Generating Behavior Equations from Explicit Representation of Mechanisms

Extended Abstract

Yumi Iwasaki

Department of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

This research is supported by the Defense Advanced Research Projects Agency, Department of Defense, ARPA Order 3597, monitored by the Air Force Avionics Laboratory under contract F33615-81-K-1539. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S Government.

1. Introduction

The methods of causal ordering [10] and comparative statics [9] provide an operational means to determine the causal relations among the variables and mechanisms that describe a device, and to assess the qualitative effects of a given disturbance to the system [5, 6, 3]. These procedures, which have been widely used in several fields of sciences, are generally consistent with and somewhat more general than the methods for determining causal relations and for propagating disturbances employed by many researchers of qualitative reasoning [1, 2, 11]. We have been developing a system to perform qualitative causal analysis of a device behavior based on these methods. For the method of causal ordering to produce the correct causal relations, equations comprising a model must come from understanding of mechanisms. This paper focuses on the issue of building a model that meets this requirement. The approach we have taken to ensure that each equation stands for a distinct mechanism is to represent explicitly one's understanding of mechanisms underlying an equation model. This representation forms, below the level of equation model, another level of model, which represents such understanding and from which equation models are generated automatically. We have developed the representation for mechanisms and a program to generate equation models from the representation. The domain in which we have been working in is that of coal power plants.

2. Causal Ordering

Causal ordering is an asymmetric relation among the variables and equations of a set of simultaneous equations. The idea of causal ordering in a system of equations can be described roughly as follows. A system of n equations is called self-contained if it has exactly n unknowns. Given a self-contained system, S, if there is a proper subset, s, of S that is also self-contained and that does not contain a proper self-contained subset, s is called a minimal complete subset. Let S_0 be the union of all such minimal complete subsets of S; then S_0 is called the set of minimal complete subsets of zero order. Since S_0 is self-contained, the values of all the variables in S_0 can, in general, be obtained by solving the equations in S_0. By substituting these values for all the occurrences of these variables in the equations of the set $(S - S_0)$, one obtains a new self-contained structure, which is called the derived structure of first order. Let S_1 be the set of minimal complete subsets of this derived structure. It is called the set of complete subsets of 1st order. Repeat the above procedure until the derived structure of the highest order contains no proper subset that is self-contained. If one denotes by V_i the set of variables in the complete subsets of ith order, where $i \geq 0$, then the variables in V_i, $(i > 0)$, are said to be *directly causally dependent* on the elements in V_{i-1}.

Equations comprising a model come from an understanding of mechanisms. The term *mechanism* is used here in a general sense to refer to distinct conceptual parts in terms of whose functions the working of the whole system is to be explained. Mechanisms are such things as laws describing physical processes or local components that can be described as operating according to such laws. An equation representing such a mechanism is called a *structural equation*, and every equation in the model should be a structural equation standing for a mechanism through which variables influence other variables.

3. Modeling

Describing a system in terms of the mechanisms that determine the values of the variables is fundamental to causal analysis. In order to apply the method of causal ordering to determine the causal structure in a model, each equation in the model must be a structural equation. Unfortunately, there is no simple formal answer to the question of how to know if an equation is structural.

Let us illustrate with a simple example of a condenser model how a choice of equations affect the causal ordering produced. A condenser has inputs of steam (STM.in) and cooling water (CWT.in) and outputs of condensed steam (STM.out) and warm cooling water (CWT.out) Only considering mass flow for now, one can write the following equations for the condenser. M_x stands for the mass of x, where x is one of the inputs or outputs of the condenser. c_1 and c_2 are some constants.

1. $M_{CWT.in} + M_{STM.in} = M_{CWT.out} + M_{STM.out}$	The overall conservation of mass.
2. $M_{STM.in} = M_{STM.out}$	Steam flow.
3. $M_{CWT.in} = M_{CWT.out}$	Cooling water flow.
4. $M_{CWT.in} = c_1$	Exogenous variable assumption.
5. $M_{STM.in} = c_2$	Exogenous variable assumption.

This set of equations is redundant as the equation (1) is a linear combination of (2) and (3). Each of the sets of equations, (1, 2, 3, 5), (1, 3, 4, 5), and (2, 3, 4, 5) is self-contained and will give rise to a different causal ordering. The equations themselves do not tell us which one of these three sets should be selected as the model for the device.

The problem here is that the equations above are not all structural equations. Concerning ourselves only with mass flows for now, we know there are two distinct mechanisms in the above situation, namely the flow of steam and the flow of cooling water. The equation (1) is clearly not a structural equation, because it mixes up the two mechanisms.

In general, given a device, we must choose from a large set of equations about the device only those equations that reflects our understanding of mechanisms to produce the a correct causal structure. Our solution to this problem is to have an explicit representation of mechanisms, from which equations can be systematically derived in such a way that only structural equations are produced.

4. Network Representation of Processes

We use a network representation of processes for the purpose of explicitly representing one's understanding of mechanisms underlying an equation model. A process network is a semantic net representation of active processes taking place in a device. Intuitively, processes are things that are happening in a device that give rise to the overall behavior of the device. Since we are working in the domain concerned with flows of matter and energy, processes are the happenings that produce the outputs from the inputs of the device. In particular, we represent processes in a device as flows of matter or energy which together are responsible for the overall input-output behavior of the device.

Figure 4-1 shows the process network for a turbo-generator. There are four different processes happening in the turbo-generator, namely the steam flow (denoted MF_{stm} in the figure), the flow of energy of the steam accompanying the steam flow (EF_{stm}), generation of electricity from the input steam energy (EF_{ele}), and heat loss into the atmosphere (EF_{htl}).

4.1. Nodes and Links

A process network contains the following four types of nodes, namely *device, material, energy*, and *process*. A device node represents the device whose behavior is described by the network. A material node represents a conceptually distinct body of matter such as different types of input and output matter of a process. An energy node represents a conceptually distinct body of energy, which is an input or output of a process. The internal energy of matter is represented as an energy node separate from the matter itself. Since a body of material always has internal energy, a material node must always have an associated energy node.

A process, as stated above, is modeled as a flow. There are material flows and energy flows. A process may be complex or simple: A complex process involves one or more other processes. A simple process involves no other processes. A process is said to *involve other processes* when the latter processes are integral parts of the former and can only exist as such.

Nodes in a process network are connected by five types of links; *has-processes* (links numbered 1 in the figure) *involves-material-flow* (2), *involves-energy-flow* (3), *source* (4), *destination* (5), and

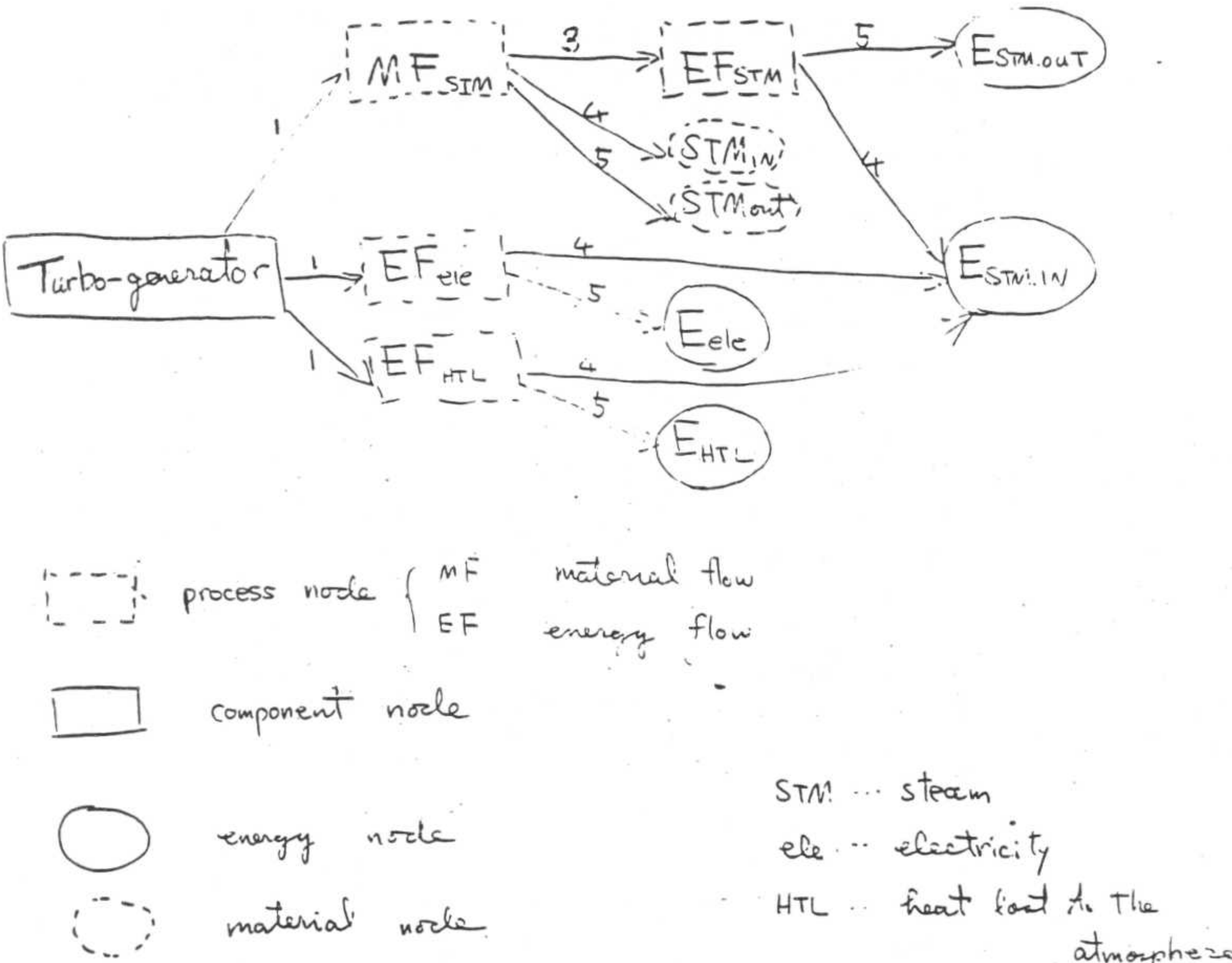

Figure 4-1: The process network for a turbo-generator

internal-energy (not shown in the figure to avoid visual clutter). A device node has has-processes links to process nodes in the network. A complex process has links of the types, involves-material-flow or involves-energy-flow, to the processes it involves. As a process is a flow, it has source and destination link to material or energy nodes. A body of matter and its internal energy are represented as separate material and energy nodes, where the material node has an internal-energy link to the energy node.

5. Generating Variables and Equations

Given a process network, relevant variables for each node in the network are automatically created. The information about what types of variables must be created for material nodes resides in the knowledge base, which contains general knowledge about different types of matter. Besides variables pertaining to material nodes, efficiency variables are generated for processes.

Once variables are generated, the system generates flow equations. For each material node that has n (>0) incoming flows, flow equation is generated, in the form of:

$$Mass = \sum_{i=1}^{n} flow-rate_i$$

where Mass is the mass variable of the node, and flow-rate$_i$'s ($1 <= i <= n$) are expressions for the flow rates of the flows into the node. An expression for the flow rate of a flow, F, is generated as follows:

If F is an active flow (i.e. has an efficiency variable),
"M_S * $EFFICIENCY_F$"
else
"$M_S \cdot \sum_{j-1}^{m} flow-rate_j$",

where S is the source node of F, M_S is the mass variable of the source, m is the total number of flows going out of S, $EFFICIENCY_F$ is the expression for the efficiency of the process F, and flow-rate$_j$'s ($1 <= j <= s$) are the expressions for the flow rates of the flows going out of S. Note that the above procedure will not terminate unless all but one of the flows sharing the same source are active processes. Likewise, energy flows equations are generated.

Besides flow equations, equations expressing characteristic properties of different types of matter and processes are generated by using general knowledge about the domain in the knowledge base. Likewise, equations to compute the internal energy of matter from its mass, and also those expressing explicit control over a variable exercised by external mechanisms are generated.

6. Results and Discussions

A program was written to generate behavioral equations from a process network as described above. The program was run on 6 such networks, which represent various components of a coal power plant to produce equations models. These networks are of complexities ranging from 2 to 13 processes, resulting in equation models of complexities from 9 to 26 variables. In 3 of the 6 cases, the program produced one more equation than the number of variables. Unless the situation modeled is a physically impossible one, the procedure should never produce a truly over-constrained system because it only produces equations that are true in the situation depicted. We discovered two types of problems causing the number of equations to exceed the number of variables.

The first case we consider is a case where the equations are not linearly independent. This problem arises for the following reason. The decision to represent material flows and energy flows separately in a process network was made based on the assumption that the two are independent. This assumption is generally true in the domain. However, cases where the energy flow is solely dependent on the material flow dq arise sometimes, and in these cases the system ends up producing redundant equations. This problem of redundant equations does not pose a fundamental difficulty to our approach and it was easily remedied by adding a checking step for this dependence when generating an energy flow equation. However, it serves to demonstrate that if one constructs a procedure to work on a certain representation of the domain, it is easy to forget the assumptions that went into making fundamental design decisions for the representation.

The second case where the equation model was over constrained was when an equation of a higher abstraction level was included in a model. For example, if a function representing a control that can only accomplished through some feedforward or feedback mechanism and that takes longer time to reach equilibrium are included in a model along with equations that describe functions that reach equilibrium in a much shorter time, one will end up with too many equations. Such an equation belongs to a more aggregate model than the original model, and one must be careful about mixing equations of different abstraction levels when constructing a model.

After the above problems were eliminated, our system produced causal dependency structures, which were in good agreement with our coal power plant expert's view of causal relations among the variables involved. As a solution to the problem of building a behavior model consisting of structural equations, our approach of explicitly representing processes from which equations are generated automatically seems successful. Though the usefulness of modeling processes as flows is limited to domains mainly concerned with flows of various things, they include many important domains. Also, this approach of automatically generating equilibrium models from explicit network representation of processes viewed as mechanisms acting on inputs to produce outputs seems generalizable to other domains dealing with other types of processes, though it remains to be demonstrated.

7. Current and Future Directions

A behavioral model produced from a process network as discussed in this paper is an equilibrium model. The methods of causal ordering and comparative statics provide an operational means to determine the causal relations among the variables in such a model and to assess the qualitative effects of a given disturbance to the system. However, if the situation modeled involves feedback, before one could determine the system's response over time to some given disturbance, one must determine stability of the dynamic behavior of the system before such assessment can be made. We are studying ways to incorporate into the system the capability to qualitatively analyze the stability of a dynamic model.

References

[1] De Kleer, J., and Brown, J.S.
Qualitative Physics Based on Confluences.
Artificial Intelligence 24, 1984.

[2] De Kleer, J.
How Circuits Work.
Artificial Intelligence 24, 1984.

[3] de Kleer, J., and Brown, J.S.
Theory of Causal Ordering.
Artificial Intelligence 29, 1986.

[4] Forbus, K.
Qualitative Process Theory.
Artificial Intelligence 24, 1984.

[5] Iwasaki, Y., and Simon, H.A.
Causality in Device Behavior.
Artificial Intelligence 29, 1986.

[6] Iwasaki, Y., and Simon, H.A.
Theory of Causal Ordering: Reply to de Kleer and Brown.
Artificial Intelligence 29, 1986.

[7] Kuipers, B.
Commonsense Reasoning about Causality: Deriving Behavior from Structure.
Artificial Intelligence 24, 1984.

[8] Kuipers, B.
Qualitative Simulation.
Artificial Intelligence 29, 1986.

[9] Samuelson, P.A.
Foundations of Economic Analysis.
Harvard University Press, Cambridge, MA, 1947.

[10] Simon, H.A.
Causes and Possible Worlds.
Boston Studies in the Philosophy of Science. Volume 54.*Models of Discovery: and other topics in the methods of science.*
D. Reidel Publishing Company, 1977, Chapter 2.1 - 2.4.

[11] Williams, B.C.
Qualitative Analysis of MOS Circuits.
Artificial Intelligence 24, 1984.