

Intelligent adaptive monitoring for cardiac surveillance

Lucie Callens¹ and Guy Carrault² and Marie-Odile Cordier³ and Elisa Fromont⁴
and François Portet⁵ and René Quiniou⁶

Abstract. Monitoring patients in intensive care units is a critical task. Simple condition detection is generally insufficient to diagnose a patient and may generate many false alarms to the clinician operator. Deeper knowledge is needed to discriminate among alarms those that necessitate urgent therapeutic action. We propose an intelligent monitoring system that makes use of many artificial intelligence techniques: artificial neural networks for temporal abstraction, temporal reasoning, model based diagnosis, decision rule based system for adaptivity and machine learning for knowledge acquisition. To tackle the difficulty of taking context change into account, we introduce a pilot aiming at adapting the system behavior by reconfiguring or tuning the parameters of the system modules. A prototype has been implemented and is currently experimented and evaluated. Some results, showing the benefits of the approach, are given.

1 INTRODUCTION

Monitoring means to process incoming data (signals) recorded by sensors in order to recognize alarming conditions. Such devices may generate alarms in huge volume that can overwhelm an operator who has to validate the alarms and take therapeutic actions. What the operator really needs is a decision support system that could help him decide whether an alarm needs some action or can be skipped safely. In the 1980's the concept of knowledge based system emerged with the aim to associate deep knowledge to diagnosis. An intelligent monitoring system integrates such a knowledge-based system into a monitoring system.

The first step of intelligent monitoring is temporal abstraction. This means transforming numerical time series into symbolic event sequences. There is a huge literature in this domain (for surveys see [26, 12]), e.g. in the cardiac domain. The second step is devoted to the reasoning task. Among proposals, model-based diagnosis [19, 3] has the main advantage of using an explicit model that can be used to diagnose the series of events observed during monitoring as well as giving comprehensible explanations to the operator. As diseases have an important temporal dimension, we have proposed to represent them by sets of events linked by temporal constraints on their occurrences. Such sets of events are called chronicles [7]. They can model (a faulty model, in this case) the evolution of a disease during time or local typical temporal phenomena, e.g. typical wave sequences of an electrocardiogram (ECG) that represent cardiac beats

characteristic of some rhythm problem. Their recognition on an input stream is based on efficient processing of temporal constraint networks [4]. This makes chronicles good candidates for monitoring.

One of the main challenge of temporal abstraction in intelligent monitoring systems is to closely couple signal processing tasks and higher level tasks involved in diagnosis. One source of difficulty is that, generally, recorded data are highly dynamic and subject to changes. For example, the patient may move letting some sensor transmit very noisy data. Also, the patient state may evolve quickly due to the effect of some drug or disease evolution. We propose to introduce a central module called a pilot that analyzes continuously the monitoring context, i.e. the nature and quality of signals as well as the hypotheses devised by the diagnosis module. The aim of the pilot is to select the best signal processing algorithms and the right abstraction level for data abstraction. The pilot makes use of decision rules in order to bring high flexibility for taking into account new monitoring conditions or new monitoring domains.

The major bottleneck of knowledge based approaches is knowledge acquisition and maintenance. Machine learning has been advocated for this task. Since monitored diseases have a temporal relational dimension, first order models are good candidates for knowledge representation. This is why we have used Inductive Logic Programming [16] for learning chronicles. Devising decision rules for the pilot could also be tedious and time consuming [17]. We have proposed decision tree learning for inducing decision rules from the performance of algorithms in a representative set of contexts.

Stacey and McGregor's survey [26] lists many intelligent monitoring systems that share several objectives and features with ours. To cite a few, RÉSUMÉ [25], VIE-VENT [13] and its successors like ASGAARD [24] have introduced general knowledge representation paradigms for temporal abstraction with a deep integration of domain knowledge. VIE-VENT could process data streams whereas RÉSUMÉ was limited to databases. For efficiency reasons, instead of being general, our work focusses on cardiac knowledge in order to extract rich information from ECG or pressure signals online. From the temporal reasoning and diagnosis point of view, NÉO-GANESH [5] shares the concept of temporal patterns, called scenarios, with our approach. In addition, we have proposed a supervised method to learn such temporal patterns automatically from annotated temporal data. In [11], a self-adaptive software is introduced to modify the processing chain according to some predefined events such as sensor loss. But this approach does not deal with high-level goals as temporal reasoning which is crucial for monitoring evolving systems. In ASGAARD [23], the data abstraction module focusses on specific input sources according to a contextual plan which can be adapted when the processed inputs contradict expected values. However, data abstraction subtasks are scheduled in rigid sequences and diagnosis information is not used for adapting the current system.

¹ INRIA Rennes, France, email: lucie.callens@irisa.fr

² LTSI, University of Rennes 1, France, email: guy.carrault@univ-rennes1.fr

³ IRISA, University of Rennes 1, France, email: marie-odile.cordier@irisa.fr

⁴ Katholieke Universiteit Leuven, Belgium, email: Elisa.Fromont@cs.kuleuven.be

⁵ Dept. of Computing Science, University of Aberdeen, Scotland, email: fportet@abdn.ac.uk

⁶ INRIA Rennes, France, email: rene.quiniou@irisa.fr

This article summarizes the work done during several years as an active collaboration with experts in biological signal processing and the department of cardiology of the local hospital and which led to the implementation of an experimental platform called Calicot. Section 2 describes the medical applicative context. In Section 3, an overall overview of the system architecture is given. In Sections 4 and 5, we present the temporal abstraction and diagnosis methods. In Section 6, a solution to adaptation to context change is detailed. Learning chronicles and decision rules are presented in Section 7. We provide some evaluation results in Section 8 before concluding.

2 APPLICATION DOMAIN: MONITORING CARDIAC PATIENTS

Several areas in medicine require monitoring and management, e.g. temporary paralysis of the respiratory center in the brain, renal damage, surgical anesthesia or myocardial infarction. In this latter case, our concern, intensive care units were initiated in the 60s, for monitoring the vital functions of a patient after a serious cardiac attack. The main goal was to prevent, detect and control lethal arrhythmias by therapeutic actions. ECG signals are displayed to an operator and analyzed in real time: the trends of the main parameters, such as the cardiac frequency, are computed and alarms are generated. All this information assists the operator who is in charge of analyzing the situation, validating the alarms and deciding what action to perform.

The main problem is to abstract the signal observation to a pathological state through several steps [15]: signal processing, alarm handling, therapy advising. To solve this it, intelligent supervision systems appeared in the 90's. Their aim was to integrate several sources of observation (numerical or symbolic) and several types of medical knowledge (surface and deep knowledge). Surface knowledge relies on the use of the so-called experiential knowledge while deep knowledge corresponds to a complete theory about a particular subject, from which all valid statements can be derived.

Recent developments have led to knowledge-based temporal abstraction (KBTA) where machine learning techniques are used to extract the most discriminating patterns which can be identified in normal and several pathological states. There is a wide literature about temporal abstraction in medical domains [12, 26]. The Calicot system [1], dedicated to cardiac arrhythmias detection, and the one proposed by Guimaraes [9] which is focused to sleep-related respiratory disorders are two examples. Collaborative knowledge discovery approaches exploiting the properties of multi-agent systems (open character, autonomy of its components) have been proposed recently [10] for the exploration of mechanical ventilation asynchronies.

3 ARCHITECTURE

Calicot has two execution modes: offline and online. The online mode, depicted in Fig. 1, is devoted to monitoring and adopts a pattern-matching approach: multivariate signals are first abstracted in series of symbolic timestamped events and then a matcher attempts to recognize, on the fly, instances of chronicles in the symbolic series. A chronicle associated to some cardiac disease is a temporal signature of this disease. The offline mode is dedicated to learning and updating the decision rule base and the chronicle base.

Contextual information is of great importance for monitoring. On the one hand, by taking the signal quality (noise) into account one can decide more accurately which is the most relevant signal processing algorithm to use in the current situation. On the other hand, by taking the patient state into account one can decide more accurately which

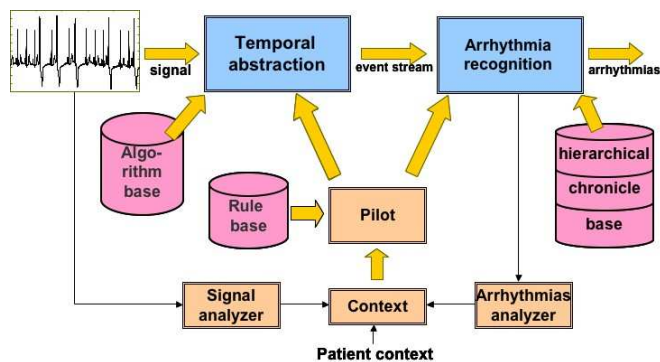


Figure 1: The architecture of the online part of the adaptive monitoring system Calicot.

are the relevant chronicles. Moreover, these two decisions are not independent: it can be the case that input signals are so noisy that it is useless to try to detect P-waves, for example. Consequently, chronicles that contain P-wave events can't be recognized and should be removed from the candidate set. Also, in the context of a particular disease, some types of event could be absent from the set of candidate chronicles and so it is useless to execute costly signal processing algorithms for detecting their related waves. Thus, low (signal processing) and high (chronicle recognition) level computations should be tightly coupled. This is why we opt for an adaptive architecture. On the one hand, a signal processing library containing many signal processing algorithms was built. Their performance have been assessed in many contexts to determine when and how to run them. On the other hand, a chronicle abstraction hierarchy was defined: more abstract chronicles contains less event types and/or less event attributes which makes them more relevant to more noisy contexts. More specific chronicles are relevant for situations where the detection of particular events could improve the diagnosis accuracy.

The decisions are taken by a central monitor (that we call a pilot) which analyzes continuously the signal and the patient context to determine the best signal processing task and algorithm to execute as well as the related chronicle abstraction level. A centralized control was adopted because it was simpler to specify via decision rules.

4 DATA ABSTRACTION

The temporal abstraction step transforms the numerical series into symbolic event sequences that are easier to process for high-level diagnosis. In coronary care units, the main problem comes from the presence of different kinds of noise (slow baseline drift, high frequency noise, impulsive noise) and from the great variability of patient dependent patterns and which can change over time. For example, multiform premature ventricular beats can combine with permanent or intermittent left or right bundle branch block. The temporal abstraction level achieves two main tasks: QRS complex (ventricular activity) and P wave (auricular activity) detection and QRS classification. Many methods have been proposed for detecting the ventricular activity, i.e. QRSs. Each one fails in specific situations and each reacts differently to the many different QRS waveforms. Hence we retain several algorithms. The proposed approach is not to merge the decisions of several algorithms but to select, on line, the most promising detector according to its performance in similar contexts. Actually, seven algorithms were selected [17].

Once a QRS has been detected it is labeled and a symbolic event is generated. The generation of its attribute values is based on the

fact a beat can be efficiently represented by a compactly supported wavelet base [21]. Each QRS is then represented by a global extrema at each decomposed level. The QRS classification consists in labeling the beats into two main classes, normal or abnormal. A probabilistic neural network (PNN) based on radial basis function has been used.

P wave detection is very hard because it has a weak amplitude and a variable morphology. We retain the QRS-T interval cancellation technique to overcome the limitations window techniques (which assumes that a P wave always occurs before some QRS) and mainly to facilitate the detection of the P wave even for arrhythmias with A-V dissociation. The proposed approach [22] mostly relies on: i) QRS-T interval detection and cancellation based on wavelet decomposition, ii) a statistical analysis of the residue for detecting P waves, not associated to a QRS, iii) an artificial neural network classifier to reject false detection which frequently occur in P wave detection.

5 DIAGNOSIS BY CHRONICLE RECOGNITION

In model-based diagnosis, a model of either normal or faulty behavior is used to detect and identify faults or diseases [19, 3]. If a normal behavioral model is used, the values reflecting the patient's state are fed into the model and the diagnoser generates an alarm if the model outputs are different from the values observed on the patient. With a faulty model, the diagnoser reasons abductively to generate disease hypotheses that could have produced the actual observations. Sometimes, such a model can be compiled into sets of discriminant patterns that can be efficiently searched for on the input stream. This method is very suited to online monitoring: the input stream is analyzed continuously and an alarm is emitted in case a set of events that can be related to a disease have been observed in some time window.

In many situations, such as in the case of dynamic systems, time is crucial [2]. The events related to the course of a disease must happen in a specific order and have to respect delay constraints. Moreover, sometimes it is easier to describe a disease by a set of successive or synchronous events respecting temporal constraints than to extract discriminant features from a vector of values recorded by several sensors. This is true in the cardiac domain: the symptom related to some disease, e.g. bigeminy, is described more naturally by the properties of several cardiac beats than by the particular features of a QRS. Such temporal patterns can be easily represented by chronicles. Fig. 2 shows an example of chronicle related to bigeminy and an example of match on two types of signal.

Since the input data streams can be huge, recognizing chronicles on the fly must be very efficient. We have used a system called CRS (Chronicle Recognition System) [6] which manages chronicle models, chronicle instances and temporal constraints between events from these instances. CRS tries to associate each incoming event with some uninstantiated event of a chronicle instance which satisfies the temporal constraints. It generates also new chronicle instances containing an event that can match the observed event. Many chronicle instances can be generated in such a way. One strength of CRS is its ability to prune instances, as early as possible, whenever one temporal constraint could not be satisfied by assessing whether an event occurrence time has elapsed. This makes it particularly adapted for applications where the detection of critical situations is essential.

6 ADAPTATION

The architecture of Calicot shares many features with self adaptive systems [20]. Since it is impossible to anticipate every situation and

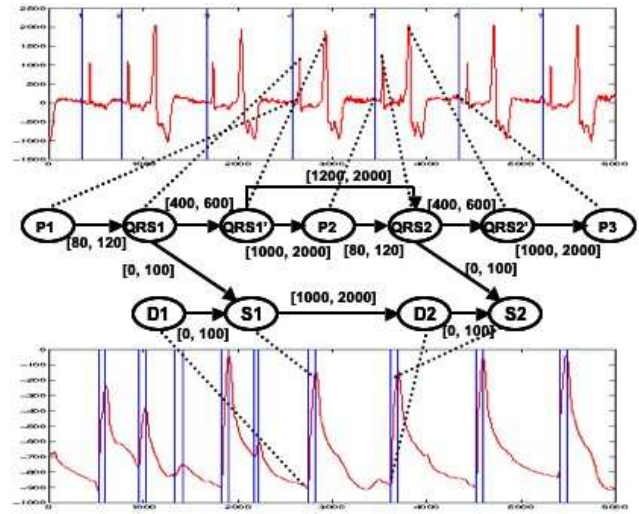


Figure 2: ECG (top) and pressure (bottom) signals of a bigeminy episode. The graph in the middle shows a bigeminy chronicle model: P stands for a P wave event occurrence, QRS for a QRS complex, D for a diastole, S for a systole. Quoted QRSs represent abnormal ones. Dotted lines indicate possible event matches that satisfy the temporal constraints.

transition, decisions concerning parameter settings and the choice of tasks and algorithms to execute next are postponed to runtime. The goal of such software, is to monitor and control themselves. In our case, the system can be viewed as a metamonitor: a monitoring system, the pilot, monitors a monitoring system. The goal is to obtain the best performance by reconfiguring the system operations, precisely choosing alternate algorithms and chronicles sets. To each processing module, algorithm or chronicle set, is associated a description which describes the way this module can be used. For example, to a signal processing algorithm is associated the task that it can achieve, the objects (types of events) that it can deliver and features describing the contexts in which it should ensure the best performance.

The pilot receives continuously two kinds of contextual information: signal context, mainly related to the signal quality, and recognition information, related to the diagnosis state. Noise type and level, event detection rate and distribution are used to estimate the signal quality (signal analyzer in Fig. 1). Chronicle recognition rate, types of chronicles recognized so far and their distribution, expected patient state, etc. are used to estimate the diagnosis quality (arrhythmias analyzer in Fig. 1). From this contextual information the pilot has to decide whether the context has changed notably and a reconfiguration is needed. In this case, the pilot must select the best algorithms and tune their parameters as well as choose the right chronicle abstraction level and the set of chronicles to be recognized.

To ensure a maximal flexibility and modularity, the pilot uses decision rules to decide when and how to perform reconfigurations. Here follows two examples of decision rules. The first one selects a detection algorithm, the second one selects the abstraction level that must be used for chronicles:

```

if paced  $\wedge$  noiseType=muscular  $\wedge$  noiseLevel  $\leq$  0 dB then algo df2
if PWaveDetection is active  $\wedge$  QRSClassification is active then abstractionLevel=4
  
```

7 MACHINE LEARNING

In our approach machine learning is used at two stages, for learning chronicles and for learning decision rules for the pilot module.

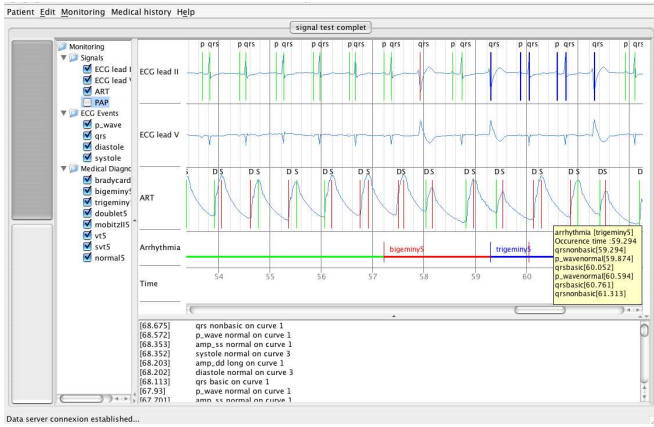


Figure 3: The graphical interface of Calicot showing the recognition of bigeminy and trigeminy episodes.

To establish accurate diagnoses efficiently the chronicles should be as discriminating as possible, i.e. they should clearly distinguish the diseases. However, the specification of discriminating chronicles is hard to do manually. Moreover, as first order (temporal) relations are concerned, we have chosen to use an ILP method. From examples describing symbolically signals related to disease episodes, ILP induces a set of first-order clauses that can discriminate the classes of the examples [16]. Fortunately, such a clause can be translated straightforwardly to a chronicle for CRS. By varying the language bias, i.e. the description language of examples and clauses, and using the related background knowledge, models at different abstraction levels can be learned. In addition, the model space is explored efficiently by searching only the most promising parts. We have also proposed a method for learning efficiently from multisource data [8].

The goal of decision rules is to aid the pilot module select the best algorithm and chronicle set according to the current signal and patient context. We have used two approaches to devise such rules. The first one made use of PCA (Principal Component Analysis) to determine the most informative context attributes [17] from performance data obtained by executing a set of signal processing algorithms in many situations with different kinds and level of noise and different diseases. Then, decision production rules conditioned by the selected attributes were built manually by experts. The second method used decision tree learning [18]. From a similar set of performance data, decision trees were learnt. The nodes of such a tree represent a partition of the values of some attribute and the leaves represent a class, here an algorithm. Every path from the root node to some leaf can be translated into a decision rule having as conditions the tests in the path nodes and as conclusion the algorithm in the leaf. It is worth noting that such learnt rules performed quite as well as expert rules.

8 EVALUATION

The Calicot prototype has been implemented in Java⁷. Its graphical interface displays monitored signals annotated with complex events related to recognized chronicles (see Fig. 3). Many experiments were conducted in order to assess its monitoring quality and evaluate its performance. Calicot has been evaluated on real clinical data recorded in ICU but, until now, has not been used in clinical routine. This section gives some results concerning the performance of the prototype on QRS detection, with and without piloting.

⁷ <http://www.irisa.fr/dream/Calicot/>

The implemented context analyser is based on a wavelet decomposition-recomposition in three subbands to obtain the triplet (ls, ms, hs) (low, medium, high subbands). This triplet together with the annotated context attributes r, n, SNR (rhythm, noise type, Signal-to-Noise Ratio) forms the context descriptor used by the pilot to decide which algorithm to use. Piloting rules have been extracted by decision tree learning from the performance results of 7 QRS detectors from the literature [18]. Three decision trees were induced: D1 (using attributes $r \times n \times SNR \times ls \times ms \times hs$), D2 (using $r \times ls \times ms \times hs$) and D3 (using the subbands only $ls \times ms \times hs$).

Ten ECGs, lasting around 30 minutes each (containing about 18.000 QRSs in total) and including ten various ventricular and supra-ventricular arrhythmic contexts, were extracted from the MIT-BIH Arrhythmia database [14]. Real clinical noise, from 5 dB to -15 dB, was introduced randomly in each ECG with probabilities $P(no_noise) = P(bw) = P(ma) = P(em) = 1/4$ and $P(5dB) = 1/2, P(-5dB) = 1/3, P(-15dB) = 1/6$ to reproduce difficult clinical ECG situations and to assess the system performance in specific contexts as well as when the context changes. The performance was evaluated from the standard TP (True Positive – correct result), FN (False Negative – missed result) and FP (False Positive – false result): the sensitivity $Se = \frac{TP}{TP+FN}$, the positive predictivity $PP = \frac{TP}{TP+FP}$ and the F-measure $FM(\beta) = \frac{(1+\beta^2)*PP*Se}{\beta^2*PP+Se}$, where $\beta = 1$ (same weight for Se and PP), were computed.

To estimate the upper bound performance reachable by Calicot with the pilot, the best detector performance (i.e. achieved by the detector with maximal FM) for each chunk of ECG was also retained. These results were used to define a gold standard (*bestChoice*).

Table 1 synthesizes the results of arrhythmia recognition without the pilot and every QRS detector algorithms, *bestChoice* (used as gold standard) and the different piloting rule sets. According to FM *pilot* D2, outperforms all the other methods. The best non piloted monitoring performance (FM=88.35%) is obtained when using algorithm *kadambe* for temporal abstraction, followed by *af2* (FM=86.67%), and *benitez* (FM=85.25%). These three algorithms outperform the others with an FM greater by 1.73%. The best piloted monitoring performance is obtained by the *pilot* D2 rule set followed by *pilot* D1 and *pilot* D3. *pilot* D1 performs better than non piloted *af2* but worst than non piloted *kadambe*. The upper bound that can be obtained is given by *bestChoice* as FM=91.71%. This shows that the piloting strategy could be notably improved with more accurate rules (at most by 3.36% of FM).

detector	sens(%)	P+(%)	FM(1) (%)	switches
<i>af2</i>	92.59	81.47	86.67	-
<i>benitez</i>	95.91	76.72	85.25	-
<i>df2</i>	80.64	86.61	83.52	-
<i>gritzali</i>	87.29	75.33	80.87	-
<i>kadambe*</i>	94.76	82.74	88.35	-
<i>mobd</i>	95.29	68.36	79.61	-
<i>pan</i>	79.66	85.70	82.57	-
<i>bestChoice</i>	92.39	91.04	91.71	1486
<i>pilot</i> D1	95.09	82.01	88.06	741
<i>pilot</i> D2	94.74	82.92	88.43	294
<i>pilot</i> D3	93.30	81.53	87.02	1478
<i>pilot</i> D1*	91.32	80.74	85.70	443
<i>pilot</i> D2*	92.33	81.38	86.51	343
<i>pilot</i> D3*	94.69	81.20	87.42	1642

Table 1: Recognition results (on 15525 cardiac beats)

kadambe associates wavelet analysis with heuristics for self-adapting to the signal, thus, it can be considered as being “piloted”. To assess the pilot more fairly, new piloting rules, `pilot D1*`, `pilot D2*` and `pilot D3*`, were learned excluding kadambe. The best performance of non piloted algorithms was obtained for `af2` followed by `benitez`. The piloted rule sets exhibited the best performance: `pilot D3*` with $FM = 87.42\%$ outperformed non piloted `af2` with $FM = 86.67\%$ improving FM by 0.75% which is considered a good score in the QRS detection field. The number of switches, `bestChoice`, shows that the best possible performance needs 1486 switches. `pilot D2` reached good scores with far less switches (294). Without using kadambe, `pilot D3*` has switched 1642 times showing that it uses the available algorithms much more.

Compared to kadambe, the advantage of using a pilot is that it uses explicit declarative rules which can be easily updated. This demonstrates the value of using a smart adaptation of QRS detection algorithms according to both signal, patient and diagnosis context.

9 CONCLUSION

We have presented an approach to intelligent monitoring with self-adaptive capabilities in the cardiac domain. Our proposition associates temporal abstraction, online diagnosis by chronicle recognition, self-adaptation to the monitoring context and automatic knowledge acquisition to learn chronicles and adaptation decision rules. A prototype named Calicot has been implemented.

Efficiency has been a constant concern during the conception and implementation of Calicot, as it was intended to run online. Thus, a temporal abstraction method taking advantage of the domain and data specificities has been proposed. Though they representing complex event, chronicles can be efficiently recognized on multiple data streams, one or two orders of magnitude less than real time in our case. To enhance the performance we have also proposed an architecture for self-adaptation, featuring a pilot which can reconfigure the processing chain or tune the module parameters when the monitoring context changes. Finally, symbolic machine learning is used, offline, to get discriminating patterns, on the one hand, and adaptation decision rules, on the other hand. Using a symbolic approach for knowledge and (temporal) reasoning makes it possible to provide understandable explanations to the user. This is very important in medicine for letting clinician operators trust such systems.

This research could not have been achieved without an active and fruitful collaboration with the medical staff. Working with clinicians in hospital is not always easy for computer scientists: experts are overbooked, getting data is sometimes difficult as protocols for recording data are very strict, especially they should not introduce any risk for the patient or any violation of data privacy. But confronting ideas and views from different research, knowledge and practice domains is particularly rewarding.

REFERENCES

- [1] G. Carrault, M.-O. Cordier, R. Quiniou, and F. Wang, ‘Temporal abstraction and inductive logic programming for arrhythmia recognition from ECG’, *Artificial Intelligence in Medicine*, **28**, 231–263, (2003).
- [2] M.-O. Cordier and C. Dousson, ‘Alarm driven monitoring based on chronicles’, in *Safeprocess’2000*, pp. 286–291, (2000).
- [3] J. de Kleer, A. Mackworth, and R. Reiter, ‘Characterizing diagnoses and systems’, *Artificial Intelligence*, **56**(2-3), 197–222, (1992).
- [4] R. Dechter, I. Meiri, and J. Pearl, ‘Temporal constraint networks’, *Artificial Intelligence*, **49**(1-3), 61–95, (1991).
- [5] M. Dojat, F. Pachet, Z. Guessoum, D. Touchard, A. Harf, and L. Brochard, ‘Néoganes: a working system for the automated control of assisted ventilation in ICUs’, *Artificial Intelligence in Medicine*, **11**(2), 97–117, (1997).
- [6] C. Dousson, ‘Alarm driven supervision for telecommunication networks. ii- on-line chronicle recognition’, *Annales des Télécommunications*, **51**(9-10), 501–508, (1996).
- [7] C. Dousson, P. Gaborit, and M. Ghallab, ‘Situation recognition: representation and algorithms’, in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 166–172, (1993).
- [8] E. Fromont, R. Quiniou, and M.-O. Cordier, ‘Learning rules from multisource data for cardiac monitoring’, in *AIME’05 (European Conference on Artificial Intelligence in Medicine)*, volume 3581 of *LNAI*, pp. 484–493, Aberdeen, Scotland, (2005). Springer.
- [9] G. Guimarães, J.-H. Peter, T. Penzel, and A. Ultsch, ‘A method for automated temporal knowledge acquisition applied to sleep-related breathing disorders’, *Artificial Intelligence in Medicine*, **23**(3), 211–237, (2001).
- [10] Thomas Guyet, Catherine Garbay, and Michel Dojat, ‘Knowledge construction from time series data using a collaborative exploration system’, *Journal of Biomedical Informatics*, **40**(6), 672–687, (2007).
- [11] G. Karsai, Á. Lédeczi, J. Sztipanovits, G. Péceli, G. Simon, and T. Kovácsházy, ‘An approach to self-adaptive software based on supervisory control’, in *Self-Adaptive Software, 2nd Int. Workshop, IWSAS*, volume 2614 of *LNCSE*, pp. 24–38. Springer, (2001).
- [12] N. Lavrac, B. Zupan, I. Kononenko, M. Kukar, and E. T. Keravnou, ‘Intelligent data analysis for medical diagnosis: Using machine learning and temporal abstraction’, *AI Communications*, **11**, 191–218, (1998).
- [13] 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**, 543–576, (1996).
- [14] G. B. Moody. ECG database applications guide. Harvard-MIT Division of Health Sciences and Technology Biomedical Engineering Center, Ninth Edition, 1997.
- [15] F. Mora, G. Passariello, G. Carrault, and J.-P. Le Pichon, ‘Intelligent patient monitoring and management systems : A review’, *IEEE Engineering in Medicine and Biology*, **12**(4), 23–33, (1993).
- [16] S. Muggleton and L. De Raedt, ‘Inductive Logic Programming: Theory and methods’, *The Journal of Logic Programming*, **19 & 20**, 629–680, (1994).
- [17] F. Portet, A.I. Hernández, and G. Carrault, ‘Evaluation of real-time QRS detection algorithms in variable contexts’, *Medical & Biological Engineering & Computing*, **43**(3), 381–387, (2005).
- [18] F. Portet, R. Quiniou, M.-O. Cordier, and G. Carrault, ‘Learning decision tree for selecting QRS detectors for cardiac monitoring’, in *AIME’07 (European Conference on Artificial Intelligence in Medicine)*, volume 4594 of *LNCSE*, pp. 170–174. Springer, (2007).
- [19] R. Reiter. ‘A theory of diagnosis from first principles’, *Artificial Intelligence*, **32**(1), 57–96, (1987).
- [20] P. Robertson and R. Laddaga, ‘Model based diagnosis and contexts in self adaptive software’, in *Self-star Properties in Complex Information Systems*, volume 3460 of *LNCSE*, pp. 112–127. Springer, (2005).
- [21] L. Senhadji, G. Carrault, and J.-J. Bellanger, ‘Comparing wavelet transforms for recognizing cardiac patterns’, *IEEE Eng. in Medicine and Biology Magazine*, **14**(2), 167–173, (1995).
- [22] L. Senhadji, F. Wang, A.I. Hernández, and G. Carrault, ‘Wavelet extrema representation for QRS-T cancellation and P wave detection’, in *Computers in Cardiology*, pp. 37–40, (2002).
- [23] A. Seyfang, S. Miksch, W. Horn, M. S. Urschitz, C. Popow, and C. F. Poets, ‘Using time-oriented data abstraction methods to optimize oxygen supply for neonates’, in *AIME’01 (European Conference on Artificial Intelligence in Medicine)*, volume 2101 of *LNCSE*, pp. 217–226. Springer, (2001).
- [24] Y. Shahar, S. Miksch, and P. Johnson, ‘The Asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines’, *Artificial Intelligence in Medicine*, **14**(1-2), 29–51, (1998).
- [25] Y. Shahar and M. A. Musen, ‘Knowledge-based temporal abstraction in clinical domains’, *Artificial Intelligence in Medicine*, **8**(3), 267–298, (1996).
- [26] M. Stacey and C. McGregor, ‘Temporal abstraction in intelligent clinical data analysis: A survey’, *Artificial Intelligence in Medicine*, **39**(1), 1–24, (2007).