REAL-TIME MONITORING OF DENSE CONTINUOUS DATA

Andrej Semrl
DePaul University, Chicago, IL
R&D United Airlines, WHQKB, Elk Grove Village, IL

Abstract: We've implemented a monitoring system that assists a perfusionist during open-heart procedures. The monitoring system supports the perfusionist by performing intelligent monitoring, including state recognition and prediction, adaptive alarming system, and data visualization supported by audio feedback. The system performs in an on-line, real-time mode in the data-intensive environments of operating rooms. The system is an open-loop system, where all interventions and decisions are made by perfusionist.

For efficient implementation of such a system, adequate knowledge representation must be chosen. By extending the semantics of generalized discrimination trees with frame-like structures, we've been able to integrate medical primitives, such as symptoms, treatments, and pathophysiological states, into discrimination trees. We utilize intermediate-depth knowledge representations or qualitative disease histories to describe interesting states of the patient, such as disease states or important transitions between different phases of the procedure. We show that histories, which in essence are shallow knowledge extended with temporal information, provide a good trade-off between computationally more expensive deep knowledge and less accurate shallow knowledge, and provide for performance required by the environment.

The reasoning process is centered around the state recognition task with short-term prediction, without any attempt to make diagnosis or generate deep explanation. As a by-product of the state recognition task, adaptive alarming is made possible, where the context and the history of the system play the major role. The reasoning process is data-driven, where each new observation starts its own discrimination process. Numerical, qualitative, and temporal reasoning are combined into one process of inference, where parameters at any given point of time can be any system quantity or system states themselves. To further eliminate the ambiguity of the reasoning, we suggest the system of expected, active, and historical system states, that also provide for the continuity of the reasoning process.

1. Introduction

Since the introduction of microprocessor technology, the clinical tasks in operating rooms (ORs) and intensive care units (ICUs) have been supported with numerous patient data, combined with audio and visual feedback. In such data-intensive environments, physicians and medical personnel can be overwhelmed by the amount of information, and making a timely and accurate clinical decision becomes so much more difficult. The use of knowledge-based systems for monitoring in critical care and operating room environments presents additional requirements and challenges in design and implementation, such as on-line data acquisition, real-time data processing and reasoning, and the utilization of the audio and visual data presentation techniques. Given a huge volume of incoming data, reasoning efficiency and focus become the major issues in development of the system.

This work presents a framework for efficient on-line reasoning in data intensive environments. The framework is implemented in an intelligent monitoring system named EHCO, applied to the problem of monitoring of patients during open-heart surgery. Although implemented in a medical setting, the same approach can be used in other domains with similar requirements, for example, in-flight on-line monitoring of aircraft engines.

The main objective of EHCO system is to collect, analyze, summarize, and efficiently interpret the on-line clinical data in order to recognize or predict possible abnormalities in parameters that are controlled and set by a perfusionist (an operator of a heart-lung machine) during an open-heart procedure. Since the system is working in the open-loop mode, where all corrective actions are made by a perfusionist, abnormal behavior and patient states are displayed along with recommended interventions and explanations.

2. Cardiac Surgery and Cardiopulmonary Bypass

Cardiopulmonary bypass (CPB) is a straightforward and logical procedure governed by basic physiological principles and reasonable technical practices [6]. The basic objective of cardiopulmonary bypass is to provide adequate blood flow to the brain and other organs while a surgical procedure is performed on the stopped and open heart. The technique was developed and put into clinical use in the early 1950s, after advances in anesthesia and concurrent advances in the development of the heart-lung machine made it possible [6].

During bypass, the perfusionist manipulates the heart-lung machine and administrates drugs to achieve:

- adequate perfusion of organs and tissues by maintaining adequate perfusion rate and perfusion pressure;
- adequate gas diffusion and transfer. In particular, partial pressures of oxygen and carbon dioxide in blood must be maintained in desired range;
- acid-base balance. This balance is measured on the pH of the blood and reflects the effects of the cellular metabolism on blood;

- adequate blood volume in the patient (PT). This includes regulation of the blood volume in patient's circulatory system and a heart-lung machine, blood suction from the operating area, and administration of other fluids;
- desired body and heart temperature, as requested per protocol. This is achieved by cooling or rewarming the patient's blood, or by the infusion of special cooled solutions (cardioplegia) into the heart;
- other desired patient's states, as required by surgeon, anesthesiologist, or protocol.

The complexity of work performed during open-heart surgery requires from perfusionist to have:

- a complete picture of the state the PT is in (state snapshot);
- understanding of the past events that gave rise to this state (history);
- knowledge about possible interventions (protocol), and
- knowledge about expected changes caused by interventions (prediction).

An AI system that would support a perfusionist during CPB would have to analyze a PT state in the past and present, and would have to maintain a knowledge about possible interventions at any given point of time. In the next two sections we present knowledge representation and reasoning required for efficient on-line monitoring in data-intensive environments. We conclude with an example and discussion.

3. Knowledge Representation for Clinical Monitoring

Data Abstraction

Input to the monitoring system is a finite set of time-stamped measurements. Each quantity in ECHO is defined by at least one quantity space, presented by its qualitative value and a derivative. Quantity space is a totally ordered list of physiologically important intervals of continuous values of the quantity, and is defined in advance in the domain model. Data abstraction is important for an efficient, on-line monitoring, because it can encapsulate incomplete information, is less complex, and is computationally more efficient than reasoning with numerical values.

ECHO utilizes the granularity of the quantity space to reduce the ambiguity of the qualitative reasoning. Each quantity in the system is defined by multiple quantity spaces which are context dependent. For example, the body temperature of the patient before bypass can be sufficiently described by three qualitative values only: hypothermia, normal, and hyperthermia. However, when the patient is on bypass, we need a more detailed mapping, and we switch to a quantity space with six values: extreme, profound, deep, moderate, and mild hypothermia, and abnormal. The switch between different quantity spaces is driven by the reasoning process as it recognizes different stages of the procedure.

An additional refinement is qualitative sensitivity which is used to reduce the number of different qualitative values of the quantity over time. The qualitative value of the quantity is fixed whenever the change between the two subsequent readings is too small (clinical interventions or instrument interference), or too big (hardware failure, probe disconnection).

Input Data Segmentation and Histories

Temporal reasoning and state recognition in EHCO is based on the input data segmentation that occurs on the quantity and the system level. Quantity segmentation generates local segments that represent the longest contiguous time interval over which a numerical parameter maps onto the same qualitative value. A numerical parameter is constant over a segment if the same qualitative value can be used to describe each measurement in the input sequence [5]. Since each measurement is time-stamped and measurements are time-ordered, the quantity segments will have their start and end time points, and adjacent segments will represent the quantity's qualitative behavior over time. A quantity history is a contiguous, non-overlapping sequence of time-ordered, adjacent segments [9].

The local segments on the quantity level are propagated and combined on the system level to form global segments, intervals over which the overall qualitative state of the system remains unchanged. Since each local segment is time-stamped and segments are time-ordered, global segments will have their start and end time points, and adjacent global segments will represent qualitative behavior of a system over time. A system history is a contiguous, non-overlapping sequence of time ordered, adjacent global segments.

Segmentation generates local and global segments that are matched against predefined segments of system transitions and faulty behaviors. Successful matches are possible hypotheses about system behavior that, in order to be validated, have to be compatible with system history and expected system behavior.

Intermediate Depth Representations

Different approaches have been explored to resolve the problems of deep and shallow models, among them multi-level presentations based on variable granularity or variable presentation type [2], and transformations between knowledge representations [1]. In medicine, where the temporal aspect of a disease and treatment is very important, Coiera suggests the use of qualitative disease histories (QDHs) [2], [3], [4]. In short, if the natural disease history (NDH) describes the usual temporal course of a disease in an average patient, then the corresponding qualitative disease history is a time ordered sequence of qualitative state descriptions [4].

EHCO utilizes shallow (discrimination trees) and intermediate depth knowledge representations (qualitative disease histories). The monitoring process starts by utilizing shallow knowledge and pattern matching to focus the search in the problem space. Reasoning with surface knowledge and qualitative data is efficient, but can generate multiple hypotheses about system transitions and faulty behaviors. To reduce ambiguity, hypotheses are matched against system history, current system state, and expected system behavior and transitions. Knowledge about the system past, present, and future behavior and transitions, represented as qualitative histories, is then used to constrain and prune ambiguous hypotheses.

Multilevel Disease Models

ECHO utilizes the three-level description of disease, first proposed in the CASNET project [7], [8]. The CASNET model consists of a descriptive component, that provides a characterization of the disease processes, and is built in three levels [8]: observations, pathophysiological states, and disease states. The second or normative

component of the model consists of production rules that allow for the inference from observations and pathophysiological states to disease states, and from pathophysiological and disease states to treatment plans.

In EHCO, we combine discrimination trees with frame-like structures, to be able to represent medical knowledge in an efficient and simplified way. Each branch has a corresponding frame with information about pathology, symptom description, and short explanation. Each leaf node, representing a disease state, includes a frame with corresponding qualitative disease history (disease state characterized with temporal and cause-effect information), explanation, and context dependent information (usually as executable code for alarm triggering, display of data or information, or limit values change).

In EHCO, each observable parameter has at least one corresponding discrimination tree which is traversed every time the new measurement is obtained. To further improve the reasoning efficiency, each observation can have multiple decision trees. For example, mean arterial pressure can have one decision tree for hypotensive perfusion, one for hypertensive perfusion, and one for normal arterial pressure transitions (to detect mean arterial pressure before, during, and after complete bypass).

As the tree is traversed from the top to the bottom, the symptoms and pathophysiological states are cached to provide, along with the node and branch descriptions, rudimentary explanations on how the decision about a particular disease or transition state has been made. At the leaf level the disease state represents a possible hypothesis that has to comply with the past, current, and expected system behavior. For that reason, each disease state contains the corresponding qualitative disease history or possible segment of the system behavior that has to be compatible with the adjacent segments and historical system behavior. If those constraints are met, the hypothesis represents a valid disease or transition state, and generates additional information about the system. If constraints are not met, the hypothesis is not valid and is not retained for further consideration.

Figure 1 represents a simple example of the tree for hypotensive perfusion. The general discrimination tree, integrated with the three-level disease model, has a single entry at the top or the root node. One discrimination tree can analyze one parameter only, but the same parameter can have multiple discrimination trees. When the new observation for a given parameter is available, all disease and discrimination trees for that parameter are checked. Each node represents a test that can be performed on a quantity or a system level. Numerical, temporal, or qualitative data and information can be queried for any quantity known to the system. The node test, <test-specification quantity>, or <test-specification system>, is essentially the way of collecting symptoms for a possible disease state. Each branch associates the symptom value v_{ij} with the corresponding pathophysiological state p_{ij}, if available. The leaf level is a possible disease state with the corresponding treatment, determined by pathology. All entities in the tree are implemented as frame-like structures, and contain additional information for efficient reasoning and explanation. The leaf node also contains the qualitative disease history the particular disease state belongs to.

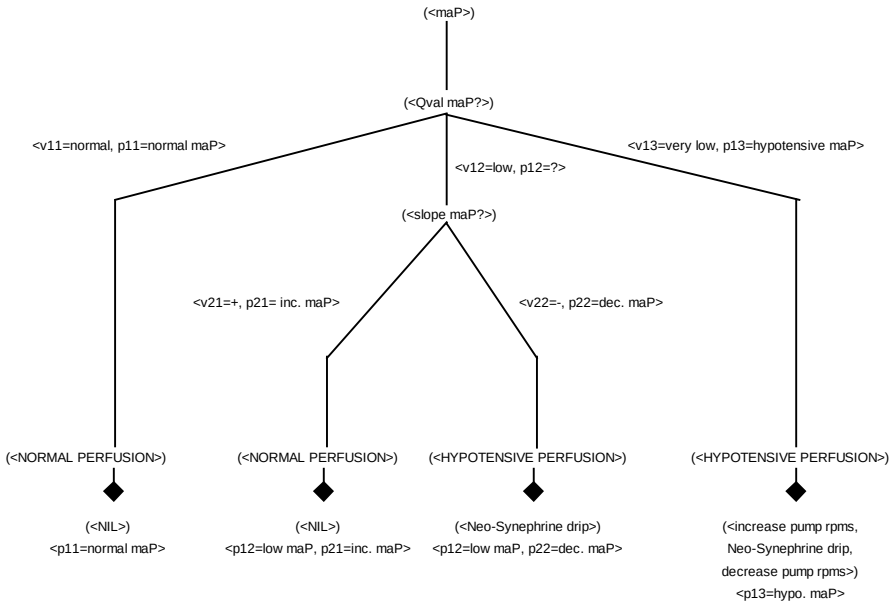

Figure 1: Discrimination Tree for Hypotensive Perfusion.

4. Reasoning in Clinical Monitoring

Qualitative and Temporal Reasoning

EHCO uses qualitative values on the quantity and the system level to perform data abstraction (local and global data segmentation), and to describe interesting segments of the system behavior over time (qualitative disease histories or QDHs). Throughout the qualitative reasoning, numerical data are used to reduce ambiguity and increase reasoning accuracy. For example, EHCO can use a sequence of raw numerical data to compute the slope for the given data segment, map the slope value to its qualitative value, and then use the qualitative value of the slope to determine the severity of a disease and a corresponding treatment.

Due to unique characteristics of local and global segmentation (segments are totally ordered by time), temporal reasoning also utilizes qualitative abstraction to generate histories on the quantity and the system level. EHCO is processing the time-stamped input parameters and events, and generates interval-based higher level temporal abstraction. If the start and the end time are associated with each interval, then the temporally ordered sequence of local or global segments represents quantity's or system's qualitative history, respectively.

In data-rich environments the efficiency of the temporal reasoning is critical, hence we suggest the use of temporal information to focus the reasoning, and to perform temporal reasoning related to quantity and system histories. In addition, the system should be able to match input behavior with qualitative disease histories or

QDHs. In short, EHCO can reason about interval durations, interval relations, and quantity or system histories and expectations.

Active, Expected, and Historical System States

EHCO uses observable patient behavior to recognize interesting states of the patient, such as disease states or important transitions among different phases of the procedure. Such observable behavior can be expressed as a time-ordered sequence of qualitative state descriptions, and can be compared against idealized natural histories, which describe the usual temporal course of a disease in an average patient [2]. The transformation of the observable patient behavior to the qualitative, temporally-ordered descriptions is performed within EHCO on the quantitative, qualitative, and temporal levels. Natural histories can be obtained and refined empirically, and then represented as qualitative disease histories (QDHs). Each QDH is characterized by preconditions that relate a particular disease state or system transition to the existing knowledge about the system, and a set of conditions that are evaluated against the current clinical findings. For example, a QDH for hypotensive perfusion requires the patient to be on a complete bypass (temporal precondition), and the observed pump RPMs to be stable and around the baseline (general precondition). Furthermore, some pathophysiological conditions must be true for a QDH to be a probable hypothesis in a given situation. For example, a hypotensive QDH can be considered as a probable hypothesis only if the qualitative value of the mean arterial pressure or maP is very low, or border line and still decreasing (clinical findings).

Recognized QDHs are considered as probable hypotheses, which have to be compatible with the existing knowledge of the system, defined with recognized, historical, and expected system states. In short, a probable hypothesis is validated only if compatible with other valid system segments. For example, observed heart warming phase can be a valid hypothesis only if heart cooling phase is one of the recognized and active system states.

Validated QDHs are then used to manage three sets of system states related to the system current conditions, system history, and system future. Active states are part of the QDH identification, like maP-hypotension or body-hypothermia, and all validated QDHs add their active states to the set of active states at the system level. As some states become active, other active and related states are moved to the historical states. For example, once the body hypothermia is recognized as an active state, the normal body temperature is moved from the set of active to the set of historical states. Also, if hypotensive maP becomes an active state, the normal maP is moved to the set of historical states. Information about what states are moved from the set of active to the set of historical states is part of each QDH definition. Finally, each active state can generate one or more expected states. For example, a normal body temperature is expected after hypothermia, and normal maP is expected after hypotension. Information about expected states can be found in the expectations slot of the QDH.

Each state manipulation is marked with a timestamp. In addition, each active state has an additional role of maintaining the current context of the system. The intended use of the context is to set up the system for the next monitoring cycle and to communicate with the user by performing alarm limit adjustments or alarm activation, setting of appropriate default values for input sequences with gaps, displaying explanation or required intervention, changing of the quantity space for a particular system quantity, or changing of the sampling interval for a particular system quantity.

In summary, both, disease and transition states are defined in advance as time-ordered sequences of events. The starting events for normal transitions are posted as expected at the beginning of the procedure, and are moved first to the active states and

then to historical states, followed by their successor events. Observed disease states start a sequence of active and expected states, that are all moved to the historical states when the disease is treated and normal or baseline values are observed.

Figure 2 is a schematic representation of how the states are managed during the monitoring cycle. Three different sets of system states are maintained during monitoring. How the states are initialized and moved around depends on whether the state in question is a disease or a phase of a procedure. At the beginning of the monitoring process, initial values for transition states are posted as expected. For example, normal body and heart temperature are expected to be observed first, so they are posted in the set of expected states. Also, all system states with baseline values can be posted as expected. Once the available observations generate a system state that has a match in the set of expected states, that system state is moved from the set of expected to the set of active states. Also, any predecessor to the activated state is moved from the set of active to the set of historical system states. For example, as body hypothermia is recognized, it is moved from expected to active states, while its predecessor, normal temperature, is moved to the historical states, and its successor, body warming or normal temperature, is posted in expected states. In addition, an active state can execute different actions to update system context. For example, during body hypothermia, the system can increase the sampling rate for heart temperature. A sequence of system states describing a disease can be defined in advance, but can not be posted as expected, because it may or may not manifest during the course of the procedure.

Once the available observations generate a system state describing a disease state, it is posted as active, and its successor system states are posted as expected. For example, a hypotensive perfusion can have two possible successors, a hypotensive perfusion with a positive slope of maP, and a normal perfusion (input sequence might potentially contain gaps, so a direct move from hypotensive to normal perfusion is possible). As the treatment is undertaken, that successor state, hypotensive perfusion with the positive slope of maP, is moved from expected to the active states. Its predecessor state, hypotensive perfusion, is moved to the historical states, and its successor, normal perfusion, is posted as expected. As the treatment is completed, the normal perfusion is stated as active, there are no successor states posted, and the predecessor state, the hypotensive perfusion with the positive slope of maP, is moved to the historical states. If expected states, posted as successors of disease states, are not removed from the set of expected states, then the treatment plan might be changed or adjusted. Expected states might be organized as an agenda, where the events expected to happen first are on the top. Timestamps and interval durations, related to the internal system time, can be used to determine when the expected state should be activated.

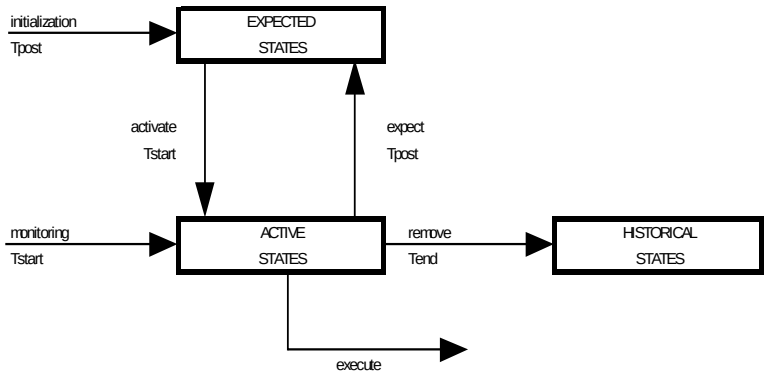

Figure 2: System States Management in EHCO system.

5. Example

Body and heart temperature during the 20 minutes of open heart surgery, with the following local segments (Data source: Semrl, March 1995, Cardiovascular Surgery, Evanston Hospital): normal body temperature, from 9:41am to 9:44am; body hypothermia, from 9:44am to 10:00am; heart cooling between 9:52am and 9:56am; and heart warming between 9:56am and 10:00am.

In the case of body and heart temperature, the only posted expected state during system initialization is normal body temperature, with the initialization timestamp value of –999999. At 9:41am, the observed qualitative value of the body temperature is normal, so that state is moved from expected to active, and two successors, body hypothermia and heart cooling are posted as expected. At 9:44am, body hypothermia is observed, so its predecessor, normal body temperature, is moved to historical states, and its successor, normal body temperature is posted as expected. At 9:52am, heart cooling is observed, so that stated is moved from expected to active, and heart worming is posted as expected. At 9:56am, heart warming is observed, so its predecessor, heart cooling, is moved to historical states, and its successor, heart cooling is moved to the expected states. By the end of the observed interval, body hypothermia and heart warming are active states, with corresponding successors and predecessors in the expected and historical states, respectively. The actual transcript of the system run is listed next and a schematic representation is given in Figure 3.

```
"--- SYSTEM EXPECT -------------------------------------------------------"
 state > body-temperature-normal  post > 94456
 state > heart-temperature-cooling  post > 95651
"--- SYSTEM ACTIVE -------------------------------------------------------"
 state > body-temperature-hypothermia  post > 94142  start > 94456
 state > heart-temperature-warming  post > 95336  start > 95651
"--- SYSTEM HISTORY ------------------------------------------------------"
 state > body-temperature-normal  post > -999999  start > 94142  end > 94456
 state > heart-temperature-cooling  post > 94142  start > 95336  end > 95651
```

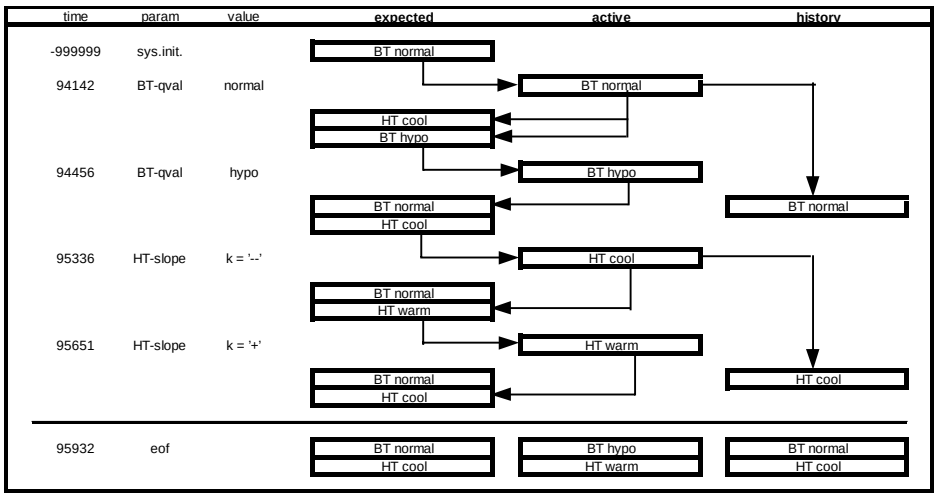

Figure 3: Maintenance of Heart and Body System States during complete bypass.

6. Conclusion

We propose and implement a paradigm for the construction of an intelligent, computer-based monitoring system used in data-rich environments. In this paradigm, normal and faulty system behaviors are defined in advance, as templates of typical system transitions, that have to be compatible with the adjacent segments of system's behavior. For efficiency purposes, medical models and primitives are integrated as frame-like structures into discrimination trees. The reasoning process is data-driven.

The major contribution of the EHCO system is a novel approach to the system state management used to focus the reasoning process and prune system states that are not compatible with the recognized present and past, or expected future system states. Also, by combining frames with decisions trees, we integrate a three-level disease model into the search process itself, resulting in an efficient system implementation.

Future research will focus on incompatible and opposing hypotheses. In particular, EHCO will not consider a hypothesis that is not compatible with any expected, active, or historical system states. Also, EHCO can not resolve hypotheses that are compatible with the current knowledge of the system, but contradict to each other.

References

[1] Bratko, I., Mozetic, I., Lavrac, N. eds. 1989. KARDIO: A Study In Deep And Qualitative Knowledge For Expert Systems. Reading, Mass.: MIT Press.

[2] Coiera, E.: Intermediate Depth Representations. Artificial Intelligence in Medicine, 4: 431-445, 1992.

[3] Coiera, E.: Monitoring Diseases with Empirical and Model Generated Histories. Artificial Intelligence in Medicine, 2: 135-147, 1990.

[4] Coiera, E.: Reasoning with Qualitative Disease Histories for Diagnostic Patient Monitoring. Ph.D. Thesis, School of Electrical Engineering and Computer Science, University of South Wales, 1989.

[5] Forbus, K. 1987. Interpreting Observations Of Physical Systems. *IEEE Trans. Systems, Man, And Cybernetics*, 17(3):113-117.

[6] Reed, C., Stafford, T.: Cardiopulmonary Bypass. Texas Medical Press, Inc., 1985.

[7] Weiss, K., Kulikowski, C. 1982. CASNET and EXPERT Consultation. *Artificial Intelligence in Medicine*, 21-55.

[8] Weiss, K., Kulikowski, C., Amarel, S., Safir, A. 1978. A Model-Based Method for Computer-Aided Medical Decision Making. *Artificial Intelligence*, 11: 145-172.

[9] Williams, B. 1986. Doing Time: Putting Qualitative Reasoning On Firmer Ground. In Proceedings Of AAAI-86, 105-112.