Compositional Model-based Design of Physical systems

Prasanta Bose **Shankar A. Rajamoney**

Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781
Email: pbose, rajamone@pollux.usc.edu

Abstract

Model-based design constructs physical systems in two stages. First, a *causal relation network* (CRN) of quantities that entails the desired behavior is constructed from a domain model. Second, a physical system is designed by assembling components such that all the causal relations specified by the CRN are imposed. The Compositional Model-based Design method, CMD, simplifies the design of complex physical systems by decomposing the specified behavior into logical portions, building CRNs for each portion, and incrementally composing the CRNs until the entire desired behavior is achieved. Importantly, the method detects potential interactions between individual CRNs that may nullify the portions of behavior already designed for by detecting violations of the closure assumptions under which each CRN was formed. The CRN is revised using operators derived from axioms that specify the conditions for a change to hold in the presence of such interference. While this paper illustrates the method in the context of a boiler control system, the approach applies to regulatory physical systems with multiple operating regions.

1 Introduction

In model-based design [11, 9], a physical system that meets input behavioral specifications is designed in two steps: 1) a network of causal relations, *causal relation network* (CRN), which entails the desired behavior, is constructed from a *domain model* describing components and their interactions, and 2) a design of the physical system, consisting of the components and their structural relations, is obtained from the design constraints that were instantiated in order to impose the causal relations in the constructed CRN representation. The intermediate CRN plays a pivotal role by explicating the causal paths that show how the designed physical system achieves the desired behavior. By bridging the specified behavior and the desired physical system, the causal relation network considerably simplifies the search for candidate design hypothesis.

However, the design of complex physical systems using this model-based approach is impractical since the construction of the CRN becomes computationally expensive due to the large number of components and interactions that must be considered. In this paper, we describe compositional model-based design (CMD), an approach for the model-based design of complex physical systems. This method adopts a divide-and-conquer strategy. It decomposes the specified behavior into logical portions, and separately constructs CRNs and

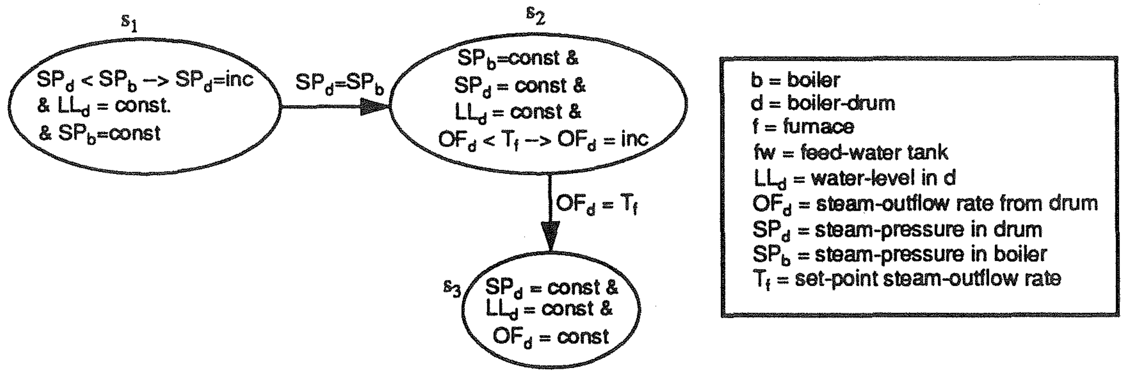

Figure 1: A partial qualitative state diagram.

their corresponding design fragments for each individual portion. The CRN (and the corresponding design) for the entire specified behavior is constructed by incrementally composing the individual CRNs and extending them, when necessary, to deal with interactions. The input to the CMD method is the desired qualitative behavior specified as a state diagram consisting of a partial sequence of states, each of which describes the relevant qualitative quantity changes (increase, decrease, or steady). The method outputs a design specifying the components of the physical system and their structural relations.

Figure 1 illustrates the behavioral specification of a system for pressure and level regulation in a steam plant. A design ([1]) that achieves the desired behavior is shown in Figure 2. In brief, the designed regulator functions as follows:

> Fluctuations in the load cause changes to the amount of steam in the turbine, thereby, causing changes to the steam outflow from the boiler drum. These changes are sensed and used to control the heat flow from the furnace to the boiler by changing the fuel-air ratio. Accordingly, the steam generation changes to balance the steam outflow from the drum, thereby, maintaining a constant pressure at the drum. Changes to the steam generation, however, affect the water-level in the drum; these changes are sensed by a level sensor which controls a valve regulating a compensatory inflow of water from a feedwater tank. Consequently, the level of water in the boiler drum is also maintained constant.

Compositional model-based design constructs such a complex system by first designing fragments which independently maintain the pressure and the water-level in the boiler drum steady and, then, by composing these two fragments together. The composition explicitly considers the potential interactions of the two separately designed fragments (e.g. the water-level may be disturbed when attempting to make the pressure at the drum steady).

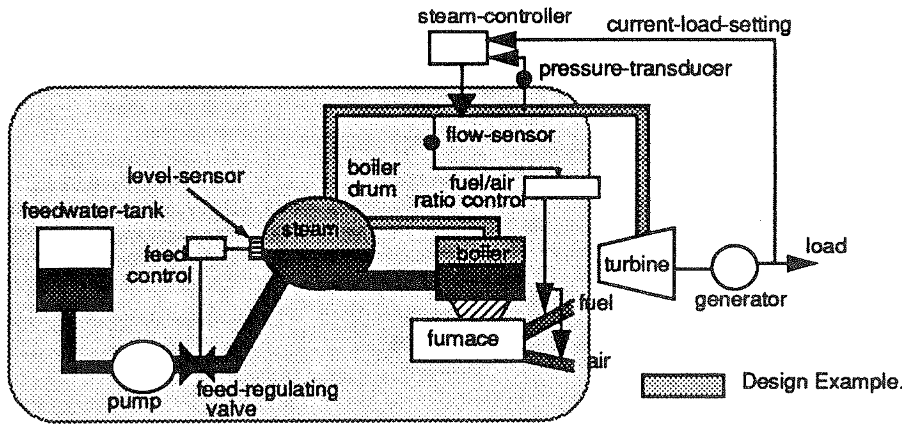

Figure 2: A boiler control system in a steam plant.

2 Knowledge Representation

The CMD method uses a domain model to construct CRNs. The domain model consists of a set of model fragments [4] that describe the domain objects and the physical mechanisms and processes. For example, in the fluids' domain, a domain model for fluids will describe objects such as liquids, gases, containers, valves, levers, and pumps, and physical processes such as heat flow, boiling, and fluid flow. We adopt the vocabulary of Qualitative Process Theory, *QP Theory*, [6] to represent the domain model. QP theory defines two types of causal relations between quantities: a) A direct influence between two quantities x and y, denoted as $x \propto_{I\pm} y$, which describes how changes in x are monotonically dependent on y. For example, the velocity of an object is directly (positively) influenced by its acceleration: if the acceleration is positive, zero or negative, the velocity is increasing, steady or decreasing, respectively. b) A qualitative proportionality between x and y, denoted as $x \propto_{Q\pm} y$, which describes how changes in y are monotonically dependent on changes in x. For example, the acceleration of an object is (positively) qualitatively proportional to the net force on the object: if the net force on the object is increasing, steady or decreasing, its acceleration is also increasing, steady or decreasing. Each model fragment consists of the *individuals* that constitute the model fragment, the *conditions* that must hold for the model fragment to become operational or *active*, and the *relations* that hold when the model fragment is active.

Examples of two model fragments, boiling and a path including a valve, are shown in Figure 3. The model fragment for boiling states that when a liquid is heated at its boiling point, it boils, resulting in a generation of steam at a rate qualitatively proportional to the heat flow rate. The model fragment for a valved-path states that the resistance offered by the path to a liquid flow is qualitatively proportional to the opening of the valve, which in turn is qualitatively proportional to the position of the valve mount.

The desired device behavior is specified by a partial qualitative state diagram describing the relevant device states and transitions. A device state includes: a) The qualitative values

Defmodel: Boiling
Individuals: ?liq, ?gas, ?heat-flow
Activity-Conditions:
temperature(?liq) > Boiling-point(?liq)
Relations:
Amount-of(?liq) α_{I-} Generation-Rate.
Amount-of(?gas) α_{I+} Generation-Rate.

Defmodel: Valved-path
Individuals: ?valve, ?valve-mount, ?path
Activity-conditions:
conn(?valve, ?path)
Relations:
opening(?valve) α_{Q+} position(?valve-mount),
path-resistance(?path) α_{Q-} opening(?valve)

Figure 3: Examples of model fragments for boiling and a valve.

of quantities (Qvals). The Qval of a quantity can be either a *landmark value*, a value at which interesting changes occur, or a value between two landmark values. b) Quantity changes in qualitative terms. The qualitative value of a quantity change can take any of three values: inc, dec, or const. c) Conditional quantity changes. A quantity change can be conditional on another quantity change or the change can be conditional on a Qval of the quantity.

An example of a partial qualitative state diagram, specifying the regulation of the steam-pressure, steam-outflow, and water-level in a boiler-drum of an idealized steam plant is shown in Figure 1. State S_1 includes a conditional quantity change, namely, if the steam-pressure, SP_d, in the boiler-drum, d, is less than the boiler-pressure, SP_b, the steam-pressure increases. Once the desired steam-pressure in the boiler drum is attained, the system must adjust the steam-outflow to attain the desired set-point steam outflow-rate of T_f. The final state, S_3, specifies that all the quantities are to be held constant once the desired steady-state values are reached. The water-level in the boiler-drum and the boiler-pressure are to be held constant in all the states. The state transition from S_1 to S_2 occurs when the steam-pressure in the boiler-drum reaches the boiler pressure. The state transition from S_2 to S_3 occurs when the rate of the steam outflow from the boiler-drum reaches T_f.

The designed device is specified by the components (e.g. open-container(P1)), and their structural relations (e.g. port-conn(bot(container-x,container-y, pipe-xy)).

3 Causal Relation Networks

A causal relation network describes how the causal relations imposed by the components and their connectivity form a causal path through which changes propagate from one quantity to another, ultimately producing a desired change in a quantity. The change in a quantity is determined from the changes in the quantities that have causal influence relations with the quantity. All the influences on a quantity are partial, and the total influence is determined by combining all the partial influences, specified by the influences in the influence closure (EC). The influence closure gives a complete enumeration of the simultaneous influences on the quantity under the *influence closure assumption* which states that there are no other influences other than those included in the EC. The ECs are made explicit in the CRN representation and play an important role in determining modifications of the CRN to handle

Let,

holds(Q_x, inc, S)== Qualitative value of change of quantity Q_x is *inc* in state S

dinfl(P, Q, V, S) == P is a direct influence on Q pushing it in direction V (inc or dec) in state S

[1.1] Establishing increase in quantity Q:

holds(Q, inc, S) $\wedge$ $G \neq 0 \Leftrightarrow$ net_infl(P, Q, inc, S)

<<Comment: Change of Q has a qualitative value of *inc* if the net influence of P on Q results in increasing Q. >>

[1.2] net_infl(P, Q, inc, S) $\Leftrightarrow$

a1) $\exists\ P_j \in P \wedge$ dinfl(P_j, Q, inc, S) $\wedge$

a2) [$\exists\ R_{j=1,k} \wedge R_j \in P \wedge$ dinfl(P_j, Q, dec, S) $\rightarrow$

$\exists\ P_{i=l,n} \wedge P_i \in P \wedge$

$\forall_{i,j},\ R_i \neq P_j \wedge$ dinfl(P_i, Q, inc, S) $\wedge$

dominates_dinfl($[P_1, ..., P_n]$,$[R_1, ..., R_k]$, Q, inc, S)]

<< Comment: Dominates-dinfl is a predicate which defines the conditions for one set of direct influences on Q to dominate over another set of direct influences causing a quantity Q to increase. >>

Table 1: A subset of the axioms for establishing increase in a quantity via use of direct influences.

interferences that arise due to composition.

A causal relation network representations consist of a set propositions that represent the following three types of information: i) qty changes and causal relations between the qtys that represents paths of interactions and lead to propagating a change to produce another change ii) the influence closures that enumerate the simultaneous set of causal influence relations relevant to a qty ii) the existence of model-fragment activities and their physical basis which impose the causal relations. For simplicity of illustration, the figures show only part 1 and 2. The design constraints are shown as part of the constructed design fragments. Given such representation, we define a set of axioms that specify a set of criteria that a CRN must necessarily meet in order to establish desired quantity changes and conditional changes. For example, the axioms given in Table 1 specify some of the criteria that a CRN must meet in order to establish an increase in a quantity based on use of direct influences.

A portion of a CRN for achieving an increase in the steam pressure of the boiler drum is shown in Figure 4. The CRN shows that the mass of the steam in the boiler drum is increasing since it is directly influenced by the rate of the steam inflow and, under the influence closure, EC_2 the direct influence due to the steam inflow is the only causal relation affecting it. The direct influence holds when the conditions for the steam inflow hold, namely, there must be a pressure difference between the boiler and the boiler drum, and the path connecting the two must permit a flow of steam. The increase in the steam mass is propagated through a causal relation causing an increase in the steam pressure.

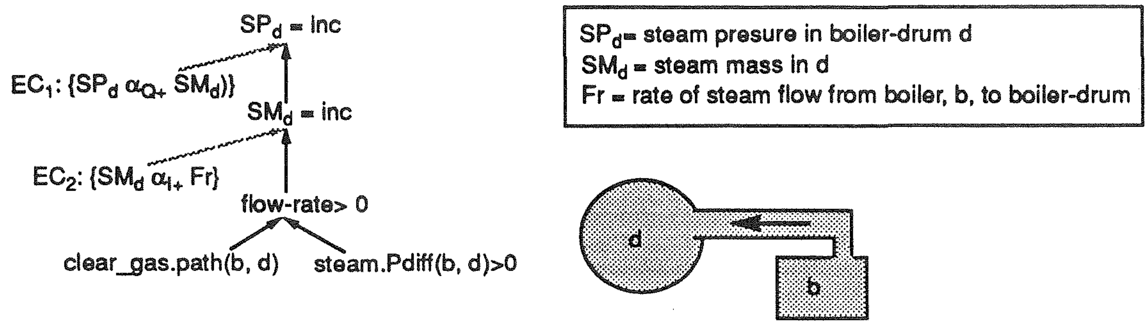

Figure 4: A CRN for increasing the steam pressure in the boiler drum.

4 Compositional Model-based Design: The Details

Compositional model-based design (CMD) decomposes the desired behavioral specification into three logical portions: 1) Individual unconditional or conditional quantity changes (e.g. $SP_d = \mathsf{inc}$ or $SP_d < SP_b \rightarrow SP_d = \mathsf{inc}$). 2) The quantity changes constituting a physical system state (e.g. $s_1 = \{SP_d < SP_b \rightarrow SP_d = \mathsf{inc}, LL_d = \mathsf{const}, SP_b = \mathsf{const}\}$). 3) The transitions between physical system states (e.g. $s_1 \rightarrow s_2$). Accordingly, the method consists of a cycle of three steps: 1) Constructing CRNs for each individual quantity change in the behavioral specification. 2) Composing the CRNs for the quantity changes constituting a physical system state. 3) Extending the CRNs for a system state to make each desired transition from that state. This last step may require additional quantity changes and causal relations to be achieved, in which case, the method iterates. Importantly, the *influence closures* play a fundamental role: they form the basis for a) detecting when the composition of two CRNs or the extension of a CRN may result in new CRNs that do not achieve the previously established behavior due to adverse interactions, and b) deciding how to revise such CRNs to re-establish the desired behavior. The body of the top-level algorithm is given in Table 2.

Step 1: Constructing a CRN for a quantity change

In this step, the CMD method independently constructs CRNs for each distinct conditional or unconditional quantity change in the qualitative state diagram. The method uses operators to propose domain model fragments that impose the causal relations for achieving a quantity change. The operators are based on axioms that specify the necessary truth conditions for a quantity change [2]. An example of such an axiom is given in Table 1. The influence closures constructed for each quantity change, gaurantee that the truth conditions are satisfied in the context of the constructed CRNs.

A CRN that achieves an increase in the steam pressure of the boiler drum in the boiler control example is shown in Figure 5. It is constructed using causal relations from the model fragments for a container, a contained gas, and a gas flow. According to this CRN, the increase in the steam pressure of the boiler drum is achieved by an increase in the steam's

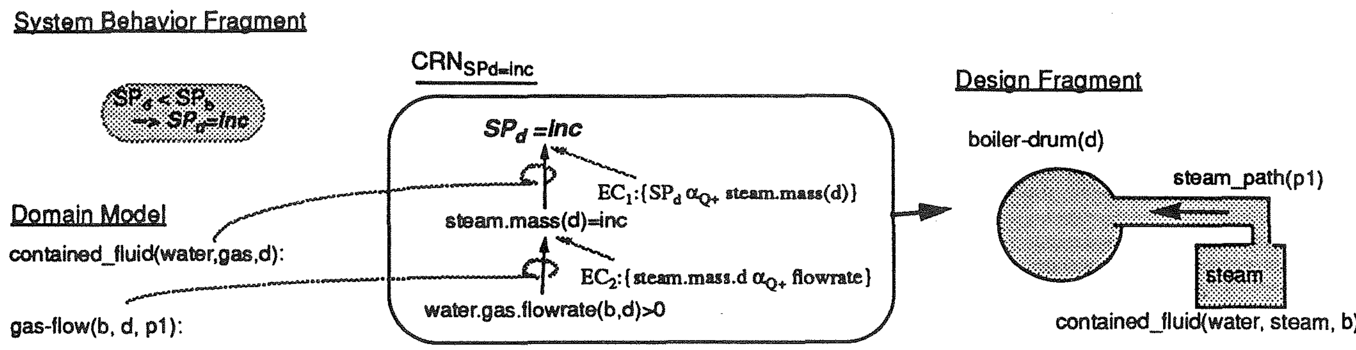

Figure 5: The construction of a CRN to achieve an increase in the steam pressure in the drum.

mass which, in turn, is caused by a gas flow (with flow-rate greater than zero) from the boiler to the boiler drum. A design fragment that imposes this CRN is also shown in the figure. Similarly, CRNs for other quantity changes stipulated in the qualitative state diagram (e.g. constant water-level in the boiler drum, constant pressure in the boiler drum, etc.) may be constructed. The CRNs for the constant water-level and constant pressure in the boiler drum are based on ECs specifying that there are no causal relations affecting these quantities (consequently, they remain constant).

Step 2: Constructing a CRN for each state

In this step, the method composes the CRNs for the individual quantity changes constituting a system state into a composite CRN that entails the system state behavior. A CRNs for each state is generated from the CRNs produced in the previous step, by composing the relevant CRNs for quantity changes in the state.

The ECs of the composite CRN are updated as follows: If under the composition of the design constraints, due to sharing of an existing component a quantity appearing in a CRN necessarily codesignates with a quantity in another CRN, then the EC of the quantity is a composite EC that consists of the union of the causal relations affecting the quantity, otherwise the EC remains unaffected. The method verifies that each of the updated ECs continues to satisfy the correctness requirements that must be met in order to achieve the individual quantity changes of the system state. If an individual quantity change no longer holds (due to adverse interactions), additional causal relations to re-establish the quantity change are added to the composite CRN by applying operators that propose the necessary revisions of the ECs. The operators are derived directly from a set of correctness axioms [2], a small subset of which is given in Table 1. An example of such an operator, stated informally is:

> If the desired quantity, Q, is to be held steady, and there exists a causal influence Q_x on Q that is causing Q to increase, then introduce another causal influence

Q_y on Q that pushes Q to decrease, such that Q_x and Q_y cancel each other.

Another means of preventing such interactions is by disallowing structure sharing. If attempts to repair the composite CRN fail, the method backtracks to an alternative candidate.

For the boiler-control example, Figure 6 illustrates how the CRNs of Figure 5 are composed to obtain a consistent composite CRN for the system state, s_1. The steam pressure in the boiler was previously established to remain constant by formulating an EC which specified that no causal relations affected it. However, in the composite CRN, the CRN for achieving an increase in the steam pressure in the boiler drum contributes a causal relation that affects the steam pressure in the boiler since, in that CRN, a flow of steam from the boiler to the boiler drum was established to increase the steam pressure in the boiler drum. Consequently, the composite CRN will achieve a decrease in the steam pressure of the boiler, instead of maintaining it constant (Figure 6[A]). There are two alternative design choices: i) alter the CRN to prevent the sharing of structure (by specifying non-sharing of the boiler with the source of the steam inflow into the boiler drum), or ii) introduce additional causal relations to cancel the effect of the steam flow, for example, by postulating a generation of steam to compensate for the loss due to the outflow from the boiler (Figure 6[B]). The steam generation rate in the boiler must be greater than zero and, consequently, the heat-flow-rate from the furnace must also be greater than zero (assuming that the temperature of the water in the boiler is at the boiling point). In order to maintain the steam-generation, the water in the boiler must be constantly replenished from another source. Choosing the boiler drum as the source leads to a revision of the EC of water-level is inconsistent with the requirement of maintaining the water-level in the boiler drum constant, and the CRN may be repaired by introducing a water inflow from the feedwater tank to compensate for the outflow to the boiler (Figure 6[C]). The revision process is repeated until all the ECs satisfy the truth conditions for the required quantity changes and hence the composite CRN achieves all the quantity changes of the system state.

Step 3: Augmenting CRNs to perform state transitions

The previous two steps result in CRNs that achieve the behavior in each individual state. In this step, the desired transitions from each state are achieved. State transitions are due to changes in the activity of model fragments in a state (e.g. a steam flow will become inactive when the pressures at the source and destination become equal, resulting in a transition to a new state). Accordingly, a transition from state s_i to state s_j requires dynamically transforming CRN_{s_i} to CRN_{s_j} by identifying the differences in the activity of the model fragments in the two CRNs. The enablement or disablement of these model fragments, as appropriate, will then lead to the desired transformation. The CMD method performs a state transition by: i) Identifying the model fragments in CRN_{s_i} that must be deactivated. ii) Identifying the model fragments in CRN_{s_j} that must be activated, and iii) Introducing causal relations that achieve the quantity condition changes for the activation and deactivation (e.g. to activate a gas flow, the pressure at the source must be made greater than that at the destination by introducing causal relations affecting one or both of the pressures). The method applies

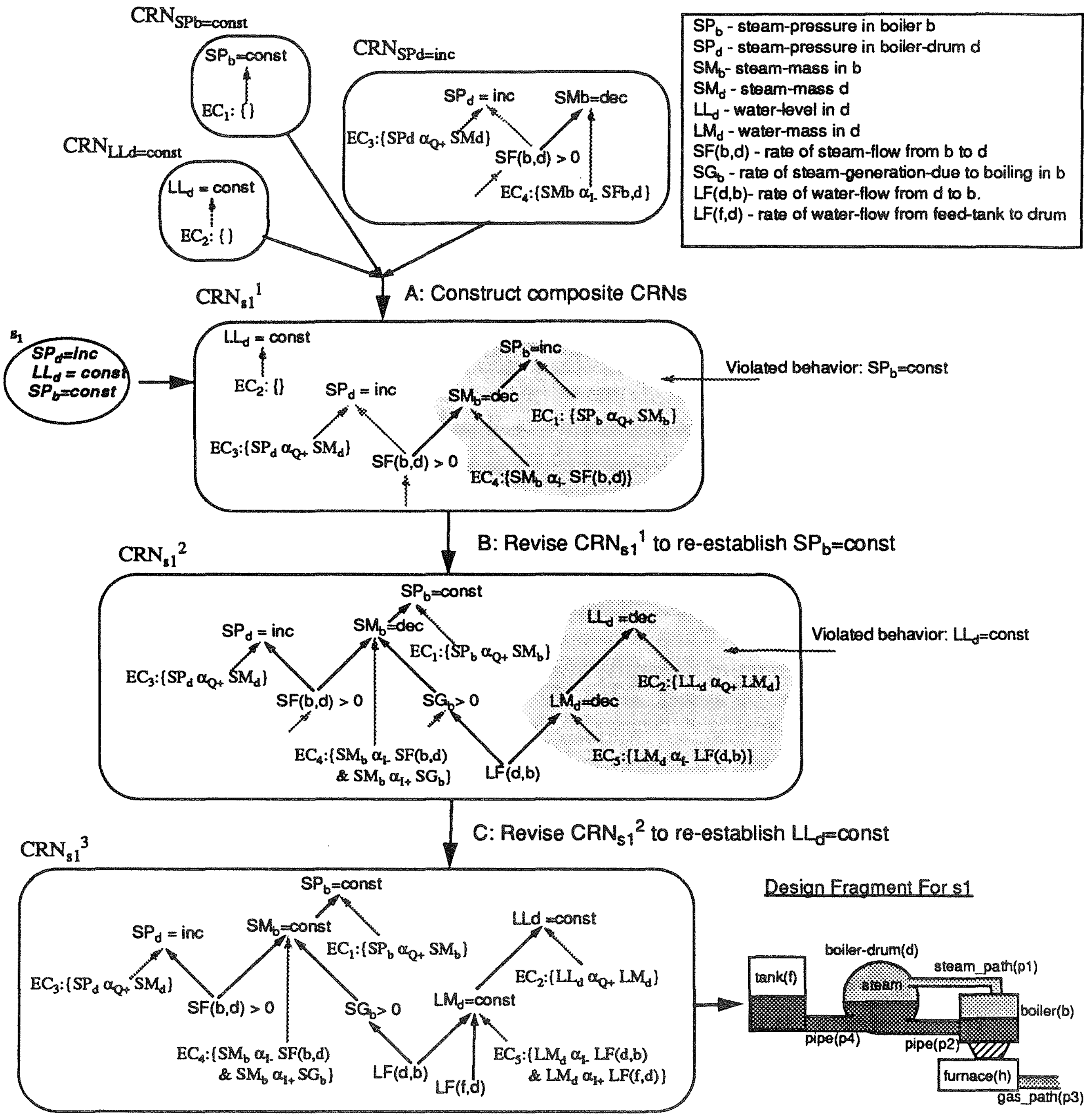

Figure 6: The construction of a composite CRN to achieve state s_1.

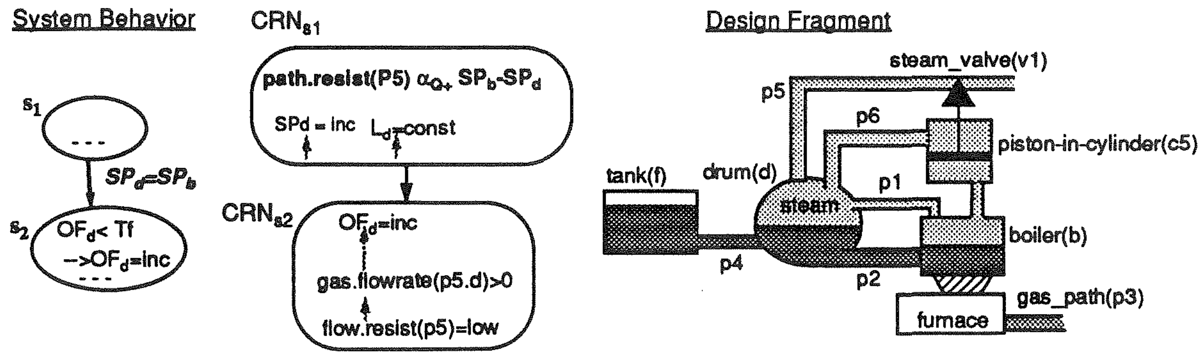

Figure 7: The extension of CRN_{s_1} to achieve a state transition from s_1 to s_2.

operators to make the necessary modifications to the CRN. An example of an operator for deactivating a model fragment is:

> If the activity condition for a model fragment M is $X > Y$ and M is active then M is deactivated by holding Y constant and decreasing X.

In the boiler-control example, the transition from s_1 to s_2 requires finding causal relations that activate a steam outflow from the boiler drum when the steam pressure of the boiler drum is equal to that of the boiler (Figure 7). Since the flow model fragment is conditioned on the path-resistance being less than or equal to some threshold value, it may be activated by decreasing the path-resistance (e.g. slowly opening a valve). The activation of the steam-outflow can be coordinated with the steam pressure of the boiler drum reaching the steam pressure of the boiler by using the changing pressure difference to decrease the path-resistance to some value below its threshold. Figure 7 illustrates how the causal relation may be achieved. The new causal relations and other quantity changes introduced in this step (e.g. decreasing the path-resistance) are similar to those initially achieved in Step 1, and are achieved by iteration over the entire method. The iteration terminates when all the posted quantity changes and transitions are achieved. Figure 8 shows the final CRN and the designed physical system.

5 Related Work

Williams' work [11] on *Interaction-based Invention*, another model-based design method, is closely related to our work. His method precompiles the domain theory into an abstract space describing the interactions between quantities. It constructs a CRN that establishes a desired functional relationship between two quantities by finding a candidate path of potential interactions in this abstract space and verifying with the detailed domain model that the path establishes the desired relationship. The CMD method, by checking for potential

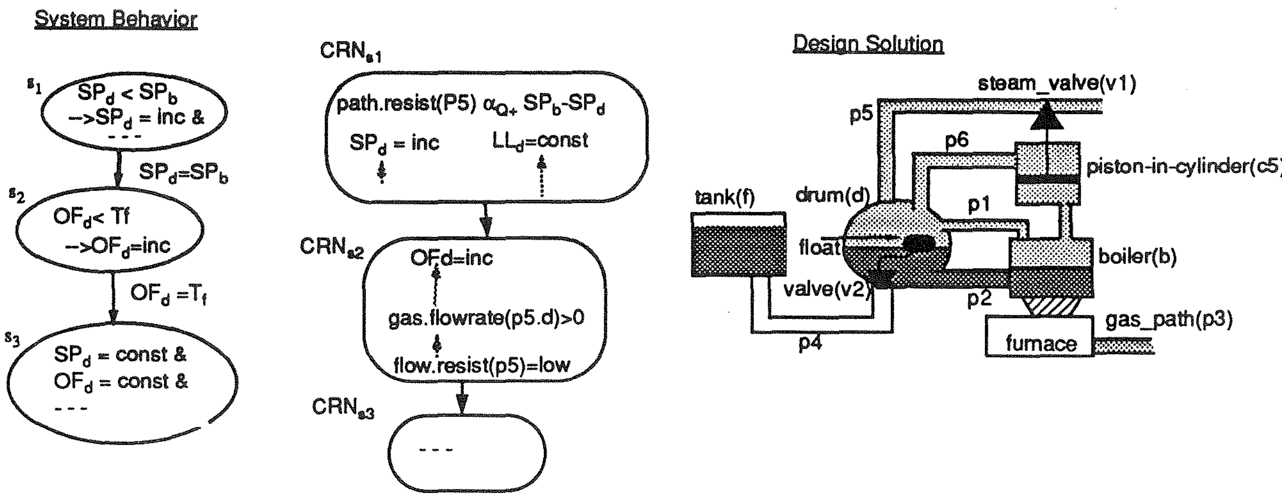

Figure 8: The final design solution that achieves the specified behavior.

interactions during the construction of the composite CRN, eliminates the need for a separate verification step. In addition, the ECs provide strong guidance for determining how to incrementally extend a CRN to achieve more of the desired behavior; in contrast, Williams' method, in extending partial paths, has limited guidance.

Ulrich [9] uses a bond-graph representation of component models to propose approximate design solutions which are refined using debugging operators. Unlike our method, his method does not construct an intermediate CRN which entails the desired behavior, and describes the causal relations that the design must impose.

Our use of truth conditions for ensuring correctness in the constructed designs is similar in spirit to the use of modal truth criteria in the planning work of Chapman [3]. Our truth conditions differ from his, in terms of having to consider interference that arise from simultaneous quantity changes and sharing of structure and function.

6 Summary

In summary, we described a compositional model-based design method for designing complex physical systems. The method decomposes the desired behavior into logical portions, constructs CRNs that achieve each of these portions independently, and incrementally extends the CRNs fragments until a composite CRN that entails the desired behavior is constructed. Finally, the method builds a physical system that imposes the causal relations of the CRN and, consequently, achieves the specified behavior. Interactions between CRNs are detected based upon violations of the necessary truth conditions by the ECs that result from the composition of CRNs, and are revised to maintain previously established behavior. The CMD approach has been implemented within the ADB rule-based framework [5] on a SPARC workstation. The method has generated designs for subproblems in the boiler-control example. In addition, it has been demonstrated in examples involving the design of control subsystems for chemical reactors and other multiple-operating region regulatory devices.

The current bottleneck in our approach is the problem of ensuring correctness of CRNs which requires checking for all potential interactions between CRNs. We are currently investigating how the search for correct design solutions can be controlled using abstractions of domain models [10, 8, 7, 4]. We are also evaluating the soundness and completeness of the method which depends on the axioms that specify conditions for necessary truths of quantity changes and quantity conditions in a state. In addition, we are investigating how the search conducted by the CMD method may be performed in a least-commitment manner by working with partial designs and extending them by posting constraints[3], a method that mixes well with our compositional approach.

References

[1] S. Addanki, R. Cremonini, and J. S. Penberthy. Graphs of models. *Artificial Intelligence*, 1991.

[2] Babcock and Wilcox. *Steam/Its Generation and Use.* Babcock and Wilcox Company, 1975.

[3] Prasanta Bose. *Design of Physical Devices: An Approach based on Partial Causal Networks.* Technical Report, University of Southern California, CA, 1991.

[4] David Chapman. Planning for conjunctive goals. *Artificial Intelligence*, 1988.

[5] Brian Falkenhainer and K. Forbus. Compositional modelling: Finding the right model for the job. *Artificial Intelligence*, 1991.

[6] K. D. Forbus. The qualitative process engine. In *Readings in Qualitative Reasoning about Physical Systems.* Morgan Kaufmann Publishers, 1990.

[7] Ken Forbus. Qualitative process theory. *Artificial Intelligence*, 1984.

[8] L. Joskowicz. Simplification and abstraction of kinematic behaviors. In *Proceedings of the IJCAI*, 1989.

[9] K. T. Ulrich. *Computation and Pre-Parametric Design.* AI-TR-1043, MIT AI Lab, Cambridge, MA, 1988.

[10] D. Weld. Approximation reformulations. In *Proceedings of AAAI-90*, 1990.

[11] Brain C. Williams. *Invention from First Principles via Topologies of Interaction.* PhD thesis, MIT AI Lab, Cambridge, MA, 1989.