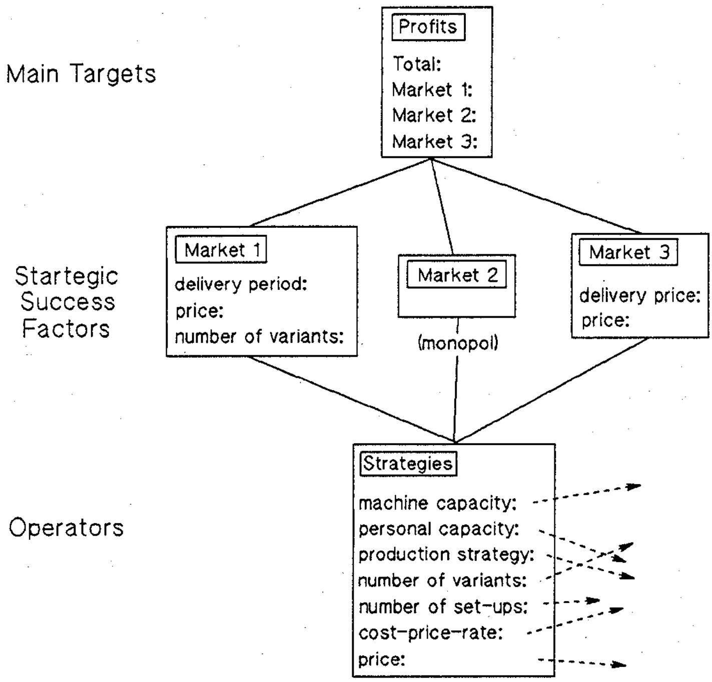

Fig. 4 Example goal hierarchy

The main goal naturally focuses on the profits which usually cannot be influenced directly. Nor can the subgoals which are expressed by the strategic success factors for each market. The strategy frame finally focuses on the operators which can be adjusted in users reality. Every attribute of the strategy frame points to the individual operator frame which holds the description of initial conditions, restrictions, operation and sensibility. This semantical net supports the user in searching efficient and realizable strategies.

Besides the necessity for representing goal hierarchies there is a need for representing different goal-oriented model versions, that is representing knowledge about the incorporated strategies of a model. Models act differently depending on which strategies are already implemented. In fact, to compare diverse policies they must rely on the same (base) model. This kind of knowledge is stored in a generic model whereby different instances of the model represent model versions with different instances.

7. Restriction handling

Restrictions are of different natures, too. They appear in rule conditions, as range limitors or simply as an evaluation constant. On principle all parameters in a simulation program are changeable, so technically there is no differentiation between the restrictions either. However in real systems not only the knowledge about the existence of restrictions is important but also information about their status. Are they physical, technical, organisational or even political in nature? Restrictions or constraints might be passed by some actions not mentioned yet. Therefore it is useful when the system may question the restrictions found during the structural and strategical analyses.

In BAMBOO II parameters, values and rules which are not definite for the user can be marked as "situational". Also a differentiation in diverse classes of restrictions is projected. Furthermore there is the possibility of storing knowledge about the consequences of the hyphothetical violation of restrictions and to keep it on hand for explanations. These features allow an adequate restriction management required by real problems.

8. Summary

It has been shown how object-oriented AI-techniques offer advantages for the differentiation and analysis of the model structure. Several hierarchies of object classes have been presented. Especially application-oriented refinements allow the representation of contextual knowledge which offers a distinct support in searching efficient and realizable strategies.

9. References

/1/ Shannon, R.E. Mayer, R. und Adelsberger, H.H. (1985). Expert Systems and Simulation. In: Simulation 44:6, 275-284.

/2/ Richmond, Barry (1985). STELLA: Software for Bringing System Dynamics to the other 98%. In: Proceedings of the 1985 International Conference of the System Dynamics Society, Keystone, Colorado, 706-718.

/3/ Young, David F. (1985). KBSIM - A Knowledge based Tool and its Use in Model Preprocessing. In: Proceedings of the International Conference of the System Dynamics Society, Keystone, Colorado, 1070-1080.

/4/ Kerckhoffs, E.J.H. und Vansteenkiste, G.C. (1986). The Impact of advanced Information Processing on Simulation - An illustrative Review. In: Simulation 46:1, 17-26.

/5/ Ören, Tuncer I. und Zeigler, Bernard P. (1987). Artificial intelligence in Modelling and Simulation. In: Simulation 48:4, 131-134.

/6/ Reddy, Y.N.R. (1987). Epistemology of Knowledge Based Simulation. In: Simulation 48:4, 162-166.

/7/ O'Kneefe, Robert (1986). Simulation and Expert Sytems - a Taxonomy and some Examples. In: Simulation 46:1, 10-16.

/8/ Moser, Jorge (1986). Integration of Artificial Intelligence and Simulation in a Comprehensive Decision-Support System. In: Simulation 47:6, 223-279.

/9/ Kleinhans, Andreas (1986). A Behavioral Analysis Expert System for System Dynamics Models. In: Aracil et al. (Hrsg., 1986), 1039 - 1048.

/10/ Kleinhans, Andreas (1989). Wissensverarbeitung im Management. Frankfurt: Lang.

/11/ Harmon, P. and King, D. (1985). Expert Systems - Artificial Intelligence in Business. John Wiley & Sons.

/12/ Puppe, Frank (1988). Einführung in Expertensysteme. Berlin: Springer.