A Theory of Mapping from Structure to Function Applied to Engineering Thermodynamics

John O. Everett
Qualitative Reasoning Group
The Institute for the Learning Sciences
Northwestern University
1890 Maple Avenue Evanston, IL 60201
U. S. A.

Abstract

This paper presents a theoretical framework for mapping from structure to function in engineering domains. We argue that a generative approach grounded in Qualitative Process Theory produces useful functional explanations. These explanations are articulate, in that they enable the user to explore their theoretical justifications and perform counterfactual reasoning. These explanations stem from a teleological representation based on goals, plans, roles, and views. We show that an ontology based on aggregated processes facilitates the recognition of recurring thermodynamic structures. We describe an implementation of this theory, a system called CARNOT that explains steady-flow thermodynamic cycles ranging in complexity from four to 24 components.

1 Introduction

Thermodynamic cycles (e.g., power plants, refrigerators) form an important class of artifacts. Devices based on them are complex and costly to operate, which provides several motivations for reasoning about them. Engineers and students need to verify that their designs will behave as desired, and plant operators need to generate and test hypotheses concerning system functions from schematics.

Each of these cases calls for reasoning about function given a structural description. This paper describes a theory of structure-to-function mapping that supports these tasks in the domain of thermodynamic cycles. We have implemented this theory in a system called CARNOT that takes as input a schematic depicting the structural configuration of a system such as a refrigerator and produces a description of the system's function, at both global and local levels.

de Kleer [1984] was the first to investigate the mapping from structure to function. He proposed, for the domain of electronic circuits, a methodology using qualitative physics to map from structure (what the artifact is) to behavior (what the artifact does) and a separate, teleological reasoning process to map from behavior to function (what the artifact is for). Thus the behaviors of a working turbine include expansion of the working fluid, cooling of the fluid and creation of shaft work. Its function, however, may be either to produce work or to cool the working fluid, and is contingent on the context in which it is embedded.

Despite the relatively greater constraints on the function of thermodynamic systems, we encountered significant ambiguities in mapping from the structure of thermodynamic cycles to their function. This paper describes how our theory resolves these ambiguities to produce teleological construals of a schematic. Section 2 presents an overview of the domain, Section 3 discusses our theory, Section 4 describes our representations, Section 5 outlines the algorithm, and Section 6 presents in detail one example and summarizes some of the more interesting results from other cycles CARNOT currently solves. We conclude with a discussion of related and future work.

2 Domain Overview

Artifacts incorporating thermodynamic cycles are pervasive. Virtually all electrical power generated today relies on a thermodynamic cycle in which massive boilers generate steam to turn turbines that drive generators. Refrigerators rely on essentially the same cycle, albeit running in reverse and supplied with a different working fluid that enables their operation at safer pressures. Automobile and jet engines operate in a so-called "open" cycle that takes in air from, and expels exhaust gases to the environment, yet they may be analyzed as cycles by treating the atmosphere as a single reservoir of air. Industry relies on thermodynamic cycles for power, for liquefying gases (e.g., natural gas, nitrogen, oxygen), and for process steam.

2.1 A Simple Heat Engine

The defining characteristic of a thermodynamic cycle is that it operates between two reservoirs of different temperatures, typically by passing a working fluid through a system of pipes and components. Figure 1 shows a simple cycle.

This basic cycle (with some modifications to increase efficiency) is commonly used to generate electricity. Heat energy obtained from combustion or nuclear reaction converts the working fluid into vapor in the boiler. This vapor

[Everett, 1995] J. Everett. A theory of mapping from structure to function applied to engineering thermodynamics. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (Montreal), pp. 1837-1843. San Mateo, CA: Morgan Kaufmann, 1995.

Simple Vapor-Cycle Heat Engine

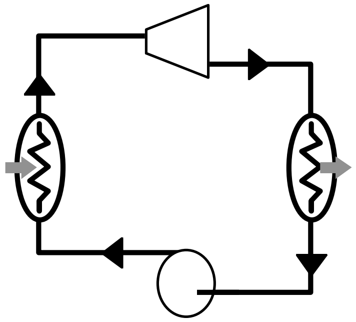

Figure 1

then expands in the turbine, causing its blades to rotate, producing work. The condenser returns the working fluid to its original state by ejecting heat to the environment. The pump ensures a steady supply of working fluid to the boiler and maintains the system's direction of flow.

Despite the fact that the constituent devices of this and other thermodynamic systems are complex artifacts designed to accomplish specific functions, we have found significant ambiguities in mapping from structure to function in this domain. For example, a turbine may function as either a work-producer or a cooler, and in cryogenic cycles the latter is the desired function. Reaching human-like conclusions with little information despite such ambiguity has been the primary motivation for the development of our theory and the design of CARNOT'S representations and algorithms.

3 Teleological Theory

The goal of this research is to automate the process of making functional inferences from structural information. To gauge our success, we need criteria for what constitutes a good functional explanation. We define such an explanation to be one that:

- Generates internally consistent construals
- Takes into account all available information
- Relates each device to at least one design goal
- Provides a certainty metric for each inference
- Enables counterfactual reasoning
- Grounds explanations in a qualitative theory of behavior

The value of functional explanations lies in the inferences that they sanction. A template-based approach could produce canned descriptions in great detail. However, such explanations would be unable to respond to user queries. A student might not understand a particular statement, and should therefore be able to backtrack through the inferential chain connecting that statement to domain theory givens.

To achieve such generativity, we minimize the size of knowledge fragments and rely on inference to assemble explanations. This avoids the redundant encoding of information endemic to template-matching approaches and enables the explanation of novel cycles. Modularizing the representation also facilitates the task of maintaining a knowledge-base large enough to support a practical teleological reasoner. Although our representation is primarily qualitative, it also supports the use of quantitative information for making more precise functional inferences.

Finally, to prevent explosive inferencing, CARNOT adopts what de Kleer (1984) calls the teleological perspective, in that we assume by default that each device contributes to the function of the system. This enables us to avoid extensive simulation because we assume that components operate within their normal parametric ranges.

4 Knowledge Representations

Perusal of thermodynamic texts and reference materials reveals no universal standards for schematics, although informal conventions do exist (e.g., turbines are generally depicted as trapezoids with vertical parallels). We use the schematic representation we designed for CyclePad (Forbus & Whalley, 1994), a system that enables students to design and experiment with thermodynamic cycles.

This representation reflects certain pedagogical considerations. To encourage students to consider modeling issues, only basic devices are explicitly represented. For example, there is no separate icon for a jet-ejector (a pump utilizing a high-velocity jet) because a mixer can function in this capacity. Devices are also constrained to a particular number of directional fluid ports by which they connect to other devices. Although in reality a turbine may have a large number of ports for bleeding steam, students must represent such turbines as sets of turbine stages connected by splitters.

CARNOT'S functional descriptions are composed of plans and roles. Plans summarize common structural configurations that have particular functional import. Roles specify which behavior of a particular component is its intended function. Intermediate view and process constructs enable the instantiation of the proper plans and roles. Views describe possible behaviors of particular devices, while processes ground explanations in a qualitative model of thermodynamics and provide a useful definition of locality. Figure 2 provides an overview of these representations. Arrows indicate constraining relationships. For example, the topology of the schematic determines which processes, roles, and views are instantiated. In the following description of these constructs, we will make reference to the cycle depicted in Figure 3. This cycle generates shaft-work by running steam through the five turbines across the top of the diagram. Because its efficiency is directly related to the average temperature at which heat is added, some of the

CARNOT'S Representations

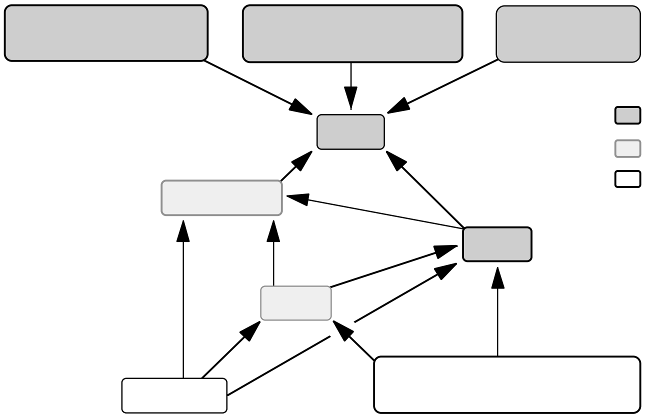

Figure 2

Regenerative Rankine Cycle

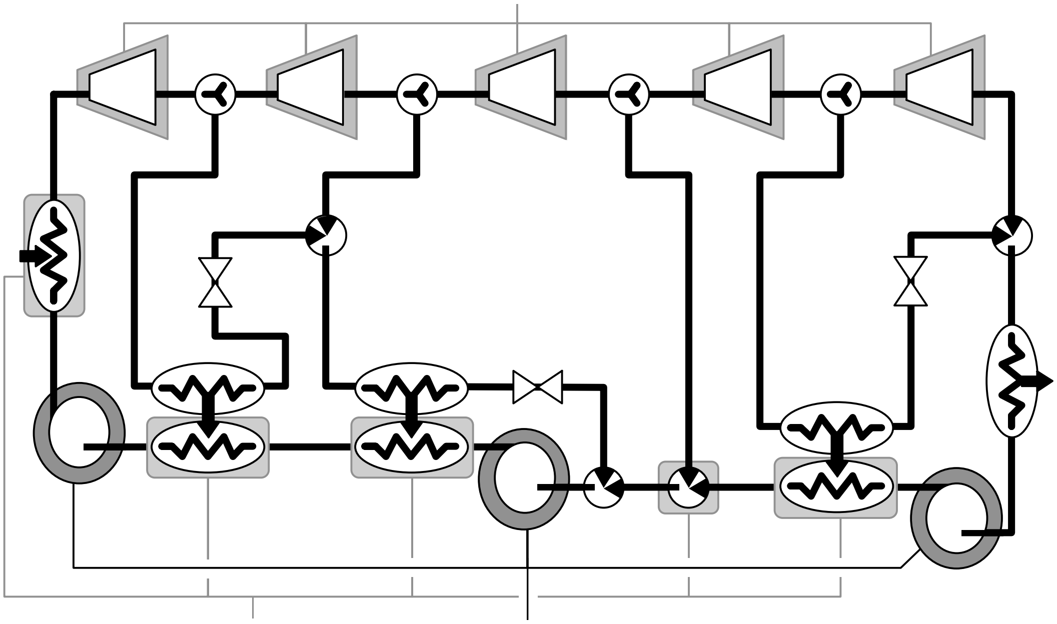

Figure 3

steam is bled from the turbine (via the four splitters) and used to preheat the feedwater flowing to the boiler.

4.1 Goals

The rational designer premise enables us to restrict our consideration to the set of goals that a rational agent would choose to pursue in the context of a design task. Of this set, we believe that three possible goals in particular provide a thorough characterization of the teleology of a system: (1) achieving a change of state in the environment, (2) doing so with a minimal input of energy, and (3) preserving the integrity of the system. In the case of a heat engine, the first goal is to convert heat energy into shaft-work, whereas in the case of a refrigerator it is to move heat from one location to another. The second goal arises from the assumption that the designer is under tight economic constraints. Finally, because systems that achieve the first two goals also create potentially damaging conditions, some devices may be present solely to prevent the occurrence of such states. For example, in Figure 3 Pump-3 acts to prevent the working fluid from vaporizing in the two heat-exchangers downstream of it, because most pumps cannot pump mixtures of liquid and gas. Should Pump-1 receive such a mixture, it would cease to supply water to the boiler, which would then fail.

4.2 Views

Views are device-specific behavioral descriptions. For example, views for a pump include (1) default, (2) coasting, (3) cavitating, and (4) losing. By default, CARNOT considers pumps to compress liquids (compressors compress gasses), so the default view sanctions inferences that input and output stuffs are liquid and that the input pressure is less than that of the output. Views thus propagate phase information, but they also serve to prevent devices known to be behaving abnormally from informing process and plan inferences.

4.3 Roles

Roles are the functional counterpart of views. For example, the potential roles of a pump include (1) flow-producer and (2) flash-preventer. The behavior of a default-view pump is to compress liquid; its function is to produce a flow. The difference is a presumption that this flow is essential to achieving one or more of the three design goals. A view is insufficient to support this presumption, because it is possible that the actual function is to act as a work-sink.

Although roles are device-specific, they generally require consideration of the structural context for a device, and thus more reasoning. Unlike views, roles are not always mutually exclusive; indeed, achieving multiple functions via a single device is often desirable from a design standpoint, for potential cost-savings and/or efficiency improvements.

4.4 Processes

Processes are central to thermodynamics; the components of a particular cycle exist solely to create and control them. Moreover, processes often span several devices, which may or may not be immediately adjacent. Reifying such processes provides CARNOT with a powerful definition of locality, as we shall see below.

There are three types of process: (1) local, (2) boundary, and (3) aggregate. Devices create one or more local processes across their fluid-paths. For example, a pump creates a local fluid-flow process from its inlet to its outlet.

We adopt the thermodynamic convention of establishing control volumes around systems and subsystems of interest. Control volumes require an accounting of all mass and energy crossing their boundaries. CARNOT explicitly notes all

Rankine Cycle Plan

- Vaporize working fluid at constant pressure
- Create a constant-entropy resisted expansion to produce shaft work
- Fully condense working fluid at constant pressure
- Pump liquid working-fluid at constant entropy to maintain flow direction

Figure 4

boundary-crossing processes. For example, the heat-flow to a boiler must cross the system boundary, so all heaters give rise to boundary heat-flow processes.

Aggregate processes provide a flexible means for matching canonical plans to cycles, because they capture critical aspects of a system without being overly sensitive to its particular topology. For example, Figure 3 shows the aggregate pumping, heating, and expansion processes that arise from the three pumps, five heaters (Mixer-2 is an open heat-exchanger, as we will see below) and five turbines. This cycle is therefore identical to the simple cycle of Figure 1 when we consider it in terms of aggregate processes.

4.5 Plans

Certain thermodynamic configurations recur so often that their idealized abstractions have been reified. For example, most electrical power generating systems use some variant of the Rankine cycle, around which a working fluid is vaporized and condensed. We refer to such named configurations as plans because they are in effect well-known strategies for the realization of design goals. Figure 4 shows the content of our representation of the Rankine cycle plan. Other common plans include the Carnot cycle, a theoretical ideal, and the Brayton cycle, used for jet engines.

Idealization simplifies analyses by assuming certain state parameters remain constant across the plan's processes. The ideal Rankine cycle is comprised of constant-pressure (isobaric) heating and cooling processes and constant-entropy (isentropic) expansion and compression processes.

CARNOT distinguishes truly ideal from stepwise-ideal processes. The latter occur when the creation of two processes is interleaved. For example, the pumps and heaters of Figure 3 are interleaved, obviously preventing the aggregate heating process from occurring at constant pressure. However, if each constituent local process is ideal, then CARNOT labels the aggregate as stepwise-ideal. This distinction enables CARNOT to differentiate practical from ideal cycles. An ideal cycle maximizes efficiency even at the cost of failing to preserve system integrity. Such cycles are useful for pedagogical reasons and as benchmarks for assessing the efficiency of practical cycles.

CARNOT'S plans vary in generality. The most general are the heat-engine and refrigerator plans, which have no ideal-process requirements. The more information CARNOT is given, the more specific the plans instantiated. CARNOT includes likely plans in the final description with a caveat that their antecedents must be true. CARNOT also recognizes inter-cycle plans, such as CASCADE-CYCLES and USE-WORK-INTERNALLY. In cascaded systems one cycle uses the heat ejected by the other, while in systems that combine heat-engine and refrigerator cycles, the heat-engine's work drives the refrigerator.

5 CARNOT'S Algorithm

CARNOT uses a logic-based truth maintenance system [Forbus & de Kleer, 1993] coupled to a pattern-directed rule engine. CARNOT'S knowledge base is encoded as a set of rules. The underlying TMS caches the resulting chains of inference, enabling CARNOT to perform counterfactual reasoning and to construct causal explanations on demand.

CARNOT alternates between propagating local inferences and global processing based on these inferences. Figure 5 summarizes the algorithm. CARNOT first instantiates models for each device it finds in the input cycle, ensuring that at least one view is active for each device. It then topologically parses the cycle into fluid loops and searches for a globally consistent functional labeling of each device. CARNOT then decides which of the fluid loops are thermodynamically meaningful subcycles, and infers the function of each subcycle. Finally, CARNOT creates aggregate process assertions and uses them to infer the presence of plans. The following explains the algorithm in greater detail.

5.1 Instantiating Domain Knowledge

CARNOT first instantiates a set of device models that describe the structure of the input system and result in the instantiation of views. In some cases there isn't enough initial information for a particular device to have an active view. For these devices CARNOT instantiates the most specific view consistent with the known information.

For example, a heater default view makes no commitment about the phase of the stuff at inlet or outlet. However, if CARNOT detects only compressors (which can only compress gasses), it will assume a Gas-Heater view, which implies that the phase of the stuffs at inlet and outlet is gas.

Algorithm Step	Example of Result

1. Assert propositions describing system	`(DEVICE TURBINE tur-1 s10 s20)`
	`(tur-1 HAS-INLET s10)`

2. Run rules to instantiate first consequences of description	`(tur-1 HAS-VIEWS (default eroding stalling))`

3. Identify fluid loops in cycle	`(FLOOP fl-1 (pmp-1 htr-1 tur-1 clr-1)`
	`(s10 s20 s30 s40))`

4. Create a consistent view structure for system	`(VIEW DEFAULT HEATER htr-1)`
5. Do dependency-directed search for roles	`(ROLE mxr-1 JET-EJECTOR)`

6. Identify routes in system	`(SUBCYCLE subc-1 (pmp-1 htr-1 tur-1 clr-1)`
	`(s10 s20 s30 s40))`

7. Refine view structure in light of new information	`(VIEW EVAPORATING HEATER htr-1)`
8. Aggregate local processes	`(PROCESS AGGR EXPANSION (tur-1 tur-2 tur-3))`
9. Run rules to identify plans	`(PLAN RANKINE-CYCLE subc-1)`

Figure 5

5.2 Identifying Topological Structures

CARNOT next parses the cycle topologically into floops (short for "fluid loops"). These are directed cycles in which neither arcs nor vertices are duplicated. CARNOT breaks floops immediately upstream of the first compressing device to be found after the last expansion device. It does so because the working fluid is closest to ambient conditions here. Automobile engines, which operate in a so-called "open" cycle, break the cycle at this point, taking in working fluid (i.e., air) immediately prior to compressing it, and exhausting it immediately after the power stroke.

Floops do not necessarily correspond to meaningful substructures in the input cycle, but merely represent routes that a piece of working fluid could traverse during the steady-state operation of the cycle. There are five floops in Figure 3, corresponding to the outermost loop (and only subcycle) and the routes originating at the four splitters. CARNOT generates a hypothesis concerning the function of each floop, which is potentially a heat-engine, a refrigerator, or an artifact of the cycle's topology.

A mixer can act as either a simple route-joiner, a heat-exchanger (if its two inputs are of different temperature) or a pump (if its two inputs are of different pressure). A splitter may either act as a route-divider or a flash-chamber, in which the working fluid evaporates, the gas leaving by one exit and the remaining liquid by the other.

This functional ambiguity means that valid subcycles may lack apparent pumps, expansion devices, heaters or coolers. To identify such floops, CARNOT uses the constraints shown in Figure 6 to conduct a dependency-directed search for a consistent set of views of each floop's devices. These constraints follow from the rational-designer premise; there is no thermodynamically sound reason to immediately undo a change. For this search, CARNOT generates sets of potential roles for each mixer and splitter on the floop, ordered such that any solutions that allow the default mixing and splitting roles of those devices will be found first.

On completion of the search, CARNOT generates a refrigerator, heat-engine, or topological-artifact hypothesis based on either the order of devices in the floop or the presence of devices which could accomplish the essential compression, heating, expansion, and cooling processes. Device order, although more persuasive evidence than mere presence, is not a certain predictor of floop type. CARNOT therefore asserts hypothesis statements that contain the inferred floop type, the justification for the inference (ORDERED or ALL-PRESENT) and the set of role assumptions required for that floop type to pertain. CARNOT postpones committing to a hypothesis, however, because this requires non-local reasoning and can be made with greater certainty later.

This search procedure only resolves situations in which a device must play a certain role. However, Mixer-2 in Figure 3 need not play a heat-exchanger role. CARNOT resolves the roles of such devices via qualitative inference.

Functional Labeling Constraints

- Processes are considered neighbors if they are consecutive on a particular route or if they are connected by one or more splitting and/or mixing processes.
- Heating and cooling processes cannot be neighbors.
- Expansion and compression processes cannot be neighbors.

Figure 6

Identifying an Open Heat-Exchanger via Inequalities

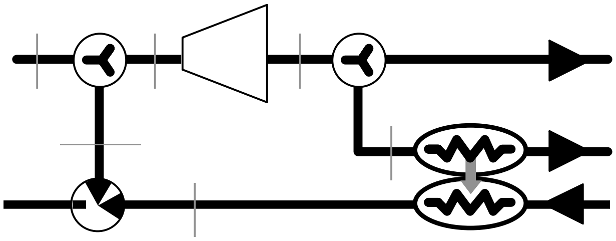

Figure 7

5.3 Resolving Roles via Qualitative Inference

Roles depend on the context in which the device is embedded. For the jet-ejector and open heat-exchanger roles, this context is limited to the state of the mixer's inputs; a temperature difference across the inputs indicates an open heat-exchanger, while a pressure difference implies a jet-ejector. When CARNOT instantiates its knowledge of mixers, it also expresses interest in finding inequalities in either pressure or temperature across the mixer's inputs.

Once CARNOT has identified the system's floops, the information necessary to find these inequalities, should it exist, will be present in the database. At this point, CARNOT attempts to assert an inequality statement for the identified stuff parameters via transitive reasoning. For example, in the cycle fragment of Figure 3 shown in Figure 7 the fact that the mixer is an open heat-exchanger can be qualitatively deduced. The transitivity reasoning proceeds as follows:

1. No temperature drop across a default-view splitter gives $T(A) = T(B) = T(X)$.
2. Temperature drop across a default-view turbine gives $T(B) > T(C)$.
3. No temperature drop across a default-view splitter gives $T(C) = T(D)$.
4. $T(D) \geq T(Y)$ because perfect heat transfer in the default-view heat-exchanger would make $T(D) = T(Y)$.
5. By transitivity, $T(Y) \leq T(C)$ and thus $T(Y) < T(B)$.
6. Because $T(X) = T(B)$, we can deduce that $T(Y) < T(X)$, thus satisfying the conditions for an open heat-exchanger.

5.4 Identifying Subcycles and Paths

CARNOT now attempts to re-parse the input system into a set of mutually exclusive and collectively exhaustive subcycles and paths, the latter starting at splitters and ending at mixers. However, CARNOT relaxes this "no-gaps/no-overlaps" constraint when a hypothesized heat-engine shares structure with a hypothesized refrigerator, because SHARED-STRUCTURE is a known plan for achieving the goals of maximizing efficiency and ensuring reliability.

CARNOT uses these heuristics to identify subcycles:

1. Should a floop exactly subsume two or more floops, consider only the subsumed floops.
2. Floops that have no structure in common with other floops are considered subcycles, as are lone heat-engine and refrigerator floops.
3. Of a set of floops sharing structure, choose the puta-

Rules For Composing Aggregate Processes

- Heating processes
  - May have an arbitrary number of intervening pumping, mixing, and splitting processes
  - Last local heating process must be downstream of last local pumping process
  - No intervening cooling, expansion, or throttling processes
- Cooling processes
  - May have an arbitrary number of intervening mixing, splitting, and expansion processes
  - No intervening heating processes
- Compression processes
  - May have an arbitrary number of intervening heating, mixing, and cooling processes
  - First local compression process must be upstream of the first heating process
  - No intervening expansion processes
- Expansion processes
  - May have any number of intervening heating, cooling, splitting, and mixing processes
  - Last local expansion process must be downstream of last heating process
  - No intervening throttling or compression process

Figure 8

tive heat engine with the greatest number of work-sources, or choose the putative refrigerator with the greatest number of heaters (i.e., refrigerator coils).

CARNOT now accepts or rejects the type hypothesis for each subcycle. If all the views required for the hypothesis to hold are true, then CARNOT simply records the relevant type statement. If a single view is false, CARNOT rejects the hypothesis. When one or more view statements are unknown, CARNOT assumes in turn that each unknown is true and looks for any resulting contradictions in its knowledge of the system. Should such a contradiction occur, CARNOT retracts the view and asserts that both the view and the hypothesized type cannot mutually pertain. Otherwise, CARNOT accepts the hypothesis.

Rules for Instantiating Rankine Cycle Plan

```
(RULE
  ((:true (PLAN HEAT-ENGINE ?name ?cmp
               ?htg ?exp ?clg) :var ?p1
             : test (every #'liquid-pump?
                           (cadr ?cmp))))
 (let ((pumps (make-pump-exprs ?cmp)))
   (assert! (:implies (:and ?p1 pumps)
              (PLAN VAPOR-POWER ?name ?cmp
                            ?htg ?exp ?clg))
       :vapor-power-cycle-inference)))

(RULE
  ((:true (PLAN HEAT-ENGINE ?name ?cmp
                ?htg ?exp ?clg) :var ?p1)
   (:true (PLAN VAPOR-POWER ?name ?cmp
                ?htg ?exp ?clg) :var ?p2)
   (:false (INTERLEAVED ?cmp ?htg) :var ?p3)
   (:false (INTERLEAVED ?htg ?exp) :var ?p4)
   (:false (INTERLEAVED ?exp ?clg) :var ?p5)
   (:true (ISOBARIC ?htg) :var ?p6)
   (:true (ISOBARIC ?clg) :var ?p7)
   (:true (ISENTROPIC ?exp) :var ?p8))
  (assert! (:implies (:and ?p1 ?p2 ?
                       (:not ?p3) (:not ?p4)
                       (:not ?p5) ?p6 ?p7 ?p8))
    (PLAN RANKINE IDEAL ?name ?cmp
                        ?htg ?exp ?clg))
      :ideal-Rankine-cycle-inference))
```

Figure 9

5.5 Aggregating Processes and Inferring Plans

The set of active views determines what processes are considered to be active. For example, a boiling process is only active if its associated heater is viewed as a Boiling-Heater.

As mentioned above, aggregate processes arise when two or more devices operate to produce a single effect. CARNOT aggregates local processes according to the set of heuristics shown in Figure 8, which are based on the rational-designer premise; there is no physical law enforcing these constraints, but violating them would serve no thermodynamic purpose, and in fact be at odds with one or more of the three teleological goals CARNOT imputes to an input system.

The assertion of plans is a local propagation based on the current set of active aggregate processes and other information cached in the database. Figure 9 shows the rules that instantiate an ideal Rankine cycle plan.

6 Examples

We present here CARNOT'S explanation (translated from the predicate calculus) of the cycle in Figure 3, and conclude this section with a brief description of other examples.

- The system is a heat engine. Given stepwise isentropic expansion in the turbines and stepwise isobaric heating in the heaters, a practical Rankine cycle, because it is a vapor power cycle. It is a vapor power cycle because it fully condenses its working fluid.
- Turbines 1-5 create the resisted expansion process of the system. Heaters HX-1, MIXER-2, HX-2, HX-3 and BOILER create the heating process. Pumps 1-3 create the compression process. CONDENSER creates the cooling process.
- MIXER-2 is an open heat-exchanger because the fluid from splitter SPL-3 has a higher temperature than the fluid from HX-1. This is done to achieve the design goal of MAINTAIN-SYSTEM-INTEGRITY, because an open heat-exchanger removes contaminants from the working fluid.
- Pumps PUMP-2 and PUMP-3 may act to prevent the working fluid from flashing. This would achieve the design goal of MAINTAIN-SYSTEM-INTEGRITY, because flashing would cause downstream pumps to cavitate, cavitation would cause the pump's fluid-flow-rate to decrease, and a decrease in fluid-flow to the boiler would cause the boiler to melt. [This inference is uncertain because it is based solely on the cycle's topology; PUMP-2 and PUMP-3 have both heaters and pumps downstream of them, so it is possible that the removal of either pump would enable a downstream heater to cause the fluid to flash into vapor. Given numeric information, CARNOT determines whether this would actually occur].
- Heaters HX-1, MIXER-2, HX-2 and HX-3 preheat the working fluid. This achieves the design goal of MAXIMIZE-SYSTEM-EFFICIENCY, because a Rankine Cycle's efficiency is directly related to the average temperature of heat addition.

Other cycles that CARNOT explains include the simple heat engine of Figure 1, a simple refrigerator, a subcooling refrigerator, a heat-driven refrigerator utilizing either a turbine and compressor or a jet-ejector, a heat-driven air-conditioning system, an intercooled gas-turbine, and a combined gas-turbine/vapor-power cycle in which the latter utilizes the waste heat of the former to increase efficiency. Each of these cycles presents a particular challenge to achieving a consistent mapping. For example, both heat-driven refrigerator systems consist of two subcycles, a heat-engine and a refrigerator, that share structure (a common condenser). CARNOT correctly identifies the two subcycles and infers that the shared structure is a plan to achieve the goal MAXIMIZE-SYSTEM-EFFICIENCY by reducing complexity and cost. In the heat-driven air-conditioning system, there are three mixers, one of which acts as a jet-ejector, and three splitters, one of which acts as a flash-chamber. CARNOT correctly identifies both of these roles. Finally, in the combined cycle, CARNOT identifies both subcycles as power cycles and correctly infers that the vapor cycle is present to achieve the goal of MAXIMIZE-SYSTEM-EFFICIENCY.

7 Related Work

Chandrasekaran has developed a theory of Functional Reasoning that is consistent with the work presented here. He has proposed that teleological knowledge be encoded in Causal Process Descriptions (CPDs) that are represented as directed graphs whose arcs are causal links [e.g., Chandrasekaran, 1994]. CARNOT'S knowledge base is organized along similar lines, although we prefer not to encode the causal links explicitly, and instead allow the inference engine to instantiate them as they become relevant, via the view and role mechanisms.

Vescovi, Iwasaki, Fikes, and Chandrasekaran have proposed a modeling language, CFRL, for integrating qualitative and functional reasoning [Vescovi et al, 1994]. CFRL composes a qualitative model from model fragments and then attempts to fit a causal story (encoded in CPDs) to a particular trajectory through the qualitative state space. Because thermodynamic cycle analysis is steady-state, we have been able to avoid the complexities arising from such explicit temporal reasoning.

Franke has proposed a rigorous language for teleological description (TeD) [Franke, 1993] that may in the future provide us with useful formalisms as we extend CARNOT. He approaches the issue of teleology from the designer's point of view, while CARNOT attempts to infer the intentions of the designer after the fact, given only the artifact.

Narayanan, Suwa and Motoda have described a system that predicts the operation of simple mechanical devices given a labeled schematic [Narayanan et al., 1994]. Their system also produces explanations but focuses more on visual reasoning, whereas CARNOT'S input is construed as a set of devices in a particular structural configuration.

8 Conclusion

We have described a set of teleological representations consisting of goals, plans, roles, and views that enable the production of functional explanations of complex thermodynamic cycles grounded in a qualitative domain theory. We have also shown that aggregate processes provide a powerful heuristic for recognizing cycles despite structural variations.

We believe the generativity of our approach will enable it to scale up to explain any thermodynamically valid system. CARNOT now explains all eight of the steady-flow cycles contained in an introductory text [Whalley, 1992], and 24 of the 32 cycles in a more comprehensive text [Van Wylen & Sonntag, 1985]. Explaining non-steady-flow systems, such as Otto and Diesel cycles is our next goal. We believe that some improvements to the algorithm combined with roughly a one-third increase in CARNOT'S current rulebase (which now contains about 140 rules) will enable the explanation of the thirty-five cycles contained in *Analysis of Engineering Cycles* [Haywood, 1980], considered to be the definitive text on thermodynamic cycles[1].

As an initial test of its capabilities, we plan to incorporate CARNOT into the coaching module of a thermodynamics tutoring system. We also intend to test the applicability of this theory to other domains, such as hydraulics.

Acknowledgments

This work has been supported by the Computer Science Division of the Office of Naval Research.

References

[Chandrasekaran 1994] B. Chandrasekaran. "Functional Representation and Causal Processes." *Advances in Computers*, Vol. 38. ed. Marshall Yovits. Academic Press.

[de Kleer 1984] Johan de Kleer. "How Circuits Work." *Artificial Intelligence* 24:205-280.

[Forbus 1984] Kenneth D. Forbus. "Qualitative Process Theory." *Artificial Intelligence* 24:85-168.

[Forbus & de Kleer 1993] Kenneth D. Forbus and Johan de Kleer. *Building Problem Solvers*. MIT Press, Cambridge, MA.

[Forbus & Whalley 1994] Kenneth D. Forbus and Peter B. Whalley. "Using Qualitative Physics to Build Articulate Software for Thermodynamics Education." In *Proceedings of the Twelfth Annual AAAI Conference*, Seattle, WA.

[Franke, D 1993] David Franke. *A Theory of Teleology*. Ph.D. dissertation, The University of Texas at Austin.

[Haywood 1980] Richard Haywood. *Analysis of Engineering Cycles*, 3rd edition. Oxford: Pergamon Press

[Narayanan et al., 1994] N. Hari Narayanan, Masaki Suwa and Hiroshi Motoda. "How Things Appear to Work: Predicting Behaviors from Device Diagrams." In *Proceedings of the Twelfth Annual AAAI Conference*, Seattle, WA. AAAI Press.

[Van Wylen & Sonntag 1985] Gordon Van Wylen and Richard Sonntag. *Fundamentals of Classical Thermodynamics*, 3rd edition. New York: John Wiley & Sons.

[Vescovi et al., 1993] Marco Vescovi, Yumi Iwasaki, Richard Fikes and B. Chandrasekaran. "CFRL: A Language for Specifying the Causal Functionality of Engineered Devices." In *Proceedings of the Eleventh Annual AAAI Conference*, Washington, DC. AAAI Press.

[Whalley 1992] Peter B. Whalley. *Basic Engineering Thermodynamics* Oxford: Oxford University Press.

[1] P.B. Whalley, personal communication.