Relations between Visual and Textual Representations of Temporal Patterns for Medical Data Abstraction*

Norman Brümmer, Joachim Baumeister, Frank Puppe
Department of Computer Science,
University of Würzburg,
Am Hubland, 97074 Wuerzburg
{bruemmer,baumeister,puppe}@informatik.uni-wuerzburg.de

In this paper we present a visual representation of temporal patterns in abstractions of numerical and time-stamped data. We provide a curve-like acquisition tool which supports domain specialists to develop and refine temporal knowledge in an intuitive and effective manner. The resulting patterns can be used to detect artifacts as well as more complex phenomena, e.g., in order to derive intelligent alarms. Some experiences obtained from the acquisition and the use of medical surgery patterns is reported.

1 Introduction

The temporal development of numerical data and its interpretation, respectively, is of prime importance when monitoring patients in the medical domain, e.g., during surgeries or in the context of an ICU. Here, the automatic abstraction and interpretation of these continuously received parameter values can support the medical staff, e.g., anesthetists, with the tracking of the patient's status. Especially for long term

*This ist an extended version of the paper: N. Bruemmer, J. Baumeister, D. Riewenherm, F. Puppe and J. Broscheit. Visual Develpoment of Temporal Patterns for Medical Data Abstraction. In Proceedings of the Workshop on Intelligent Data Analysis in Biomedicine and Pharmacology (IDAMAP 2006), pages 37 - 38, 2006.

signal developments a computer-enabled monitoring is helpful, since for the medical staff it is often not possible to recall signal developments lasting longer than a certain time period (e.g. half an hour).

Furthermore, the interpretation of parameter values and their development is often difficult because they are superimposed by artifacts, e.g., an accidently dropped pulse sensor. In consequence, the validation of received parameter values preceding the actual interpretation is a crucial issue. The formalization of the required abstraction and validation knowledge is a difficult and costly task and commonly has to be accomplished by domain specialists.

In this paper we present an approach for a visual representation of temporal abstraction and validation knowledge allowing for an intuitive and precise formalization. The applied visual patterns were adopted from knowledge engineering interviews taken with the domain specialists and were refined in order to allow for a formal and precise interpretation of the entered temporal knowledge.

The context of our work is an intelligent monitoring and alarm system to be used during surgeries or in IC units, there supporting the work of anesthetists.

The paper focusses on the development of high–level abstractions, i.e., deriving meaningful alarms or artifacts, although the handling of basic abstractions derived from raw data streams is also an important task. Low-level abstractions, such as symbolization and trends will not be discussed in this paper, e.g., in [Miksch *et al.*, 1996] a detailed approach is presented handling different types of trends.

The derivation of intelligent alarms is two-folded in our system: In the first step we try to detect defined states of artifacts within the symbolized data stream and annotate the data with possibly found artifacts (thus enabling for a high-level data validation). In the second step we investigate the annotated data stream for defined alarm states, e.g. *insufficient anesthesia* and *hypovolemia*.

The presented approach was implemented in a case study investigating the development of important artifacts and alarms. The formalized patterns were evaluated on more than 300 cases in a subsequent step. These cases were collected from surgeries undertaken in the University of Würzburg hospitals in 2005/2006.

The rest of the paper is organized as follows: In Section 2 we present a visual representation of temporal patterns that allow for an intuitive, curve-like acquisition of the temporal behavior of abstract parameters. Section 3 defines the semantics of those patterns that can be expressed by the visual elements. We introduce a textual representation for temporal patterns and provide a translation from visual to textual patterns. In Section 4 we report the experiences that we have made in a medical monitoring project with respect to the knowledge acquisition and use of the developed patterns. The paper concludes in Section 5 with a discussion and an outlook for future work.

This paper extends the work of Bruemmer et al. [Bruemmer *et al.*, 2006] by defining the textual representation in more detail and by presenting the translation from the visual to the textual concept in a descriptive manner.

2 A Visual Representation for High-Level Abstractions

Before introducing a visual representation of abstraction knowledge we provide some general definitions and notations that are used for the rest of the paper.

General Definitions and Notations For the acquisition of temporal high-level abstractions we define parameters consisting of variables having a discrete domain. We denote the universe of *abstract temporal parameters* with $\Omega_{\mathcal{P}}$, e.g., $heartrate \in \Omega_{\mathcal{P}}$. Abstract parameters P are characterized by a discrete domain $dom(P)$, e.g. $dom(heartrate) = \{low, normal, high\}$. We denote the universe of all possible values of parameters $P \in \Omega_{\mathcal{P}}$ as $\Omega_{\mathcal{V}} := \cup_{P \in \Omega_{\mathcal{P}}} dom(P)$.

Abstract Temporal Curves The manual definition of complex patterns of the particular parameter changes over time is a difficult and error–prone task. For this reason we introduce an intuitive and visual representation for describing such patterns, i.e., *Abstract Temporal Curves (ATC)*, which can be interpreted as conditions for temporal rules for deriving high-level abstractions, e.g. artifacts or alarms. The representation offers a curve-like description of the temporal behavior of parameters, thus describing certain phenomena.

In the following, we introduce simple graphical elements that enable a description of basic events occurring in abstract parameter courses. Thereafter, we define temporal constraints that can be applied to events in order to describe the temporal behavior. A more complex temporal pattern is described by a set of events, attached constraints and a maximum duration restricting the whole pattern.

The modeling basis of an ATC contains layers for each involved parameter. Horizontal lines denote the corresponding abstract parameter values. There exist two basic elements that can be combined in different ways in order to describe the possible events.

Edges are horizontal lines describing a persistent value the specified parameter may take. Parallel edges of the same parameter define alternative and possible values for the parameter.

Nodes are markers placed on edges at arbitrary positions. They define changes of parameter values and thus basically declare temporal constraints.

Changes in the specified parameter behavior must be separated by nodes. Additionally, further nodes can be placed at any position, as it has been done with the first defined node in Figure 1. Here the decrease of the abstracted parameter value *AP* (arterial blood pressure) is specified starting with value *critical* then falling to either *high* or *normal* and finally decreasing to the value *low*.

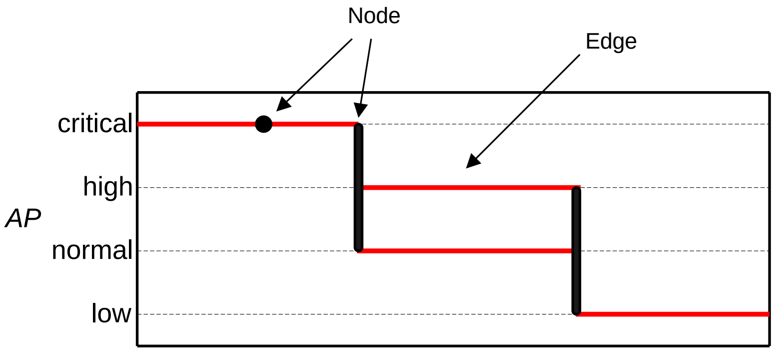

Figure 1: Edges and nodes in a parameter sequence. Nodes can be defined at arbitrary positions on edges. They denote events expressing value changes and value persistence.

We extend the notation by *temporal constraints* between the nodes. A temporal constraint consists of a pair of nodes connected by a period. It denotes that the enclosed events need to occur within the specified time range. There are three alternatives to connect nodes by temporal constraints; Figure 2 depicts the different types of constraints: A *sequence* defines in which time span a certain event flow, occurring on a single parameter, is required to occur. A second alternative for a definition of temporal constraints is the use of *intervals*. An interval connects two nodes of different parameter courses, which means that the corresponding events need to occur in the given time span. With the third alternative, i.e., the *point-interval*, we directly connect nodes in order to express that the corresponding events are required to occur simultaneously. The additional global constraint T_{max} depicted

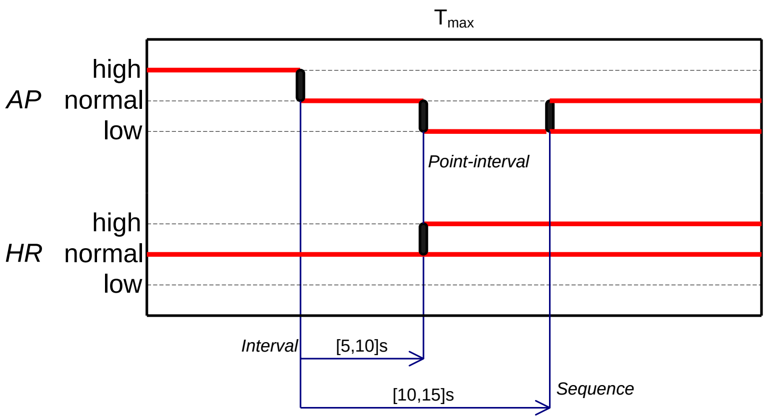

Figure 2: Temporal constraints defined between some nodes on the courses of the parameters *AP* (arterial blood pressure) and *HR* (heart rate).

on top of the pattern description implies that the duration of the entire scenario is restricted to the time-interval $(0, T_{max}]$.

The following examples demonstrate temporal patterns in visual representation which have been extracted from the ATC illustrated in Figure 2.

Examples

The first example (Figure 3) explains the sequence depicted in Figure 2. First, the value of arterial blood pressure (*AP*) has to change from *high* to *normal*. Within 10 to 15 seconds its value must change from *normal* to *low* and then from *low* to *low* or *normal*, whereas it is allowed to oscillate in the latter case.

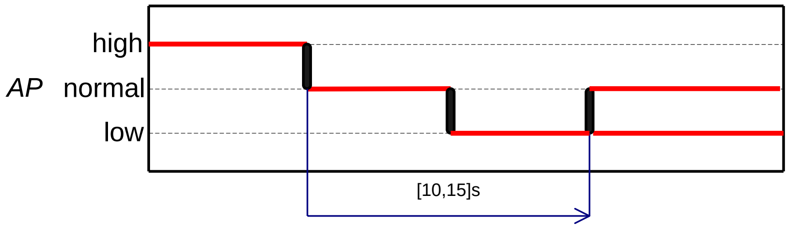

Figure 3: Simple example for a temporal event sequence.

The second example in Figure 4 explains the interval constraint illustrated in Figure 2 in more detail. After the value of arterial blood pressure (*AP*) has changed from *high* to *normal*, the value of heart rate (*HR*) must change from *normal* to *high* within 5 to 10 seconds. This implies that both *AP* and *HR* hold their values at *high* and *normal*, respectively, for at least 5 seconds.

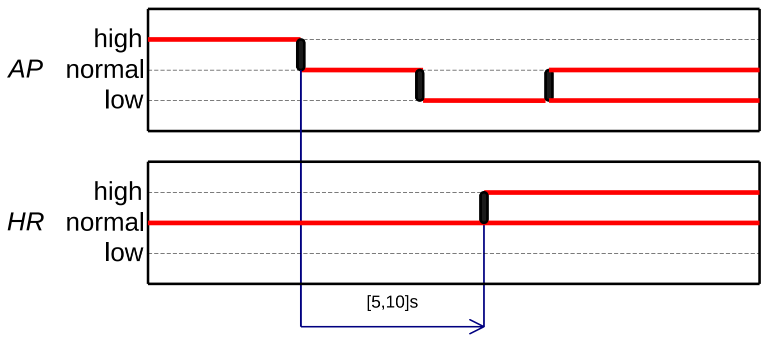

Figure 4: Simple example for a temporal event interval.

The presented visual notation allows for an intuitive description of temporal events. However, very complex events can be defined by combining these basic elements. In the next section, we introduce an explicit semantics of the described patterns: *Event predicates* and *temporal expressions* are defined for the particular patterns that allow for an integration to a rule-based formalism. In consequence, we can use efficient inference techniques, and we can combine the temporal predicates with non-temporal parts of the rule base. We further provide a translation method that compiles ATCs to event predicates and temporal expressions.

3 Application of ATCs to Temporal Rules

We integrate the presented abstract temporal curves into a rule-based formalism by defining a rule predicate semantics for the particular patterns. With the given semantics we are able to integrate the new representation into existing and established techniques. In this section, we first define temporal rule conditions consisting of event predicates and temporal expressions. After that, we describe how *Abstract Temporal Curves* are translated into temporal rule conditions.

3.1 Complex Temporal Conditions

The most common events occurring in abstract parameter streams can be classified into *value–changed* and *value–in–range* events: A *value–changed* event occurs, if a parameter value changes its value according to a defined and corresponding condition. A *value–in–range* event occurs, if the parameter values change only in a given value range. In the following, the event types and the corresponding temporal constraints are defined and explained by examples.

3.1.1 Event Predicates

The change of abstract parameter values should be detected by an appropriately defined condition.

Definition 3.1 (Value–changed predicate) Let $P \in \Omega_{\mathcal{P}}$ be an abstract parameter value with the values $\{x_1, ..., x_m\} \cup \{y_1, ..., y_n\} \subseteq \mathit{dom}(P)$.
The predicate $\mathit{vc}\big(P, \{x_1, ..., x_m\}, \{y_1, ..., y_n\}\big)$ is satisfied if P changes its value from $x \in \{x_1, ..., x_m\}$ to $y \in \{y_1, ..., y_n\}$. In this case a *value–changed event* occurs.

A value–changed predicate defines a connection of two non-identical sets of edges by a node in an ATC, as it is depicted in Figure 1. We write the corresponding event predicate as

$$\mathit{vc}\big(\mathit{AP}, \{\mathit{critical}\}, \{\mathit{high, normal}\}\big).$$

Further, a predicate for the detection of parameter values persisting in a certain range is necessary.

Definition 3.2 (Value–in–range predicate) Let $P \in \Omega_{\mathcal{P}}$ be an abstract parameter and $\{x_1, ..., x_n\} \subseteq \mathit{dom}(P)$ are values of P. The predicate $\mathit{vir}\big(P, \{x_1, ..., x_n\}\big)$ is satisfied, if the current value of P is contained in $\{x_1, ..., x_n\}$. Then, a *value–in–range event* occurs.

A value–in–range predicate expresses a collection of expected and possible values of a parameter, that is visually represented as a set of edges fixed by a node in an ATC. An example is given by the left-most node depicted in Figure 1. We denote the corresponding event predicate as

$$\textit{vir}\big(\textit{AP}, \{\textit{critical}\}\big).$$

Event predicates can be related to each other by the definition of temporal constraints. They specify in which order and periods the events have to occur to match a certain temporal pattern.

3.1.2 Use of Temporal Constraints

The following definitions introduce two variants of temporal constraints relating event predicates to each other. A set of events considering only one parameter can be constrained by a *temporal event sequence*. This constraint defines the order in which the events need to occur. In contrast, a *temporal event interval* defines a temporal relation between exactly two event predicates for distinct parameters.

Definition 3.3 (Temporal event sequence) Given a list of event predicates $E_1, ..., E_n \in \{\textit{vc}, \textit{vir}\}$ defined for the same abstract parameter $P \in \Omega_{\mathcal{P}}$. Then, a *temporal event sequence*

$$\textit{seq}\big((E_1, ..., E_n), [t_l, t_u]\big)$$

denotes that the predicates $E_1, ..., E_n$ have to be satisfied in the specified order. The period $[t_l, t_u]$, with $0 < t_l \leq t_u$, restricts the sequence to the specified time interval.

The restriction to the period $[t_l, t_u]$ is necessary to prevent the domain specialist from defining inconsistent sequences. A consistent example is given in Figure 3 and is written as

$$\begin{aligned}\textit{seq}\big(&\textit{vc}\big(\textit{AP}, \{\textit{high}\}, \{\textit{normal}\}\big),\\ &\textit{vc}\big(\textit{AP}, \{\textit{normal}\}, \{\textit{low}\}\big),\\ &\textit{vc}\big(\textit{AP}, \{\textit{low}\}, \{\textit{normal, low}\}\big), [10s, 15s]\big).\end{aligned}$$

It is satisfied, if in a time span between 10 and 15 seconds the arterial blood pressure (*AP*) changes from *high* to *normal*, then from *normal* to *low*, and finally to *low* or *normal*.

Definition 3.4 (Temporal event interval) Let $E_1, E_2 \in \{\textit{vc}, \textit{vir}\}$ be two event predicates defined for distinct parameters. A *temporal event interval*

$$\textit{int}\big(E_1, E_2, [t_l, t_u]\big)$$

specifies that both E_1 and E_2 have to be satisfied in the given period $[t_l, t_u]$. For a simultaneous occurrence of events, i.e., $t_l, t_u = 0$, the constraint is called *point interval*, and it is abbreviated by *int*(E_1, E_2).

For example, the temporal event interval

$$\mathit{int}\Big(\mathit{vc}\big(\mathit{AP}, \{\mathit{normal}\}, \{\mathit{low}\}\big),\\ \mathit{vc}\big(\mathit{HR}, \{\mathit{normal}\}, \{\mathit{high}\}\big), [5s, 10s]\Big)$$

is equivalent to the interval illustrated in Figure 4. It is satisfied, if the value of arterial blood pressure (*AP*) changes from *normal* to *low*, and if 5 to 10 seconds later the value of heart rate (*HR*) increases from *normal* to *high*.

As we have suggested earlier, temporal events and constraints can be used in rule conditions, thus defining simple temporal rules. More complex rules can be created using *Complex Temporal Conditions (CTC)* that group event predicates and temporal expressions. Furthermore, a maximum duration T_{max} is attached to the generated rule condition in order to restrict the entire temporal pattern.

A CTC is satisfied, if all temporal expressions are valid; it is not satisfied, if any event did not occur or if there is an expression that is not fulfilled. Further, the global maximum duration T_{max} must be satisfied by all expressions.

3.2 Translation from ATCs to CTCs

In this section we explain the semantics for abstract temporal curves using a translation from ATCs to event predicates and temporal constraints. For a better understanding we provide examples for all steps of the translation process.

3.2.1 Basic Structures

First we have a set of simple edges defined for one parameter $P \in \Omega_{\mathcal{P}}$ with $dom(P) = \{v_1, ..., v_n\}$. Let $\{v'_1, ..., v'_m\} \subseteq dom(P)$ be the values of P for which an edge is defined. This structure will be translated to the predicate

$$\mathit{vir}\big(P, \{v'_1, ..., v'_m\}\big).$$

The next basic structure contains value changes in the series of P. Let $\{v_{j_1}, ..., v_{j_m}\} \cup \{v_{i_1}, ..., v_{i_k}\} \subseteq dom(P)$. The modeled value change of P from $\{v_{j_1}, ..., v_{j_m}\}$ to $\{v_{i_1}, ..., v_{i_k}\}$ will be translated to the predicate

$$\mathit{vc}\big(P, \{v_{j_1}, ..., v_{j_m}\}, \{v_{i_1}, ..., v_{i_k}\}\big).$$

Up to now, the presented basic structures do not contain any information about the temporal behavior of the parameters. Further, relations between events occurring in the series of distinct parameters are still missing. In the following, the translation of such more complex patterns is outlined.

3.2.2 Temporal Constraints

More complex scenarios evolve from defining temporal constraints for events in parameter courses. Figure 5 shows the alternatives for declarations of temporal events. In the following the translation of the whole set of curves will be performed in five steps.

In (a) an edge of parameter P with value v_3 and a node on a value–change for parameter Q are constrained by the period $[t_1, t_2]$. This means, that the value of P must hold at least for a duration of t_1 and at most for t_2. Then, the value of Q must change from w_2 to w_1. In summary, this yields the expression

$$\textit{int}\big(\textit{vir}\big(P, \{v_3\}\big), \textit{vc}\big(Q, \{w_2\}, \{w_1\}\big), [t_1, t_2]\big).$$

In the scenario (b) and (c) a value–change in P from $\{v_3\}$ to $\{v_1, v_2\}$ must be followed by a value–change in Q from w_1 to w_2. Simultaneously, the latter event must occur with a value–change in P from $\{v_1, v_2\}$ to $\{v_1\}$. A translation to event predicates and temporal expressions will be performed as follows:

$$\begin{aligned}
&\textit{seq}\big(\textit{vc}\big(P, \{v_3\}, \{v_1, v_2\}\big), \textit{vir}\big(P, \{v_1, v_2\}\big),\\
&\quad \textit{vc}\big(P, \{v_1, v_2\}, \{v_1\}\big), [t_3, t_4]\big)\\
&\textit{int}\big(\textit{vc}\big(P, \{v_1, v_2\}, \{v_1\}\big), \textit{vc}\big(Q, \{w_1\}, \{w_2\}\big)\big)
\end{aligned}$$

Finally, in (d) and (e) for both parameters P and Q the particular event flow must be translated to maintain the order of the defined events. The corresponding sequences are:

$$\begin{aligned}
&\textit{seq}\big(\textit{vir}\big(P, \{v_3\}\big), \textit{vc}\big(P\{v_3\}, \{v_1, v_2\}\big),\\
&\quad \textit{vir}\big(P, \{v_1, v_2\}\big), \textit{vc}\big(P\{v_1, v_2\}, \{v_1\}\big),\\
&\quad \textit{vir}\big(P, \{v_1\}\big), [0, T_{max}]\big)\\
&\textit{seq}\big(\textit{vir}\big(Q, \{w_2\}\big), \textit{vc}\big(Q\{w_2\}, \{w_1\}\big),\\
&\quad \textit{vir}\big(Q, \{w_1\}\big), \textit{vc}\big(Q\{w_1\}, \{w_2\}\big),\\
&\quad \textit{vir}\big(Q, \{w_2\}\big), [0, T_{max}]\big).
\end{aligned}$$

The resulting temporal expressions and event predicates are combined in a *Complex Temporal Condition (CTC)* representing the entire ATC. Thus, temporal rules r of the form

$$r = c(r) \rightarrow a(r)$$

can be built, where $c(r)$ describes the rule condition and $a(r)$ is the rule action. For a temporal abstraction, $c(r)$ may be a Complex Temporal Condition and $a(r)$ may be an action assigning a value $v \in dom(P)$ to an abstract parameter $P \in \Omega_{\mathcal{P}}$, e.g., $a(r) = (P := a)$.

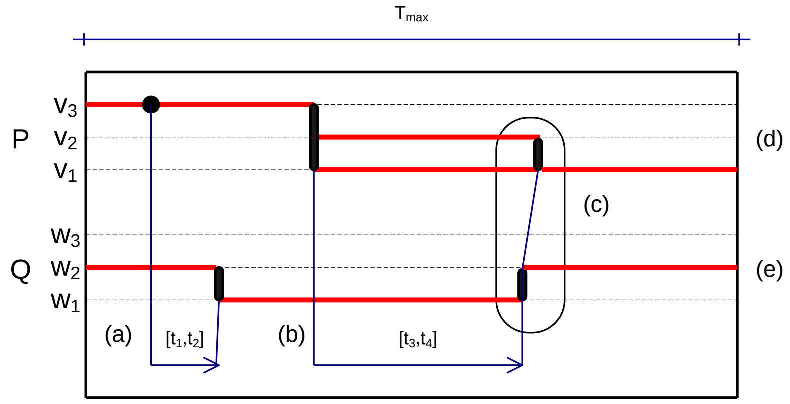

Figure 5: Parameter courses with temporal constraints. (a)-(e) mark intervals and sequences that will be translated to temporal expressions and predicates.

4 Case Study

The presented work was implemented in the context of a medical project aiming for an intelligent detection of artifacts and alarms during surgeries. The knowledge base was developed with a specialized editor of the knowledge modeling tool d3web.KnowME [Baumeister, 2004] suitable to create and edit the presented temporal patterns and rules. Two medical experts were involved with the formalization and refinement of the temporal knowledge including a variety of typical artifact and alarms. In the context of this case study we focus on the evaluation of five artifacts and one alarm in two variants. The knowledge was validated using real cases. Up to now, 319 surgeries were recorded in operation rooms of the Neurosurgery department of the Clinical Hospitals University Würzburg. The case durations differ from about half an hour to more than 16 hours. The effect of the artifact module on the alarm recognition was evaluated by enabling and disabling the corresponding knowledge for the artifact detection.

4.1 Knowledge Acquisition

The knowledge base contains temporal knowledge for the identification of 7 artifacts and 3 alarms variants. However, in the recorded cases not all artifacts and alarms occurred in a frequency that allow for a reasonable validation. Thus, we here consider only the most common five artifacts and one alarm in two variants occurring in the cases.

It turned out that the domain specialists very quickly got familiar with using the visual knowledge representation. This was not surprising since the representation was adopted from the observations made during knowledge acquisition interviews conducted with the domain specialists. In fact, similar curves were drawn on paper during the interview sessions to explain the particular events.

Consequently, almost no training phase was required and the initial model was discussed and implemented in about 5 minutes for each pattern. However, the main effort consisted in testing and refining the collected patterns. The most complex task was the appropriate definition and refinement of temporal constraints included in the patterns. These adaptations required frequent testing cycles. For simple patterns, e.g., the most artifacts, the refinement phase took about 10 minutes, whereas more complex patterns required more than 30 minutes for adaptation and testing. Figure 6

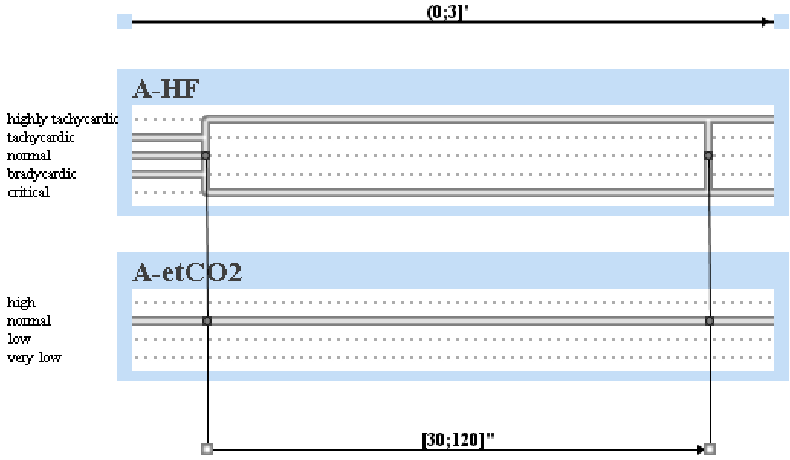

Figure 6: A screen shot taken from d3web.KnowME defining a ATC for an ECG artifact.

shows a screen shot of a temporal box defining an ATC for the detection of the ECG artifact. The pattern is satisfied if the value of the exhaled CO_2 is steady while the value of the measured heart frequency drifts to either very high or very low values, and if the parameters hold their values in a time span between 30 and 120 seconds. The whole pattern is restricted to a maximum duration of three minutes.

4.2 Modeled Artifacts and Alarms

In the following, we show the considered artifacts together with the amount of parameters involved in the ATC and the number of rules generated from the ATC.

Artifact	Parameters	Rules
ECG artifactial	6	3
Pulseoxymetry disturbed	5	2
ABP measurement attenuated	2	1
BIS unusable	2	1
MAP unusable	2	2

The artifacts *ECG artifactial* and *Pulseoxymetry disturbed* required temporal knowledge in the form of simple trends as well as abstract temporal curves. Both of them are expressed by two complex rules combining temporal and non-temporal knowledge. The remaining artifacts could be modeled with non–temporal but rather constraint-checking rules.

We have tested two types of the alarm *Insufficient Narcosis*, varying in the considered parameter set. Each alarm variant required one temporal rule whose antecedents consist of a combination of temporal and non–temporal conditions. The first alarm variant generally considers trends in the measured arterial blood pressure parameters combined with abstracted heart frequency values. The second alarm variant considers arterial blood pressure and bispectral index (BIS) measurements. The second variant was of limited use, since only about 10% of the available cases contained appropriate BIS recording.

4.3 Validating Inspection of the Patterns

In the following, we present the artifact and alarm detection results obtained from test runs with 319 neurosurgical cases. When executing pure artifact detection, it was not surprising that *ECG artifactial* and *Pulseoxymetry disturbed* occur in nearly all cases. In general, these are the most frequent artifacts in typical surgeries. *ABP measurement attenuated* and *MAP unusable* only occurred in 100 cases and *BIS unusable* in 36 surgeries, because the measurement of the arterial blood pressure and the BIS was only used in this limited number of surgeries. Otherwise, a non inversive blood pressure measurement has been used, for which the recorded data is usually not affected by artifacts. Table 1 presents detection results obtained from test runs we performed with both disabled and enabled artifact detection. In the middle column the number of case and total occurrences are listed. The right column denotes the true positive rate for the corresponding total matches. The values in braces denote results obtained with disabled artifact detection knowledge.

Alarm	**Cases/Total w/ (w/o) artifact det.**	**% TP**
Insufficient Narcosis	19/33 (20/38)	97 (87)
Insufficient Narcosis (BIS)	2/6 (2/13)	100 (46)

Table 1: Detection results from test runs with both disabled and enabled artifact detection.

It is not surprising that enabling the artifact detection module decreases the amount of alarm occurrences and significantly improves the true positive rate.

The most frequently filtered artifacts were *ABP measurement attenuated* and *BIS unusuable*. For the first artifact the increase of the arterial blood pressure has often been misinterpreted because of a preceding attenuation. For the latter both artifacts were essential.

Currently, we are working on the installation of more data recording stations in further clinical departments. With the variety of different clinical departments we expect a higher variance of occurring artifacts and alarms, respectively. Then, the relevance of a more diverse set of artifacts and alarms can be tested and refined.

5 Discussion and Future Work

In this paper, we have introduced a visual representation for the description of temporal scenarios in abstracted medical monitoring data. The visual representation is compiled into a rule–based formalism to be used in an efficient inference procedure and allowing for a combination of temporal and non–temporal rule predicates. The visual curve-like representation is close to the mental model of the domain specialists when discussing the development of medical parameters. It simplifies the knowledge acquisition and refinement tasks in an effective and intuitive way. We have implemented and evaluated the approach as a module for the knowledge–based diagnosis system *d3web*. The development process was evaluated with 2 medical domain specialists, whereas the detection of modeled artifacts and alarms was tested with about 300 cases recorded at neurosurgery operation rooms.

In the literature, representations for temporal high-level abstractions are mostly graph-based or, in a less complex case, threshold-based. In [Silvent *et al.*, 2003] temporal patterns can be defined providing thresholds for time and domain axis, defining gaps between parameter means and relative time points. These patterns allow for the discrimination of e.g. blood oxygen desaturation and sensor disconnects. In our approach, we can realize such patterns by defining ATCs with appropriate temporal constraints and edges expressing corresponding abstracted parameter means. The graph-based formalism provided in [Frydman *et al.*, 2001] enables the definition of scenarios using basic events obtained by discretization [Le Goc and Frydman, 2004]. The authors describe a recursive abstraction process starting at the extraction of *signal events* from single parameters. The highest abstraction level consists of *process phenomena* that combine the behavior of multiple parameters. The acquisition of temporal patterns is done by defining a graph-based model in ELP language [Frydman *et al.*, 2000], which will be transformed to a DEVS automaton. A textual representation of signatures are *chronicles* [Le Goc and Frydman, 2004] subsuming events, assertions and temporal constraints for the description of certain phenomena. The discrete event paradigm and signatures are related to event predicates and temporal expressions presented in this paper. With our approach we are also able to manage several abstraction levels including a recursive abstraction process. Indeed, our visual representation is not that general, but it is more comprehensive and more intuitive to be used by domain specialists. A further graph–based representation can be found in [Dojat *et al.*, 1998], where temporal constraint networks are used to define complex temporal scenarios. These networks are adopted in the scenario recognition module of *DéjaVu*, a system for monitoring patients in intensive care units. Our approach also provides the expressiveness of a graph-based representation, but as mentioned above, we claim that the curve-like representation is easier to understand and to adopt for medical domain specialists.

In the future we are planning to integrate a fuzzy definition of abstraction thresholds and temporal expressions, since an exact definition turned out to be one of the most difficult tasks. In consequence, the resulting fuzzy quantifications need to be propagated to higher abstraction levels, yielding a fuzzy derivation of diagnoses. In a first step we develop a non–binary pattern matching method that allows for the evaluation of partly matching patterns. Furthermore, we consider the semi-automation of the refinement process of the patterns by adapting discovery algorithms for this task. Related work for this task was undertaken e.g. in [Atzmueller *et al.*, 2005; Bellazzi *et al.*, 2005].

References

[Atzmueller *et al.*, 2005] M. Atzmueller, J. Baumeister, A. Hemsing, E.-J. Richter, and F. Puppe. Subgroup Mining for Interactive Knowledge Refinement. In *Proceedings of the 10th Conference on Artificial Intelligence in Medicine (AIME 05)*, pages 453–462. Springer, 2005.

[Baumeister, 2004] J. Baumeister. *Agile Development of Diagnostic Knowledge Systems*. AKA Verlag, DISKI 284, 2004.

[Bellazzi *et al.*, 2005] R. Bellazzi, C. Larizza, P. Magni, and R. Bellazzi. Temporal Data Mining for the Quality Assessmant of Hemodialysis Services. *Artificial Intelligence in Medicine*, 34(1):25–39, 2005.

[Bruemmer *et al.*, 2006] N. Bruemmer, J. Baumeister, D. Riewenherm, F. Puppe, and J. Broscheit. Visual Develpoment of Temporal Patterns for Medical Data Abstraction. In *Proceedings of the Workshop on Intelligent Data Analysis in Biomedicine and Pharmacology (IDAMAP 2006)*, pages 37–38, 2006.

[Dojat *et al.*, 1998] M. Dojat, N. Ramaux, and D. Fontaine. Scenario Recognition for Temporal Reasoning in Medical Domains. *Artificial Intelligence in Medicine*, 14(1-2):139–155, 1998.

[Frydman *et al.*, 2000] C. Frydman, N. Giambiasi, and L. Torres. Acquisition of Knowledge Based DEVS Models using Extended Event Graphs. In *AIS-IEEE 2000, Artificial Intelligence, Simulation, and Planning in High Autonomy Systems*, Tucson, USA, 2000.

[Frydman *et al.*, 2001] C. Frydman, M. Le Goc, L. Torres, and N. Giambiasi. The Diagnosis Approach Used in SACHEM. In *DX'2001, 12th International Workshop on Principles of Diagnosis*, pages 63–70, Sansicario, Italy, 2001.

[Le Goc and Frydman, 2004] M. Le Goc and C. Frydman. The Discrete Event Concept as a Paradigm for the 'Perception Based Diagnosis' of Sachem. *Journal of Intelligent & Robotic Systems*, pages 1–26, 2004. Kluwer Academic Publishers.

[Miksch *et al.*, 1996] S. Miksch, W. Horn, C. Popow, and F. Paky. Utilizing Temporal Data Abstraction for Data Validation and Therapy Planning for Artificially Ventilated Newborn Infants. *Artificial Intelligence in Medicine*, 8(6):543–576, 1996.

[Silvent *et al.*, 2003] A.S. Silvent, C. Garbay, P.Y. Carry, and M. Dojat. Data, Information and Knowledge for Medical Scenario Construction. In *Proceedings of the Workshop Intelligent Data Analysis in Medicine and Pharmacology*, Protaras, Cyprus, 2003.