CAUSAL AI MODELS

Steps Toward Applications

CAUSAL AI MODELS

Steps Toward Applications

Edited by

Werner Horn

Department of Medical Cybernetics and Artificial Intelligence
University of Vienna
and the Austrian Research Institute for Artificial Intelligence
Vienna

HEMISPHERE PUBLISHING CORPORATION
A member of the Taylor & Francis Group
New York Washington Philadelphia London

NOTICE

Because this volume is appearing as an issue of *Applied Artificial Intelligence*, as well as a hardbound book, it was necessary to use a numbering system that would incorporate both the continuous journal page numbers and the book page numbers. The pages are double-numbered as follows: page 1 of the book is equivalent to page 85 of the journal. The numbers appear on the pages as book page number followed by the journal page number in brackets.

CAUSAL AI MODELS: Steps Toward Applications

1 2 3 4 5 6 7 8 9 0   B R B R   8 9 8 7 6 5 4 3 2 1 0 9

This book was set in Times Roman by Hemisphere Publishing Corporation. The editors were Mary Prescott and John P. Rowan; the managing editor was Ralph Eubanks; the production supervisor was Peggy M. Rote, and the typesetter was Sandra F. Watts. Cover design by Sharon M. DePass.
Braun-Brumfield was printer and binder.

Library of Congress Cataloging-in-Publication Data

Causal AI models.
1. Artificial intelligence—Mathematical models.
2. Reasoning—Mathematical models. I Horn, Werner, Dipl.-Ing. Dr.
Q335.C38 1989 006.3'01'5118 89-19853

ISBN 1-56032-048-6

CONTRIBUTORS

Stig K. Andersen
Nordjysk Udviklingscenter and Department of Medical Informatics and Image Analysis, Aalborg University, Aalborg, Denmark

Steen Andreassen
Nordjysk Udviklingscenter and Department of Medical Informatics and Image Analysis, Aalborg University, Aalborg, Denmark

Bert Bredeweg
Department of Social Science Informatics, University of Amsterdam, Amsterdam, The Netherlands

Tom Bylander
Department of Computer and Information Science, The Ohio State University, Columbus, Ohio 43210

B. Chandrasekaran
LAIR, CIS Department, The Ohio State University, Columbus, Ohio 43210

Jean-Marc David
Laboratoires de Marcoussis, C.G.E. Research Center, Marcoussis, France

Richard J. Doyle
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91109

Bjørn Falck
Nordjysk Udviklingscenter, Aalborg, Denmark, and Department of Clinical Neurophysiology, Turku University Hospital, Turku, Finland

Tim Finin
Paoli Research Center, Unisys Corporation, Paoli, Pennsylvania 19301

John R. Frederiksen
Bolt Beranek & Newman Inc., Cambridge, Massachusetts 02138

David E. Hirsch
Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Werner Horn
Department of Medical Cybernetics and Artificial Intelligence, University of Vienna, and Austrian Research Institute for Artificial Intelligence, Vienna, Austria

Finn V. Jensen
Department of Mathematics and Computer Science, Aalborg University, and Judex Datasystemer, Aalborg, Denmark

Frank Jensen
Nordjysk Udviklingscenter and Department of Mathematics and Computer Science, Aalborg University, Aalborg, Denmark

Uffe Kjaerulff
Judex Datasystemer, Aalborg, Denmark

David Klein
IBM T.J. Watson Research Center, Yorktown Heights, New York 10598

Phyllis A. Koton
Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Jean-Paul Krivine
Laboratoires de Marcoussis, C.G.E. Research Center, Marcoussis, France

Marc Linster
Expert Systems Research Group, Gesellschaft für Mathematik und Datenverarbeitung, St. Augustin, Federal Republic of Germany

William Long
Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Kristian G. Olesen
Nordjysk Udviklingscenter and Department of Mathematics and Computer Science, Aalborg University, Aalborg, Denmark

Ramesh S. Patil
Clinical Decision Making Group, MIT Laboratory for Computer Science, Cambridge, Massachusetts 02139

Mallory Selfridge
Department of Computer Science and Engineering, University of Connecticut, Storrs, Connecticut 06268

Oksana Senyk
Division of Clinical Decision Making, Tufts–New England Medical Center, Boston, Massachusetts 02111

Sheldon R. Simon
Department of Surgery, The Ohio State University, Columbus, Ohio 43210

Frank A. Sonnenberg
Division of Clinical Decision Making, Tufts–New England Medical Center, Boston, Massachusetts 02111

Luc Steels
Artificial Intelligence Laboratory, Vrije Universiteit Brussel, Brussels, Belgium

Jon Sticklen
AI/KBS Group, CPS Department, Michigan State University, East Lansing, Michigan 48824

Peter Szolovits
Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Hans Voss
Expert Systems Research Group, Gesellschaft für Mathematik und Datenverarbeitung, St. Augustin, Federal Republic of Germany

Michael A. Weintraub
Department of Computer and Information Science, The Ohio State University, Columbus, Ohio 43210

Barbara Y. White
Bolt Beranek & Newman Inc., Cambridge, Massachusetts 02138

☐ PREFACE

Research on causal modeling has received strong attention in the last two years. Johan de Kleer received the "Computers and Thought Award" at the IJCAI-87 in Milan for his work on qualitative physics, and the Special Volume on Qualitative Reasoning about Physical Systems of the Artificial Intelligence journal has become a "classic." Qualitative physics—describing function and behavior of technical systems in a qualitative way—is only one area of the causal modeling field. Reasoning from first principles, reasoning from structure and behavior, and causal reasoning are often associated with the term "causal modeling." In addition, second-generation expert systems are trying to integrate associative "surface"-level reasoning with causal "deep"-level reasoning. This "deepening" of systems has stimulated a great amount of discussion regarding the "deepness" of systems and the interrelations between causal models and deep systems.

Seeing this enormous interest in research on building AI programs with an adequate understanding of the systems they are modeling—both for technical and for biological systems—we decided to collect articles from around the world into a single volume. This book should report how far research efforts have brought us towards causal AI models in practice. In the past, great emphasis has been put on basic methods of causal modeling. *Causal AI Models* should demonstrate how to extend and utilize these methods to build applied systems. It should give us examples from the application of the technology, with first insights drawn from these applications. Naturally, the applied systems will bring up new problems which will lead us to improve theoretical results and to obtain new ones. In July 1988, I started inviting researchers to contribute to this collection. The response was very positive. I am now able to present *Causal AI Models: Steps Toward Applications*, which gives an excellent view on research results in the field.

Causal modeling is a rich field with many dimensions, as demonstrated by the many aspects addressed by the papers collected herein. All these aspects, and perhaps others, must be explored if we want to arrive at a full picture of the capabilities and implications of causal models. I hope this book will help to broaden our view and, finally, will help us to build applied systems with greatly increased power and competence.

Werner Horn

☐ CONTENTS

INTRODUCING META-LEVELS TO QUALITATIVE REASONING

BERT BREDEWEG
Department of Social Science Informatics, University of Amsterdam, 1016 BS Amsterdam, The Netherlands

Current approaches to qualitative reasoning are largely based on a fixed framework for modeling the physical world and concentrate on the reasoning methods that support qualitative reasoning. This paper argues that we need several levels of abstraction and different viewpoints on how to model the physical world, in order to create systems that reason about the physical world in a flexible way. We present a framework that integrates the three basic approaches to qualitative reasoning and show how this framework can be used as a basis for a flexible qualitative reasoning system.

INTRODUCTION

Despite the fast growth of research in qualitative reasoning (Bobrow, 1984), building qualitative models of systems from the physical world is still a major bottleneck. Problems such as finding the right level of abstraction and the appropriate set of modeling primitives are largely left up to the creativity of the designer. In fact, it is often the case that the resulting model is quite trivial, whereas the process of arriving at a particular model is highly complex. It is this modeling process that we want to support and to make more flexible and more closely related to commonsense reasoning. With this long-term perspective in mind, in this paper we address the question of how a system can perform qualitative reasoning with a variety of ontological models of the physical world.

The contents of this paper are structured as follows. In the next section we describe a number of problems, arising from building qualitative models, that we would want our qualitative reasoning approach to be able to deal with. In the third section we describe the framework that we have developed for qualitative prediction of behavior. This framework integrates the three best known approaches to qualitative reasoning. In the fourth section we explain how different operationalizations of domain knowledge within this framework allow us to solve the problems mentioned in the second section. Finally, in the conclusions

Bob Wielinga, Martin Reinders, Zeger Karssen, Huub van der Wouden, and Paul de Greef provided helpful discussions and observations on the research reported here.

This document is part of a research project funded by the Esprit Programme of the Commission of the European Communities as project number 1098 and of a research project funded by the Foundation of Knowledge Based Systems (SKBS), which is supported by the Dutch Government through the SPIN program.

we summarize the results and point out some of the research issues that are still open.

OBJECTIVES TO BE ADDRESSED

While going through a number of modeling exercises, we formulated the following conclusions:

1. When a system can be modeled with one particular approach to qualitative reasoning, it can also be modeled with another approach to qualitative reasoning (*multiple models*).
2. It is sometimes desirable to use modeling primitives from different approaches to qualitative reasoning within one model (*integrated models*).
3. It is sometimes desirable to use different levels of detail within one model (*levels of abstraction*).

Each of these issues is described in more detail in one of the following sections.

Multiple Models

The multiple models issue can be illustrated by the two-tanks problem (Fig. 1). A process-centered model for this system has been described by Forbus (1984) and a constraint-centered model has been described by Kuipers (1986). We assume that the reader is familiar with these models. In Bredeweg (1989) we have, among others, described a component-centered model (de Kleer and Brown, 1984) for the two-tanks system. In this component-centered model two containers and a valvelike connector are distinguished. Each of them has four qualitative states. For the containers they are given in Fig. 2. A container can be *steady*, which means that the total amount of liquid remains constant;* the total amount can be *decreasing* or *increasing*; and the container can be *empty*, which means that there is no liquid present in the container. Each of these behaviors

*To keep the model simple we assume that no liquid flows in or out at the top of the container.

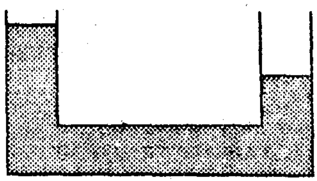

FIGURE 1. Two connected tanks.

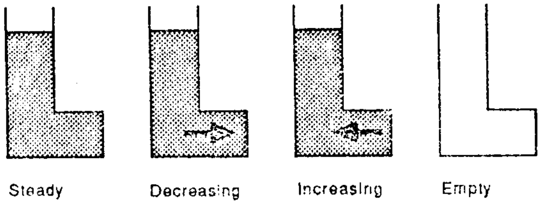

FIGURE 2. A component model for the two-tanks system—part 1.

has been modeled with a separate qualitative state. Note that in this model the amount of liquid, and how it changes, has been the guideline for defining the qualitative states. Other parameters might have been used, which would probably have resulted in a different model.

For the valvelike connector a similar set of qualitative states has been defined (Fig. 3). The amount of liquid is *steady*, flows from *left to right*, flows from *right to left*, or there is no liquid (*empty*). It is easy to see how the cross-product of these qualitative states results in 64 overall state descriptions (most of them inconsistent, of course) and how the final behavior can be derived from this as a sequence of valid state descriptions.

Obviously, the three models mentioned above (process, constraint, and component-centered) differ and can be classified as being good, bad, or somewhere in between. However, this classification can be done only by using the criteria provided by a particular approach. A *liquid flow process* is an incorrect model when judged by the *no function in structure principle*, but it is a very attractive model from a cognitive modeling point of view. Note that there are no fundamental arguments to favor a particular model.

Apparently none of the approaches models qualitative reasoning in its full extent: that is, there are qualitative models, based on the ontology provided by one approach, that cannot be reasoned about by another approach. Moreover, humans *can* reason with models based on different ontologies. If an approach to qualitative reasoning is to be more than just a special-purpose tool, it should be able to support this reasoning. It should have knowledge about these different models. It should know how these models are related to each other and know how they can be used.

Integrated Models

The second issue, the use of integrated models, can be illustrated by a model of the refrigerator (Fig. 4). The behavior description that goes along with this model is the following:*

*This description is taken from *Sesam Technische Encyclopedie* by T. Bosch and G. Keuning (1979). We translated the Dutch description into the English one given here.

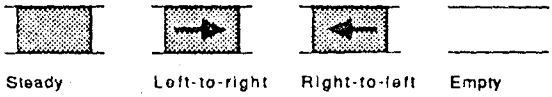

FIGURE 3. A component model for the two-tanks system—part 2.

> By a refrigerator based on the compression principle, gas is sucked out of the evaporator and compressed by the compressor. The compressed gas is then transformed into liquid in the condenser by cooling it with air or water. Next, the liquid goes through the throttle valve, which decreases its pressure, and arrives in the evaporator. In the evaporator the liquid evaporates as a result of this low pressure and, by doing so, withdraws heat. This is where the actual cooling takes place.

At first glance the refrigerator seems typically something to model with the component-centered approach, in particular the behavior of the compressor and the throttle valve. However, modeling the behavior of the evaporator and the condenser is not that straightforward. First, the behavior of these components depends on the interaction with the environments in which they operate. As a result, additional components and conduits must be defined in order to model their behavior. Second, the notion of transforming a gas into a liquid, and the other way around, cannot be modeled *explicitly* with the component-centered approach. Both these objections disappear when a process definition is used to describe the behavior taking place between the components and their environments. Therefore, although in this particular example it is possible to model the

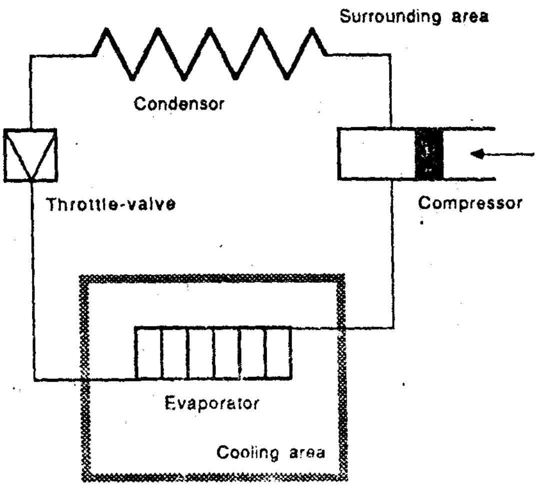

FIGURE 4. A model of the refrigerator.

behavior of the condenser and the evaporator with the component-centered approach, it is rather counterintuitive, and from a modeling point of view it is more convenient to use a process definition. Moreover, in some situations process definitions are an ontological necessity. Some behaviors in the physical world are not enforced by a component, but happen because objects, in a particular configuration, interact and facilitate behavior. Examples are heat flow, liquid flow, and gravity.

Levels of Abstraction

The final issue, the use of different levels of abstraction, will be explained with a model of a relay (Fig. 5). We came across the following description of a relay:

> If the input current is turned on, then the relay immediately produces an output current equal to the input current. This output current lasts for a particular time and is then turned off by the relay. The output current remains off until the input current is turned off and on again. After the input current has been turned on again the process repeats itself.

This behavior description cannot be modeled properly with one of the prevailing approaches to qualitative reasoning, because it uses *both* discrete and continuous parameter values. The input and output current are described as having a discrete character, as they are either on or off, whereas the delay time is described as having a continuous character; it starts increasing after the input current is turned on, until it reaches a certain point, after which the output current is turned off. To model this behavior description properly, an approach should be able to handle both discrete and continuous descriptions of values simultaneously.

Summarizing the Objectives

Following the discussion above, we came to the conclusion that there is no such thing as *the* ultimate model for a physical system, in particular not in commonsense reasoning. An approach for reasoning about physical systems should therefore be able to support reasoning with multiple models, based on different ontologies, at different levels of abstraction and with different viewpoints within one level of abstraction. To address this objective, an approach should have a meta-level understanding of the available modeling primitives and reasoning processes. We have described a framework (Bredeweg and Wielinga, 1988a, 1988b; Bredeweg, 1989) that integrates qualitative reasoning approaches and provides the first input to meta-level reasoning in this field. In

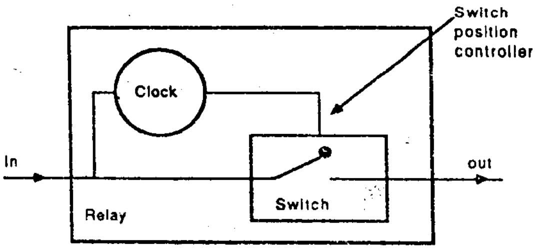

FIGURE 5. A model of the relay.

this paper we investigate how domain models can be operationalized within this framework, which allows more flexible and intuitive modeling of the physical world.

GARP: AN INTEGRATED APPROACH

GARP (Bredeweg, 1989) is an acronym for General Architecture for Reasoning about Physics and refers to the program that has been implemented in Prolog, based on the framework put forward in the generic task (see Breuker et al., 1987; Chandrasekaran, 1987; Clancey, 1986; Wielinga and Breuker, 1986) that we have described for qualitative prediction of behavior. GARP is able to simulate qualitative reasoning as described by the original approaches. A description of the framework is given below.

Theory Underlying the Framework

The framework is based on the premise that knowledge used by people during reasoning processes can be distinguished according to several types, corresponding to the different roles the knowledge plays in the reasoning process. Wielinga and Breuker (1986) identify four types of knowledge—domain, inference, task, and strategy—and therefore describe models of expertise in four layers. The first layer, the domain layer, describes the knowledge of a particular domain. In the domain of electronics, for example, this layer might embody knowledge about transistors, wires, switches, and so forth. The second layer, the inference layer, describes the canonical inferences that can be made on the basis of the first layer. Two types of objects are used at the inference layer: *meta-classes* and *knowledge sources.* Meta-classes represent the role domain objects can play in the inference process. For example, a domain concept like *faulty transistor* may play the role of a *finding* but may also play the role of a *hypothesis.* Knowledge sources describe what kind of inferences can be made on the basis of the relations in the domain layer. Examples of knowledge sources are *specification* and *abstraction*, which both might use the subsumption relation in the domain. The third layer, the task layer, specifies task structures that are typical for the domain. Task structures are sequences of knowledge sources and meta-classes that can be used to achieve a particular goal. The fourth layer, the strategic layer, contains knowledge to deal dynamically with the knowledge at the other layers. It should, for instance, plan a particular task structure, monitor its execution, and, if needed, diagnose, repair, or even replace the current task structure with another task structure, until the desired problem-solving goal is reached.

Domain and Inference Layer

According to this four-layer model, the original approaches to qualitative reasoning are domain theories. They provide ontological primitives to model domain-specific knowledge, such as processes, component models, and constraints. The inference layer abstracts from these domain-specific modeling primitives by (1) describing the *canonical* inferences used in the reasoning process and (2) pointing out the *role* the modeling primitives play in this reasoning process.

Objects of the Inference Process

The meta-classes (roles) that can be identified in qualitative prediction of behavior are the following*:

System model description: Central to qualitative reasoning is the way in which a system is describe during *a period of time in which the behavior of the system does not change*. The notion of change is rather subtle because the actual (real-world) system may change whereas from a qualitative point of view its behavior remains constant. A constant state of behavior is therefore characterized by parameters that describe the behavior of a particular system qualitatively and that do not change within the time elapsed during the state. In GARP such a description is called a *system model description*. A system model description is a composition of one or more of the modeling primitives referred to by the meta-classes described in this list.

System elements: System elements refer to entities from the real-world system that are represented symbolically in the qualitative reasoning program. Examples are containers, components, and heat paths.

*Note that approaches do not necessarily use all the modeling primitives described here. Sometimes they are implicit (like system elements in the constraint-centered approach) and sometimes they are not used at all (like system structures in the constraint-centered approach).

Parameters: Parameters are used to describe properties of system elements. Examples are temperature, amount-of, and pressure.

Parameter values: Parameter values represent the values parameters can take on. Well-known examples are [+, 0, −].

Quantity spaces: To arrive at qualitative values, the quantitative values that a parameter can have are divided into a small set of intervals with relevant distinctive characteristics. Such a set is called a quantity space.

Parameter relations: Parameter relations are used to describe dependencies between parameters. Examples are influences, arithmetic equations, and proportional relations.

Qualitative calculi: Qualitative calculi are used to define the semantics of a relation. They express how a relation should be interpreted.

Mathematical model: The relations that hold at a particular moment represent a mathematical model of the behavior of the system in the real-world.

System structures: System structures are templates that are used to augment a system model description. Examples are *views* and *processes* in the process-centered approach and *qualitative states* in the component-centered approach. In order to apply system structures, the qualitative reasoning program must know when a particular structure holds. System structures therefore have an *if-then* nature. The if part specifies the parts of the system model description that must exist for the structure to be applicable, whereas the then part specifies the new parts that must be added to the system model description when the structure is applicable. Basically, system structures are used to find mathematical models that represent the behavior of system elements.

Transformation rules: Transformation rules are used to store knowledge about identifying successive states of behavior.

Behavior descriptions: Finally, a behavior description is a set of system model descriptions ordered in time. It represents the possible behavior of some real-world system.

Some Examples

It is beyond the scope of this paper to give detailed examples of how the meta-classes described in the previous section are implemented. However, some brief examples are listed below.

The first example illustrates a *system structure* that represents the process *liquid flow* from the process-centered approach.

```
system_structures( liquid_flow( (Con1, Con2) ), type( process ),
    conditions([
        parameters([
            pressure( Liquid1, Press1 ),
```