Temporal Logic Based Monitoring of Assisted Ventilation in Intensive Care Patients

Citation for published version:
Bufo, S, Bartocci, E, Sanguinetti, G, Borelli, M, Lucangelo, U & Bortolussi, L 2014, Temporal Logic Based Monitoring of Assisted Ventilation in Intensive Care Patients. in T Margaria & B Steffen (eds), *Leveraging Applications of Formal Methods, Verification and Validation. Specialized Techniques and Applications: 6th International Symposium, ISoLA 2014, Imperial, Corfu, Greece, October 8-11, 2014, Proceedings, Part II.* Lecture Notes in Computer Science, vol. 8803, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 391-403. https://doi.org/10.1007/978-3-662-45231-8_30

Digital Object Identifier (DOI):
10.1007/978-3-662-45231-8_30

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Leveraging Applications of Formal Methods, Verification and Validation. Specialized Techniques and Applications

Temporal Logic based Monitoring of Assisted Ventilation in Intensive Care Patients

Sara Bufo[1], Ezio Bartocci[2], Guido Sanguinetti[3,4]
Massimo Borelli[1], Umberto Lucangelo[5], Luca Bortolussi[1,6,7]

[1] Department of Mathematics and Geosciences, University of Trieste, Italy
[2] Faculty of Informatics, Vienna University of Technology, Austria
[3] School of Informatics, University of Edinburgh, UK
[4] SynthSys, Centre for Synthetic and Systems Biology, University of Edinburgh, UK
[5] Department of Medicine, University of Trieste, Italy
[6] Computer Science Department, Saarland University, Saarbrücken, Germany
[7] CNR/ISTI, Pisa, Italy

Abstract. We introduce a novel approach to automatically detect ineffective breathing efforts in patients in intensive care subject to assisted ventilation. The method is based on synthesising from data temporal logic formulae which are able to discriminate between normal and ineffective breaths. The learning procedure consists in first constructing statistical models of normal and abnormal breath signals, and then in looking for an optimally discriminating formula. The space of formula structures, and the space of parameters of each formula, are searched with an evolutionary algorithm and with a Bayesian optimisation scheme, respectively. We present here our preliminary results and we discuss our future research directions.

1 Introduction

Temporal logic (TL) has proved to be a powerful and natural framework to describe complex temporal properties of systems. In fact, temporal logic formulae describe temporal patterns between events in a form which is close to our way of thinking, and as such they are intelligible and suitable to represent behavioral specifications. The availability of efficient verification and monitoring algorithms, that can check if a property is satisfied by a model or an observed run of a system, has further fostered this logical approach as a tool for design.

Monitoring, in particular, is applied mainly to engineered systems, as TL can naturally be used to encode the desired behavioural specifications the system should satisfy, which are provided by the designer. The availability of software and hardware for real time verification, however, makes this approach very attractive also in medicine, for instance to monitor the ECG signal or the flow/pressure curves of assisted ventilation of a patient in intensive care.

The main obstacle in this respect is that the behavioural specification we should observe are unknown: how can we describe by a TL formula the emergence of a dangerous clinical condition? The experience of physicians can help us identify situations in which the observed signals are prodrome to the insurgence of clinical complications, but a precise characterisation of these conditions in TL is by no means easy to obtain. Such a description would enable practitioners to use available monitoring tools, constructing devices that can support physicians in critical care choices.

The alternative to the unfeasible manual derivation of such specifications is to learn TL formulae from observed data, in the form of (manually) annotated signals. For instance, we can have as input from physicians a set of flow/pressure curves in which abnormal respiratory acts have been identified. Learning a TL specification of such abnormalities essentially means to construct a TL classifier of signals, that can separate normal breaths from critical ones. An appealing aspect of classifying by TL formulae would be the ease of interpretation of results, and the possibility of obtaining actionable physiological insights from the classifier. While statistical classifiers such as support vector machines often achieve impressive accuracy, this comes at the cost of developing opaque non-linear maps which offer little in the way of physiological insight.

Learning TL specifications from data is a problem that has recently received a certain attention in the literature [2,12,25,18,40,41,23], and which will be discussed in the related work section (Section 5). A frequently encountered problem with these approaches is the very large amount of data needed to learn inductively properties which are robust to noise in the observations [2,23]. The approach we consider here, which has been introduced in [4,5], tries to recast the learning problem within a solid statistical framework. Our strategy, instead, is to first infer a generative statistical model of the observed data, and then learn temporal specifications that have a high probability of being true in the so obtained model. This naturally keeps the effects of noise under control in a systematic way, but also solves the data shortage problem, as we can generate as much synthetic data as needed. In this work we consider a variant of this learning problem in which we aim at distinguishing two sets of signals, the good and the bad ones. This is obtained by constructing a statistical model for each class of signals, and then assigning to each TL formula a score which is high when the formula is true with high probability in a model and false with high probability in the other one.

From a medical perspective, we have started applying this framework to the identification of respiratory problems in patients in intensive care, which are breathing under assisted ventilation. In particular, our goal is to classify single respiratory acts into normal and abnormal. In principle, we want to look for different types of abnormality, although at this stage we focussed on ineffective triggering efforts, i.e. on the asynchrony between the flow of the ventilator and the attempt of the patient to start a new breath. Although a single occurrence of such event per se is not dangerous, and as such is largely ignored in practice, a long sequence of them can lead to severe clinical complications. This, and the fact that most ventilators in the market are not equipped with monitoring routines, motivates the investigation of this problem. More details, also from a biological and clinical perspective, will be given in Section 2.

In Section 3, instead, we discuss the basic steps of our methodological approach, namely the construction of statistical generative models of the signals we consider, which here take the form of a Stochastic Hybrid System (Section 3.1), the TL we use, which is the time-bounded fragment of Metric Temporal Logic (Section 3.2), the procedure to learn the structure of TL classifiers (based on an Evolutionary Algorithm, Section 3.4), and the method to learn the best formula parameters, based on Bayesian optimisation (Section 3.5). Some results are presented in Section 4, while conclusions will be drawn in Section 6.

2 Assisted Ventilation and Patient Ventilator Asynchronies

Pulmonary ventilation is the process of air flowing into and out of the lungs and occurs because the pressure of the atmosphere and of the gases inside the lungs differ.

During inspiration, the diaphragm and the external intercostal muscles contract, leading to an increase in volume of the thoracic cavity. As a result, the pressure within the lungs decreases and falls below atmospheric pressure and air flows into the lungs. On the contrary, in the expiratory phase the relaxation of the diaphragm decreases the thoracic volume and the sign of the pressure gradient changes (becomes positive), causing the direction of flow to be reversed. As air moves when breathing is accomplished, oxygen gas and carbon dioxide are exchanged.

In patients suffering from acute respiratory failure, such gas exchange is inadequate and normal pulmonary ventilation is augmented or replaced by a mechanical ventilator. Mechanical ventilators are machines that generate a controlled flow of gas and constantly measure the airway pressure (*Paw*), the quantity of air that enters the lungs per unit time (flow Q) and how much air enters and leaves the lungs (volume V). Despite the possibility of continuously monitoring such ventilatory parameters (*Paw*, Q and V), one of the major clinical concerns in mechanical ventilation is represented by asynchronies, a generic term describing a wide class of 'poor interactions' between the mechanical ventilator and human breathing. Asynchronies affect more than one third of mechanically ventilated patients [35,37,13] and, despite the debated question of cause-effect relation to poor outcome [10], they generate stress and discomfort for the patient, providing uncontrolled delivery of large volumes or high pressures to the patient respiratory system. Asynchronies potentially contribute to ventilator induced lung injury [39,36,30,38,22]. Asynchronies can appear during all the phases of the respiration: the triggering phase; the pressure-delivery phase; the cycling-off phase [21]. During the initial triggering phase, triggering delay, ineffective inspiratory effort and auto-triggering may occur. During the pressure-delivery phase the ineffective triggering is the major concern, but also inadequate or excessive ventilator assist is a problem, as well as the lack for an optimal setting of pressure rise time. During the cycling-off phase the premature opening (inadequate assist and double triggering) and the late opening (triggering delay and ineffective effort) of the expiratory valves are the major concerns [36,22,21]. Asynchronies can also interact appearing into one breathing act [27]. Only sophisticated ventilators are currently equipped with supplementary devices (e.g. neurally adjusted ventilation, [32]) that reveal and quantify [31] the presence of such phenomena. A human intervention is therefore often required to analyse and interpret data. For this reason, simple algorithms based on standard waveforms of pressure, flow and volume able to detect anomalies will be useful tools to automatise the diagnostic process [21]. Currently, various algorithm have been presented. While in [15] the ineffective inspiration triggering efforts have been addressed by a FORTRAN procedure evaluating phase portrait flow loops, other authors [13,28,29,8] directly investigate on numerical or analytical aspects of flow Q waveform.

In this context, two problems call for consideration, i.e. the classification of single breathing acts and the recognition of sequences of breaths exhibiting a pattern leading to severe respiratory failure.

Learning logical formulae discriminating between different conditions is a possible line of research in both cases and could be easily put into practice implementing monitoring algorithms in cheap hardware such as FPGA-based devices.

In this paper the focus is set on the first problem. In particular, we are interested in learning temporal logic properties that characterise single breathing acts. The methodology that we illustrate is then applied to a specific case, i.e. the recognition of ineffective inspiratory efforts considering flow data. An ineffective inspiratory effort (IE) is a condition that arises when a patient receiving mechanical ventilation tries to inspirate when the pressure gradient is positive and the drop in pressure related to the activation of the inspiratory muscles is unable to change the sign of the gradient, causing inspiration and triggering of a new ventilation cycle not to occur. A single breath may be affected by one or more IE and the presence of each IE may be revealed by the presence of a hump in the flow curve, see Figure 1.

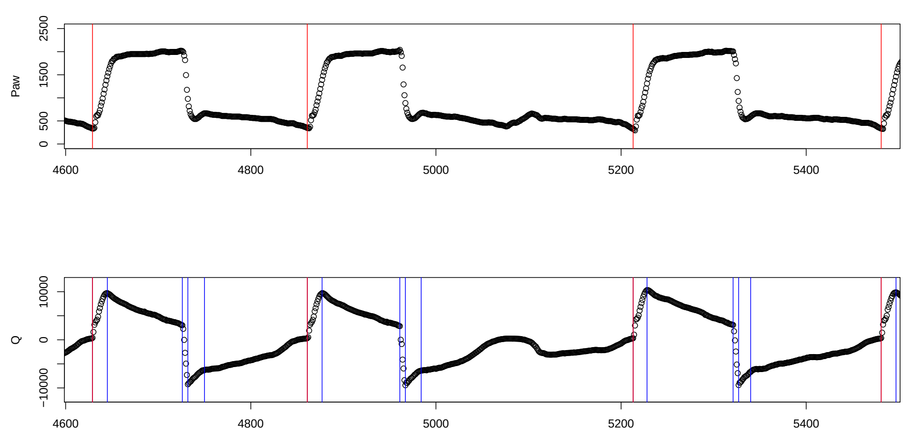

Fig. 1. *Paw* and Q tracings of two standard breaths and a breathing act with an IE divided into single respiratory acts (red lines). The different phases used to build the stochastic models of flow curves (Section 3.1) are also highlighted (blue lines)

3 Methodology

The general problem of learning temporal properties of a system $\mathcal{A}$ can be rephrased and recast within specific contexts, in accordance with the available data and the final

objective. We assume that the system is observable and system observations are available and conceive properties as logical statements. Within this framework, we consider a discriminative variation of the learning problem, i.e. a second system $\mathcal{B}$ is introduced and properties that best discriminate between $\mathcal{A}$ and $\mathcal{B}$ (i.e. logical formulae that are satisfied by $\mathcal{A}$ and not by $\mathcal{B}$) are searched for. Different approaches to this problem are possible. At a high level, our methodology starts by devising a data-driven statistical abstraction of each system. In this way, systems are represented by generative models which can be simulated *ad libitum* (preventing the occurrence of data shortage problems) and properties describe the trajectories sampled from the models. The second step is the property synthesis phase, where learning of formulae is performed. In more detail, a score function $R(\varphi)$, depending on the formula φ, based on the simulation of both models and representative of the discriminating power of each formula is introduced and optimised. Even though other choices are possible, we have decided to consider structure and parameter formulae components separately and tackle these suboptimisation problems using a local search algorithm and a Bayesian optimisation approach, respectively.
In the following sections, the methodology introduced above will be applied to learn properties of flow curves of MV breathing acts with an ineffective effort (IE, system $\mathcal{A}$). In order to capture properties that are related to the IE only, standard breath flow curves are considered as system $\mathcal{B}$. The statistical models used to represent $\mathcal{A}$ and $\mathcal{B}$ are Stochastic Hybrid Systems (Section 3.1) and the logic chosen to specify properties is $\mathrm{MITL}_{[a,b]}$ (Section 3.2). The score function $R(\varphi)$ is based on the log odds ratio (Section 3.3) and is optimised considering structure and parameter formulae components separately. Structural learning is accomplished with an Evolutionary Algorithm (Section 3.4) and formula parameters are refined resorting to a Bayesian optimisation routine (Section 3.5).

3.1 Statistical modelling of ventilation signals

The models chosen to represent flow curves are Stochastic Hybrid Systems [11,16] devised from clinical data of a patient assisted through mechanical ventilation. The training data used to build our models were organised as discrete time series and sampled flow values of 46 breathing acts with an IE in the expiration phase ($\mathcal{A}$ training data) and 251 standard breaths ($\mathcal{B}$ training data), from a single patient. We will now briefly illustrate how the model of system $\mathcal{B}$ was built. Then, we will explain how we derived the model of system $\mathcal{A}$ from this. Looking at the flow curve structure, we can see that each breath can be naturally divided into five parts or phases (see Figures 1 and 2). Within each phase, representing the discrete skeleton of the hybrid model, we described the evolution by continuous components representing the flow value and the duration of the phase. Time is kept discrete, to mimic the sampling frequency of real data, equal to 100Hz. We supposed that the length of each phase was normally distributed, and devised mean and variance parameters of each discrete sub-model from training data. A new duration (truncated to the closer time step) is sampled every time the system changes phase, hence this operation can be formally modelled as part of the reset function attached to each discrete transition. The flow component was instead treated as a discrete dynamical model: the flow value at a time instant t is a function of the flow value at the time instant $t-1$. Visual inspection of normal patient traces (shown in e.g. Figure 2) suggested that, within each breathing phase, a linear first-order autoregressive model may be appropriate. The resulting model of system $\mathcal{B}$ is therefore

$$\begin{cases} flow(t+1) = f_k(flow(t)) + \varepsilon_k \\ f_k(flow(t)) = a_k \cdot flow(t) + b_k \\ \varepsilon_k \sim \mathcal{N}(0, \alpha_k^2) \\ length_k \sim \mathcal{N}(\mu_k, \sigma_k^2) \end{cases}$$

Parameters a_k, b_k and α_k are calculated using linear regression from $\mathcal{B}$ training data. Whereas for $k = 1, \ldots, 4$ the slope a_k and the intercept b_k are constant, for $k = 5$ these coefficients depend on the length (i.e. on the realization of $length_5$). This choice is based on the observation that length, intercept and slope of final parts are highly correlated.

The model of system $\mathcal{A}$ was built in a similar way from the correspondent training data. Inspection of sample IE trajectories revealed a conspicuous anomaly in phase 5 of the breathing act; we therefore introducted a novelty factor in the phase 5 submodel to capture the presence of the IE. We decided to tackle this part introducing a hierarchical model, i.e. to describe an IE signal as a normal signal plus a perturbation, which for IE in the expiration phase is a sinusoidal-like hump, see Figure 3.1. We constructed a statistical model of such a hump by fitting a polynomial curve (whose degree, equal to seven, was selected by optimising the Aikake information content [7]).

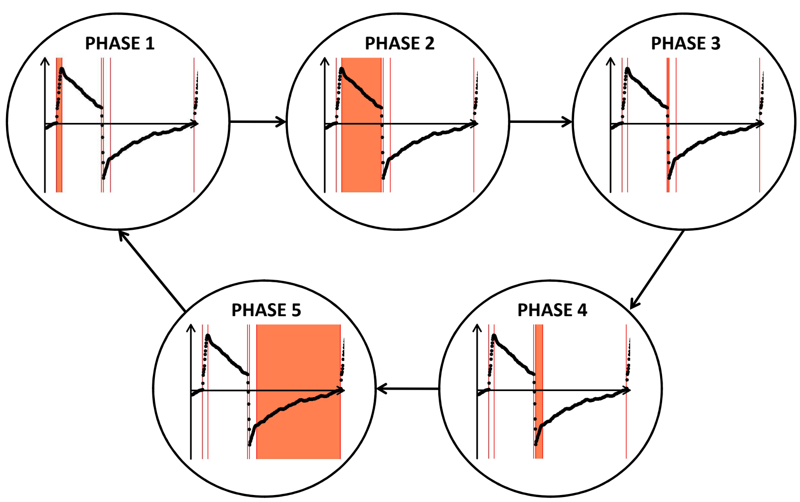

Fig. 2. Scheme of the stochastic hybrid model of standard breath flow curves. Each phase corresponds to the highlighted segment of the flow curve.

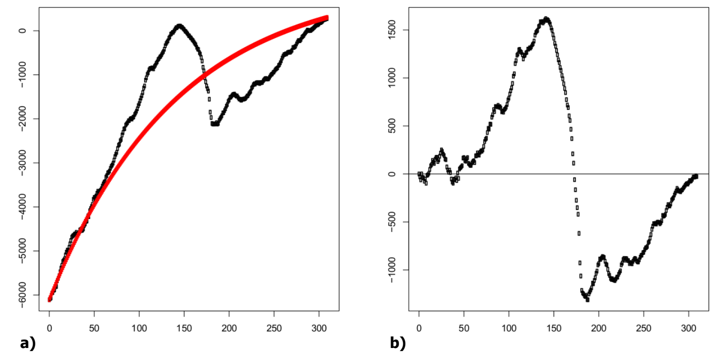

Fig. 3. (a) Flow signal f (part 5, black) of a breathing act with an IE with overlapped the normal signal n (red). (b) Perturbation of the signal , computed as $f - n$.

3.2 Metric Interval Temporal Logic

We consider here $\mathrm{MITL}_{[a,b]}$ [1,26] a fragment of the Metric Temporal Logic [24] with linear time-bounded temporal operators which has proven to be an efficient formalism to characterise properties of real-valued signals evolving in continuous time. For this reason, we have decided to adopt this logic to specify properties of the trajectories sampled from our models.

The syntax of $\mathrm{MITL}_{[a,b]}$ is given by the following grammar:

$$\varphi ::= \top \mid \mu \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \mathrm{U}_{[a,b]} \varphi_2$$

where $\top$ is the true formula and temporal modalities are restricted to intervals of the form $[a,b]$ with $0 \leq a < b$ and $a, b \in \mathbb{Q}_{\geq 0}$. Formulae are built from atomic propositions μ using boolean operators $\neg$, $\wedge$ and time-constrained versions of the until operator U. Atomic propositions are boolean predicate transformers, i.e. operators transforming real-valued functions into boolean signals, which provide a true ($\top$) or false ($\bot = \neg\top$) value to the formula at each time instant.

Further temporal modalities are derived from the $\mathrm{MITL}_{[a,b]}$ syntax and commonly used. As an example, time-bounded eventually $\Diamond_{[a,b]}\varphi \equiv \top\ \mathrm{U}_{[a,b]}\ \varphi$ and time-bounded globally $\Box_{[a,b]}\varphi \equiv \neg\Diamond_{[a,b]}\ \neg\varphi$ can be defined as usual from the until operator. $\mathrm{MITL}_{[a,b]}$ formulae are interpreted over a time instant t and a real-valued function $\mathbf{x}$, and the satisfaction relation is given in a standard way, see e.g. [26]. We recall that a stochastic model induces a distribution on the space of trajectories, hence we can compute the probability of the set of trajectories that satisfies a given $\mathrm{MITL}_{[a,b]}$ formula φ. We will refer to such probability $p(\varphi)$ as the satisfaction probability of φ, see e.g. [3] for further details. In the context of this work, we estimated such a probability by statistical means, resorting to Statistical Model Checking [14,42].

3.3 Discrimination Function

The problem of finding formulae that are likely to be satisfied by trajectories sampled from the model of system $\mathcal{A}$ but not by trajectories sampled from the model of system $\mathcal{B}$ is translated into an *optimisation problem* of the *discrimination function* $R(\varphi)$ associated with a $\mathrm{MITL}_{[a,b]}$ formula φ. A possible choice for such function is $R(\varphi) = L(\varphi)$, the log odds ratio between the satisfaction probabilities

$$L(\varphi) = \log \frac{p(\varphi \mid \mathcal{A} \text{ model})}{p(\varphi \mid \mathcal{B} \text{ model})} \tag{3.1}$$

In this case, penalty terms could be introduced to favour formulae which satisfy certain properties (e.g. having a small size, thus penalising complex formulae over simple ones).

3.4 Structural Learning

We will now present how the structure of the discriminating $\mathrm{MITL}_{[a,b]}$ formulae (i.e. the formulae which optimise φ) was found. As previously mentioned, we decided to tackle this optimisation problem using an Evolutionary Algorithm (EA) [19]. EAs are a class of search and optimisation algorithms inspired by models of the natural selection of species. The main idea of an EA is to consider a starting (usually randomly chosen) population of candidate solutions (the starting generation) and iteratively evolve it towards better solution sets. Each iterative step produces a new generation by manipulating the previous one using stochastic operators (the genetic operators) and the procedure ends when a fixed number of generations has elapsed or some form of convergence criterion has been met. The most simple EAs are based on the use of three genetic operators which resemble the biological principles of survival of the fittest (selection operator), reproduction (recombination operator) and gene mutation (mutation operator). In the framework of EAs, selection is used to choose the individuals (parents) that will pass the information they contain to the next generation, recombination to generate new (and possibly better) individuals by combining parental individuals information and mutation to introduce innovation in the population. One of the main attraction of EAs is that operators are practically implemented by simple algorithms and usually finds very quickly good solutions.

When learning discriminating formulae in our case study, we considered populations of $\mathrm{MITL}_{[a,b]}$ formulas, represented by their parsing trees. Within this framework, recombination and mutation are simply implemented by performing with a certain probability an exchange of parental subformulas (recombination) and a modification of a node (mutation, e.g. of a boolean or temporal operator).

3.5 Parameter Learning

We now turn to the issue of tuning the parameters of formulae to maximise their satisfaction probability. More specifically, suppose to have a $\mathrm{MITL}_{[a,b]}$ formula φ_θ which depends on some continuous parameters θ. We aim to maximise its discriminative power

$R(\varphi_\theta)$ defined in equation (3.1). Naturally, this quantity is an intractable function of the formula parameters; yet its value at a finite set of parameters can be noisily estimated using a stochastic model checking procedure, i.e. by simulating the model for a certain number n of times, checking the formula in each run, and then estimating $R(\varphi_\theta)$ from the so generated data. The problem is therefore to identify the maximum of an intractable function with as few (approximate) function evaluations as possible. This problem is closely related to the central problem of reinforcement learning of determining the optimal policy of an agent with as little exploration of the space of actions as possible. We therefore adopt a provably convergent stochastic optimisation algorithm, the GP-UCB algorithm [33], to solve the problem of continuous optimisation of formula parameters. Intuitively, the algorithm interpolates the noisy observations using a stochastic process (a procedure called emulation in statistics) and uses the uncertainty in this fit to determine regions where the true maximum can lie. This algorithm has already been used in a formal modelling scenario in [9].

4 Results: monitoring ineffective respiratory acts

We present here the results obtained by applying our learning procedure on discrimination of IE occurring during expiration. Taking into account only the expiratory phase, only the last part of the trajectories sampled from the statistical models (i.e. phase 5) is considered. Accordingly, the time instant 0 of the $\mathrm{MITL}_{[a,b]}$ formulae refers to the time instant when phase 5 is entered. The set of $\mathrm{MITL}_{[a,b]}$ formulae examined is built over the set of atomic propositions

$$\mathcal{P} = \{flow \leq \lambda\} \cup \{flow \geq \lambda\} \cup \{flow' \leq \mu\} \cup \{flow' \geq \mu\}$$

where $flow'(t) = flow(t+1) - flow(t)$. We search for short formulae maximising the discrimination function $R(\varphi)$ associated with a $\mathrm{MITL}_{[a,b]}$ formula φ, described in 3.3. Since a trajectory sampled from a statistical model does not have a fixed duration (it is thus not always possible to know *a priori* if its truth value over a $\mathrm{MITL}_{[a,b]}$ formula φ is definable), a penalty term $U(\varphi)$ is introduced to keep track of the number of non-sufficiently long trajectories generated during the calculation of the value of φ over φ. As a result, $R(\varphi) = L(\varphi) - S(\varphi) - U(\varphi)$, where $L(\varphi)$ is the log odds ratio between the satisfaction probabilities and $S(\varphi)$ is a size penalty. We experimented our learning algorithm by testing different parameters and settings, such as different variants of the Evolutionary Algorithm operators, the frequency of utilisation of GP-UCB within the evolutionary algorithm (i.e., we optimised all elements of a population, only best candidate solutions, only best solutions at the end of the algorithm), and the values of the penalty terms. The best formulae obtained are

$$\varphi_1 = \Box_{[0.4518,0.8609]}(\Diamond_{[0.7853,0.9394]}(\Box_{[0.6370,0.8222]}(\Diamond_{[0.7923,0.8070]}(flow \geq -4554.0))))$$
$$\varphi_2 \equiv \Diamond_{[0.3966,1.6705]}(flow' \leq -144.2708)$$

Their satisfaction probabilities $p_{\mathcal{A}}(\varphi) = p(\varphi \mid \mathcal{A}\text{ model})$ and $p_{\mathcal{B}}(\varphi) = p(\varphi \mid \mathcal{B}\text{ model})$, summarised in the table below, were estimated by statistical model checking [42,14].

	φ_1	φ_2
$p_{\mathcal{A}}$	0.5040	0.88523
$p_{\mathcal{B}}$	$< 10^{-3}$	$< 10^{-3}$

If we inspect these two formulae, we can easily understand their meaning. Formula φ_1 roughly forces the signal to be longer than 3 seconds (forcing the flow to be defined at that time), and captures the fact that IE respiratory acts tends to last longer than normal ones. Formula φ_2, instead, detects a quick drop in the flow, corresponding the decreasing part of the hump, which is generally not present in a normal breath.

Formulae were then validated on real data from the same patient considered in the training phase, specifically on a test set of 345 standard breaths and 77 breathing acts with an IE. In this phase, φ_1 was able to recognise 33 ineffective efforts, whereas φ_2 26. False positives (i.e. normal breaths satisfying formulae) were detected during validation of φ_1 only. We decided to merge these two formulae using logical disjunction and validate the obtained formula $\varphi_1 \vee \varphi_2$. As a result, 58 ineffective efforts (75.3%) and 336 standard breaths (97.4%) were correctly classified.

5 Related Work

Mining temporal logic specifications from data is an emerging field of computer aided verification [2,12,25,18,40,41]. Generally, this task is predicated on the availability of a fully specified model, enabling a quantitative evaluation of the probability that a certain formula will hold. This enables the deployment of optimisation based machine learning techniques, such as decision trees [18] or stochastic optimisation methods [41,40]. Learning temporal logic specifications directly from observed traces of the system is considerably more challenging. In general, solving the full structure and parameter learning problem is infeasible, due to the intractability resulting from a hybrid combinatorial/continuous optimisation problem. Heuristic search approaches have been proposed in [12]; while these may prove effective in specific modelling problems, they generally do not offer theoretical guarantees, and can be prone to over-fitting/vulnerable to noise. Geometric approaches such as the one proposed in [2] rest on solid mathematical foundations but can also be vulnerable to noise, and require potentially very large amounts of data to permit identification. The work of [23], instead, employs a notion of robustness of satisfiability of a formula to guide an optimisation based mining procedure. While this approach can be applied also in a model-free scenario, empirical estimation of the robustness of a formula may require the observation of a large number of traces of the system. Furthermore, the approach is based on some monotony properties of a subset of formulae which does not hold for the log-odd ratio score.

Our approach instead combines statistical modelling ideas from machine learning with formal verification methods. In this respect, our work is related to a number of other recent attempts to deploy machine learning tools within a verification context [6,34,20]. Similar ideas to the ones used in this paper have been deployed on the parameter synthesis problem in [9,3], where the GP-UCB algorithm was used to identify the parameters of a model which maximised the satisfaction/robustness of a formula. Statistical abstractions draw their roots in the *emulation* field in statistics: within the context of dynamical systems, emulation has been recently used in [17] to model compactly the interface between subsystems of complex gene regulatory networks.

6 Conclusions

In this paper we presented a method to learn temporal properties that discriminate two classes of temporal signals, by first constructing generative statistical models of the two sets, and then exploring the formula space searching for good discriminating formulae with a combination of evolutionary algorithms and bayesian optimisation strategies. This method has been applied to detect patient/ventilator asynchronies in patients in intensive case, and exemplified on the detection of ineffective respiratory efforts during expiration.

The method we presented is still in a preliminary development stage, and has some limitations. First of all, the trained formulae consider only the flow; keeping track of pressure should increase its performance. Secondly, the parameters of the formulae depend on properties of input signals like the range of the flow and the average phase duration, so that they tend to be patient specific. One way to attack this problem would be to optimise again the (key) parameters while starting monitoring a new patient. A more interesting alternative can be to normalise flow and pressure signals so that their duration and range becomes the same for any patient. We are currently investigating the benefits and limits of this idea. More generally, a difficulty we found is that the hard time bounds of formulae conflict with the different durations of breaths even for a single patient. Possible solutions we are investigating include adding more discrete phases to the generative models or checking properties of signals in the flow/pressure phase space, rather than of the time-flow/ pressure representation.

Another issue with the current approach is the score function. The log odd ratio, in fact, tends to privilege the decrese of the satisfaction probability of the formula in the second model rather than its increase in the first one, i.e. to decrease the false positive rate rather then the false negative one. The reason for this is readily explained: if the probability in the second model passes from 10^{-3} to 10^{-2} then the log odd ratio decreases by an additive term of $-\log 10$, while if the satisfaction probability of the first model passes from 0.5 to 1, the log odd ratio in increased only by $\log 2$. Hence, better scoring function are needed. Indeed, this is confirmed by the following experiment with the formula φ_2 of Section 4: we run the GP-UCB algorithm optimising only its satisfaction probability in the first model, varying the threshold $\theta_0 \approx -144$ in the range $[-300, -30]$. In this case, the problem resulted monotonic and the optimum is obtained for $\theta^* = -30$. With this new parameter, the discriminative power of the formula φ_2 alone on the validation set passed from a false negative (false positive) rate of 60% (of 0%) to a rate of 8% (of 3.3%).

The presented method can be further extended in trying to detect other kinds of asynchronies and surely requires extensive testing before reaching one of our final goals, i.e. its implementation in a dedicated hardware.

Acknowledgements. L.B. acknowledges partial support from the EU-FET project QUANTICOL (nr. 600708) and by FRA-UniTS. G.S. acknowledges support from the ERC under grant MLCS306999.

References

1. Alur, R., Feder, T., Henzinger, T.A.: The benefits of relaxing punctuality. J. ACM 43(1), 116–146 (1996)
2. Asarin, E., Donzé, A., Maler, O., Nickovic, D.: Parametric Identification of Temporal Properties. In: Proc. of RV 2011. LNCS, vol. 7186, pp. 147–160 (2012)
3. Bartocci, E., Bortolussi, L., Nenzi, L., Sanguinetti, G.: On the robustness of temporal properties for stochastic models. In: Proc. of HSB 2013. pp. 3–19 (2013)
4. Bartocci, E., Bortolussi, L., Sanguinetti, G.: Learning temporal logical properties discriminating ECG models of cardiac arrhytmias. CoRR abs/1312.7523 (2013)
5. Bartocci, E., Bortolussi, L., Sanguinetti, G.: Data-driven statistical learning of temporal logic properties (2014)
6. Bartocci, E., Grosu, R., Karmarkar, A., Smolka, S.A., Stoller, S.D., Zadok, E., Seyster, J.: Adaptive runtime verification. In: Proc. of RV 2012. LNCS, vol. 7687, pp. 168–182 (2012)
7. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2006)
8. Blanch, L., Sales, B., Montanya, J., Lucangelo, U., Garcia-Esquirol, O., Villagra, A., Chacon, E., Estruga, A., Borelli, M., Burgueño, M., Oliva, J., Fernandez, R., Villar, J., Kacmarek, R., Murias, G.: Validation of the better care system to detect ineffective efforts during expiration in mechanically ventilated patients: A pilot study. Intensive Care Med. (In Press)
9. Bortolussi, L., Sanguinetti, G.: Learning and Designing Stochastic Processes from Logical Constraints. In: Proc. of QEST 2013. LNCS, vol. 8054, pp. 89–105 (2013)
10. Branson, R.: Patient-ventilator interaction: the last 40 years. Respir. Care 56(1), 15–24 (2011)
11. Bujorianu, M.L., Lygeros, J.: General stochastic hybrid systems. In: IEEE Mediterranean Conference on Control and Automation MED. vol. 4 (2004), `http://www.nlr.nl/public/hosted-sites/hybridge/documents/C2.4%201213.pdf`
12. Calzone, L., Chabrier-Rivier, N., Fages, F., Soliman, S.: Machine learning biochemical networks from temporal logic properties. Trans. on Comp. Sys. Bio. VI pp. 68–94 (2006)
13. Chen, C., Lin, W., Hsu, C., Cheng, K., Lo, C.: Detecting ineffective triggering in the expiratory phase in mechanically ventilated patients based on airway flow and pressure deflection: feasibility of using a computer algorithm. Crit Care Med. 36(2), 455–461 (2008)
14. Clarke, E., Donzé, A., Legay, A.: On simulation-based probabilistic model checking of mixed-analog circuits. Formal Methods in System Design 36(2), 97–113 (2010)
15. Cuvelier, A., Achour, L., Rabarimanantsoa, H., Letellier, C., Muir, J., Fauroux, B.: A noninvasive method to identify ineffective triggering in patients with noninvasive pressure support ventilation. Respiration 80(3), 198–206 (2010)
16. Davis, M.: Markov Models and Optimization. Chapman & Hall (1993)
17. Georgoulas, A., Clark, A., Ocone, A., Gilmore, S., Sanguinetti, G.: A subsystems approach for parameter estimation of ode models of hybrid systems. In: Proc. of HSB 2012. EPTCS, vol. 92 (2012)
18. Grosu, R., Smolka, S.A., Corradini, F., Wasilewska, A., Entcheva, E., Bartocci, E.: Learning and detecting emergent behavior in networks of cardiac myocytes. Commun. ACM 52(3), 97–105 (2009)
19. Hoos, H.H., Stützle, T.: Stochastic local search: Foundations & applications. Elsevier (2004)
20. Kalajdzic, K., Bartocci, E., Smolka, S.A., Stoller, S., Grosu, G.: Runtime Verification with Particle Filtering. In: Proc. of RV 2013. LNCS (2013)
21. Kondili, E., Akoumianaki, E., Alexopoulou, C., Georgopoulos, D.: Identifying and relieving asynchrony during mechanical ventilation. Expert Rev. Respir. Med. 3(3), 231–243 (2009)
22. Kondili, E., Prinianakis, G., Georgopoulos, D.: Patient-ventilator interaction. Br. J. Anaesth. 91(1), 106–119 (2003)

23. Kong, Z., Jones, A., Ayala, A.M., Gol, E.A., Belta, C.: Temporal Logic Inference for Classification and Prediction from Data. Proc. of HSCC 2014 (2014)
24. Koymans, R.: Specifying real-time properties with metric temporal logic. Real-Time Syst. 2, 255–299 (1990)
25. Lee, C., Chen, F., Roşu, G.: Mining parametric specifications. In: Proc. of ICSE 2011. pp. 591–600. ACM (2011)
26. Maler, O., Nickovic, D.: Monitoring temporal properties of continuous signals. In: Proc. of FORMATS. LNCS, vol. 3253, pp. 152–166 (2004)
27. Mellott, K., Grap, M., Munro, C., Sessler, C., Wetzel, P., Nilsestuen, J., Ketchum, J.: Patient ventilator asynchrony in critically ill adults: Frequency and types. Heart Lung 43(3), 231–43. (2014)
28. Mulqueeny, Q., Ceriana, P., Carlucci, A., Fanfulla, F., Delmastro, M., Nava, S.: Automatic detection of ineffective triggering and double triggering during mechanical ventilation. Intensive Care Med 33(11), 2014–2018 (2007)
29. Mulqueeny, Q., Redmond, S., Tassaux, D., Vignaux, L., Jolliet, P., Ceriana, P., Nava, S., Schindhelm, K., Lovell, N.: Automated detection of asynchrony in patient-ventilator interaction. In: Conf. Proc. IEEE Eng. Med. Biol. Soc. pp. 5324–5327 (2009)
30. Sassoon, C., Foster, G.: Patient-ventilator asynchrony. Curr. Opin. Crit. Care 7(1), 28–33 (2001)
31. Sinderby, C., Liu, S., Colombo, D., Camarotta, G., Slutsky, A., Navalesi, P., Beck, J.: An automated and standardized neural index to quantify patient-ventilator interaction. Critical Care 17, 239 (2013)
32. Sinderby, C., Navalesi, P., Beck, J., Skrobik, Y., Comtois, N., Friberg, S., Gottfried, S.B., Lindström, L.: Neural control of mechanical ventilation in respiratory failure. Nat Med. 5(12), 1433–1436 (1999)
33. Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.W.: Information-theoretic regret bounds for gaussian process optimization in the bandit setting. IEEE Transactions on Information Theory 58(5), 3250–3265 (2012)
34. Stoller, S.D., Bartocci, E., Seyster, J., Grosu, R., Havelund, K., Smolka, S.A., Zadok, E.: Runtime Verification with State Estimation. In: Proc. of RV 2012: the Second International Conference on Runtime Verification. pp. 193–207. Springer-Verlag (2012)
35. Thille, A., Rodriguez, P., Cabello, B., Lellouche, F., Brochard, L.: Patient-ventilator asynchrony during assisted mechanical ventilation. Intensive Care Med. 32(10), 1515–1522 (2006)
36. Tobin, M., Jubran, A., Laghi, F.: Patient-ventilator interaction. Am. J. Respir. Crit. Care Med. 163(5), 1059–1063 (2001)
37. Vignaux, L., Vargas, F., Roeseler, J., Tassaux, D., Thille, A., Kossowsky, M., L., B., P., J.: Patient-ventilator asynchrony during non-invasive ventilation for acute respiratory failure: a multicenter study. Intensive Care Med. 35(5), 840–846 (2009)
38. de Wit, M., Miller, K., Green, D., Ostman, H., Gennings, C., Epstein, S.: Ineffective triggering predicts increased duration of mechanical ventilation. Crit Care Med 37(10)(10), 2740–2745 (2009)
39. Wrigge, H., Reske, A.: Patient-ventilator asynchrony: adapt the ventilator, not the patient! Crit Care Med. 41(9), 2240–1. (2013)
40. Xiaoqing, J., Donzé, A., Deshmukh, J.V., Seshia, S.A.: Mining Requirements from Closed-loop Control Models. In: Proc. of HSCC 2013. pp. 43–52. ACM (2013)
41. Yang, H., Hoxha, B., Fainekos, G.: Querying Parametric Temporal Logic Properties on Embedded Systems. In: Proc. of ICTSS 2012, LNCS, vol. 7641, pp. 136–151 (2012)
42. Younes, H.L.S., Kwiatkowska, M.Z., Norman, G., Parker, D.: Numerical vs. statistical probabilistic model checking: An empirical study. In: Proc. of TACAS 2004. LNCS, vol. 2988, pp. 46–60 (2004)