Functional and Teleological Knowledge in the Multimodeling Approach for Reasoning About Physical Systems: A Case Study in Diagnosis

Luca Chittaro, Giovanni Guida, Carlo Tasso, and Elio Toppano

Abstract—**This paper first presents the basic concepts of the multimodeling approach to the representation of physical systems. The key point of this approach is the exploitation of many, diverse models of a system for the execution of complex problem solving tasks, such as interpretation, diagnosis, design, simulation, etc. The considered models are based on different ontologies, representational assumptions, epistemological types, and aggregation levels. After a brief survey of the techniques adopted for representing structural and behavioral knowledge, attention is focused on function and teleology. A novel approach is proposed for defining, representing, and using these two types of knowledge which play a fundamental role both from the representation and reasoning perspectives. The fundamental claim here is that while teleological knowledge concerns the specific purposes for which the system has been designed, functional knowledge is devoted to bridge the gap between such abstract purposes and the actual structure and behavior of the system, through the concepts of phenomena, processes, and functional roles. Moreover, the paper provides a clear definition of all the various epistemological and ontological links existing between the different models, which allow the execution of complex reasoning activities by cooperation among all the models. Finally, the proposed approach is applied to the specific task of diagnosis. An experimental system called DYNAMIS is described, which has been used as a testbed for original diagnostic strategies exploiting the problem solving power offered by the multimodeling approach and, in particular, by functional and teleological knowledge. Three sample diagnostic sessions with DYNAMIS are illustrated, which focus on the issues of operator diagnosis, diagnosis focusing, and functional conflict recognition.**

I. Introduction

MODEL-BASED reasoning has been, in the last decade, a very active research field. The major efforts have concentrated on paradigms for qualitative modeling and qualitative reasoning about physical systems [81]. Milestones of this research field are the component-based approach of ENVISION [25], the process-based approach of qualitative process theory [32], and the constraint-based approach of QSIM [49], [51], [54]. All these approaches have exploited structural and behavioral knowledge in order to support a variety of tasks, such as prediction, diagnosis, causal explanation, and design [9], [45], [82]. A commonly referenced principle for device modeling using structural and behavioral knowledge is the "no function in structure" principle [25], which states that the behavior of a system should be described without considering its potential functionalities or intended use, i.e., in a context independent way. Accordingly, past work on qualitative modeling does not deal explicitly with function and teleology. The only exception is represented by the teleological analysis proposed in [24]. More recently, some researchers deliberately renounced the "no function in structure principle" and focused on theories of qualitative reasoning based on the use of knowledge about functions and goals [28], [33], [47], [48], [71], [75]. The main claim of these approaches is that functional and teleological knowledge can provide important additional information for understanding and reasoning about the structure and the behavior of a system.

At the same time, increasing attention was devoted to the issue of cooperation of multiple models of the same system in order to improve the effectiveness and efficiency of reasoning processes. Literature proposals span a wide range of possibilities and perspectives: using models of different aggregation levels [22], [36], [58], [66] or featuring different approximations [3], [50], [80]; using different ontologies [21], [39], [57], [65]; representing variable values at multiple resolutions and using different behavioral qualitative and quantitative models [35], [61], [64]; and, finally, using multiple types of knowledge to support specific reasoning tasks, such as supervision and control [35], [67], diagnosis [1], [16], [31], [60], and design [10], [15], [33], [37]. These research directions have been strongly supported by cognitive motivations. In fact, experimental activity in the cognitive field provides substantial evidence that human experts typically use multiple representations in complex problem solving tasks and are able to switch from one representation to another whenever appropriate [5], [17], [52], [53]. Several open issues still exist with these approaches. First of all, no clear and organic framework has been pro-

Manuscript received January 9, 1992; revised October 17, 1992. This work was supported in part by "Progetto Finalizzato Robotica" of CNR (Italian National Research Council) under Grant 91.01966.PF67.

L. Chittaro, C. Tasso, and E. Toppano are with the Department of Mathematics and Computer Science, University of Udine, Via Zanon 6, 33100 Udine, Italy.

G. Guida is with the Department of Electronics for Automation, University of Brescia, Via Branze, 25124 Brescia, Italy.

IEEE Log Number: 9209636.

posed, so far, to support a disciplined, effective, and coherent representation of different models of the same system. Although some progress in this direction has been made—consider, for example, the work reported in [29], [30], and [55]—the proposed solutions are weak from a methodological point of view. None of the above mentioned approaches takes the issue of cooperation of many, diverse models as the epistemological foundation for the design of general and powerful reasoning mechanisms, nor exploits the consequences of multiple modeling with full generality. More specifically, the question of how partial results obtained within a model can be exported to other models, in order to progress in the problem solving activity, has been tackled only in very specific situations.

In recent years we have proposed a novel methodology for the representation of physical systems, called *multimodeling* [11], [13], [18], [19], [38]. This approach is based on the key idea of considering the task of reasoning about a physical system as a cooperative activity which exploits the contribution of many diverse models, each one encompassing a specific type of knowledge and representation. In this frame, several critical problems have been studied, including the design of a general framework suitable for supporting multiple modeling, the representation languages appropriate for the various models, the specific reasoning tasks which can be performed within each model, and the cooperation strategies used to integrate the contributions of the different models into the overall problem solving task. Particular attention has been devoted to the investigation of the relationships among different models: the links between different representations, the ways for ensuring that the set of available models consistently describe the physical system considered, the mechanisms for exporting partial results from the model where they have been obtained and importing them into the model where they will be exploited.

In this paper we focus on the representation and use of teleological and functional knowledge in the frame of the multimodeling approach. Although it is widely recognized that teleological and functional knowledge play a fundamental role in understanding the behavior of physical systems and reasoning about them, the problem of how to represent and use it effectively has been confronted only in a partial and inadequate way. Moreover, among the approaches exploiting the use of multiple models, the consideration of functional and teleological knowledge is relatively new and no cohesive theory has been proposed yet to integrate these models together with structural and behavioral knowledge. The main goal of this paper is to illustrate the nature of functional and teleological knowledge, discussing the relationships existing between function and teleology, and their role in supporting complex reasoning tasks.

In order to illustrate the advantages and limitations of the multimodeling approach in a realistic task we have considered the diagnostic problem. Diagnosis has been widely studied in the past in the field of model-based reasoning and several approaches have been proposed [22], [23], [26], [36], [68]. Model-based diagnosis has been proposed to overcome the deficiencies (brittleness, device dependence, limited explanation power, lack of generality, etc.) shown by the early diagnostic systems based on surface knowledge [20], [41], [46], [59], [72] by exploiting principled knowledge about the structure and behavior of the system. In this frame, a primary role has been played by the consistency based theory of diagnosis [26], [27], [68], [77]. This theory which defines a diagnosis to be a minimal set of components—assuming that each of these components is faulty—together with the assumption that all other components are behaving correctly, is consistent with available observations about actual system behavior. Considerable research has been focused on the computation of all possible diagnoses using a model of the system to be diagnosed, which accounts only for the normal behavior of its components. Although this approach to model-based diagnosis provides an elegant and general framework for diagnostic reasoning, a number of open problems remain to be addressed. First, model-based diagnosis, focusing only on structural and behavioral models, features a very high computational complexity when applied to real world systems with hundreds or thousands of components. Second, the consistency based definition of diagnosis may provide diagnoses which are logically acceptable but physically meaningless [77]. Third, the consistency based approach is scarcely plausible from a cognitive point of view: modeling principles and reasoning algorithms proposed are only loosely constrained by the way human experts actually solve diagnostic problems. In the multimodeling approach, the use of functional and teleological knowledge, in addition to structural and behavioral, allows some of the above limitations to be overcome by providing more abstract mechanisms for focusing diagnosis, for avoiding erroneous (i.e., not physically founded) diagnoses, and for accomplishing a better cognitive adequacy.

II. The Multimodeling Approach

A. Motivations and Requirements

Several of the approaches so far proposed for the representation and reasoning about physical systems suffer from two main limitations:

1) Although based on formally sound and powerful theories, they are scarcely plausible from a cognitive point of view: modeling principles and reasoning algorithms are only loosely constrained by the way humans actually solve problems. Therefore, their problem solving power is often spoiled by their scarce capability of producing acceptable justifications, both in the case of successful reasoning and of failure.
2) Although supported by effective reasoning strategies in specific problem domains of limited complexity, they are generally weak at dealing with large and complex systems, as most real artifacts and natural systems usually are.

We claim that these two limitations are not independent from each other: adding a sound cognitive background to the way computer programs reason about physical systems can greatly contribute to improve, not only their transparency and their cognitive coupling, but also their effectiveness and efficiency. Accordingly, the following specific *requirements* turn out to be of primary importance for a computer program devoted to reasoning about a physical system:

- *representation adequacy*—the capability of appropriately supporting the representation of many diverse, large, and highly structured knowledge sources (models) about a physical system;
- *problem-solving power*—the capability of exploiting the various knowledge sources available in a cooperative and effective way for problem solving tasks;
- *problem solving economy*—the property of using only the relevant and necessary knowledge in any step of a problem solving task;
- *multiple use of knowledge*—the possibility of exploiting the same knowledge sources for a large variety of problem solving tasks, such as interpretation, diagnosis, design, simulation, etc.;
- *cognitive coupling*—the capability of ensuring that the computational models and reasoning mechanisms implemented in the system can be assumed as plausible (partial) representations of the mental models and processes of a human reasoner;
- *effici ncy*—the capability of dealing with large and complex problems using limited computing resources.

The main point in this list is certainly the requirement about the availability and use of a large variety of knowledge sources (models) about the physical system which is the object of the reasoning activity. In fact, it is widely recogni ed that no single model is adequate for a wide range of problem solving tasks. For example, design optimization problems may be very difficult using structural or behavioral knowledge, but can be greatly simplified resorting to functional knowledge, which can support reasoning about alternative system structures [33]. On the other hand, however, behavioral and structural models are needed for all types of analytic tasks, such as simulation or diagnosis [10], [22], [25]. Moreover, it is known [67] that adopting a single model generally conflicts with an economic use of knowledge and a high cognitive coupling. Finally, experience has shown that the computational effort needed to achieve a specific goal (for example, in the case of envisioning, the prediction of all possible behaviors of a system) grows very fast (exponentially, in the example considered) with the number of variables of the model. Efficiency cannot be achieved, in general, using only one model: an appropriate problem decomposition and the cooperation of a variety of knowledge sources organized at different levels of aggregation, and accessible under appropriate views is possibly the only way of adequately coping with complexity issues [22], [29], [30], [36], [76]. According to the above stated requirements, we have proposed in recent years a novel methodology for the representation of physical systems called *multimodeling* [11], [18]. The multimodeling approach is characterized by the representation of many diverse, explicit models of a system which are used in a cooperative way in specific problem solving tasks as it will be illustrated in the following sections. Note that the fundamental assumptions about knowledge modeling and reasoning mechanisms, which characterize the multimodeling approach, do not identify a unique way of representing a physical system and reasoning about it. On the contrary, the multimodeling approach is an abstract and general framework which allows for a variety of concrete implementations. When a specific application domain is considered, several decisions remain to be made in defining the particular instance of the multimodeling approach which is considered appropriate for the considered case.

B. Knowledge Modeling

The concept of *model* we assume here is that of a symbolic system designed to provide a representation of a physical system appropriate for a given purpose. So, a model is only a partial representation of reality and depends on subjective decisions of the model designer. In particular, modeling requires going through four fundamental choices concerning ontologies, representational assumptions, epistemological types, and aggregation levels. These concepts are illustrated in detail below.

1) Ontologies: Building a model requires a commitment about the kind of entities we assume the real system is made up of and that the model must designate. This decision defines the *ontology* of the model. We distinguish between two main ontologies:

1) *Object centered ontology* assumes that reality is made up of individual objects whose properties can be stated in an objective, context independent, and general way. This ontological perspective enforces modularity and reusability of the representation. According to the granularity of the individual objects we assume reality is made of, we further distinguish between 1) *macroscopic ontologies*, which assume that reality is made up of individual macroscopic objects, and 2) *microscopic ontologies*, which assume that reality is made up of elementary individuals at an atomic or molecular level.
2) *System centered ontology* assumes that reality is made up of systems, intended as organized units, whose elements cannot be defined in isolation. This ontological perspective enforces the representation of a system in terms of specific and context dependent properties.

The component-centered approach [25] to qualitative modeling of physical systems is a good example of a theory based on a macroscopic object-centered ontology. The charge-carrier ontology [57] and the microscopic theories used in [65] for reasoning about physical systems at the molecular level are examples of microscopic object-centered ontologies. Examples of the use of a system-centered ontology can be found in the functional modeling approaches [48], [67], [71].

In some cases, both ontologies may be present in a model, such as for example, in the process-centered approach [32], developed on top of the component-centered approach [25] by modeling explicitly not only individual components, but also the processes acting on them. An ontological perspective which mixes aspects of both object-centered and system-centered ontologies is called a *hybrid ontology*.

Models of the same system based on different ontologies can be related to each other through *ontological links*, which explicitly connect corresponding knowledge elements in different ontologies.

2) Representational Assumptions: Another early decision to be made in the modeling activity concerns what to represent of the real system in the model. This decision involves two basic aspects:

- the *scope* of the model, i.e., the aspects of the real system which are considered relevant to the purpose of the model and, therefore, must be included in the representation;
- the *precision* of the model, i.e., the degree of accuracy of the representation.

These choices are referred to as *representational assumptions*. Different representational assumptions lead to different models of the same system which are called *approximations*. Note that scope and precision are independent dimensions of an approximation: we may have models with the same scope but featuring different precision (for example, a behavioral quantitative model of a system and its qualitative version), and models having the same precision but different scopes (for example, models which consider or do not consider friction, or models which assume or do not assume the hypothesis of rigid body). Models of the same system based on different representational assumptions can be related to each other through *representational links*, which explicitly specify i) the representational assumptions that must be added or retracted in order to switch from one approximation to another [3], and ii) the relationships connecting corresponding knowledge elements in different approximations.

In our approach, the scope of a model is organized according to physical views. The concept of *physical view* [76] represents a feature of knowledge organization that allows the indexing of the elements of a representation according to the physical perspectives they are related to, for example, thermal, electrical, mechanical, etc. Physical views allow the reasoning process to focus only on those parts of the scope of a model which are relevant to the current problem solving step, discarding other useless details.

3) Epistemological Types: By *epistemological type* we mean the class of epistemological features the model represents about the real system. We consider five epistemological types:

- *Structural knowledge*—knowledge about system topology. This type of knowledge describes which components constitute the system and how they are connected to each other (their adjacency).
- *Behavioral knowledge*—knowledge about potential behaviors of components. This type of knowledge describes how components can work and interact in terms of the physical quantities, (variables and parameters) which characterize their state and of the laws which rule their operation.
- *Functional knowledge*—knowledge about the roles components may play in the physical processes in which they take part. This type of knowledge relates the behavior of the system to its goals, and deals with functional roles, processes, and phenomena.
- *Teleological knowledge*—knowledge about the goals assigned to the system by its designer and about the operational conditions which allow their achievement through correct operation. This type of knowledge concerns the high level reasons which are behind the system concept and which have determined its actual structure.
- *Empirical knowledge*—knowledge concerning the explicit representation of system properties through empirical associations. This type of knowledge may be derived from observation, experimentation, and experience, and may include, in particular, the subjective competence that usually human experts acquire through direct interaction with a system.

Let us note that the five epistemological types defined above can be approximately grouped into three categories. Structural and behavioral knowledge are *fundamental knowledge*, i.e., basic knowledge used to reason about a system using the objective and neutral language of natural sciences. Functional and teleological knowledge are *interpretative knowledge*, i.e., knowledge derived from a subjective interpretation of fundamental knowledge in terms of functions and goals of system components. This knowledge does not have the same generality and objectivity of fundamental knowledge. For example, when we say "component X is devoted to . . ." we express a relationship between a system component (its structure and behavior) and a goal, which is generally not valid for other components of the same type in the same or other systems. Finally, *empirical knowledge* is a separate category which concerns explicit statement of system properties and may refer to both fundamental and interpretative knowledge.

Models of the same system based on different epistemological types can be related to each other through *epistemological links*, which explicitly connect corresponding knowledge elements in different models.

4) Aggregation Levels: By *aggregation level* of a model we mean the degree of granularity of the represented knowledge. For example, a structural model of a plant may be represented at the level of major subsystems or may be further refined at the level of elementary components. Of course, for a physical system (once representational assumptions have been defined and specific ontologies and epistemological types have been chosen) several models featuring different aggregation levels may generally be identified.

Models of the same system based on different aggregation levels can be related to each other through *aggregation links*, which explicitly connect corresponding knowledge elements in different models.

In the multimodeling approach any choice about ontology is allowed, as well as any kind of representational assumptions, epistemological types, and aggregation levels. The only restrictions we impose to the organization of models are the following:

1) Models are *separate*—any individual model may encompass only one specific choice about ontology (possibly a hybrid ontology), representational assumptions, epistemological types, and aggregation levels;
2) Models must be *interconnected*—any individual model must be explicitly and appropriately interconnected to the others with appropriate ontological, representational, epistemological or aggregation links.

As far as 1) is concerned, note that its primary motivation—in addition to the generic issue of modularity—is the requirement of multiple use of knowledge. In fact, according to the specific problem solving task considered, different types of knowledge may be useful in different moments and with different roles, and, therefore, their representation must be as far as possible separate. Note that this assumption is not shared by several authors who allow knowledge of different epistemological types (for example, structural and behavioral or functional and behavioral) to be mixed together within a single model [16], [25], [32].

As far as 2) is concerned, its primary motivation is the requirement of effective use of available knowledge in a cooperative way. Note that models are not required to be complete: they can represent only parts of a system.

C. Reasoning Mechanisms

The execution of a problem solving task (for example, interpretation, diagnosis, design, simulation, etc.) within the multimodeling approach is based on two fundamental mechanisms:

1) *Reasoning inside a model*, which exploits knowledge available within a single model by using *basic reasoning utilities* provided by the model; and
2) *Reasoning through models*, which supports opportunistic navigation among models in order to allow each individual step of the problem solving activity to exploit the most appropriate knowledge source.

The overall reasoning process is then constituted at the domain level, by a sequence of "reasoning inside a model" steps which are guided at the control level, by a "reasoning through models" activity, which continuously monitors and directs the use of knowledge at the domain level. This clearly requires that appropriate mechanisms for translating (exporting and importing) partial results from one model to another are available.

The control regime of the "reasoning through models" mechanism is determined by two main types of knowledge which are described below.

1) Knowledge About the Tasks the System is Requested to Solve: The multimodeling approach is, in principle, suitable to reasoning about a large variety of *tasks*, such as interpretation, diagnosis, design, etc. When the reasoning mechanism is tailored to a specific application, knowledge about the involved task and the relevant *task-specific problem solving methods* has to be provided [14], [74]. These are used for decomposing the task at hand into subtasks until *elementary tasks* are identified, which can be solved by exploiting appropriate *models-based problem solving methods* specifying the execution of basic reasoning utilities provided by the models.

2) Knowledge About Effective Exploitation of Available Domain Level Knowledge: During system operation several control problems must be solved concerning the most suitable use of models. These include:

- Choosing the most suitable model to be used for solving an elementary task, when several alternative models (for example, based on different aggregation levels or on different representational assumptions) are available (*initial model selection*);
- Selecting a new model where the reasoning activity can continue after a failure has occurred during the execution of a basic reasoning utility (*failure-driven model selection*);
- Monitoring the execution of a models-based problem solving method and deciding when it might be appropriate to switch from the currently used model to another which is supposed to be more suitable to continue the reasoning activity (*opportunity-driven model selection*);
- Determining the appropriate focus of attention for the current step of the reasoning process by activating the relevant physical views and choosing the most appropriate part of the model to work upon (*focus control*).

For all these problems, appropriate control knowledge must be provided when a specific application is developed.

D. Comparison of the Multimodeling Approach With Related Work

In recent years several research efforts have been spent to address the issue of physical system modeling with multiple representations. However, most of these approaches focus only on one type of models such as behavioral models [55], or are stated in very general terms [2], or refer to specific tasks such as design [37]. A proposal which is close to ours is compositional modeling by

Falkenhainer and Forbus [29], [30]. Compositional modeling is a technique for organizing multigrain, multiperspective models of physical systems (more specifically of physical phenomena) in order to manage complexity. Although we share the same broad intention, there are important differences between the two approaches:

1) Compositional modeling mainly focuses on modeling. The goal is to automatically and dynamically construct a system model (called a scenario model) by instantiating and composing together general purpose domain models and using the current task for guidance. Although automatic modeling is a very important issue, our work is, at present, mainly focused on the different issues of integration and cooperation between different models.
2) Compositional modeling proposes an organization of domain knowledge that allows controlling granularity, ontology, approximation, and perspectives (perspectives are similar to views) by the use of simplifying assumptions. However, the framework is limited to the consideration of only fundamental knowledge: other types of knowledge such as teleological and empirical are simply not allowed.

Recently, some authors [10], [33], [79] proposed a representation approach for physical systems which, like multimodeling, maintains a clear separation between knowledge of structure and behavior on one side and knowledge of function or purposes on the other side. Furthermore, these approaches, like multimodeling, are domain independent and can be reused unaltered for several applications including design, simulation, diagnosis, and explanation. The main difference is in the absence of a physical foundation for the relationships existing among the different models, more specifically, the relationship between behavior and teleology which, in our approach, is represented by functional knowledge (see Sections V and VI). In particular, the FBS (Function Behavior State) Diagram proposed by Umeda *et al.* [79] as a modeling strategy, although based on some common goals (for example, to rigorously define concepts of function and behavior, to provide a unified framework in which fundamental and interpretative knowledge are integrated, to deal with structural and functional hierarchies), does not consider the problem of reasoning with multiple models as the central issue of the approach. Moreover, the proposal is stated in general terms and lacks an explicit theory for the representation of systems, i.e., a specification of which modeling primitives and which relations should be used to represent both fundamental and interpretative knowledge, how these two types of knowledge have to be related, etc.

III. Fundamental Knowledge: Structural and Behavioral Models

A. The Structural Model

The *structural model* focuses on the topology of a system; it describes which parts constitute the system and their interconnections [18]. Interconnections are intended here in the general sense of physical adjacencies, i.e., possible pathways for interaction [22]. The structural model is based on the object-centered ontology and it is represented using the following three primitives:

1) *Components* represent the physical entities that constitute the system and determine its behavior. Examples of components are solid objects such as electronic devices or mechanical parts, but also fluid entities such as the air between the sheets of an insulator, or the water flowing in a hydraulic circuit. Components have *terminals*, which are passive channels supporting possible interactions with the outside environment. A terminal supports just one kind of physical interaction (for example: thermal, electrical, mechanical, etc.) which identifies its *type*.
2) *Nodes* are used to connect together two or more components. A node has two or more *terminals*, all of the same type. Nodes do not correspond to any physical entity present in the modeled system: their only purpose is to provide an explicit representation of the possible connections among components.
3) *Connections* describe how components are connected together through appropriate nodes. More specifically, a connection is a declaration of identity between two terminals of the same type: one of a component and one of a node. Connections are undirected links.

The structural model of a system is made up of a collection of components and nodes: each node terminal must be linked through a connection to a component terminal of the same type.

In order to illustrate the above concepts let us introduce here a sample system which will be used throughout Sections III, IV, V, and VI. The system considered is a simplified version of a temperature gauge [62]. The schema of the device is represented in Fig. 1. It consists of a battery, a switch, a wire, a bimetallic strip, a thermistor, and a pointer on a scale. Its operation is quite intuitive. The temperature to be measured is sensed by the thermistor which is sensible to temperature changes: a small increase of the temperature causes a large decrease of its resistance and vice versa. If the switch is closed (i.e., the gauge is operating), the resistance of the thermistor determines a current in the circuit, which in turn causes heat to be generated by the wire and transferred through air to the bimetallic strip. This is constituted by two strips made of different metals welded together: temperature changes cause the two strips to expand differently, causing the bimetallic strip to bend. Therefore, the temperature of the bimetallic strip determines its bending, which eventually determines the angular position of the pointer on the scale.

Fig. 2 shows two structural models of the temperature gauge at two different aggregation levels: STR-0 and STR-1. The component AR in the structural model STR-1 refers to the air between the wire WR and the bimetallic strip BS, which allows a heating exchange among them. Aggregation links between structural models of different

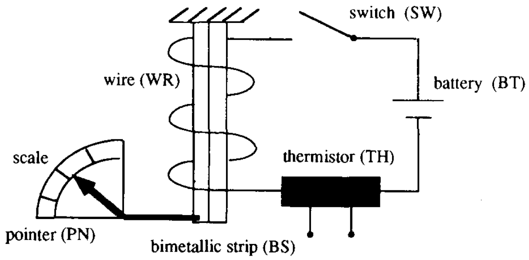


Fig. 1. Schema of the temperature gauge.

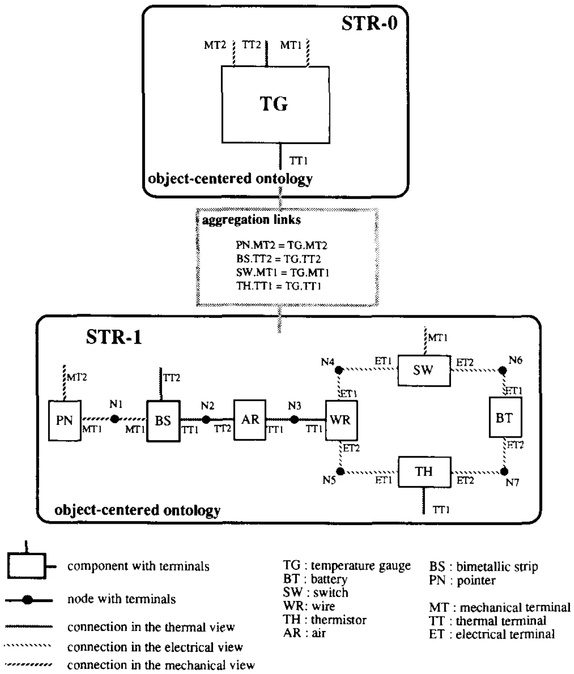


Fig. 2. Structural models of the temperature gauge.

aggregation levels relate the terminals of the representation of a component in a model with the corresponding terminals of the representation of the same component in the finer or coarser model. The physical view mechanism is realized in the structural model through terminal types: a physical view is identified by the set of terminals of a given type. Note that since aggregation links relate typed terminals, physical views extend across all available models of different aggregation levels.

B. The Behavioral Model

The *behavioral model* is devoted to represent the potential behavior of a system; it describes how components operate and interact with each other through relationships among physical quantities [18]. The behavioral model makes it possible to generate the actual behavior of the system, in terms of the sequence of values system quantities assume over time. The behavioral model is based on the object-centered ontology and is represented using the following three primitives:

1) *Physical quantities* are the basic entities used to capture the nature, the state, and the behavior of a system. They can be grouped into three classes:
   a) *constants* are independent from time and from the specific system being modeled (for example, gravitational acceleration, light speed in the vacuum, Boltzmann constant, etc.);
   b) *parameters* represent specific attributes of a component, for example, the section of a pipe, the thickness of an insulator, the thermal conductivity of a material, the density of a fluid, etc.;
   c) *variables* characterize the state of a component with reference to the physical phenomena in which it can take part (for example, the temperature of a substance, the voltage drop across a resistor, the pressure of a fluid, etc.).

   Physical quantities may assume either quantitative or qualitative values [49] and have an associated *type* which characterizes the physical domain to which they refer. So, for example, temperature and heat flow are thermal variables, while tension and electrical current are electrical ones; resistance is an electrical parameter, etc.
2) *Physical equations* represent the relationships existing among physical quantities and characterizing the potential behavior of a component or of a node. Their role is to constrain the possible values the physical quantities can assume over time.
   Two main types of physical equations have been identified, namely:
   a) *structural equations* are relationships that do not involve parameters; structural equations usually represent general principles of conservation (*balance equations*), such as the Kirchhoff principles in the electromagnetic domain and the equilibrium equations in mechanics, or represent definitions of new concepts (*definitional equations*), such as the concept of velocity defined as $v = dx/dt$ in mechanics;
   b) *phenomenological equations* represent relationships that do involve parameters; they represent physical laws either characterizing a single specific physical domain (*constitutive equations*) or establishing relationships between different physical domains (*coupling equations*); for example, Ohm's law is a typical constitutive equation in the electromagnetic domain, while Seebeck's law is a typical coupling equation which bridges electrical and thermal variables.

   Physical equations can be described either quantitatively or qualitatively [49].
3) *Operating modes* provide a representation of mutually exclusive operating regions of a component [25], for example, the active, saturation or cutoff regions of a transistor; an operating mode is described through 1) a set of relations among physical quantities, called *characteristic condition*, which specify the ranges of variables values within which

the component can be assumed to be in the considered operating mode, and 2) the set of physical equations which represent the component behavior in that mode.

Since the concept of *actual behavior* of a system refers to temporal sequences of states where a state is intended as the vector of all instantaneous values of system quantities in a specific time instant, a *temporal model* is needed. In the case of quantitative models, we assume that there exists a continuous time line, isomorphic to the real axis, and that a mapping between time points and system states can be defined. In the case of qualitative models, we assume the qualitative ontology of time proposed by Kuipers [49].

The behavioral model of a system is made up of a collection of physical quantities and physical equations holding among them. The physical equations are possibly organized according to a set of operating modes. Physical views are realized by clustering the physical equations in the model according to the types of the quantities involved. These clusters are then indexed by the views to which they refer.

Aggregation links between behavioral models of different aggregation levels are described through structural equations relating physical quantities of the representation of a component in a model with the corresponding quantities of the representation of the same component in a finer or coarser grained model.

The epistemological link between behavior and structure is realized by associating behavioral primitives (physical quantities, physical equations, and operating modes) to structural ones (terminals, components, and nodes).

Considering again the example of the temperature gauge, Fig. 3 shows four behavioral models: BEH-0, BEH-1.1, BEH-1.2, and BEH-1.3. BEH-0 represents the device at the highest aggregation level while BEH-1.1, BEH-1.2, and BEH-1.3 show a finer aggregation level. All the models are based on the object-centered ontology. However, BEH-1.1 and BEH-1.2 show different approximations based on a macroscopic ontology (approximation A2, for example, includes thermal effects on resistance R1, which are ignored in approximation A1), while BEH-1.3 is based on a microscopic ontology (the physical equations used are a qualitative version of the well known Maxwell laws of the electrodynamic theory). Fig. 3 also shows the links existing between the models: aggregation links between BEH-0 and the three finer models BEH-1.1, BEH-1.2, and BEH-1.3; representational links between BEH-1.1 and BEH-1.2; ontological links between BEH-1.3 (microscopic ontology) and BEH-1.1 and BEH-1.2 (macroscopic ontology), and, finally, epistemological links relating all behavioral models to structural ones. Note that links may describe complex relationships between the entities represented in the models. In the above example, the ontological link between BEH-1.3 and BEH-1.1 is described by a set of equations. These equations relate changes in macroscopic entities (such as the voltage drop V4-V5 and the electrical current I3 represented in BEH-1.1) to changes in microscopic entities (such as the electrical field E1 and the current density j3) represented in BEH-1.3.

C. Reasoning With Structural and Behavioral Knowledge

Structural knowledge may be used to obtain information about the connectivity of the modeled system. The only model-specific basic reasoning utility available in the structural model is "path finding." *Path finding* finds all paths—made up of components, terminals, nodes, and aggregation links—between any pair of components. Path finding is affected by the physical view mechanism: once a specific focus of attention has been determined, the paths obtained contain only components, terminals, nodes and aggregation links belonging to the selected views.

Three basic reasoning utilities are available for behavioral knowledge. These are briefly illustrated below:

1) *Behavioral prediction* infers the future states of a system (or of a subsystem) given its actual state and a perturbation of this state. Behavioral prediction is performed by numerical simulation in the case of quantitative models and by qualitative simulation when qualitative models are used [49].
2) *Causal dependency analysis* finds all variables whose possible deviations may influence a given variable. Parameters are considered as constants that could be neglected. The method we used for causal dependency analysis is based on causal ordering [6], [45], and assumes a stationary equilibrium state of the system.
3) *Sensitivity analysis* finds all parameters whose possible deviations may influence a given variable. Causal dependency analysis is similar to sensitivity analysis in that it is based on causal ordering.

D. Comparison with Related Work

The concept of structural model introduced above includes several ideas originally developed in [25] and [22]. However, some basic differences are worth stressing:

1) System constituents are not classified into materials, components, and conduits. This characterization, which is based on functional knowledge, is captured and generalized in the multimodeling approach through the concepts of functional role and generalized substance (see Section V-B).
2) In the above mentioned approaches, terminals are simple component attributes, which describe channels of possible interaction with the outside world. In the multimodeling approach, terminals are typed objects, which exactly characterize the class of interactions they can support (electrical, thermal, mechanical, etc.). Only terminals of the same type can be connected together, so type checking can help to avoid or identify inconsistent connections.
3) The concept of node is used to represent connec-

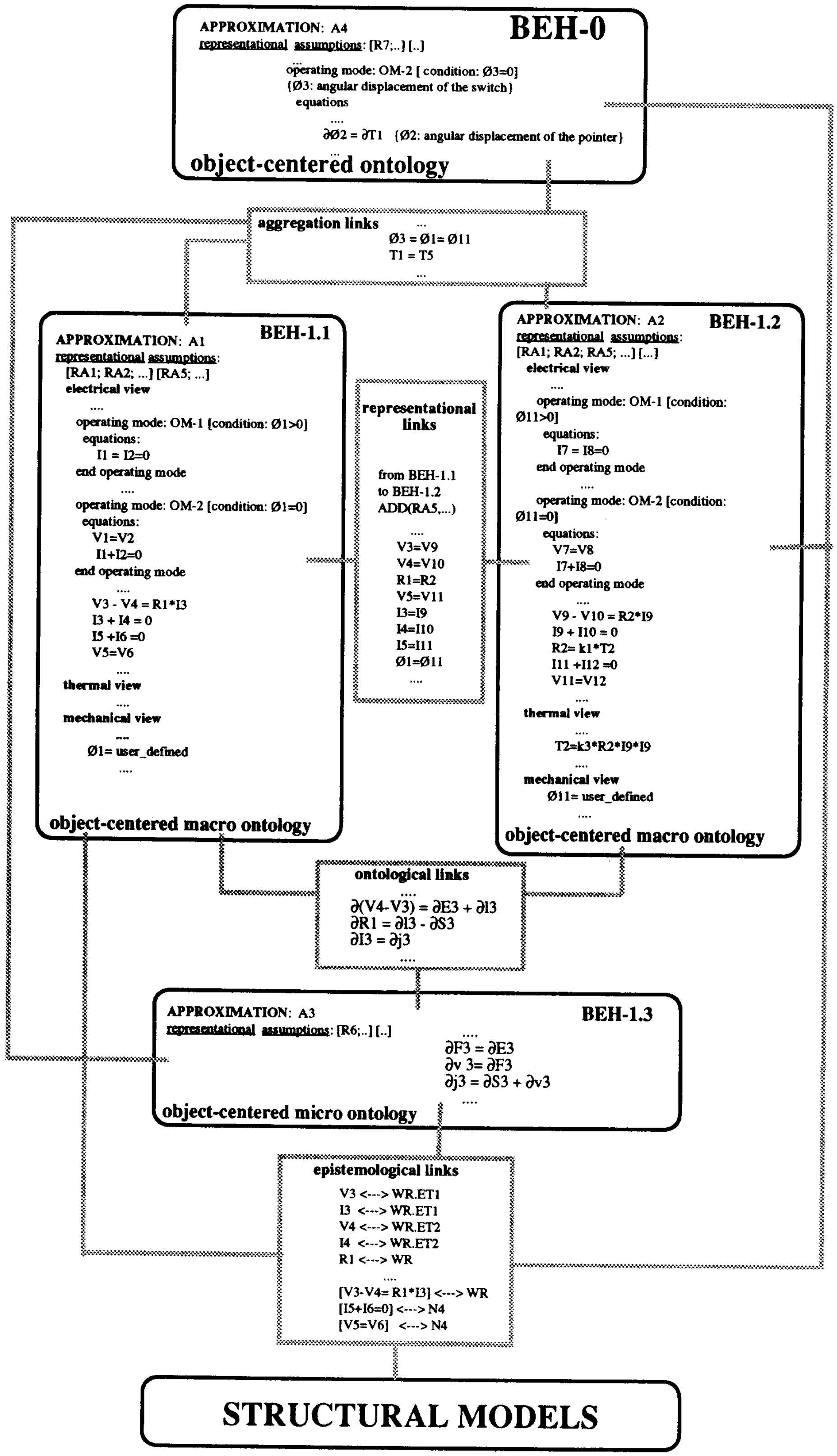


Fig. 3. Behavioral models of the temperature gauge.

tions among components. This makes the structural model more explicit and transparent.

In defining the behavioral model the work reported in [25] and [49] has been taken as a starting point. The main difference is that, in the multimodeling approach, a clear separation between parameters and variables is introduced, and, as a consequence, between constitutive and structural equations. This distinction allows specification of explicit epistemological links between the behavioral and structural models. Moreover, physical views are not considered at all in the above mentioned approaches.

IV. Interpretative Knowledge: Open Problems

Interpretative knowledge derives from an interpretation of fundamental knowledge in terms of system functions and goals. Several problems about the representation and use of functional and teleological knowledge may be identified in the approaches so far proposed in the literature. These are briefly discussed below:

1) Difference between function and purpose. For some authors the concept of function is strongly related to teleology: the function of a system is identified by the task the system accomplishes or should accomplish [73], or by its intended use or purpose [24], [28], [33], [47], [71], [75]. Other authors distinguish more accurately between function and teleology and introduce the concept of function as a mapping between behavior and teleology. This approach is taken in [9], where function is defined as the relationship existing between the behavior of a system and the goals of a human user. A similar position is also proposed in [2], where functional knowledge is devoted to link components to the processes in which they take part, and in [67], where function is considered an abstraction of behavior, performed by taking into account the goals of the system, which maps behaviors into processes. However, in these approaches the issue of how functional and teleological knowledge can be actually represented and used is discussed only in very general terms and no concrete proposal is presented.
2) Relativity of teleological knowledge. The concept of purpose of a system is multifaceted and seems to escape a precise definition. In fact, teleological knowledge does not concern objective features of a system but rather its relationship to some external reference. However, there is no general agreement on what should be considered as the reference—the user [1], the designer [10], [33], or somebody else.
3) Function primitives. Some authors consider the function of a component as an interpretation of its behavior in terms of primitive actions that the component performs on substances flowing through it [7], [31], [56]. These approaches however, raise critical problems concerning the concept of functional primitive: How are they chosen? How can their semantics be specified? How can their expressiveness, completeness, and minimality be evaluated?
4) Relationship between fundamental and interpretative knowledge. Most approaches fail to explicitly represent the relationships existing between teleology, function, behavior, and structure [67], [71]. This is, however, extremely important both for consistency reasons and for an effective exploitation, in the reasoning process, of all available knowledge sources.

In the following two sections we will illustrate how these problems have been tackled in the frame of the multimodeling approach.

V. Representation and Use of Functional Knowledge

A. The Concept of Function

In the multimodeling approach, the *function* of a system is defined as the relationship between its behavior and the goals assigned to it by the designer. The concept of function is therefore understood as a bridge between behavioral and teleological knowledge [12]. In a sense, it is not a primitive concept (such as structure, behavior or teleology), but a concept which only exists as a relationship between two other concepts.

Accordingly, the functional representation of a system is aimed at describing how the behaviors of individual components contribute to the achievement of the common goal assigned to the system by its designer. Therefore, functional modeling mainly focuses on system organization and implies a teleological explanation of behavior; hence, the behavior of an individual component is justified by looking at the final causes of the system as a whole, rather then on the basis of the efficient causes that generate it in a mechanistic way [25], [67].

Moreover, since the behavioral model is based on the object-centered ontology and the teleological model is based on the system-centered ontology, the concept of function which realizes the transition between these two models must share some aspects of both ontologies. Therefore, as it will be illustrated in the following sections, functional knowledge is represented through three kinds of models: the model of functional roles which is based on an object-centered ontology, the model of processes which has a two-fold nature since it is based on a hybrid ontology, and finally, the model of phenomena which is based on a system-centered ontology. In this way the mapping between behavior and teleology is realized in a gradual way by progressively introducing in the representation knowledge elements which are more and more context dependent. The above proposal is not unique; other proposals may be acceptable as well, possibly comprising a lower number of intermediate models. The main goal of our proposal has been that of a smooth and gradual transition.

B. Generalized Interpretation of Behavior

It is quite common in several physical domains to interpret the behavior of a system in terms of flow structures, i.e., in terms of networks of operators acting on substances flowing through the structure of the system, such as heat, electrons, liquids, etc. and in terms of the forces responsible for their flow, such as thermal gradient, electrical tension, pressure gradient, etc. This interpretation is useful to identify and exploit analogies among phenomena pertaining to different physical domains, but ruled by physical laws featuring the same formal structure. This is the case of the well known similarities between thermal conduction in thermodynamics, electric conduction in electrodynamics, and laminar flow in hydrodynamics, which are all ruled by equations of the form $y = Kx$ (the well known Fourier's law, Ohm's law, and Poiseuille's law, respectively).

In our approach, physical variables are classified on the basis of the role they play in physical phenomena interpreted as flow structures. From this perspective, it is possible to identify two types of *generalized variables* [25], [70] common to different physical domains:

1) *Generalized substance* (gs), represents the abstract *entities* which *flow* through a system. The concept of generalized substance can be further decomposed into two subtypes, namely: *generalized displacement* (q) and *generalized impulse* (p). The rationale behind this distinction is that substances flowing through a system are usually associated with energy. Since there are two fundamental types of energy, potential and kinetic, we also need two types of generalized substances. Thus, generalized displacement can be intuitively associated with potential energy and generalized impulse with kinetic energy.
2) *Generalized current* (gc) represents the amount of a generalized substance which flows through a unitary surface in a time unit, i.e., more formally: $gc = d(gs)/dt$. Therefore, according to the type of generalized substance which is flowing, we distinguish between *generalized flow* (f) intended as flow of displacement (dq/dt) and *generalized effort* (e), intended as flow of impulse (dp/dt). The product $e * f$ intuitively represents the amount of energy which flows through a unitary surface in a time unit (i.e., power).

Generalized variables are, of course, independent of any specific physical domain. When they are specialized in a specific physical domain we obtain usual *physical variables*. For example, in the electromagnetic domain, q represents electrical charge; p, magnetic flux; f, electrical current; and e, electrical tension. In the rotational mechanical domain, q represents angular displacement; p, angular momentum; f, angular velocity; and e, torque.

After having classified physical variables, it is possible to exploit the well known concept of *Tetrahedron of State* (*TOS*) [63], [70] in order to identify a set of abstract relationships among generalized variables, which are called *generalized equations* and are common to a large class of physical theories. Fig. 4 presents the organization of the TOS and two of its possible instantiations in the mechanical and electromagnetic domains.

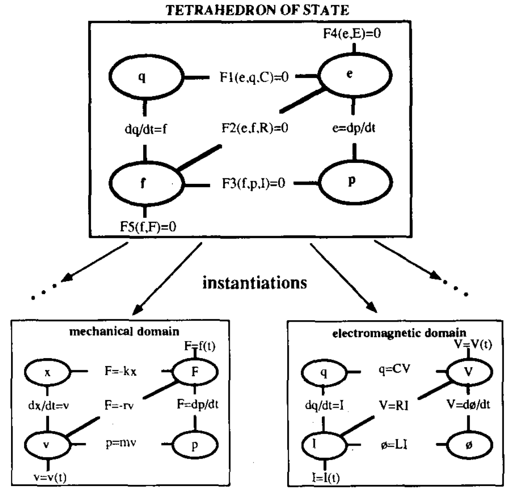


Fig. 4. The organization of the TOS and two sample instantiations.

The TOS comprises five generalized equations among the generalized variables q, e, p, and f. Two of them are structural equations describing the relations between q and p and their flows f and e. The remaining three equations are constitutive equations which describe:

- an explicit relationship $F1(e, q, C) = 0$ between generalized effort e and generalized displacement q, involving the parameter $C = dq/de$ which represents a *generalized capacity*;
- an explicit relationship $F2(e, f, R) = 0$ between generalized effort e and generalized flow f, involving the parameter $R = de/df$ which represents a *generalized resistance*;
- an explicit relationship $F3(f, p, I) = 0$ between generalized impulse p and generalized effort f, involving the parameter $I = dp/df$ which represents a *generalized inductance*.

Finally, two more generalized equations have been introduced that are not considered in the original concept of TOS: the relationship $F4(e, E) = 0$ involving the generalized effort e and the parameter E which represents a *generalized electromotive force*, and the relationship $F5(f, F) = 0$ involving the generalized flow f and the parameter F which represents a *generalized electromotive flow*. These equations represent the application of generalized effort or flow at the boundary of open systems and are necessary to represent generators of effort and flow in a correct and natural way.

If the generalized equations appearing in the TOS, in particular, constitutive equations, are specialized in a specific physical domain we obtain usual *physical equations*.

For example, the generalized equation $F2(e, f, R) = 0$ in the electrical domain corresponds to Ohm's law ($\Delta V - RI = 0$), while in the thermal domain it corresponds to Fourier's law ($Q - K\Delta T = 0$).

C. Functional Roles

The components of a system have a primary role in determining the flow structures which constitute its function, since their behaviors can be interpreted as operators on generalized variables. We define the *functional role* of a component as an interpretation of its behavior—more precisely, of the physical equations governing its behavior—aimed at characterizing how the component contributes to the realization of the flow structure in which it takes part.

The concept of TOS introduced in the previous section can be used as a sound basis for identifying a finite set of *functional roles*, which are assumed to be sufficient for interpreting the behavior of a large set of systems of practical interest [70]. Nine functional roles can be defined in relation to the five generalized equations $F1$, $F2$, $F3$, $F4$ and $F5$. Their definitions are reported below.

From the generalized equation "$F2(e, f, R) = 0$" we identify the following five functional roles:

f-conduit (C^f): An f-conduit enables a generalized flow from one point to another in the structure of a system with dissipation of generalized effort. A series-resistor in the electrical domain is an instance of an f-conduit since it transmits electrical current from one terminal to the other, but it dissipates electrical tension (i.e., there is a tension drop across the resistor).

e-conduit (C^e): An e-conduit enables a generalized effort from one point to another in the structure of a system with dissipation of generalized flow. A parallel resistor is an e-conduit of electrical tension which dissipates electrical current, i.e., the electrical current that enters the component at one terminal is not the same that exits the component at the other terminal.

Note that the concept of conduit embodies the intuitive meaning of resistive conduit (or generalized resistor), since it transmits without changes only one type of current (f or e) but not both together, i.e., it dissipates power.

purely conductive conduit (*CC*): A purely conductive conduit enables both generalized currents (f and e) from one point to another in the structure of a system, without power dissipation. For example, the ideal junction between two electrical components can be described using a purely conductive conduit, since it features continuity of current and compatibility of electrical tension.

f-barrier (B^f): An f-barrier prevents a generalized flow from one point to another in the structure of a system. In the electrical domain, for example, an open circuit is an f-barrier.

e-barrier (B^e): An e-barrier prevents a generalized effort from one point to another in the structure of a system. In the electrical domain, for example, a short circuit is an e-barrier.

From the generalized equations "$F1(e, q, C) = 0$" and "$F3(f, p, I) = 0$" we identify the following two functional roles, respectively:

q-reservoir (R^q): A q-reservoir (or generalized capacitor) enables accumulation of a generalized displacement, q. A spring in the mechanical domain or a capacitor in the electrical domain are examples of q-reservoirs.

p-reservoir (R^p): A p-reservoir (or generalized inductor) enables accumulation of a generalized impulse p. A mass m which moves with velocity v in the mechanical domain, or an inductor in the electrical domain are examples of p-reservoirs.

From the generalized equations "$F4(e, E) = 0$" and "$F5(f, F) = 0$" we identify the following two functional roles, respectively:

e-generator (G^e): An e-generator causes a generalized effort between two points in the structure of a system.

f-generator (G^f): An f-generator causes a generalized flow from one point to another in the structure of a system.

Note that of the roles defined above only e-generator and f-generator are *active roles* which denote the capability of a component to cause something, while all others are *passive roles* that can only enable or prevent something to occur.

Two types of relations between functional roles can be identified. These are defined below:

1) *Mutual dependency:* two functional roles FRi and FRj, which refer to physical equations PEi and PEj respectively, are mutually dependent if PEi and PEj share a physical variable (*direct mutual dependency*) or if there exists a structural equation that links a physical variable of PEi with a physical variable of PEj (*indirect mutual dependency*). For example, the functional roles associated with two electrical resistors connected in series are directly mutually dependent since they refer to two physical equations (Ohm's law) that share the physical variable representing electrical current. The functional roles associated to the spring and to the mass of an oscillating mass–spring system (without friction) are indirectly mutual dependent since they refer to two physical equations ($F = -kx$ and $p = mv$ respectively) which are related by a structural equation ($F = dp/dt$ or $v = dx/dt$).
2 *Influence:* a functional role FRi, which refers to physical equations PEi, influences a functional role FRj, which refers to physical equations PEj if a physical variable of PEi is a parameter of PEj. For example, the functional role q-reservoir associated with the screw of a tap influences the functional role of the tap viewed as a f-conduit since the amount of angular displacement stored in the reservoir is coupled with the section of the tap. Therefore, it interacts with the generalized resistance of the tap viewed as a conduit.

Functional roles can be connected together using the above relations to yield a graph, called *functional role network*.

D. The Functional Role Model

The *functional role model* of a system describes the potential functional roles of its components and the ways the functional roles are related. It is represented by a functional role network. Of course, since functional roles and relations are generalized entities, they have to be specialized in the appropriate physical domains when they are used to represent a specific system.

The epistemological link between the functional role model and the behavioral one is defined by associating functional primitives (functional roles and relations) to behavioral ones (constitutive and structural equations). This is done in such a way that an appropriate functional role is associated, in the behavioral model, to each constitutive equation which is a specialization of a generalized equation in the TOS, and, analogously, a relation (i.e., a mutual dependency or an influence) is associated to each structural equation. Since behavior is related to structure (see Section III-B), the association between functional roles and physical equations results in an indirect assignment of functional roles to structural components. Therefore, the association between functional roles and components can be of two types:

1) One to one, when a component is bound to a single functional role in a single view and operating mode; this is the case, for example, of a pipe viewed as a conduit in the hydraulic view;
2) many to one, when one of the following three cases occurs:
   a) a component is bound to several coexisting functional roles in the same view and operating mode; as an example, consider an oscillating pendulum: it is at the same time an inductor and a capacitor in the mechanical view, since, at any time, it is either charging with potential energy and discharging kinetic energy, or vice versa;
   b) a component is bound to several coexisting functional roles in different views but in the same operating mode; as an example, consider a wire: it is at the same time a conduit in the electrical view and an f-generator in the thermal one;
   c) a component is bound to several coexisting roles in the same view but in different operating modes; as an example, consider a valve represented in the hydraulic view by two operating modes (open and closed): in the open mode it is a conduit, in the closed mode it is a barrier.

Note that a component may dynamically change its functional role in a given view depending on the values of physical variables which determine its operating mode. Thus, in the electrical view, a fuse is a conduit until the current flowing through it is below a specified threshold; afterwards, it becomes a barrier.

Considering again the example of the temperature gauge, Fig. 5 shows a functional role model represented in terms of a functional role network: FUN.R-1.1. This model is based on the same set of representational as-

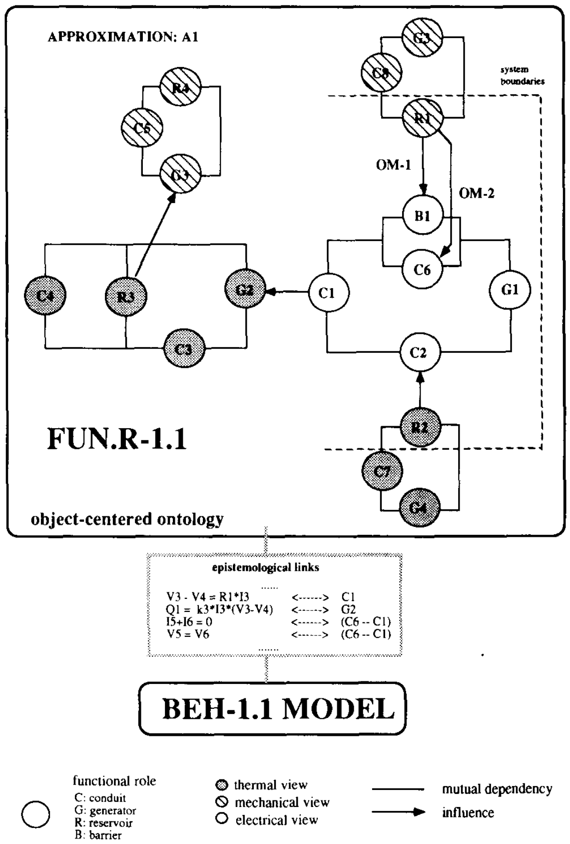


Fig. 5. Functional role model of the temperature gauge.

sumptions adopted for BEH-1.1, and describes the device at the same aggregation level. The model can be better understood if one takes into account the epistemological links that indirectly relate function to structure through behavior. By considering these links, note that some component is bound to more than one role in different views: the thermistor TH, for example, results to be a reservoir (of heat) in the thermal view (R2) and a conduit (of electrical current) in the electrical view (C2). Between these two roles an influence relation holds: the amount of heat stored in the reservoir influences (negatively) the resistance of the thermistor viewed as a conduit of electricity. Other components have alternative roles associated with them due to the existence of different operating modes in the same view: the switch SW, for example, in the electrical view, is a conduit (C6) when it works in the mode "CLOSED" (i.e., OM-2) and a barrier (B1) when it works in the mode "OPEN" (i.e., OM-1). Roles G3, C8, G4, and C7 in Fig. 5 are not associated with any component of the temperature gauge but have been introduced to describe explicitly the application of external generators at the boundaries of the system: G3 is a generator of torque acting on the switch SW through conduit C8, and G4 is a generator of heat driving the thermistor TH through C7. Fig. 5 also shows the epistemological links between FUN.R-1.1 and BEH-1.1. For example, the functional role C1 in the electrical view is associated to the physical equation V3 − V4 = R1 * I3 (Ohm's law applied to the wire WR) in the same view, the direct mutual dependency between functional roles C6 and C1 in the electrical view is associated to the structural equations I5 + I6 = 0 (same current through wire WR and switch SW) and V5 = V6 (same voltage in the node between wire WR and switch SW), in the same view.

E. Processes and Phenomena

Functional roles participate in physical processes. Let us introduce first of all the concept of process. A *process* is a four-tuple ⟨cofunction, precondition, effect, posteffect⟩, where:

- *cofunction* is a functional role network which specifies which functional roles are necessary and how they must be related together in order to enable the occurrence of the process;
- *precondition* is a logical predicate which characterizes the situation which enables the process to occur;
- *effect* is a logical predicate which characterizes the situation which is true during the occurrence of the process;
- *posteffect* is a logical predicate which characterizes the situation which is true after the process has ended, i.e., when its precondition ceases to hold.

Note that preconditions, effect, and posteffect may involve values of physical variables which are instances of generalized variables associated to the functional roles of the cofunction.

Of course, not every functional role network can be the cofunction of a process. For example, a series of conduits linked together by mutual dependencies, although constituting a functional role network, does not support any process at all. To be a cofunction a functional role network must include at least a generator and provide a path through which a generalized substance can flow. Moreover, a cofunction must be thought of as the representative of an equivalence class including all equivalent functional role networks (i.e., functional role networks which are indistinguishable from a functional point of view).

For the class of systems whose behavioral model can be interpreted in terms of the TOS, a finite set of three possible processes can be identified. These are described below. Note that in the definition of the process the symbol {. . .} denotes an option, the symbol [. . .] an alternative, val(PV, GV, FR) denotes the value of the physical variable PV which is an instance of the generalized variable GV of the functional role FR, m-dependent (FR1, FR2) denotes that the functional roles FR1 and FR2 are mutually dependent, and der(PV, GV, FR) denotes the sign of the time derivative of PV. Note also that we refer, in general, to minimal cofunctions, i.e., functional role networks which are the cofunction of a process and are composed by the strictly necessary number of functional roles.

transporting (TRANS)

cofunction	$(G1^{[e,f]}, C^{[f,e]}, \{G2^{[e,f]}\})$
	m-dependent$(G1^{[e,f]}, C^{[f,e]})$
	{m-dependent$(C^{[f,e]}, G2^{[e,f]})$}
precondition	val$(PV_1$, [e, f], G1) > 0
	{AND val$(PV_1$, [e, f], G1) > val$(PV_2$, [e, f], G2)}
effect	val$(PV_3$, [f, e], C) > 0
posteffect	val$(PV_3$, [f, e], C) = 0 AND
	val$(PV_1$, [e, f], G1) = 0
	{OR val$(PV_1$, [e, f], G1) = val$(PV_2$, [e, f], G2)}

Transporting involves one or two generators, either G^e or G^f depending on the generalized substance involved, and a conduit, C^f or C^e, respectively. The functional roles are linked together by mutual dependency relations. Its precondition is that the generator pushes the generalized substance, or, in the case of two generators, that one pushes and the other pulls, i.e., there must be a difference between the values of the physical variables of type effort (or flow) associated to the generators. The resulting effect is a current flowing through the conduit. The posteffect is that the current becomes zero and the generator does not push anymore, or, in the case of two generators, that one does not push and the other does not pull anymore, i.e., the difference between the values of the physical variables associated to the generators becomes zero.

reservoir charging (RESC)

cofunction	$(G^{[e,f]}, C^{[f,e]}, R^{[q,p]})$
	m-dependent$(G^{[e,f]}, C^{[f,e]})$
	m-dependent$(C^{[f,e]}, R^{[q,p]})$
precondition	val$(PV_1$, [e, f], G) > val$(PV_2$, [e, f], R)
effect	val$(PV_3$, [f, e], C) > 0
	AND der$(PV_4$, [q, p], R) > 0
posteffect	val$(PV_1$, [e, f], G) = val$(PV_2$, [e, f], R)
	AND val$(PV_3$, [f, e], C) = 0
	AND val$(PV_4$, [q, p], R) > 0

Reservoir charging involves a generator, either G^e or G^f depending on the generalized substance involved, a conduit, either C^f or C^e, respectively, and a reservoir, either R^q or R^p, respectively. The functional roles are linked together by mutual dependency relations. Its precondition is that there must be a difference between the values of the physical variables of type effort (or flow) associated to the generator and to the reservoir. The resulting effect is a current flowing through the conduit and an increasing amount of substance in the reservoir. The posteffect is that the difference between the values of the physical variables associated to the generator and to the reservoir and the current flowing through the conduit become zero, and the amount of substance within R has a value which is greater than zero.

reservoir discharging (RESD)

cofunction	$(R^{[q,p]}, C^{[f,e]}, \{G^{[e,f]}\})$
	m-dependent$(R^{[q,p]}, C^{[f,e]})$
	{m-dependent$(C^{[f,e]}, G^{[e,f]})$}
precondition	val(PV$_1$, [q, p], R) > 0
	{AND val(PV$_2$, [e, f], R) > val(PV$_3$, [e, f], G)}
effect	val(PV$_4$, [f, e], C) > 0
	AND der(PV$_1$, [q, p], R) < 0
posteffect	val(PV$_4$, [f, e], C) = 0 AND
	(val(PV$_1$, [q, p], R) = 0
	{OR val(PV$_1$, [q, p], R) $\geqq$ 0})
	{AND val(PV$_3$, [e, f], G) = val(PV$_2$, [e, f], R)}

Reservoir discharging involves a reservoir, either R^q or R^p depending on the generalized substance involved, a conduit, either C^f or C^e respectively, and a possible generator, either G^e or G^f respectively. The functional roles are linked together by mutual dependency relations. Its precondition is that the reservoir is initially not empty, and, in the case of presence of a generator, that there is a difference between the values of the two physical variables of type effort (or flow) associated to the reservoir and to the generator, respectively. The resulting effect is a current flowing through the conduit and a decreasing amount of substance in the reservoir. The posteffect is that the substance in the reservoir and the current flowing through the conduit becomes zero or, if a generator is present, that the difference between the values of the two physical variables associated to the reservoir and to the generator is zero.

Processes have an associated *functional state* which may assume one of the following two values: active, not active. We say that a process is active if its cofunction is present in some view and its precondition is satisfied. The process is deactivated when either its cofunction is no more present or its precondition is no more satisfied.

Three types of relation between processes can be identified. These are defined below:

- *Direct causation:* a process Pi directly causes a process Pj if the effect or posteffect of Pi entails the precondition of Pj. For example, water pumping (reservoir charging) causes water transportation (transporting), since the effect of pumping is to create a pressure drop, which is the precondition for water transportation.
- *Regulation:* a process Pi regulates a process Pj if there exists an influence relation between two functional roles belonging to their cofunctions which involves a parameter which is a generalized resistance. For example, the process of turning a tap on a water conduit changes its section and thus modifies the rate of the process of water transportation.
- *Support:* a process Pi supports a process Pj if there exists an influence relation between two functional roles belonging to their cofunctions which involves a generalized substance on one side and a parameter which is a generalized capacitance or a generalized inductance, according to the type of substance considered, on the other side. Consider, for example, two hydraulic tanks A and B connected through a pipe. Suppose that tank A is filled with hot water while B is empty. Two processes coexist process-1 is a discharging capacitor process in the hydraulic view (water flows from tank A to B), process-2 is a discharging capacitor process in the thermal view (heat flows from A to B). We can now say that process-1 supports process-2 because of the influence relation holding between water and heat: the capacity of the water considered as a reservoir of heat is influenced by the volume of water present in the tank.

Processes can be connected together using the above relations to yield a graph, called *process network*. By *functional state* of a process network we mean the specification of which processes of the network are active and which are not active at a given time instant.

Processes participate in the definition of physical phenomena. A *phenomenon* is a four tuple ⟨organization, precondition, effect, posteffect⟩, where:

- *organization* is a process network which defines which processes are necessary and how they must be related together in order to enable the occurrence of the phenomenon;
- *precondition* is a logical predicate which characterizes the situation which enables the phenomenon to occur; the precondition involves the functional state of the processes specified in the organization of the phenomenon;
- *effect* is a logical predicate that characterizes the situation which is true during the occurrence of the phenomenon;
- *posteffect* is a logical predicate which characterizes the situation which is true after the phenomenon is terminated, i.e., when its precondition ceases to hold.

Note that effect and posteffect may involve values of physical variables which are instances of generalized variables associated to the functional roles belonging to the cofunction of processes specified in the organization of the phenomenon. Phenomena whose organization is constituted by a single process are called *elementary phenomena*.

The set of possible phenomena is open ended. Several types of commonly occurring phenomena can be easily identified, such as natural oscillation, damped oscillation, homeostasis, dynamic equilibrium, etc. As an example of a phenomenon, we illustrate below the phenomenon of damped oscillation. Note that in the definition the symbol *d*-causes(P1, P2) denotes that the process P1 directly causes the process P2, while state(P) denotes the functional state of the process P.

damped oscillation (DMPOSC)

organization: ($RESD1^q$, $RESC1^p$, $RESD2^p$, $RESC2^q$)
d-causes($RESC2^q$, $RESD1^q$) AND
d-causes($RESD1^q$, $RESC1^p$) AND
d-causes($RESC1^p$, $RESD2^p$) AND
d-causes($RESD2^p$, $RESC2^q$)

precondition [state($RESC1^p$) = active, state($RESC2^q$) = active]

effect [$der(PV_1, p, R_1) \neq 0$, $der(PV_2, q, R_2) \neq 0$] AND
$der(PV_1, p, R_1) + der(PV_2, q, R_2) < 0$

posteffect $val(PV_1, p, R_1) = val(PV_2, q, R_2) = 0$

The organization of a damped oscillation comprises four processes: a reservoir discharging process of displacement q ($RESD1^q$), a reservoir discharging process of impulse p ($RESD2^p$), a reservoir charging process of impulse p ($RESC1^p$), and a reservoir charging process of displacement q ($RESC2^q$). The processes are linked together by four relationships of direct causation: $RESC2^q$ directly causes $RESD1^q$, $RESD1^q$ directly causes $RESC1^p$, $RESD2^p$ directly causes $RESC2^q$, and $RESC1^p$ directly causes $RESD2^p$. The precondition of the phenomenon is that either $RESC1^p$ or $RESC2^q$ are in the active state. The effect is a change in the amount of substance p within reservoir R1 belonging to the cofunction of $RESC1^p$ and $RESD2^p$ or a change in the amount of substance q within reservoir R2 belonging to the cofunction of $RESC2^q$ and $RESD1^q$ and a decreasing of the total mechanical energy associated to the system (i.e., the sum of the amounts of displacement and impulse associated respectively to reservoirs R2 and R1). The posteffect is that the system will be at rest with no kinetic energy.

When the above phenomenon is specialized, for example, in the mechanical domain, it can be used to represent the oscillation of a spring–mass system. In this case, the four processes constituting the organization of the damped oscillation are the processes of charging and discharging of position and velocity, respectively. Analogously, when it is specialized in the electrical domain, it can be used to describe the four processes of charging and discharging of electrical charge and magnetic flux in a RCL circuit.

Finally, two phenomena P and Q are *related* if they share a process in their organization or if two processes belonging to their respective organizations are related through direct causation, regulation or support. Phenomena can be connected together using the above relations to yield a graph, called *phenomenon network*.

Phenomena can be directly associated to the intended goals of the designer that are represented in the teleological model, thus completing the link between behavior and teleology.

F. The Process Model

The *process model* of a system describes the set of processes that may occur in the system and their relationships. It is represented by a process network. Of course, since processes are generalized entities, they have to be specialized in the appropriate physical domains when they are used to represent a specific system.

The process model is directly related to the functional role model through an ontological link. This link, which relates a model which is based on the object-centered ontology to a model which is based on a hybrid ontology, is represented by specifying which functional roles and relations constitute the cofunction of the processes of the process model. Of course, since the functional role model is indirectly associated to the structural model through the behavioral model, the association between processes and functional role networks results in an indirect assignment of processes to structural components.

Considering again the example of the temperature gauge, Fig. 6 shows a functional representation of the device using a process network (FUN.P-1.1) which includes several processes and their relationships. By traversing the ontological links described in Fig. 6 (towards the functional role model) and then the epistemological links specified in Figs. 5 and 3, it is possible to identify the structural components which participate to processes described in FUN.P-1.1. More specifically, for example, the battery BT, the switch SW (operating in the CLOSED mode), the wire WR and the thermistor TH fulfill the cofunction of a transporting process (TRANS1) in the electrical view. In the thermal view, the wire WR, the air surrounding the wire AR, and the bimetallic strip BS accomplish the cofunction of a reservoir charging process of heat from the wire to the bimetallic strip (RESC1). The influence relation existing between the functional role C1 in the electrical view and G2 in the thermal view, both associated to the wire WR, is represented by a causal relationship between the TRANS1 and RESC1 processes. Note that Fig. 6 also describes exogenous processes which occur when the operator acts on the switch SW or an external generator of heat is connected to the thermistor TH. These processes are represented by reservoir charging processes (RESC3, RESC4) which are related to the electrical transporting process by a regulation relation.

G. The Phenomenon Model

The last model considered in the functional representation of a system is the *phenomenon model*. This model represents the set of phenomena which may occur in the system. It is represented by a phenomenon network. Of course, since also phenomena are generalized entities, they have to be specialized in the appropriate physical domains when they are used to represent a specific system.

The phenomenon model is directly related to the process model through an ontological link. This link, which relates a model based on the system centered ontology to a model based on a hybrid ontology, is represented by specifying which processes and relations constitute the organization of each phenomena of the phenomenon model. Of course, since the process model is indirectly associated

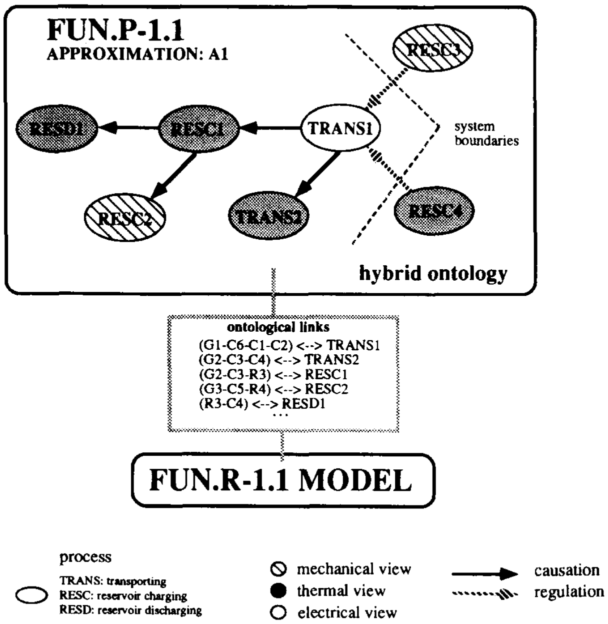


Fig. 6. Process model of the temperature gauge.

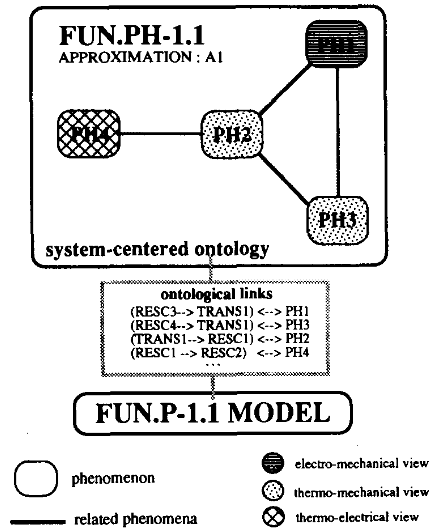


Fig. 7. Phenomenon model of the temperature gauge.

to the structural model through the functional role model and the behavioral model, the association between phenomena and process networks results in an indirect assignment of phenomena to structural components.

Fig. 7 shows a partial phenomenon model, representing the temperature gauge without considering heat dissipation. The model represents four phenomena: PH1, PH2, PH3, and PH4 which can be better understood if one considers the ontological link that relates the phenomenon model to the process model and the link existing between this last model and the structural model through the functional role model and the behavioral model. PH1 describes the functioning of the switch in the electrical circuit while PH2 describes the functioning of the thermoelectrical coupling between the wire and the bimetallic strip. PH3 describes the functioning of the thermistor in the electrical circuit and PH4 describes the functioning of the thermomechanical coupling between the bimetallic strip and the pointer. The phenomenon model also specifies how the phenomena are related to each other: phenomena PH2 and PH4, for example, are related through process RESC1, which belongs to their organizations. Finally, Fig. 7 shows the ontological link existing between the phenomenon model and the process model.

H. Using Functional Knowledge

Three basic reasoning utilities are available for functional knowledge. These are briefly illustrated below.

- *Functional prediction* derives the future functional state of a process network, given its actual state and a perturbation in the state of some of its constituent processes.
- *Functional dependency analysis* identifies, within a given process network, all the sets of processes whose functional state may influence a given process.
- *Process detection* identifies all the potential processes that may occur inside a system given a representation of the system in terms of a functional role network.

I. Comparison with Related Work

Considering the concept of function as a mapping between behavior and teleology, we have taken as a starting point of our approach the proposals presented in [2] and [9]. However, these proposals are both very synthetically stated, and the problem of how this mapping can be concretely represented is not faced at all.

Similar to [67], we propose to describe function at various levels of abstraction. However, unlike the work described in [67] which does not propose any modeling language, we introduce specific primitives to model each aspect of functional knowledge together with the ontological links between the various representations.

The set of functional roles defined in Section V-C is partially inspired to the primitive functions proposed in [7], [31], [56]. The main difference is that in our proposal, functional roles are derived from the generalized equations described in the TOS, while in the above mentioned approaches functional primitives are derived only intuitively, and are specific to the particular physical domain considered.

The concept of process defined in Section V-D, is partially related to [32]. However, our concept of processes is more general. Moreover, in our approach the concept of cofunction is explicit and can be used not only to establish the presence of a potential process in a system in analytic tasks, but also to suggest a possible functional configuration able to achieve a primitive goal in synthetic tasks. Finally, in our approach, processes are used to define phenomena which represent a further level of interpretation of behavior which is not explicitly represented in [32].

The method used for process detection was originally inspired to the parsing mechanism used in [24] for the teleological analysis of a system. It also shares some similarities with consolidation [7], [8].

Finally, our approach to functional modeling is also partially related to the work on bond graphs [70] developed in the field of system dynamics. However, our goals are very different. Our aim is to explicitly represent the link existing between behavior and teleology in the frame of multimodeling, where models are assumed to be separate. Therefore, we only focus on functional aspects. Behavioral and structural aspects, which are mixed with the functional ones in bond graph theory, are separately represented in our proposal in the behavioral and structural models respectively.

VI. Representation and Use of Teleological Knowledge

A. The Concept of Teleology

In the multimodeling approach the *teleology* of a system is defined as the specification of the goals assigned to it by the designer [13]. System goals are assumed to be achieved through phenomena. However, while all artifacts achieve their goals by performing some kind of physical phenomenon, for example, transporting electrical signals, transducing power, etc., not all goals can be directly associated to physical phenomena. In fact, goals are often at the information level, i.e., they concern the acquisition, processing, and distribution of information, independently of the underlying physical phenomena. Therefore, two kinds of goals can be identified:

physical goals, which describe system purposes in terms of physical phenomena;
information goals, which define system purposes in terms of information processing.

Of course, these two kinds of goals are not independent. In fact, the information level can be viewed as the result of an interpretation performed upon an underlying physical level: thus, information goals always refer to appropriate physical goals which constitute their background. In this paper, we focus only on physical goals, disregarding their possible higher level interpretation in terms of information processing.

B. The Concept of Goal

The fundamental concept in the teleological model is that of goal. A goal represents a generic purpose. Consider, for example, a ram, a pump, and a single phase alternator; all these devices may be considered as power transducers since they convert power from one physical domain to another. However, the ram specifically converts power from the hydraulic to the mechanical domain, the alternator from the electrical to the mechanical domain, etc. "To transduce power from one physical domain to another" is, thus, the generic purpose, i.e., the common goal of these devices.

A *goal* is a triple ⟨goal pattern, conditions, intended behavior⟩, where:

- *Goal pattern* assigns a *name* to the goal and specifies its *arguments*, which are typed variables relevant to the definition of the goal;
- *Operational conditions* specify the necessary operational conditions for the achievement of the goal; they include:
  Inputs, which specify what should be provided as input to the system in order to enable it to achieve its intended goal. Inputs are expressed in terms of admissible ranges of values for exogenous variables and possibly their derivatives, i.e., system variables whose values can be set by the user or are determined by the environment where the system operates;
  Settings, which specify how system parameters have to be adjusted in order to enable it to achieve its intended goal. Settings include the specification both of *operating modes* and of the appropriate *reference values* for parameters. Settings can be expressed using logical operators and temporal relations [4];
- *Intended behavior* specifies the behavioral effects that are expected from the goal; it includes:
  Outputs, which specify the system variables relevant to the definition of the goal and their admissible values;
  Transformation, which specifies the relationships expected to hold among the values of output and operational conditions when the goal is achieved.

Goals are generalized entities. Therefore, they have to be specialized in the appropriate physical domains when they are used to represent a specific system. A goal is specialized when:

- Generalized variables, parameters, and components in the intended behavior and operational conditions of the goal are replaced by physical variables, parameters and actual components of the behavioral and structural model of a given physical system;
- The values (or admissible ranges of values) of variables and parameters are specified;
- The transformation between variables and parameters is explicitly stated.

Our approach to teleological modeling is based on the system centered ontology. It considers a system as an organized unit characterized only by parameters, input variables, and output variables. The model specifies what is expected from the system in terms of goal pattern, inputs and settings necessary to achieve the goal and the intended output behavior. Therefore, our perspective is that teleology embodies a specification of possible behaviors, not the description of a single behavior. In other words, there may exist several alternative behaviors that satisfy a given teleology.

C. Primitive Goals

Primitive goals are goals whose intended behavior can be directly accomplished by elementary phenomena. In general, the correspondence between primitive goals and elementary phenomena is *many to one*. This means that more than a single primitive goal may correspond to the same elementary phenomenon, according to which variables and parameters in the phenomenon definition are considered as inputs and outputs in the specification of the purpose. Examples of primitive goals are:

1) TO-TRANSFER x: generalized current FROM y: component TO z: component.
The generic purpose of this goal is to move a generalized current from a point to another of a system. Specific examples of this goal are "TO-TRANSFER angular velocity FROM main wheel TO escape wheel" associated to the wheel train of a watch. "TO-TRANSFER electrical current FROM battery TO light bulb" associated to an electrical circuit, etc. This goal can be achieved by an elementary phenomenon whose organization is constituted by a transporting process.
2) TO-ACCUMULATE x: generalized substance {INSIDE y: component}.
The generic purpose of this goal is to increase the amount of a generalized substance, possibly specifying the component of the system where the accumulation takes place. Specific examples of this goal are "TO-ACCUMULATE heat INSIDE water" associated to a boiler, "TO-ACCUMULATE angular momentum INSIDE flywheel" associated to a mechanical device, etc. This goal can be achieved by an elementary phenomenon whose organization is constituted by a reservoir charging process.
3) TO-SENSE-RATE-OF x: generalized current CHANGE.
The generic purpose of this goal is to sense the rate of a generalized current change in a system. Specific examples of this goal are "TO-SENSE-RATE-OF pressure CHANGE" associated to a pressure gauge, "TO-SENSE-RATE-OF velocity CHANGE" associated to an accelerometer, etc. This goal can be achieved by an elementary phenomenon whose organization is constituted by a reservoir charging process.

The epistemological link between primitive goals and elementary phenomena is represented by a mapping between the arguments of the primitive goal and the generalized variables associated to the functional roles belonging to the cofunction of the single process which constitutes the organization of the elementary phenomenon that realizes that goal.

Note that primitive goals should not be confused with a simple relabeling of the elementary phenomena. In fact, goals and elementary phenomena encode different knowledge: a goal describes, in very general terms, what is expected from the system without making any commitment on how it can be achieved. On the contrary the elementary phenomenon associated to the goal describes how these expectations can be realized in terms of which process and preconditions must be provided.

D. Nonprimitive Goals

Goals which are not primitive represent purposes that can be achieved by nonelementary phenomena. Several different nonprimitive goals may correspond to a single phenomenon and, of course, the set of nonprimitive goals is open ended. Examples of nonprimitive goals are:

1) TO-TRANSDUCE x: generalized variable INTO y: generalized variable.
The generic purpose of this goal is to convert a generalized current or substance from a physical domain to another. Specific examples of this goal are "TO-TRANSDUCE force INTO pressure" associated to a hydraulic ram, "TO-TRANSDUCE electrical current INTO torque" associated to an asynchronous motor, etc. This goal can be achieved, for example, in the asynchronous motor by a phenomenon whose organization is composed of two coupled storage processes (i.e., two reservoir charging processes) of magnetic energy influencing a mechanical generator.
2) TO-CONTROL x: generalized current BY y: generalized substance.
The generic purpose associated to this goal is to regulate a generalized current flowing out of a system, proportionally to the amount of a generalized substance accumulated inside the system. Specific examples of this goal are "TO-CONTROL electrical current BY position" associated to an electrical switch, "TO-CONTROL water flow BY angular displacement" associated to the tap of a faucet, etc. This goal can be achieved, for example, by a phenomenon whose organization is composed by a reservoir charging process that regulates a transporting process.
3) TO-KEEP x: generalized current OF y: component AT z: reference value.
The generic purpose associated to this goal is to maintain a specific partial state in time, described in terms of the values which some relevant physical variable describing the purpose of the system must hold. Specific examples of this goal are "TO-KEEP temperature OF room AT 18C°" associated to a thermostat controlled home heating system, "TO-KEEP angular velocity OF platter AT 45 RPM" as-

sociated to the control system of a turntable etc. This goal can be achieved by a phenomenon whose organization is represented by a complex network of interrelated processes constituting a feedback loop. The organization must comprehend a set of feedforward processes having a positive gain through the feedback loop (i.e., aimed at increasing the rate of flow of some substance) and a set of feedback processes having a negative gain through the loop (i.e., aimed at decreasing the same rate of flow, for example, through a regulation).

Goals can be obtained by composing together primitive and nonprimitive goals. Therefore, a goal may be described through its teleological decomposition, i.e., by explicitly specifying the primitive and nonprimitive goals upon which it is based and the ways they are interconnected. A *subgoal* relationship relates a goal to its constituent subgoals. Generally, a goal may be decomposed in several different ways, i.e. there exist alternative decompositions of the goal into constituent subgoals. Therefore, the definition of a goal can easily be visualized through an AND/OR tree, where OR nodes represent alternative decompositions of a parent goal and AND nodes represent subgoals which are all needed to achieve the parent goal. Moreover, subgoals of a given goal may be constrained by mutual temporal dependencies which can be described through an appropriate set of temporal relations inspired by [4].

The epistemological link between goals and phenomena is analogous to that illustrated in Section VI-C between primitive goals and elementary phenomena. Since there may exist alternative decompositions of a single goal into subgoals, the correspondence between goals and phenomena is, in general, many to many. In other words, the same phenomenon can be used to achieve different goals, while a single goal might be potentially achieved by different phenomena. Fig. 8 describes in generalized terms (i.e., without referring to a specific system) the epistemological links existing between the primitive goals TO-ACCUMULATE INSIDE and TO-TRANSFER, which constitute the leaves of the decomposition of the goal TO-CONTROL, and two elementary phenomena PH-1.1 and PH-1.2 that accomplish the goals. The organization of the two elementary phenomena is represented by a reservoir charging process and a transporting process, respectively. The figure illustrates how the arguments of the primitive goals are mapped into the generalized variables associated with the functional roles belonging to the cofunction of the processes constituting the organization of the two phenomena.

E. The Teleological Model

The *teleological model* of a system describes the purpose of the system by specifying the goals associated with it and their relationships. Each goal is represented in terms of goal pattern, operational conditions, and intended output behavior. Consider, for example, the temperature gauge. The purpose of the gauge, at the information level, is that of measuring temperature. At the physical level, the purpose is achieved by a transduction phenomenon which converts temperature deviations into the angular displacement of a pointer on a scale. Therefore, the purpose can be described by the goal pattern "TO-TRANSDUCE temperature INTO angular displacement" which is an instance of the generalized pattern "TO-TRANSDUCE x: generalized current INTO y: generalized substance" where the two arguments x and y have been specialized in the thermal and mechanical (rotational) domains, respectively. The operational conditions specify that the temperature gauge should be fed with an electrical voltage having a well specified value and frequency (for example, a voltage of 12 V DC) and should be used only to measure temperatures whose values lie within a specific range (for example, T within $[-20, +50]$ °C). Moreover, to enable the device to achieve its goal, the operating mode "CLOSED" for the switch of the temperature gauge should be selected. In this simple example, the selection of reference values is not necessary. The intended behavior associated with the temperature gauge specifies the transformation between the input and the output. The output variable is represented by the angular displacement ϕ of the pointer with reference to a given starting point. The value of ϕ is constrained to remain within a given range and is related to temperature deviations by a linear relationship, i.e., $\phi = K\Delta T$ where K (rad/C°) represents the gauge sensitivity. Of course, many other relationships and constraints may be used, and are actually provided, in order to specify the intended behavior of the system such as its linearity, stability, and response time.

Fig. 9 shows three teleological models of the temperature gauge: TEL-0, TEL-1, and TEL-2. Each model describes a specific level of the decomposition of a goal into subgoals. Aggregation links between teleological models at different aggregation levels are represented by subgoal relationships which relate goals in a model with the corresponding goals in a finer or coarser grained one. Model TEL-0 represents the goal G-0.1 (i.e., TO-TRANSDUCE temperature INTO angular displacement) associated by the designer to the temperature gauge. Model TEL-1 represents a finer aggregation level in which the goals associated to the gauge are G-1.1 (i.e., TO-CONTROL current BY heat), and G-1.2 (i.e., TO-TRANSDUCE current INTO angular displacement). Model TEL-2 represents a further refinement in which the teleology associated to the temperature gauge is described through four goals: the primitive goals G-2.1 (i.e., TO-TRANSFER current FROM BT TO WR) and G-2.2 (i.e., TO-ACCUMULATE heat INSIDE TH), and the nonprimitive goals G-2.3 (i.e., TO-TRANSDUCE current INTO heat) and G-2.4 (i.e., TO-TRANSDUCE temperature INTO angular displacement). The figure also shows the temporal dependencies which are held between the goals represented in each model. In general, the teleological decomposition of the system goals into subgoals and the structural decomposition of the same system into components and subcomponents leads to two hierarchies that may be quite dif-

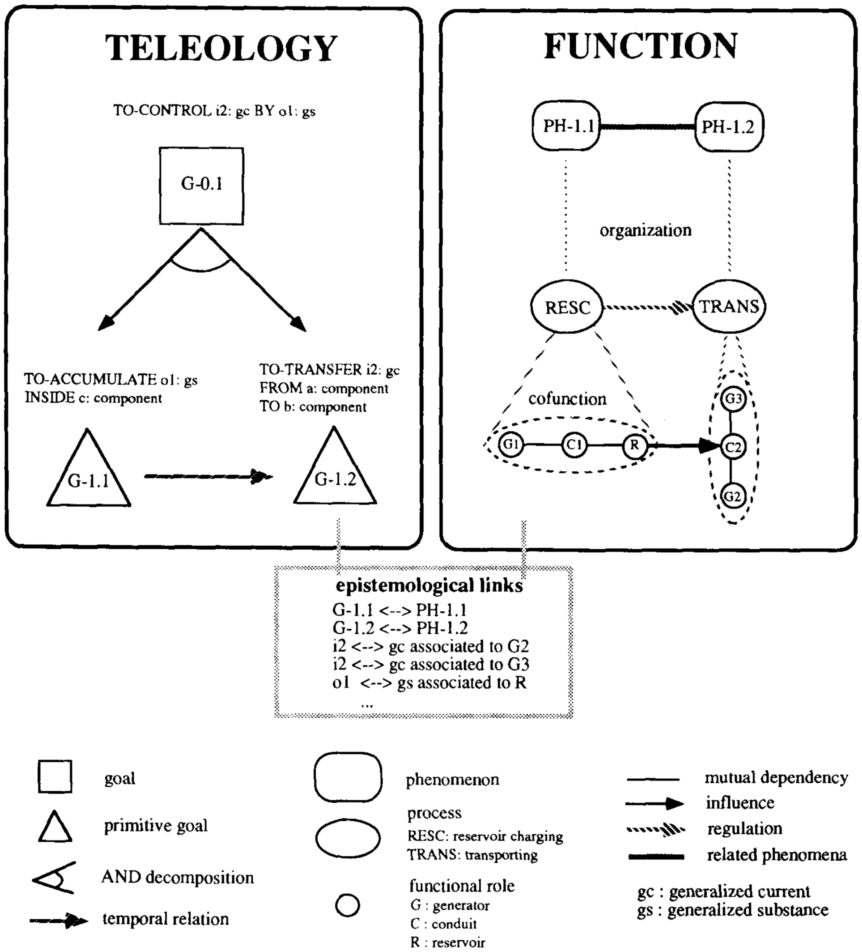


Fig. 8. TO-CONTROL: epistemological links between goals and phenomena.

ferent, especially in those devices exhibiting a great deal of function sharing [78].

The epistemological link between the teleological model and the phenomenon model is represented by associating goals (primitive and nonprimitive) to phenomena (elementary and not elementary, respectively). For example, the nonprimitive goal G-2.4, represented in the TEL-2 model, is associated with the phenomenon PH4 represented in the FUN.PH-1.1 model. The primitive goals G-2.1 and G-2.2 in the TEL-2 model are associated with two elementary phenomena (not explicitly represented in the phenomenon model) whose organizations are represented by a transporting process of electricity (i.e., the process TRANS1 shown in Fig. 6) and a reservoir charging process of heat (i.e., the process RESC4 also shown in Fig. 6), respectively. Note that there are some elementary phenomena which do not correspond to any goal. For example, the two elementary phenomena whose organizations are represented by the transporting process of heat from the wire to the air (TRANS2) and the reservoir discharging process of heat from the strip to the air (RESD1) respectively, have no goals associated with them in the teleological model. Since they actually represent side effects of other phenomena, they cannot be ascribed to any real designer's intention.

F. Using Teleological Knowledge

Two basic reasoning utilities are available for teleological knowledge. These are briefly illustrated below.

- *Interpretation of actual use* identifies which goals of the teleological model should be achieved as a result of user actions. It takes in input current inputs and settings, and provides in output a list of possible operator's goals.
- *Definition of proper use* infers the actions the user should perform to achieve a desired goal. It takes in input a list of operator's goals, and provides in output the specification of system inputs and settings required to achieve them. "Definition of proper use" fails when it leads to contradictory actions or contains inconsistent temporal information; then, a jus-

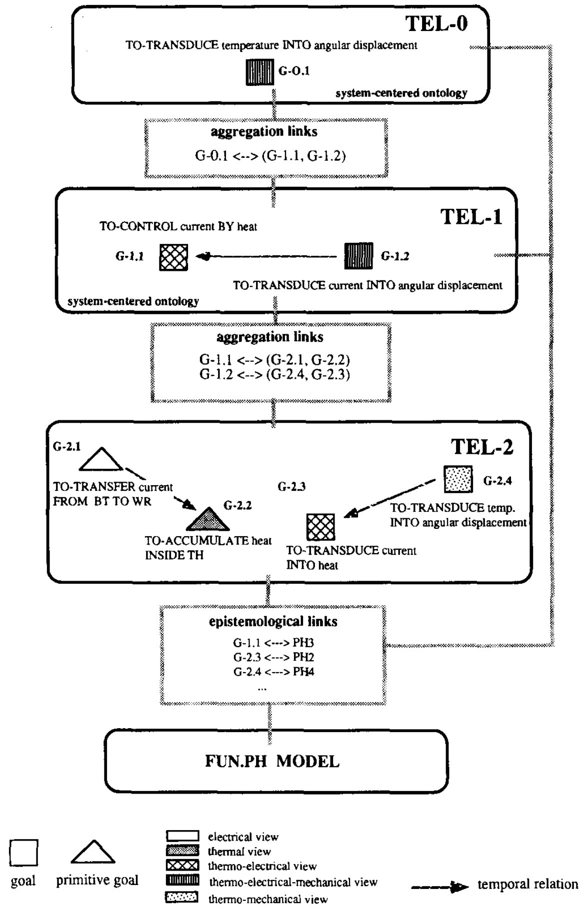


Fig. 9. Teleological models of the temperature gauge.

tification of the failure is provided to show which goals are inconsistent or unachievable and why.

G. Comparison with Related Work

The concept of goal is partially inspired by [47], however our approach to teleological modeling is more general. Goals identify generic purposes in a way that abstracts from physical domains; goals are structured entities per se, characterized by operational conditions and intended behavior; they may be composed together to yield more complex goals or decomposed until their primitive components are obtained.

In [10] a teleological model is proposed which is somehow similar to our approach. In particular, the characterization of a goal, in terms of the behavior which is expected when the system is in some specified state, is quite close to our distinction between the intended behavior associated to a goal and its operational conditions. However, our specification of such conditions is more articulated and includes aspects of temporal sequencing of settings that have been ignored in [10]. Furthermore, in our approach goals are generalized entities which are associated to phenomena in the phenomenon model. So, given a goal, the epistemological link with functional knowledge allows us to retrieve the possible functional organizations that enable the achievement of the goal. In this way, we explicitly represent the intuitive notion of "general engineering template" which is used in [10] to denote "a general arrangement of devices that is well understood" and can be used to attain a purpose.

The teleological language proposed in [33] is used to represent the relation existing between design modifications (for example, the addition of a component of an initial structure or the modification of a specific behavioral parameter) and the effects that these changes produce in terms of behaviors which are prevented, guaranteed or introduced in a system. This perspective is different from ours, because in our approach purposes are associated to systems as a whole and not to design modifications.

VII. Diagnosis Based on Multimodeling: A Case Study

A. Model-Based Diagnosis

The basic paradigm of *model-based diagnosis* can be stated as follows [23], [26], [68]. Given:

- A model of a system, including a description of the structure of the system and of the expected (normal) behavior of each component,
- A set of observations about actual system behavior, obtained through direct inspection or measurement, and
- A set of symptoms, i.e., discrepancies between actual behavior (observations) and expected normal behavior (predicted by the model),

a *diagnosis* is a set of components whose abnormal behavior can explain all the observed symptoms. A diagnosis is minimal if it has no proper subset that is also a diagnosis. The diagnostic task is aimed at discovering all minimal diagnoses.

A fundamental work in model-based diagnosis is constituted by the General Diagnostic Engine (GDE) [26], which provides a general, domain independent architecture for diagnosing any number of simultaneous faults in a system. Its diagnostic strategy is based on three main activities:

- Prediction: use observations and component models to make predictions about system behavior;
- Conflict recognition: recognize symptoms (i.e., discrepancies between predictions and observations) and construct the set of minimal conflicts. A conflict is a set of assumptions which support a symptom and lead to an inconsistency: a set of components is thus a conflict if the assumption that they are all correctly functioning leads to an inconsistency with some of the observations;
- Candidate generation: construct the set of minimal candidates. A candidate is a set of assumptions such that if all of them fail to hold, then all symptoms are

explained: a set of components is thus a candidate if hypothesizing an abnormal behavior for all of them can explain all the observed symptoms.

This process is performed incrementally in response to every new observation. Furthermore, GDE introduces a sequential probing strategy that uses probabilistic information and a minimum entropy technique to discriminate among remaining candidates by performing as few measurements as possible.

B. Limitations of Current Approaches to Model-Based Diagnosis

The basic paradigm of model-based diagnosis refers to the use of a single monolithic model of the system to be diagnosed representing only structural and behavioral knowledge. This approach suffers from several limitations:

1) Representation of symptoms. Symptoms can be represented only in terms of structural and behavioral knowledge. However, in real-world diagnosis symptoms are often formulated at a more abstract level using functional or teleological knowledge; for example, in terms of a missing functional role or a missing process or in terms of unachieved goals.
2) Representation of faults. Faults can be represented only in structural or behavioral terms thus causing the following problems:
   a) Fault representation can be inappropriate. Consider, for example, the case of a fault to the cooling system in a nuclear power plant. In this case, the representation of the fault in structural or behavioral terms could simply be at the wrong abstraction level and would fail to help the operator in the compensation task which requires, instead, information about mass and energy flow.
   b) Fault representation can be impossible. This happens, for example, when the symptomatic behavior of a system can be ascribed to unproper use or unproper operational conditions.
3) Detection of symptoms. As it has already been pointed out by other researchers [1], not every behavioral discrepancy is necessarily a true symptom. The observed behavior can be different from the expected one, for example, an observed wave form is different in shape from the expected one. However, the normal functioning of the system may be achieved since the goals behind the two behaviors are the same. For example, the observed wave form carries the same energy of the expected one.
4) Computational complexity. Diagnostic approaches based only on structural and behavioral models feature a very high computational complexity, caused by the combinatorial explosion exhibited by the activities of prediction, conflict recognition, and candidate generation in diagnosing complex systems.
5) Physically erroneous diagnoses. The consistency based definition of diagnosis may provide diagnoses which are logically but not physically acceptable [77].

Attempts to tackle these problems have followed two main directions.

One direction has been to model a system at a more abstract level by explicitly representing knowledge about the functions, or the purposes associated to the system, by the designer [1], [31], [73]. However, diagnostic approaches exploiting only functional or teleological knowledge also present some serious limitations. For example, since these approaches are able to consider symptoms represented only in functional or teleological terms, a symptom has to manifest itself as a loss of function or purpose; otherwise, it cannot even be identified. Analogously, since faults can be represented only in functional or teleological terms it is impossible to diagnose, for example, parameter shifts if these shifts do not cause a loss of function or purpose. Moreover, diagnoses represented in terms of purpose and functions may be too abstract when a more detailed solution of the diagnostic problem is required, for example, at structural or behavioral level.

Another direction has been aimed at combining different models of the system to be diagnosed in order to exploit their cooperative problem solving power. These proposals have investigated a wide range of possibilities and perspectives such as

- Using a hierarchy of structural and behavior models of different aggregation levels [36];
- Exploiting the integration of both qualitative and quantitative behavioral models [34];
- Extending the GDE approach by incorporating fault models [42], [77];
- Extending the GDE approach by exploiting models of different behavioral modes for each component [27];
- Integrating empirical and functional models [31] or teleological and fault models [73];
- Exploiting the integration of several models based on various knowledge types, namely: structural, behavioral, functional, and empirical [1], [10], [16], [33], [43], [60].

Most of these approaches are characterized, however, by the following limitations:

1) Absence of a general modeling approach. These approaches usually achieve their integration goal by including functional or teleological knowledge within the behavioral and structural model, or by establishing compiled associations between different models of the system. These solutions lead to very specific, hardly reusable models. Moreover, in the case of empirical associations, one of the main advantages of model-based reasoning, namely justification and explanation of diagnostic results, is impaired.

2) Absence of a solid theoretical framework for model integration and cooperation. These approaches do not provide a framework where all the models are integrated in a clear and structured way. As a consequence, they have only a limited problem solving power. For example,

a crucial problem with multiple models is the reformulation of results obtained within a model in terms of other models. Without a clear theoretical framework, current approaches are unable to tackle this problem in general terms. Usually the reformulation is hardwired in the representation and is only suitable for the specific application considered.

3) Absence of a general theory of diagnosis based on multiple models. The formalization of the concept of diagnostic task with multiple models is still an open problem.

C. The Multimodeling Approach to Diagnosis

The multimodeling approach to diagnosis extends the model-based approach by exploiting the use of all the available knowledge about a system, i.e., fundamental, interpretative, and empirical knowledge which is appropriately framed into separate but interconnected models. As a consequence, the concept of diagnosis assumes a greater generality since observations, symptoms, conflicts and faults are not required to be only of behavioral type but may concern any epistemological type. Moreover, in the multimodeling approach, the availability of different models where it may be possible to perform the same elementary diagnostic task, e.g., conflict recognition, and the clear separation existing between the domain level competence, i.e., competence relevant to reasoning "inside" each model and the control level competence, i.e., competence relevant to reasoning "through" models, allows for a greater flexibility. It is possible, for example, to experiment different diagnostic strategies or to adapt the diagnostic process to situations which impose specific constraints on the reasoning process in terms of limited computational resources, precision and accuracy of solutions, etc. by selecting the models that better satisfy the constraints.

More specifically, the integration of interpretative and fundamental models offers several advantages which are briefly listed below:

1) Diagnosing the operator behavior: the use of teleological models allows us to consider faults which do not have a structural nature, such as unproper use by the operator. This capability is important in several real world applications where human errors have been recognized as the major cause of observed symptoms.
2) Focusing the diagnostic activity: the use of interpretative knowledge allows the diagnostic task to be performed in a focused way. It is possible, for example, to start the diagnostic activity at the teleological level and then, by exploiting the bridge between teleology and behavior represented by functional knowledge, to consider only those parts of the structural and behavioral models which are responsible for the unachieved goals. This may result in a considerable refinement of the conflict recognition activity.
3) Avoiding physically erroneous diagnoses: the use of functional models gives a global point of view of the system behavior which is interpreted in terms of processes and phenomena. In the consistency based approach to diagnosis, this may help in rejecting those diagnoses which are logically but not physically acceptable.
4) Representing symptoms and faults at different levels of abstraction: symptoms and faults can be represented with reference to any available model thus allowing a diagnosis to be started or a fault to be formulated using different epistemological types or at different levels of aggregation in such a way as to match the specific features of the application domain and the cognitive requirements of the user.

The above advantages result in a diagnostic process which is more plausible from a cognitive point of view and in a better capability of producing acceptable justifications, both in the case of successful reasoning and of failure.

In order to experiment with the multimodeling approach we have developed a prototype system called DYNAMIS. Its general architecture is inspired to the control blackboard approach [40]. DYNAMIS includes several knowledge sources, called *specialists*, and two shared memories, called *domain blackboard* and *control* blackboard, respectively. The specialists are divided into two classes:

- *Control specialists*, which manage the overall organization of the problem solving activities at the domain level. The control specialists perform the "reasoning through models" activity;
- *Domain specialists*, which implement the basic reasoning utilities within the various models used for representing the system considered. The domain specialists perform the "reasoning inside a model" activity.

A description of the DYNAMIS architecture (blackboard data structures, specialists, control) has been given in [19].

DYNAMIS has been implemented in PROLOG on a SUN-4 workstation. The DYNAMIS system has been used so far to experiment with three diagnostic applications concerning a thermostat controlled home heating system [18], a turntable, and a lighting system.

D. A Sample Diagnostic Session

In the following, we present a sample diagnostic session with DYNAMIS, focusing on the use of teleological and functional knowledge. The diagnostic session concerns the lighting system depicted in Fig. 10. The lighting system is devoted to the lighting of three separate rooms of a building and is constituted by four switches, four power regulators, eight bulbs, and some wires connecting these components. Fig. 11 shows the portions of the structural (STR) and behavioral (BEH) models relevant to the diagnostic session. The part of the lighting system represented in these models comprises the set of components which are common to the three circuits providing

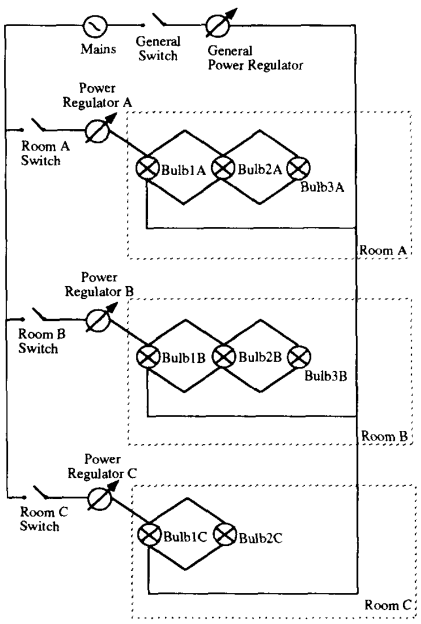


Fig. 10. Schema of the lighting system.

electricity to the three rooms, i.e., mains, general switch, general power regulator, and wire1, together with the components that constitute the specific lighting subsystem of room B. The component "ROOM B" explicitly represents the environment which is the destination of the light flow. The epistemological links that relate the structural to the behavioral model and the links relating the behavioral model to the FUN.R model allow identifying the functional roles associated to components in the different views.

Fig. 12 shows the portions of the teleological (TEL-0, TEL-1) and functional (FUN.R, FUN.P, FUN.PH-0, FUN.PH-1) models of the system that will be used in the diagnostic session. Note that some components are bound to different functional roles in different physical views: the bulbs, for example, are conduits in the electrical view and generators of light flow in the optical view. By considering only the electrical and optical views, six transporting processes are identified and represented in the FUN.P model: TRANS1, TRANS2, TRANS3 in the electrical view, and TRANS4, TRANS5, TRANS6 in the optical view. Each transport process in the electrical view directly causes one of the transport processes in the optical view. Moreover, each of these three pairs of processes represents in the FUN.PH-1 model the organization of a phenomenon, which couples the electrical and optical views and represents conversion of electrical power into light. The three phenomena (PH2, PH3, PH4) are aggregated into the single phenomenon PH1 which is represented in the FUN.PH-0 model. This phenomenon is associated to goal G-1.1 in the TEL-1 model which specifies the desired relation between the electrical power provided to the bulbs in room B and the obtained light. Goal G-1.2 specifies the desired relationship between the settings (knob position) of two power regulators (general power regulator and power regulator B) and the electrical power provided to the bulbs in room B. Finally, goal G-1.1 and G-1.2 are aggregated into the single goal G-0.2 in model TEL-0 which specifies the desired relation between the setting of power regulators and the obtained light in room B.

The sample session presented in the following is organized around three cases devoted to illustrate three main features of the DYNAMIS system: an elementary task devoted to identify possible errors of the operator (i.e., operator diagnosis), an application of opportunity driven model selection to diagnosis (i.e., diagnosis focusing), and an application of functional knowledge to diagnosis (i.e., functional conflict recognition).

When DYNAMIS is started, the reasoning through models activity is devoted to identify what kind of problem the operator would like to solve, in order to allow the selection of a proper task specific problem solving method.

DYNAMIS: Do you have any problem with the lighting system?
OPERATOR: Yes.
DYNAMIS: What is the problem?
OPERATOR: Malfunction.
DYNAMIS: Enter all information you have about the lighting system.
OPERATOR:

general switch	ON	general power regulator	0
switch A	ON	power regulator A	0
switch B	ON	power regulator B	50
switch C	ON	power regulator C	0

Since the operator presents a diagnostic problem where settings of controls (i.e., switches positions and knobs positions of power regulators) are the only available data, DYNAMIS selects a task specific problem solving method which consists in first analyzing whether the operator is using the lighting system properly (by exploiting the elementary task *operator diagnosis*). If no errors of the operator are found, a diagnosis of the physical system is performed, by gathering initial symptoms (by exploiting the elementary task *symptom gathering*), and then by generating and discriminating diagnoses (by iteration of the three elementary tasks *conflict recognition*, *candidate generation*, and *candidate discrimination*).

E. Operator Diagnosis

In order to solve the elementary task "operator diagnosis," the teleological model at the highest level of ag-

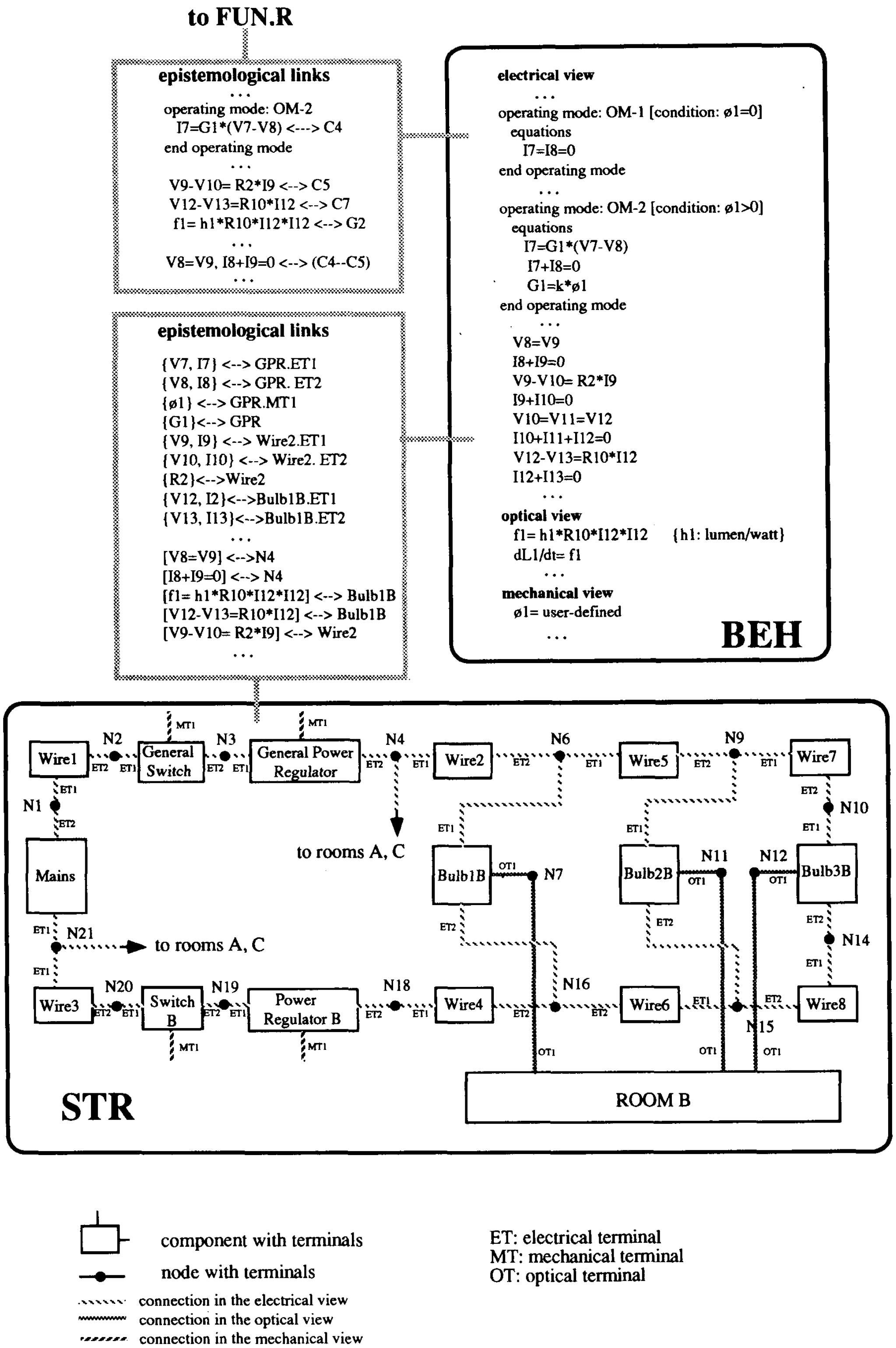


Fig. 11. Partial structural and behavioral models of the lighting system.

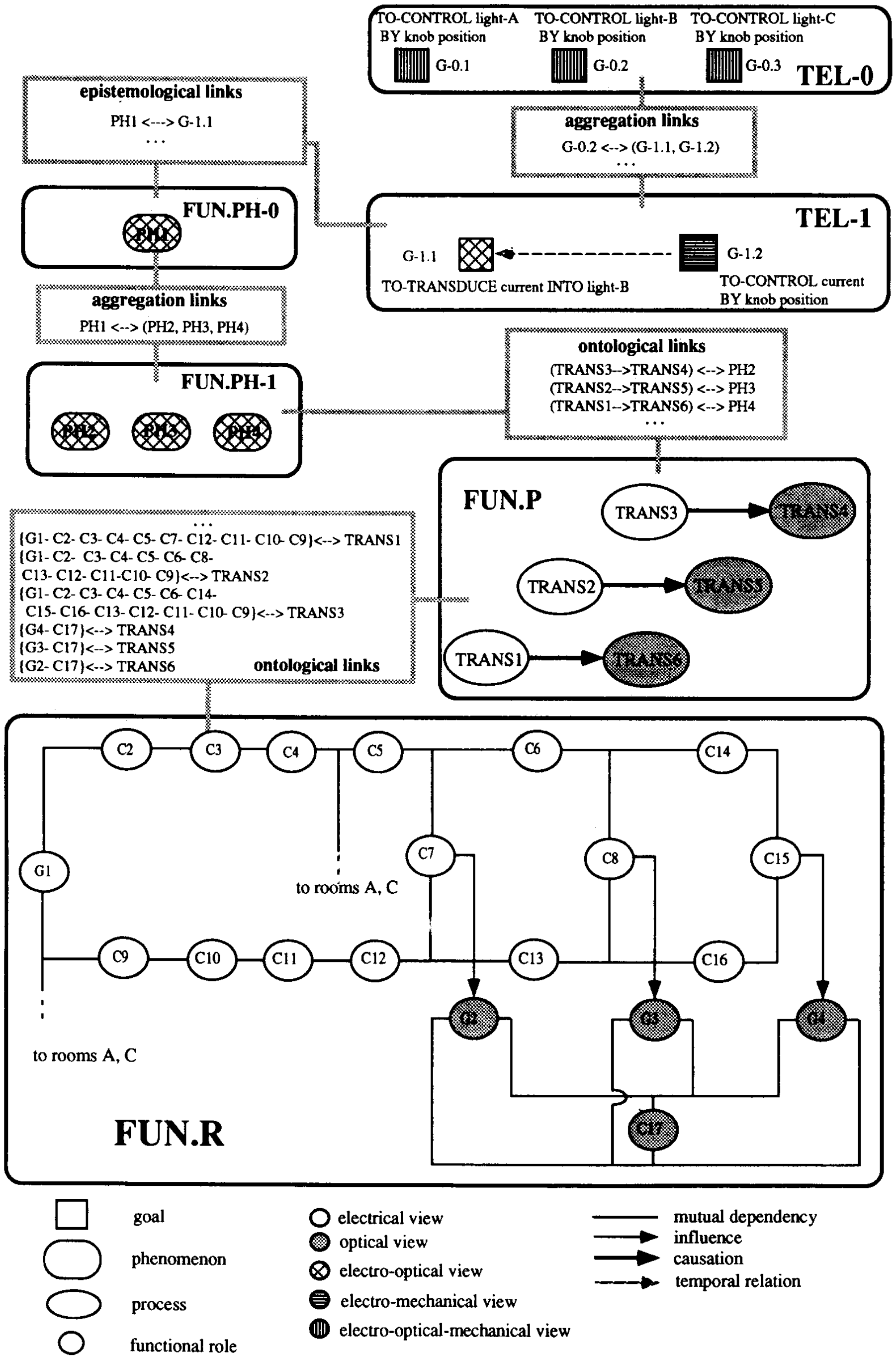


Fig. 12. Partial teleological and functional models of the lighting system.

gregation (i.e., TEL-0) and the following models-based problem solving method are selected. There are cases where other models-based problem solving methods can be selected, for example, when complex temporal relations are contained in the description of operation, or if there is need to take cognitive aspects into more consideration:

MODELS-BASED PROBLEM SOLVING METHOD OD/01

```
COMMENT: this method requires SettingOfControls
  to be known and, using a teleological model, tries to
  identify
WrongSettings, i.e., which settings the operator failed
  to set properly.
BEGIN
      ask the operator which goals of the selected
        model he wants to achieve and
             let OperatorGoals be the set of those goals
      definition_of_proper_use(IN OperatorGoals,
        OUT ProperSettings)
      IF SettingOfControls 〈〉 ProperSettings THEN
             WrongSettings := SettingOfControls
                − Proper Settings
      ELSE
             WrongSettings := ∅
      ENDIF
END
```

As it can be noticed, this models-based problem solving method uses "definition of proper use," the basic reasoning utility which considers the set of operator goals and tries to derive the specification of the system settings required to achieve them. There are two possible outcomes for this basic reasoning utility: i) a consistent specification for system settings is found or ii) the specification of required system settings contains contradictory actions (for example, a system control should be set simultaneously to ON and OFF) or inconsistent temporal information (for example, system control X must be set after system control Y which, in turn, must be set after X). In the first case, "definition of proper use" succeeds and the models-based problem solving method proceeds. If the actual settings of controls is different from the proper one, the difference (obtained by set subtraction) between actual and proper settings of controls identifies an error of the operator. In the second case, "definition of proper use" fails (leading to failure of the models-based problem solving method) and records an explanation of its failure, which can be used for a failure-driven model selection and the choice of the next models-based problem solving method. In this way, teleological knowledge can be further exploited to engage the operator in a training dialogue aimed explicitly at showing him which goals are inconsistent and the reasons why they are inconsistent. The operator could be then supported in the revision of his goals.

In the current session, goals found by DYNAMIS in TEL-0 are G-0.1, G-0.2 and G-0.3. Therefore, the operator is engaged in the following dialogue:

DYNAMIS: Select which are your goals from this list:
light room A light room B light room C
OPERATOR: light room B
DYNAMIS: So, your goals are:
NOT light room A light room B NOT light room C
Is this right?
OPERATOR: Yes

"Definition of proper use" is then executed, obtaining the proper settings of controls required to achieve the goals stated by the operator. Then, the obtained proper settings of controls are compared with the actual settings, in order to find which are wrong. An incorrect match is found (i.e., general power regulator is set to zero, while it should have been set higher) and the following message is displayed to the operator:

DYNAMIS: Fault in the settings detected:
general power regulator must be set higher than zero.

Since a wrong setting has been found, the submitted problem is considered solved and the operator is told:

DYNAMIS: Your problem is solved. Do you want to terminate the session?

The operator, before terminating the session, tries the suggested intervention to check if it can actually solve the problem. So, he sets the general power regulator to 50. Unfortunately, the lighting system does not yet operate as expected, therefore he answers negatively to the above question:

OPERATOR: No

DYNAMIS then tries to identify the reasons why the operator wants to continue the session:

DYNAMIS: Have you corrected the settings as suggested?
OPERATOR: Yes
DYNAMIS: After having corrected the settings appropriately, is the lighting system working as you want?
OPERATOR: No

Since the answer of the operator shows that he considers the system still malfunctioning and since settings of controls have now been set properly, the current task-specific problem solving method focuses next on diagnosing the lighting system.

Before proceeding further, let us stress that operator diagnosis, made possible through exploitation of teleological knowledge, allows considering the operator as a part of the diagnosed system and as a possible cause of malfunctions. This capability is of primary importance in several real world applications. For example, in the field of plant supervision and operation, human errors of control room operators have often been recognized as a major cause of faults or critical events. Also in the field of after-sales support of consumer or professional devices installed in homes, offices, or factories, service personnel often receive calls for intervention from people who are simply misusing a system which is functioning perfectly.

F. Diagnosis Focusing

The next elementary task "symptom gathering" in the current task-specific problem solving method consists in finding initial symptoms. At the moment, DYNAMIS is not yet aware of any symptom. In order to execute "symptom gathering," the reasoning through models activity selects a models-based problem solving method which consists of 1) identifying which goals the operator considers as symptoms, (i.e., those goals whose unachievement or achievement does not match operator expectations) and 2) asking for initial observations concerning those goals, thus verifying if operator judgement about them is correct.

```
MODELS-BASED PROBLEM-SOLVING METHOD
  SG/01
COMMENT: this method requires OperatorGoals to
  be known and, using a teleological model, identifies the
  set
TelSymptoms (i.e., those goals the operator considers
  as symptoms) and verifies if operator judgement about
  those goals is correct
BEGIN
      ask the operator to pinpoint those goals in
                         OperatorGoals which do not
                           match his expectations and
                         let TelSymptoms be the set of
                           those goals
      FOREACH Goal in TelSymptoms DO
            ask the operator to observe Outputs of
              Goal
            IF (user answer = Outputs of Goal)
                  THEN fail
            ENDIF
      ENDDO
END
```

This method can end with success or failure. In the first case, results of the observations help to better qualify the symptoms. For example, knowing that room B is too much, too little, not at all or only partially lit is more useful in focusing the diagnostic process than knowing only that the operator deems the "light room B" goal unachieved. In the second case, some observations contradict what the operator stated. For example, the operator deems room B excessively lit and reports the problem to the system, but the level of lighting obtained is exactly what the model predicts for the actual settings of controls. In the latter case, an interesting situation arises: a correct system behavior is deemed a symptom by the operator. This requires a finer operator diagnosis aimed at identifying operator's misconceptions about the physical system (operator diagnosis as performed before was aimed only at identifying wrong settings of controls).

Following the models-based problem solving method described above, DYNAMIS starts by asking the operator which of his goals he considers as symptoms:

DYNAMIS: Please, tell me which of your goals
NOT light room A light room B NOT light room C
are not satisfied.
OPERATOR: light room B

The set of goals considered as symptoms by the operator consists then of a single goal. So, DYNAMIS asks the operator for some easily observable facts relevant to that goal:

DYNAMIS: Is bulb 1 in room B lit up?
OPERATOR: No
DYNAMIS: Is bulb 2 in room B lit up?
OPERATOR: No
DYNAMIS: Is bulb 3 in room B lit up?
OPERATOR: Yes

All three bulbs should have been lit up, so goal G-0.2 and at a lower level goal G-1.1 are actually not achieved (while they should have been achieved). DYNAMIS has now a better description of the symptom which will be used to focus the following steps of the diagnostic activity.

Since goals which are actually symptoms have been found, the possibility of performing an opportunity-driven model selection (i.e., switching to the functional representation to perform the next elementary task of the current task-specific problem solving method) is detected in the reasoning through models activity. So, the available observations are first translated from the teleological model into the functional models. The translation is accomplished by traversing the links connecting them. More precisely, considering the functional models in Fig. 12, the available informations in the teleological model will be translated into:

- Phenomena PH1, PH3 and PH4 are not satisfied (this identifies three symptoms in the FUN. PH-0 and FUN.PH-1 models), while phenomenon PH2 is satisfied;
- Processes TRANS5 and TRANS6 are inactive (this identifies two symptoms in the FUN.P model), while process TRANS4 is active;
- Generators G2 and G3 are not generating light (this identifies two symptoms in the FUN.R model), while generator G4 is generating light.

The currently available information about settings of controls is also translated (e.g., switch B set to ON corresponds to conduit C10, etc.).

The teleological model is used to focus the diagnostic process in the functional models. Focusing consists here in identifying the portion of the functional models which corresponds to those goals which constitute the most detailed symptoms in the teleological model (goal G-1.1 in the current case). In particular, focus control which switches the diagnostic reasoning from the teleological model to the functional models is carried out by: 1) selecting the goals at the lowest levels of aggregation among those goals which are symptoms and 2) starting from the selected goals and traversing links to reach and select corresponding parts in the functional models.

Fig. 12 shows the functional role network and process network selected in the current case: they describe only those roles and processes supporting the phenomena linked to goal G-1.1, which is unachieved.

G. Functional Conflict Recognition

As shown in the previous part of the session, DYNAMIS has just switched to the functional models and selected only a portion of these models, where the next elementary task, "conflict recognition," has to be executed in order to generate the conflict set. In functional terms, a conflict is a set of functional roles where at least one role must be violated. That role is expected in the model of the physical system but would not be found in the real system. The following models-based problem solving method is devoted to construct the conflict set:

```
MODELS-BASED PROBLEM-SOLVING METHOD
  CR/02
COMMENT: this method requires at least one symptom
  in FUN.P to be known and, using the selected aggre-
  gation level of FUN.R and FUN.P, tries to build
  FunConflictSet, i.e., the functional conflict set.
BEGIN
      FunConflictSet := { }
      OKRoles := { }
      FOREACH Symptom in FUN.P DO
            functional_dependency_analysis(IN
              Symptom, OUT SetsofProcesses)
            FOREACH SetofProc in SetsofProcesses
              DO
                  Roles := union of the sets of
                    functional roles that are con-
                    nected by epistemological links
                    to the processes in
                    SetOfProc
                  FunConflictSet := FunConflict-
                    Set ∪ {Roles}
            ENDDO
      ENDDO
      FOREACH Obs in FUN.P : Obs is NOT a
        symptom DO
            functional_dependency_analysis(IN Obs,
              OUT Processes)
            Roles := union of the sets of functional
                     roles that are connected by
                     epistemological links to the
                     processes in Processes
            OK Roles := OK Roles ∪ Roles
      ENDDO
      FOREACH Set in FunConflictSet DO
                     Set := Set − OKRoles
      ENDDO
END
```

The models-based problem solving method constructs the conflict set by first using symptoms in the FUN.P model and identifying the processes and functional roles which may be responsible for the symptoms in order to generate an initial conflict set (this activity is carried out by the first and second FOREACH structures in the models-based problem solving method). Then, it tries to prune, if possible, the initial conflict set using functional observations which are not symptoms (referred to as *normality observations* in the following): 1) by identifying the processes and functional roles which are necessary to support the normality observations (this activity is carried out by the third FOREACH structure in the models-based problem solving method), and 2) by subtracting such roles from the current conflicts (this activity is carried out by the fourth FOREACH structure in the models-based problem method).

In the current session, functional symptoms in FUN.P are "process TRANS5 is inactive" and "process TRANS6 is inactive." There is only one normality observation in FUN.P, namely "process TRANS4 is active." The two symptoms in FUN.P are considered in the models-based problem solving method CR/02 in order to generate the initial conflict set. "Functional dependency analysis" determines that TRANS2, if inactive, can be the cause of the symptom "TRANS5 is inactive," and therefore, the basic reasoning utility returns the set {{TRANS2, TRANS5}}. Then, all the roles belonging to the cofunction of TRANS2 and TRANS5 are identified by means of epistemological links. These roles are taken as elements of a conflict, meaning that at least one role involved in supporting TRANS2 or TRANS5 must be violated. The generated conflict is therefore:

$$CFT1 = \{G1, C2, C3, C4, C5, C6, C8, C13, C12, C11, C10, C9, G3, C17\}.$$

Following the same line of reasoning and considering the symptom "TRANS6 is inactive," the following second conflict is generated:

$$CFT2 = \{G1, C2, C3, C4, C5, C7, C12, C11, C10, C9, G2, C17\}.$$

The initial conflict set is constituted by the two sets CFT1 and CFT2. Now, the conflict set is possibly pruned, by searching for roles which cannot be violated. Since there is one normality observation ("TRANS4 is active") in FUN.P, functional dependency analysis is invoked to determine which processes are necessary to support the normality observation. The invoked basic reasoning utility determines that TRANS3 must be necessarily active in order to support the normality observation "TRANS4 is active." Then, all those roles belonging to the cofunction of TRANS4 and TRANS3 are identified by means of epistemological links. In order to have TRANS4 and TRANS3 active, the identified roles cannot be violated. The set of roles thus generated is:

$$\{G1, C2, C3, C4, C5, C6, C14, C15, C16, C13, C12, C11, C10, C9, G4, C17\}.$$

Since these roles cannot be violated, they are now subtracted from conflicts CFT1 and CFT2. Thus the execution of the models-based problem solving method CR/02 terminates by generating the pruned conflict set:

$$\{CFT1 = \{C8, G3\}, CFT2 = \{C7, G2\}\}.$$

Being the conflict set now available, the possibility of performing an opportunity driven model selection (i.e., switching to the structural model for executing the next elementary task "candidate generation" within the task-specific problem solving method) is detected. The conflict set is first translated in structural terms, by traversing the links between FUN.R and BEH and the links between BEH and STR: in our case, CFT1 becomes {BULB2B} and CFT2 becomes {BULB1B}. Then reasoning is switched to the structural model, where the minimal sets which have a nonempty intersection with every conflict are computed, following [26]. The only candidate which can be generated from {BULB1B} and {BULB2B} is thus the common superset {BULB1B, BULB2B}. Since in this case there are not multiple candidates, the next elementary task (i.e., candidate discrimination) does not need to be executed and the diagnostic process terminates and provides the generated candidate as the solution of the current problem. The following message is then displayed to the operator:

DYNAMIS: The cause of malfunction has been located: bulb 1 and bulb 2 in room B are faulty.

DYNAMIS: Your problem is solved. Do you want to terminate the session?

At this point the operator may, for example, ask for further explanations about system conclusions, which DYNAMIS can provide by resorting again to functional knowledge.

It is interesting to note that the use of functional knowledge allows the system to avoid the generation of physically erroneous diagnoses, a problem which affects the GDE method [26]. As indeed demonstrated in [77], the diagnoses generated by GDE in the examined case would include, in addition to the correct one, other physically erroneous diagnoses, such as:

- {MAINS, BULB3B}: there is a fault in the mains power supply (bulb1B and bulb2B are not lit up) and a fault in bulb3B (it is lit up even if no electrical current flows through it);
- {WIRE1, WIRE7}: wire 1 is faulty (bulb1B and bulb2B are not lit up) and wire 7 is faulty (it produces power).

To avoid the generation of physically erroneous diagnoses in GDE, the approaches proposed in [27] and [77] rely on the use of fault models of system components. However, this solution requires:

- To have complete knowledge of the various ways components may fail (otherwise false conclusions may be drawn again);
- To manage the combinatorial explosion which arises in considering the various combinations of different possible faults.

Avoidance of physically erroneous diagnoses is made possible using functional knowledge without requiring that a complete set of fault models is available and that all possible combinations are tested, because of the following three reasons:

1) The functional model is based on the representation of the generalized substances flowing through the system. Processes give a global point of view of the paths which these flows follow and cofunctions of processes identify all functional roles involved in supporting a specific process. The epistemological link (established by the cofunction) between functional roles and the processes they support ensures that physical consequences of local role changes are globally reflected. For example, in the considered lighting system a change of role C2 from conduit to barrier could not explain the fact that processes TRANS1 and TRANS2 are inactive if TRANS3 has been observed to be active. Since C2 would be responsible for stopping electrical flow, thus deactivating TRANS1 and TRANS2, it would also deactivate TRANS3.

2) The functional model is based on generalized functional roles which can undergo only a small set of physically legitimate changes. Thus, for example, a conduit cannot become a generator, while it can become a barrier.

3) The functional model provides an explicit representation of dependence relations between different physical views. Thus, for example, a bulb cannot be a light generator (in the optical view) if electrical current does not flow through it (in the electrical view).

All these three features have been used implicitly in the above example to avoid generation of physically erroneous diagnoses. Diagnosis {MAINS, BULB3} is not generated thanks to points 1 (there is no electrical current through BULB3 if MAINS is not providing power) and 3 (BULB3 cannot generate light if there is no electric current through it). Diagnosis {WIRE1, WIRE7} is not generated thanks to point 2 (WIRE7 cannot become a generator). Note that even admitting that WIRE7 can become a generator, the diagnosis {WIRE1, WIRE7} would still not be generated, since point 1 would highlight that electrical current through BULB1, BULB2 cannot be stopped, in this case, by WIRE1 being faulty.

In case the diagnostic application would require explicit fault models to provide precise descriptions of faults (for example, "BULB1 is burned"), they can easily be represented in the multi-modeling approach as part of the empirical model.

VIII. Conclusion

In this paper we have illustrated three main topics:

- The main concepts of the multimodeling approach;
- The representation and use of functional and teleological knowledge within the multimodeling approach;
- The exploitation of multimodeling in diagnosis.

The main contributions of the presented work are:

- A general and theoretically founded framework for organizing and using many diverse models of a physical system in a cooperative way, thus providing a novel concept of model-based reasoning;
- A clear definition of the concepts of function and teleology and of their relations to structure and behavior;
- The design of a novel knowledge organization and reasoning paradigm which can support a large variety of complex problem solving tasks, such as interpretation, diagnosis, design, simulation, etc.

Moreover, from a cognitive perspective, the multimodeling approach can support the design of more adequate model-based systems, thus improving the level of man-machine coupling. In particular, it provides a concrete basis for:

1) Choosing (or enabling the user to choose) the most appropriate model to be used in a specific problem solving task. In this way, the system can help the user in reducing the complexity and opacity of the computer representation of the physical system at hand, by acting as a logical filter [44] focusing user attention only on the relevant knowledge. Moreover, the user can evaluate different interpretations of the situation at hand and focus on particular data, domains, or hypotheses. As a result, he can release part of his mental workload and understand his errors. The possibility of experimenting with several different models can also increase user conceptualization capabilities, as pointed out in [83];
2) Accounting for user actions, identifying erroneous interventions, deducing their consequences, and suggesting possible recovery operations;
3) Offering an appropriate environment for the study and simulation of human errors [69]. In fact, when the user formulates a specific action plan, slips can be characterized as an erroneous execution of a correct plan (due to stress, fatigue, etc.), while mistakes can be ascribed to erroneous knowledge present in the models or to erroneous use of the models leading to the formulation of an incorrect plan.

Finally, in the multimodeling approach, the disciplined and structured exploitation of functional and teleological knowledge turned out to be fully adequate for supporting innovative strategies in the specific domain of system diagnosis, such as operator diagnosis, diagnosis focusing, and functional conflict recognition.

Future activity will focus on the following main issues:

- Design of further general diagnostic strategies based on multimodeling;
- Application of the multimodeling approach to design tasks;
- Extension of the approach to functional and teleological modeling so far proposed to deal with new domains where the flow structure paradigm is not appropriate, for example, phenomena concerning static equilibrium;
- Application of the multimodeling approach to systems other than physical artifacts, such as, natural systems (e.g., rivers, lakes, soil, volcanoes, etc.) or conceptual systems (e.g., social, economic, etc.);
- Exploitation of the capability of the multimodeling approach to deal with incomplete models.

References

[1] A. Abu-Hanna, R. Benjamins, and W. Jansweijer, "Device understanding and modeling for diagnosis," *IEEE Expert.*, vol. 6, no. 2, pp. 26-31, 1991.

[2] S. Addanki and E. Davis, "A representation for complex physical domains," in *Proc. 9th Int. Joint Conf. Art. Intell.*, Los Angeles, CA, 1985, pp. 443-446.

[3] S. Addanki, R. Cremonini, and J. S. Penberthy, "Graphs of models," *Art. Intell.*, vol. 51, pp. 145-177, 1991.

[4] J. Allen, "Towards a general theory of action and time," *Art. Intell.*, vol. 23, pp. 123-154, 1984.

[5] M. I. Bauer and B. Reiser, "Incremental envisioning: The flexible use of multiple representations in complex problem solving," in *Proc. 12th Conf. Cogn. Sci. Soc.*, Cambridge, MA, 1990, pp. 317-324.

[6] A. Bertin, M. Gallanti, A. Stefanini, and G. Tornielli, "Diagnostic reasoning based on dependency analysis," in *Proc. 1st Cong. Ital. Assoc. Art. Intell.*, Trento, Italy, 1989, pp. 274-279.

[7] T. Bylander, "A critique of qualitative simulation from a consolidation viewpoint," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-18, no. 2, pp. 252-263, 1988.

[8] T. Bylander and B. Chandrasekaran, "Understanding behavior using consolidation," in *Proc. 9th Int. Joint Conf. Art. Intell.*, Los Angeles, CA, 1985, pp. 450-454.

[9] D. G. Bobrow (Ed.), "Special Volume on Qualitative Reasoning about Physical Systems," *Artificial Intell.*, vol. 24, 1984.

[10] J. A. Bradshaw and R. Young, "Evaluating design using knowledge of purpose and knowledge of structure," *IEEE Expert*, vol. 6, no. 2, pp. 33-39, 1991.

[11] G. Brajnik, L. Chittaro, G. Guida, C. Tasso, and E. Toppano, "The use of many diverse models of an artifact in the design of cognitive aids," in *Proc. 2nd European Met. Cogn. Approaches to Process Control*, Siena, I, 1989, pp. 69-79.

[12] G. Brajnik, L. Chittaro, C. Tasso, and E. Toppano, "Epistemology, organization and use of functional knowledge for reasoning about physical systems," in *Proc. AVIGNON '90: 10th Int. Workshop on Expert Systems and Their Application: General Conf. Second Generation Expert Systems*, Avignon, France, 1990, pp. 53-66.

[13] G. Brajnik, L. Chittaro, G. Guida, C. Tasso, and E. Toppano, "Representation and use of teleological knowledge in the multi-modeling approach," in *Trends in Artificial Intelligence*, E. Ardizzone, S. Gaglio, and F. Sorbello, eds. Berlin, FRG: Springer Verlag, pp. 167-176, 1991.

[14] B. Chandrasekaran, "Generic tasks in knowledge-based reasoning: High-level building blocks for expert system design," *IEEE Expert*, vol. 1, no. 3, pp. 23-30, 1986.

[15] —, "Design problem solving: A task analysis," *AI Magazine.*, vol. 11, no. 4, pp. 59-71, 1990.

[16] B. Chandrasekaran, J. W. Smith, and J. Sticklen, "Deep models and their relation to diagnosis," *Art. Intell. Med.*, vol. 1, pp. 29-40, 1989.

[17] M. T. H. Chi, P. J. Feltovitch, and R. Glaser, "Categorization and representation of physics problems by experts and novices," *Cogn. Sci.*, vol. 5, pp. 121-152, 1982.

[18] L. Chittaro, C. Costantini, G. Guida, C. Tasso, and E. Toppano, "Diagnosis based on cooperation of multiple knowledge sources," in *Proc. AVIGNON '89: 9th International Workshop on Expert Systems and Their Application*: Specialized Conference Second Generation Expert Systems, Avignon, France, 1989, pp. 19-33.

[19] L. Chittaro, G. Guida, C. Tasso, and E. Toppano, "Developing diagnostic applications using multiple models: The role of interpretative knowledge," in *Industrial Applications of Knowledge-Based Diagnosis*, G. Guida and A. Stefanini, eds. Amsterdam, NL: Elsevier, 1992.

[20] W. J. Clancey, "Heuristic Classification," *Art. Intell.*, vol. 27, pp. 289-350, 1985.

[21] J. Collins and K. Forbus, "Reasoning about fluids via molecular collections," in *Proc. 6th Nat. Conf. Art. Intell*, Seattle, WA, 1987, pp. 590-594.

[22] R. Davis, "Diagnostic reasoning based on structure and behavior," *Art. Intell.*, vol. 24, pp. 347-410, 1984.

[23] R. Davis and W. Hamscher, "Model-based reasoning: Troubleshooting," A.I. Memo No. 1059, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Cambridge, MA, 1988.

[24] J. De Kleer, "How circuits work," *Art. Intell.*, vol. 24, pp. 205-280, 1984.

[25] J. De Kleer and J. S. Brown, "A qualitative physics based on confluences," *Art. Intell.*, vol. 24, pp. 7-83, 1984.

[26] J. De Kleer and B. C. Williams, "Diagnosing multiple faults," *Art. Intell.*, vol. 32, pp. 97-130, 1987.

[27] —, "Diagnosis with behavioral models," in *Proc. 11th Int. Joint Conf. Art. Intell*, Detroit, MI, 1989, pp. 1324-1330.

[28] K. Downing, "The qualitative criticism of circulatory models via bipartite teleological analysis," *Art. Intell. Med.*, vol. 2, no. 3, pp. 149-171, 1990.

[29] B. Falkenhainer and K. D. Forbus, "Setting up large scale qualitative models," in *Proc. 7th Nat. Conf. Art. Intell.*, St. Paul, MN, 1988, pp. 301-306.

[30] B. Falkenhainer and K. D. Forbus, "Compositional modeling: Finding the right model for the job," *Art. Intell.*, vol. 51, pp. 95-143, 1991.

[31] P. K. Fink and J. C. Lusth, "Expert systems and diagnostic expertise in the mechanical and electrical domains," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, no. 3, pp. 340-349, 1987.

[32] K. D. Forbus, "Qualitative process theory," *Artificial Intell.*, vol. 24, pp. 85-168, 1984.

[33] D. W. Franke, "Deriving and using descriptions of purpose," *IEEE Expert*, vol. 6, no. 2, pp. 41-47, 1991.

[34] M. Gallanti, L. Gilardoni, G. Guida, and A. Stefanini, "Exploiting physical and design knowledge in the diagnosis of complex industrial systems," in *Proc. 7th European Conf. Artificial Intell.*, Brighton, England, 1986, pp. 335-349.

[35] M. Gallanti, L. Gilardoni, G. Guida, A. Stefanini, and L. Tomada, "Integrating shallow and deep knowledge in the design of an on-line process monitoring system," in *Cognitive Engineering in Complex Dynamic Worlds*, E. Hollnagel, G. Mancini, and D. D. Woods, eds. London, England: Academic, 1988, pp. 191-214.

[36] M. R. Genesereth, "The use of design descriptions in automated diagnosis," *Artificial Intell.*, vol. 24, pp. 11-436, 1984.

[37] J. S. Gero, "Design prototypes: A knowledge representation schema for design," *AI Mag.*, vol. 11, no. 4, pp. 27-36, 1990.

[38] G. Guida, "Reasoning about physical systems: Shallow versus deep models," in *Expert Systems and Optimization in Process Control*, A. Mamdani and J. Efstathiou, Eds. Aldershot, England: Technical Press, 1986, pp. 135-159.

[39] P. Hayes, "Naive physics 1: Ontology for liquids," in *Formal Theories of Commonsense World*, J. Hobbs and B. Moore, Eds. New York: Ablex, 1985, p. 71-107.

[40] B. Hayes-Roth, "A blackboard architecture for control," *Art. Intell.*, vol. 26, pp. 251-321, 1985.

[41] F. Hayes-Roth, "The Industrialization of Knowledge Engineering," in *Artificial Intelligence Applications for Business*, W. Reitman, ed. Norwood, NJ: Ablex, 1984, pp. 159-177.

[42] W. Hamscher, "Model-based troubleshooting of digital systems," Ph.D. Thesis, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Cambridge, MA, 1989.

[43] J. Hunt and C. Price, "Diagnosis of electro-mechanical subsystems using multiple models," in *Proc. 2nd International Workshop on Principles of Diagnosis*, Milano, Italy, 1991, pp. 171-180.

[44] E. Hollnagel, "Information and reasoning in intelligent decision support systems," *Int. J. Man-Machine Studies.*, vol. 27, pp. 665-678, 1987.

[45] Y. Iwasaki and H. A. Simon, "Causality in device behavior," *Artificial Intell.*, vol. 29, pp. 3-32, 1986.

[46] G. Kalm and J. McDermott, "The MUD System," *IEEE Expert*, vol. 1, no. 1, pp. 23-32, 1986.

[47] A. Keuneke, "Device representation. The significance of functional knowledge," *IEEE Expert*, vol. 6, no. 2, pp. 22-25, 1991.

[48] A. Keuneke and D. Allemang, "Exploring the No-Function-In-Structure Principle," *J. Exper. Theo. Artificial Intell.*, vol. 1, pp. 79-89, 1989.

[49] B. Kuipers, "Qualitative simulation," *Artificial Intell.*, vol. 29, pp. 289-338, 1986.

[50] —, "Abstraction by time-scale in qualitative simulation," in *Proc. 6th Nat. Conf. Artificial Intell.*, Seattle, WA, 1987, p. 621-625.

[51] B. Kuipers and C. Chiu, "Taming intractable branching in qualitative simulation," in *Proc. 10th Int. J. Conf. Artificial Intell.*, Milano, Italy, 1987, pp. 1078-1085.

[52] J. Larkin, J. McDermott, D. P. Simon, and H. A. Simon, "Expert and novice performance in solving physics problems," *Science*, vol. 208, pp. 1335-1342, 1980.

[53] J. H. Larkin, F. Reif, J. Carbonell, and A. Gugliotta, "FERMI: A flexible expert reasoner with multi-domain inferencing," *Cognitive Science*, vol. 12, pp. 101-138, 1988.

[54] W. Lee and B. Kuipers, "Non intersection of trajectories in qualitative phase space: A global constraint for qualitative simulation," in *Proc. 7th Nat. Conf. Artificial Intell.*, Saint-Paul, MN, 1988, pp. 286-291.

[55] R. Leitch, M. Wiegand, and H. Quek, "Coping with complexity in physical system modeling," *AI Commun.*, vol. 3, no. 2, pp. 48-57, 1990.

[56] M. Lind, "Multilevel flow modeling of process plants for diagnosis and control," Report RISØ-M-2357, RISØ National Laboratory, Roskilde, Denmark, DK, 1982.

[57] Z. Liu and A. M. Farley, "Shifting ontological perspectives in reasoning about physical systems," in *Proc. 8th Nat. Conf. Artificial Intell.*, Boston, MA, 1990, pp. 395-400.

[58] Z. Liu and A. M. Farley, "Structural aggregation in common-sense reasoning," in *Proc. 9th Nat. Conf. Artificial Intell.*, Anaheim, CA, 1991, pp. 868-873.

[59] R. A. Miller, H. E. Pople, and J. D. Myers, "Internist-I: An experimental computer-based diagnostic consultant for general internal medicine," *New Eng. J. Med.*, vol. 307, pp. 468-476, 1982.

[60] R. Milne, "Strategies for diagnosis," *IEEE Trans. Syst. Man, Cybern.*, vol. SMC-17, no. 3, pp. 333-339, 1987.

[61] S. Murthy, "Qualitative reasoning at multiple resolutions," in *Proc. 7th Nat. Conf. Artificial Intell.* Saint-Paul, MN, 1988, pp. 296-300.

[62] P. P. Nayak, "Causal Approximations," in *Proc. 10th Nat. Conf. Artificial Intell.*, San Jose, CA 1992, pp. 703-709.

[63] H. M. Paynter, *Analysis and Design of Engineering Systems*. Cambridge, MA: MIT Press, 1961.

[64] O. Raiman, "Order of magnitude reasoning," in *Proc. 5th Nat. Conf. Artificial Intell.*, Philadelphia, PA. 1986, pp. 100-104.

[65] S. Rajamoney and S. Koo, "Qualitative reasoning with microscopic theories," in *Proc. 8th Nat. Conf. Artificial Intell.*, Boston, MA, 1990, pp. 401-406.

[66] S. Rajamoney and S. Koo, "Behavioral aggregation within complex situations: A case study involving dynamic equilibria," in *Proc. 9th Nat. Conf. Artificial Intell.*, Anaheim, CA, 1991, pp. 862-867.

[67] J. Rasmussen, *Information Processing and Human-Machine Interaction. An Approach to Cognitive Engineering*. Amsterdam: North-Holland, 1986.

[68] R. Reiter, "A theory of diagnosis from first principles," *Artificial Intell.*, vol. 32, pp. 57-95, 1987.

[69] A. Rizzo, S. Bagnara, and M. Visciola, "Human error detection processes," in *Cognitive Engineering in Complex Dynamic Worlds*, E. Hollnagel, G. Mancini, and D. D. Woods, eds. London, England: Academic, 1988, pp. 99-114.

[70] R. C. Rosenberg and D. C. Karnopp, *Introduction to Physical System Dynamics*. New York: McGraw-Hill, 1983.

[71] V. Sembugamoorthy and B. Chandrasekaran, "Functional representation of devices and compilation of diagnostic problem solving systems," in *Experience, Memory and Reasoning*, J. L. Kolodner and C. K. Riesbeck, Eds. Hillsdale, NJ: Lawrence Erlbaum, 1986, pp. 47-73.

[72] E. H. Shortliffe, *Computer-based medical consultations: MYCIN*. New York: Elsevier, 1976.

[73] L. Steels, "Diagnosis with a function-fault model," *App. Artificial Intell. J.*, vol. 3, no. 2-3, 1989.

[74] —, "Components of expertise," *AI Mag.*, vol. 11, no. 2, pp. 29-49, 1990.

[75] J. Sticklen and E. Bond, "Functional reasoning and functional modeling," *IEEE Expert*, vol. 6, no. 2, pp. 20-21, 1991.

[76] P. Struss, "Extensions to ATMS based diagnosis," in *Proc. 3rd Int. Conf. Appl. Artificial Intell. Eng.*, Palo Alto, CA, 1988, pp. 878-882.

[77] P. Struss and O. Dressler, "Physical Negation—Integrating fault models into the general diagnostic engine," in *Proc. 11th Int. J. Conf. Artificial Intell.*, Detroit, MI, 1989, pp. 1318-1323.

[78] K. Ulrich and W. Seering, "Function sharing in mechanical design,"

in *Proc. 7th Nat. Conf. Artificial Intell.*, Saint-Paul, MN, 1988, pp. 342-346.

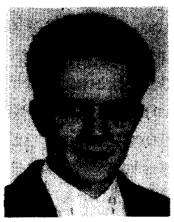
[79] Y. Umeda, H. Takeda, T. Tomiyama, and H. Yoshikawa, "Function, behavior, and structure," in *Applications of Artificial Intelligence in Engineering V*, vol. 1, J. S. Gero, Ed. New York: Springer-Verlag, 1990, pp. 177-193.

[80] D. Weld, "Approximation reformulations," in *Proc. 8th Nat. Conf. Artificial Intell.*, Boston, MA, 1990, pp. 407-412.

[81] D. Weld and J. De Kleer, eds., *Readings in Qualitative Reasoning about Physical Systems*, San Mateo, CA: Morgan Kaufmann, 1990.

[82] B. Williams, "Interaction-based invention: Designing novel devices from first principles," in *Expert Systems in Engineering. Principles and Applications*, G. Gottlob and W. Nejdl, Eds. Vienna, Austria: Springer-Verlag, 1990, pp. 118-134.

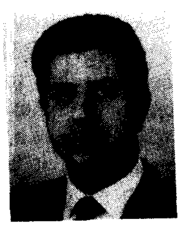
[83] D. D. Woods, "Commentary: Cognitive engineering in complex and dynamic worlds," *Int. J. Man-Machine Studies*, vol. 27, pp. 571-585, 1987.



Luca Chittaro received the Dottore degree in computer science from the University of Udine, Italy, in 1988.

Since 1988, he has been with the Department of Mathematics and Computer Science, University of Udine, where in 1991, he became a Researcher and Assistant Professor. His research interests are in the area of knowledge-based systems, particularly in diagnosis, physical systems modeling, temporal reasoning, qualitative reasoning, distributed problem solving, and human communication in knowledge acquisition.

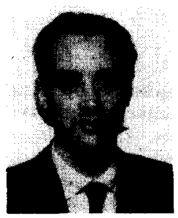
Dr. Chittaro is a member of AAAI and AIIA (Italian Association for Artificial Intelligence).



Giovanni Guida received the Dottore degree in electronic engineering (computer science) from the Politecnico di Milano, Italy, in 1975.

He is a Full Professor of Computer Science in the Department of Electronics for Automation, University of Brescia, Italy. His research interests are in the area of artificial intelligence, and include problem solving and search, distributed artificial intelligence, modeling of physical systems and model-based reasoning, applications of knowledge-based systems (industrial diagnosis, scheduling, design), knowledge-based system life cycle and design methodologies, and evaluation of knowledge-based systems. Also, he has carried out research activity in the fields of formal languages, programming languages and systems, natural language processing, man-machine interfaces, and user modeling. He is active as a Scientific and Technical Advisor in the area of Artificial Intelligence Technologies.

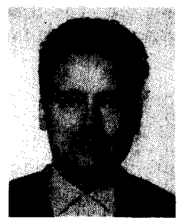
Dr. Guida is a member of AAAI and ACL.



Carlo Tasso received the Dottore degree in electronic engineering (computer science) from the University of Trieste, Italy, in 1980.

He has been at the University of Udine, Italy, since 1981, where he is currently Associate Professor of Computer Science. He has been with the Master in Computer Science Program of the University of Houston in cooperation with the University of Udine. His research interests are in the area of knowledge-based systems. More specifically, his research projects have included intelligent interfaces to information retrieval systems, natural language interfaces, automatic knowledge acquisition, user modeling, multiple modeling of physical systems, model-based diagnosis, intelligent tutoring systems, knowledge-based system life cycle and design methodologies. He is the author of several scientific papers and is active as a Scientific and Technical Advisor in the area of Knowledge-Based Systems technology.

Dr. Tasso is a member of AAAI, ACL and AIIA (Italian Association for Artificial Intelligence).



Elio Toppano received the Dottore degree in electronic engineering (computer science) from the University of Trieste, Italy, in 1981.

Since 1984 he has been Assistant Professor in the Department of Computer Science, University of Udine, Italy. His research interests are in the area of artificial intelligence, particularly in physical systems modeling, qualitative reasoning, approximation and abstraction methods in model-based systems, knowledge-based diagnosis, and design.

Dr. Toppano is a member of AAAI and AIIA (Italian Association for Artificial Intelligence).