Consistency-Based Diagnosis in Physiological Domains

Keith L. Downing *
Department of Computer and Information Science, Linköping University,
S-58183 Linköping, Sweden

Abstract

This research attempts to span the gap between the AI in medicine (AIM) and consistency-based diagnosis (CBD) communities by applying CBD to physiology. The highly-regulated nature of physiological systems challenges standard CBD algorithms, which are not tailored for complex dynamic systems. To combat this problem, we separate static from dynamic analysis, so that CBD is performed over the steady-state constraints at only a selected set of time slices. Regulatory models help link static inter-slice diagnoses into a complete dynamic account of the physiological progression. This provides a simpler approach to CBD in dynamic systems that (a) preserves information-reuse capabilities, (b) extends information-theoretic probing, and (c) adds a new capability to CBD: the detection of dynamic faults (i.e., those that do not necessarily persist throughout diagnosis).

1 Introduction

The consistency-based approach to diagnosis (CBD) provides a general formal technique for deep-model diagnosis [5, 16]. This research extends a very popular CBD approach, the GDE paradigm [5, 4], to physiology. Most CBD work has focused on discrete static domains, such as simple digital electronics. Additionally, CBD has been successfully extended to discrete dynamic systems [11, 9], but applications to continuous dynamic systems have confronted more obstacles [2]. We argue that these problems stem from the frailty in dynamic domains of *referential transparency* [1], a key prerequisite to information reuse in GDE.

We preserve referential transparency by separating static from dynamic analysis so that the CBD system deals only with steady-state analysis at selected time points. In this way, we perform model-based diagnose of continuous dynamic systems without performing dynamic simulations. Our system, IDUN, combines the GDE methodology with data interpretation to diagnose continuous dynamic systems that exhibit dynamic faults.

*Author's current affiliation: SINTEF, Automatic Control Division, N-7034 Trondheim, Norway (email: keith.downing@itk.unit.no)

2 Monotonicity in GDE

A key to GDE's success is the ability to evaluate many candidates (i.e., diagnostic hypotheses) quickly. Since many candidates are often quite similar, they support many of the same predictions. The caching and reuse of these predictions (via the ATMS [6]) is critical to an efficient search of the candidate space.

Although the ATMS supports nonmonotonic reasoning during the search of an implicit context tree, ATMS usage within GDE requires two key monotonic assumptions for making predictions within any particular node of that tree: if a particular candidate (Cand) and a set of observed findings (Obs) support a set of predictions, P, then one derives a superset of P by either (a) adding a new observation (M) to Obs, or (b) adding a new assumption about the behavioral mode of some component (C) to Cand. ATMS environments typically contain observation and mode assumptions, so the above monotonicities enable the ATMS to quickly determine an environment's implications by considering predictions made earlier by subset environments. [1]

Component mode assumptions usually include "working", "broken" and often more specific fault modes. Deep models that support the above two monotonic assumptions have the property that a component's outputs are strictly a function of its inputs and its mode. This property is a type of *referential transparency* [1], which promotes inference reuse in the obvious way: if a set of inputs and mode derive a set of outputs at time t, then given the same inputs and mode at a later time t', one can immediately assume the same outputs, without redoing the potentially expensive derivation.

Two key factors contribute to the preservation of referential transparency: (a) components are highly modularized so that other components cannot affect them through channels other than the recognized input pathways, and (b) components have few or no state variables.

Essentially, the modular state-less components of simple digital circuits have been the key to GDE's previous success. The state-less requirement indicates that dynamic systems will be more problematic for GDE. For example, given a particular input fluid flow (F) to a vessel (V), one cannot ascertain the output flow value (O)

[1] In ATMS terminology, an *environment* is a set of assumptions.

as a strict function of F, if V has a modeled capacitance and fluid volume. The values of these parameters must also be accounted for, typically as extra ATMS assumptions, in the justification of O. Unfortunately, larger antecedents decrease reusability, since more conjuncts must hold in order to reassert O at t'. Similarly, larger value spaces for antecedent parameters decrease reusability. Hence, it is no surprise that GDE has had the most success in digital domains.

3 Physiological GDE

The extension of GDE to physiology demands (a) careful attention to modularity in modeling primitives, and (b) a means to combat the decline in reusability associated with state variables.

Patil [14] envisions many problems with applying CBD to medicine. One of his main concerns is that current medical knowledge provides few causal pathways from anatomical structures (i.e., physical components) to physiological behaviors. Hence, organs or bones could not serve as the input-output modules indigenous to most GDE domains. However, modules need not correspond to physical objects but can instead represent structural or behavioral abstractions of a physical system. Compartmental models, for instance, have no necessary mapping to physical space, yet they abstract physiological systems to a level of interacting modules - a level at which model-based diagnosis has been performed [15, 12]. Our system, IDUN, also employs compartmental models [7]. However, we view *mechanisms* (see below) as the fundamental modules for GDE applications in physiology.

A second design decision further insures referential transparency: IDUN handles static and dynamic analysis separately. IDUN inputs observations from different time slices, but it assumes that the system can be approximated by steady-state equations at each slice. This helps preserve referential transparency by fixing most state variables during a time slice. For instance, in the simple-vessel example given earlier, a steady-state assumption insures that the output flow equals the input flow while the internal fluid volume and capacitance hold steady. In compartmental models, this enables us to assume that the total inputs of a material to a compartment equal the total outputs. As detailed below, IDUN performs GDE-style diagnosis only at the steady-state level, where referential transparency is more easily preserved.

The steady-state assumption is relatively common and accurate in physiological modeling due to the homeostatic nature of biological systems. Additionally, homeostatic systems, when perturbed, often evolve through a series of (relatively stable) states, each characterized by a unique set of faults. IDUN diagnoses such progressions.

4 Modeling in IDUN

4.1 Variables

Variables in IDUN are classified as either *dependent* or *fault*, where changes in the latter normally affect the former through *basic* mechanisms, whereas fault variables are affected only by *regulatory* mechanisms. Variable values are expressed as ranges over the reals, similar to [17]. Each variable has a normal point value, as well as normal, low and high ranges about that point. [2] This allows for the imprecision in both physiological models and measuring techniques, while avoiding the ambiguity of qualitative simulation in feedback-rich domains.

4.2 Mechanisms

We define a mechanism as "a group of system variables and a causal relationship between them, where the causal connection is as direct/primitive as the granularity of the system model permits." Each mechanism in IDUN is represented by one or more steady-state constraints (SSCs) such as conservation laws or ohm's law (as applied to fluids). Each constraint may contain one or more fault variables. A mechanism can be faulted in one of two ways: (1) one of its fault variables is in a high or low range, or (2) one or more of its steady-state constraints simply does not hold, regardless of fault modes.

Mechanisms are either *basic* or *regulatory*. Basic mechanisms are only represented by SSCs that are presumably always satisfied . Otherwise they are necessarily faulted by criterion 2 above. Table 1 presents the mechanisms used in the cardiovascular example. [3]

Regulators have temporal *delays* and *durations*, which respectively denote (a) the normal time lag between a regulator-triggering perturbation and the time at which its SSCs become satisfied, and (b) the normal amount of time which the SSCs remain satisfied. Some regulators, such as the baroreceptors, lose their sensitivity to persistent triggers. An unsatisfied regulatory SSC indicates either (a) a regulator fault, or (b) an inactive period (i.e., a pre-delay or post-duration point in time). [4]

Dynamic qualitative relationships among variables, *influences*, are associated with regulators. These capture the effect of the trigger variable's value upon a fault

[2] For instance, in the cardiovascular example discussed below, these ranges are: (-10% +10%), (-40% -10%), and (+10% +40%), respectively, denoting percentages above and below the normal point value.

[3] The cardiovascular variables in this model are: Arterial Pressure (PA), Blood Volume (BV), Cardiac Output (CO), Right Atrial Pressure (RAP), Systemic Pressure Drop (DP-SYS), Systemic Resistance (R-SYS), Urine Output (UO), Venous Compliance (VC), Venous Pressure (VP), Venous Pressure Drop (DP-VEN), Venous Resistance (R-VEN), and Venous Return (VR).

[4] In this paper, we disregard faults in regulators and therefore always assume case b. However, the IDUN code supports either case as well as fault modes in regulators.

Mechanism	Systemic Ohms	Active Volume	Venous Ohms	Flow Conservation
Constraints	DP-SYS = PA - PV DP-SYS = CO * R-SYS	PV = BV/VC	DP-VEN = PV - RAP DP-VEN = VR * R-VEN	CO = VR
Fault Vars	R-SYS	BV,VC	R-VEN	

Table 1: Basic Circulatory Mechanisms

variable's **derivative**. For instance, as part of autoregulation, the sign of the normalized value of CO equals the sign of the derivative of R-SYS (Notationally, CO $\rightarrow$ R-SYS in Table 2) [5]. Thus, high CO causes R-SYS to increase. IDUN estimates derivatives by comparing a variable's values across time slices.

5 A Cardiovascular Example

In the circulatory condition known as volume-loading hypertension (VLH), increased water and salt intake coupled with renal insufficiency (e.g., the inactivity of many nephrons) cause a rise in blood volume (BV), which in turn raises cardiac output (CO) and arterial pressure (PA). The barorecceptors (BARO) immediately react to rising PA by decreasing systemic resistance (R-SYS). Thus, the earliest stage of VLH is characterized by two faults: high BV and low R-SYS, and symptoms such as high PA and high CO. Table 3 summarizes the stages of VLH as described in [10].

After a day or two, the long-term autoregulator mechanism (AUTO) reacts to an overabundance of cellular oxygen (caused by high CO) by raising R-SYS. So after 3 days, AUTO has gradually elevated R-SYS back to normal, while the baroreceptors have basically lost their ability to push R-SYS down. Thus, stage two of VLH has a single fault: high BV.

After a week or two, the kidneys (although damaged) manage to excrete enough salt and water (via urine) to reduce BV almost to normal, which in turn decreases CO. However, AUTO responds to the formerly-high CO by raising R-SYS well above normal. The typical effects of AUTO take time to reverse, since they involve changes to the size and topology of capillary beds. Hence, the characteristic fault of VLH's third stage is high R-SYS.

Variables	Day 1	Day 3	Day 14
BV	+10%	+18%	+5%
CO	+35%	+37%	+5%
R-SYS	-15%	0%	+33%
PA	+15%	+35%	+40%

Table 3: The Progression of Volume-Loading Hypertension

If an automated diagnostician first "observes" a patient two weeks after VLH onset, it might wrongly assume that the root cause of high PA is high R-SYS. Thus it might prescribe various vasodilators, when in fact a low-salt diet and/or kidney dialysis would be more appropriate. Hence, an understanding of the entire regulatory sequence is critical to (a) differentiating primary from secondary faults, and (b) recommending treatments for the root causes and not merely their consequences.

[5] "$\rightsquigarrow$" denotes an inverse relationship in Table 2

6 The IDUN System

IDUN performs the following sequence of activities in diagnosing time-varying physiological models:

1. The mechanisms and initial observations are input.
2. Within each *relevant* (see below) time slice, conflicts and rated candidates are generated.
3. Rated links are formed between candidates in adjacent time slices.
4. The k highest-ranking chains (of candidates and links) are generated by best-first search.
5. Chain ratings combine with candidate predictions to support information-theoretic testing, whereby IDUN determines **what** and *when* to measure.

IDUN moves freely about the time slices gathering data and refining candidates and chains. This time-hopping is analogous to a doctor's ability to check a patient's history or to further analyze samples or pictures taken of the patient at earlier times. [6]

The following sections address each of IDUN's activities in detail while tracing through IDUN's diagnosis of a volume-loading hypertension scenario.

6.1 Initialization

At the outset, IDUN receives general descriptions of the variables and mechanisms of the physiological model. For the VLH example, IDUN inputs all the information in Tables 1 and 2. In addition, we provide three observations of CO and PA in accordance with Table 3. [7] These observations tell IDUN to perform diagnosis in three different time slices: 1,3 and 14 days.

[6] This scenario also occurs during data-intensive types of diagnosis such as core-dump analysis, where the diagnostician has ready access to time-tagged memory values but certainly does not want to look at ALL of them.

[7] IDUN inputs the absolute measurements corresponding to the percentages about the normals in Table 3. Three values for RAP (all zero) are also inputted for this example.

Regulator	Baroreceptor	Long-Term Autoregulation	Renal Fluid Balance
Constraint	R-SYS = -.17*PA + 34	R-SYS = 3.6*CO	UO = 3*PA-240
Triggers	high(PA), low(PA)	high(CO), low(CO)	high(PA), low(PA)
Influences	PA $\rightsquigarrow$ R-SYS PA $\rightarrow$ VC	CO $\rightarrow$ R-SYS	PA $\rightsquigarrow$ BV
Delay	0-hours	48-hours	24-hours
Duration	72 hours	∞	∞

Table 2: Circulatory Regulators

6.2 Intra-slice Candidate Generation: Standard CBD with Fault Modes

The conflicts and candidates of IDUN each pertain to a particular time slice. Each slice has a set of potentially active mechanisms (PAMs) which consists of all the basic mechanisms plus any regulatory mechanism R such that:

$$delay(R) \leq time(slice) \leq delay(R) + duration(R)$$

Each candidate diagnosis, C, in IDUN is characterized by a set of active mechanisms, AM(C), and a set of fault-variable modes, FVM(C), which exactly accounts for each fault variable in AM(C). The *fault modes* of C, FM(C) $\subseteq$ FVM(C), are *high* and *low* modes. All other modes are *normal*/unfaulted.

Within a time slice, S, IDUN begins diagnosis with a single candidate, the *null candidate*, in which $FM(C_{null}) = \emptyset$ and $AM(C_{null}) = PAM(S)$. IDUN enters each of the slice's observed values into the constraint network and tests C_{null} via interval-based constraint propagation over $AM(C_{null})$. Contradictions (i.e., the assignment of two non-overlapping intervals to the same variable) lead to conflict generation, candidate refinement, and further testing as more fully described in [7].

An *explained* fault mode of C is one whose real-valued interval overlaps a value predicted by a regulator, in AM(C), for the same fault variable. IDUN rates candidates between 0 (worst) and 1 (best), with favorable ratings going to those with fewer unexplained fault modes and smaller set differences between PAM(S) and AM(C).

Given the aforementionned initial observations, IDUN generates the time-tagged candidates of Figure 1. [8]

6.3 Inter-Slice Candidate Linkage

The likelihood of a candidate depends not only on local factors, but also on the possibility of its causal linkage to candidates in neighboring time slices. Hence, chains of candidates, which extend from the first to the last slice, become the focus of diagnosis. The first step in building chains is the formation of links between candidates.

A link, L, connects an earlier candidate, C1, to a later one, C2. Rating(L) is simply the percentage of qualitative fault-mode changes between FM(C1) and FM(C2) that can be explained by the dynamic properties of a relevant regulator, i.e., one whose delay and duration would allow it to be active sometime during the open interval between time(C1) and time(C2):

$$delay(reg) < time(C2) \wedge$$
$$delay(reg) + duration(reg) > time(C1)$$

Furthermore, the explaining regulator's trigger must be satisfied in C1. Triggers can be satisfied either by observed values or by values predicted by C1.

For the VLH example, eight of the 12 links between candidates appear in Figure 1. [9]

6.4 Global Explanation Chains

After forming local links, IDUN begins at the earliest slice and performs a best-first search for the k highest-rated chains, where the rating of a chain is the product of the ratings of its candidates and links. In the VLH example, there are 9 chains; five appear in Figure 1.

Notice that the middle chain in Figure 1 (call it CH1) matches our previous description of VLH progression:

- Day 1: low(R-SYS), high(BV)
- Day 3: high(BV)
- Day 14: high(R-SYS)

CH1 receives the highest rating because (a) all of its links are completely explained by dynamic regulator models, and (b) every fault mode, except high(BV) on day 1, is explained by a static regulator model. Conversely, the chain to the immediate right of CH1 receives a lower rating since nothing explains the increase in BV from "normal" (on day 1) to "high" (on day 3).

[8] In Figure 1, candidates appear within rectangles, while links have rounded edges. An "L" ("H") in front of a candidate's fault variable denotes "low" ("high"), and a regulator name that follows a fault mode (in square brackets) signifies that the regulator explains the fault mode; hence the fault mode's presence will not decrease the candidate's rating. The regulators in curly brackets are those in AM(C), which, in all cases, includes the 4 basic mechanisms as well. The parenthesized number is the candidate's rating.

[9] Each link in Figure 1 includes (a) the significant qualitative changes (i.e., increase, I, or decrease, D) in fault modes between the candidates, (b) the regulators that explain those changes, and (c) a link rating.

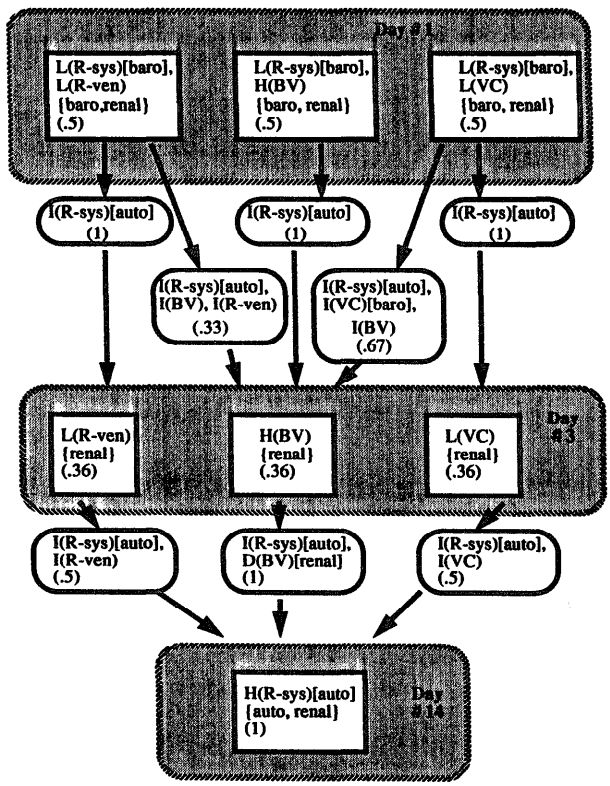

Figure 1: Candidates and Links for Volume-Loading Hypertension Scenario

6.5 Information-Theoretic Testing

The likelihood of a candidate is best estimated by the ratings of its chains, since these reflect the likelihood that the candidate participates in a global diagnosis/explanation. As fully described in [7], IDUN uses a variant of [5]'s entropy-based testing that (a) uses chain ratings as the basis for the probabilities of measurement outcomes, and (b) determines the best variable AND TIME SLICE for the next measurement.

Using information-theoretic testing in the VLH example, IDUN determines that PV at day 3 is best to measure. An inputted value of 20 mmHg then causes a reduction in chains from 9 to 6. IDUN then asks for PV's value at day 1. Entering 20 mmHG (i.e., +33%) reduces the set of chains to four. Unfortunately, each of these chains has identical predictions, so IDUN simply produces the chain with the highest rating, CH1, as its global diagnosis. CH1 corresponds to the standard medical explanation of VLH [10].

7 Evaluating IDUN

IDUN currently runs in Common Lisp on a SUN 4. The above VLH example requires only a few seconds of runtime. To test IDUN on more sizeable models, we built a compartmental modeling interface, wherein the user enters a description of compartments, substances, transfers and chemical reactions. IDUN then generates the necessary steady-state constraints and fault variables.

As detailed in [7], a 5-compartment physiological model of acid-base balance compiles into 39 mechanisms and 69 variables (16 of which are fault variables). IDUN then requires 5 minutes to diagnose a 4-time- slice acidosis-compensation scenario. IDUN's leading chains (after 7 rounds of testing) are similar to those in the physiology literature but are weakened by the presence of excess/"spurious" faults in many candidates.

Spurious faults stem from difficulties in accurately estimating (a) the "requested" (by information-theoretic testing) values of different substance concentrations during different stages of, for instance, acidosis compensation, and (b) the normal, low and high ranges of fault variables, such as the permeability coefficient of the kidneys to bicarbonate. IDUN's use of intervals allows for imprecise estimates, but large ranges hinder CBD conflict detection. Accurate parameter estimation is currently the major obstacle to scaling-up IDUN.

We have no data on information reuse in other CBD systems, so comparisons are impossible; but IDUN's numbers for the acidosis scenario appear quite impressive. Of the over 2000 candidates generated, 44% were analyzed. [10] 90% of the analyzed candidates were eliminated because they subsumed contradictory environments - leaving only 4% of the original candidates for actual testing (i.e., constraint propagation). During testing, an average of 94% of the variables were assigned ATMS-cached values (derived earlier by similar candidates), thus saving considerable recomputation.

8 Discussion

IDUN's key steps to extending GDE to physiology are (a) a dissection of physiological systems into modular mechanisms, (b) the representation of these mechanisms in terms of steady-state constraint equations coupled with dynamic models of regulators, and (c) the analysis of candidates across multiple time slices in order to account for the temporal progression of medical conditions.

[10] IDUN maintains a maximal size, M, for candidates, where size is the number of fault-mode plus inactive-mechanism assumptions. Candidates larger than M are often generated but not analyzed until all candidates of size $\leq$ M are ruled out.

Recent GDE-type systems [2, 9, 11] have employed dynamic component models to predict time-varying behaviors, which derive conflicts that static models alone would miss. These systems rely on complete dynamic simulations and sophisticated temporal reasoning to diagnose dynamic systems. We strongly believe that dynamic simulation should not be a prerequisite to the diagnosis of dynamic systems. IDUN embodies this tenet by separating static from dynamic analysis, thus reducing the dynamic aspects to simple qualitative reasoning. Furthermore, in the above three systems, candidates themselves are static: they presumably hold throughout diagnosis. IDUN extends GDE to handle dynamic candidates, an important new topic in CBD [8].

IDUN is perhaps most similar to DeCoste's DATMI [3], since both systems produce slice-dependent explanations (i.e., p-interps in DATMI, candidates in IDUN) and then compare them across slices to provide global explanations. However, [3] uses only qualitative models, requires an a-priori complete envisionment, and diagnoses faulty measurements rather than mechanisms.

Compared to other first-principle diagnostic approaches in medicine, IDUN is unique in its use of the GDE paradigm. As in [13], we account for the delays and persistences of physiological events during diagnosis. However, while [13] mixes causal and temporal reasoning via temporally enhanced causal rules, IDUN performs static and dynamic causal analysis separately in order to facilitate the information reuse indigenous to GDE.

Currently, IDUN contributes more to CBD research than to AIM. It exploits the highly-regulated nature of physiological domains to illustrate the use of a hybrid static/dynamic, quantitative/qualitative extension of GDE to diagnose dynamic faults in dynamic systems, but without performing detailed dynamic simulations. For the AIM community, we hope this research shows that CBD should not be matter-of-factly discounted from medical applications due to some inherent modeling differences between physiology and engineering. Rather, the problem lies in (a) connecting the physiological and medical levels, and (b) finding the right combination of qualitative and quantitative reasoning so that physiological CBD systems can avoid the combinatoric explosion of qualitative simulation without demanding a plethora of a-priori quantitative information about obscure physiological variables.

Acknowlegement

I am very grateful to Larry Widman, Adam Farquhar and Oskar Dressler for their advice.

References

[1] H. Abelson and G. Sussman. Structure and interpretation of computer programs. The MIT Press, Cambridge, Massachusetts, 1985.

[2] P. Dague, P. Devès, P. Luciani, and P. Taillibert. Analog Systems Diagnosis. *Proceedings ECAI*, 173–178, Stockholm, Sweden, 1990.

[3] D. DeCoste. Dynamic across-time measurement interpretation. *Proceedings AAAI*, 373–379, 1990.

[4] J. deKleer and B. Williams. Diagnosis with behavioral modes. *Proceedings IJCAI*, 1324–1330, 1989.

[5] J. deKleer and B. Williams. Diagnosing multiple faults. *Artificial Intelligence*, 31(1):97–130, 1987.

[6] J. deKleer. An Assumption-based TMS. *Artificial Intelligence*, 28, 127–161, 1986.

[7] K. Downing. Physiological applications of consistency-based diagnosis. Submitted to *Artificial Intelligence in Medicine*, special issue on intelligent monitoring and control of dynamic physiological systems, 1992.

[8] G. Friedrich and F. Lackinger. Diagnosing temporal misbehavior. *Proceedings IJCAI*, Sydney, Australia, 1991.

[9] T. Guckenbiehl, and G. Schäfer-Richter. SIDIA - Extending prediction-based diagnosis to dynamic models. *International Workshop on Principles of Diagnosis*, Stanford University, 1990.

[10] A. Guyton, C. Coleman, R. Manning, and J. Hall. Some problems and solutions for modeling overall cardiovascular regulation. *Mathematical Biosciences*, 72:141–155, 1984.

[11] W. Hamscher. Modeling Digital Circuits for Troubleshooting. Artificial Intelligence - Special Issue on Qualitative Reasoning, 1991.

[12] L. Ironi, M. Stefanelli, and G. Lanzola. Qualitative models in medical diagnosis. *Artificial Intelligence in Medicine* (2), 85–101, 1990.

[13] W. Long. Reasoning about state from causation and time in a medical domain. *Proceedings AAAI*, 251–254, 1983.

[14] R. Patil. Artificial Intelligence for diagnostic reasoning in medicine. In H. Shrobe, editor, *Explorations in Artificial Intelligence*, pages 347–379, Morgan Kaufmann Publishers, 1988.

[15] R. Patil, P. Szolovits, and W. Schwartz. Modeling knowledge of the patient in acid-base and electrolyte disorders. In P. Szolovits, editor, *Artificial Intelligence in Medicine*, pages 191–226, Westview Press, 1982.

[16] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1), 1987.

[17] L.E. Widman. Semi-quantitative "close enough" systems dynamics models: An alternative to qualitative simulation. In L.E. Widman, K.A. Loparo, and N.R. Nielsen, editors, *Artificial Intelligence, Simulation and Modeling*, pages 159–188, John Wiley & Sons, New York, 1989.