ADAPTIVE NIBP LOW-PULSE DETECTION
Detecting Low-pulse Adaptively

Ricardo Henrique Gracini Guiraldelli and Ricardo Luis de Azevedo da Rocha
Polytechnic School, University of São Paulo, Prof. Luciano Gualberto Avenue, tv. 3, 158 , São Paulo, Brazil

Keywords: Non-invasive blood pressure, Low pulse amplitude, Adaptive decision trees, Adaptive technology.

Abstract: This paper presents a proposal of an adaptive device for low-pulse detection on NIBP measurements using an adaptive decision tree algorithm (AdapTree) and a probabilistic methodology, besides featuring learning related to expert knowledge.

1 INTRODUCTION

Non-invasive blood pressure (NIBP) is a must-have measure on critical and day-by-day diagnosis, providing fast and contiguous feedback about general health aspects from patients through a simple method of arterial pulse observation (electronically, pneumatically or acoustically way), supporting the medical staff decisions over one treatment. But its "noninvasibility" has some side effects such as amortized signal, what is oddly problematic on low pulse amplitudes cases. For instance, it has been the "neck" of NIBP and the increasingly dependence on automatic electronic measurement just enlarged the problem.

As as example, there is a correlate research from (Lin et al., 2003) that deals with noise compensation generated by movements artifacts in NIBP measures. It uses fuzzy logic to calculate the smoothness of oscillometric curve. We are using adaptive technology, on the other hand, somehow complementing it.

Aiming this specific problem, the present work proposes a technique for low pulse amplitude recognition and detection, as well as dynamic learning of these cases, seeking to improve the facility brought by NIBP, particularly on intensive care units (ICUs).

For text comprehension, this paper is organized as follows: from section 2 to section 4 we introduce the background techniques behind our proposal; at section 5 we describe the designed solution and at the 6the section we summarize our conclusion and expectations.

2 NON-INVASIVE BLOOD PRESSURE

Measurement of the arterial pressure is of greater importance on patient's diagnostics because of the high density of information about the body condition — especially hemodynamic system — it provides. Although the ideal arterial pressure is made invasively, this methodology is too much costly in health aspect to the patient and a very risky procedure. For this reason some non-invasive blood pressure (NIBP) methods were developed, like *auscultatory*, *Korotkoff sounds* and *oscillometric*, the latter one we will focus on.

The oscillometric method is widely used on automatic biomedical devices because it is independent of a human expert for proceeding with the measurement and it is extremely simple, based on descendant step-pressures (as shown on figure 1). At the end of the measure, the signals collected allow the plot of a *pressure (mmHg)* $\times$ *pulse amplitude)* chart, where is possible to extract the *mean arterial pressure (P_m)*, the pressure where occurs the global maximum of the curve. Once we have the P_m, we can calculate the *systolic pressure (P_s)* and *diastolic pressure (P_d)* by the equation that follows:

$$P_m = \frac{2}{3} \cdot P_d + \frac{1}{3} \cdot P_s \quad (1)$$

Although the oscillometric method seems as an ideal one, it is less efficient when the measure presents artifacts or low pulse amplitudes. Even thou it continues to be the best method for automatic biomedical devices nowadays.

Gracini Guiraldelli R. and de Azevedo da Rocha R. (2010).
ADAPTIVE NIBP LOW-PULSE DETECTION - Detecting Low-pulse Adaptively.
In *Proceedings of the Third International Conference on Biomedical Electronics and Devices*, pages 113-116
DOI: 10.5220/0002741101130116
Copyright © SciTePress

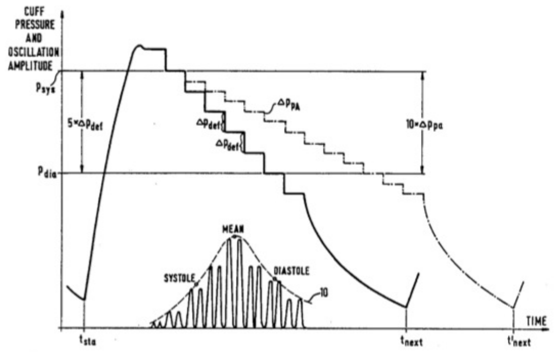

Figure 1: NIBP curve through oscillometric method (Frankenreiter, 1990).

3 PROBABILISTICS METHODS

The probabilistic techniques have widespread use on artificial intelligence because of their capability of information inference from small (or even lack of) data about the decision subject and are sustained by mathematical formalisms giving, this way, some reliable information.

Nowadays these techniques are even more popular because of the "smart software" development, i.e. software that executes a whole (and complex) task without rational human intervention (e.g. buying suggestion software at online bookstores). And these applications are getting so evolved that they are already at some specialized fields.

Resulting from that era of software development, a probabilistic method called *certainty factor* (Pearl, 1988) was designed focusing on bringing some expert knowledge to probabilistic reasoning, especially medical support one. Even more, it was thought to be simple to implement and provide fast inferences.

Certainty factor works with a set of hypothesis-evidences rules based on expert knowledge (interdependent or not) that, interconnected, infer some system's information. For reasoning purposes, each hypothesis or evidence has a *certainty degree (c_d)* such that $-1 \leq c_d \leq 1$, pointing out total unbelief (-1) or total belief (1) on the events veracity.

4 ADAPTIVE TECHNOLOGY

For complex systems development, some flexibility during decision making is an essential requirement. Today, biological systems are almost the unique ones that have this ability but some (mathematical) adaptive formalisms were developed, and they are capable of changing themselves at runtime (Neto, 1993; Shutt, 1995).

At the present work, *adaptivity* means *dynamic modification of a rules set that controls a specific device*, i.e. given a device, your transition function (or rules set) is dynamically modified at runtime of such device. It's a rough simplification of the Adaptive Technology concept presented on (Neto, 1993), but keeps the main meaning of it.

Adaptive technology has wide uses, especially in context change systems, as voice recognition and user-personalized systems (as biomedical software). In the attempt to satisfy the wide niche of applications, a number of adaptive devices were developed, all of them based on pre-existing and known models; e.g., adaptive finite automata(Neto, 1993), statecharts (Neto et al., 1998), decision tables (Pedrazzi et al., 2005) and trees(Pistori and Neto, 2002; Pistori and Neto, 2003). We are going to focus particularly on the last one.

4.1 AdapTree

When classifying data, decision trees are efficient devices for the task; based on this concept, there are numbers of algorithms(Quinlan, 1996).

Traditional decision trees needs some training over a solved body of "problem cases". This training is a huge limitation for systems that have a learning requirement and, for instance, there are some algorithms like ID3 that envisions some kind of learning based on re-training or become almost ineffective, though. Targeting to improve it, the AdapTree (Pistori and Neto, 2003) algorithm was developed using adaptive technology, putting together decision trees and adaptive finite state automata.

AdapTree still requires some training, receiving an input string with an additional data field for class representation (figure 2), what we will call as *static learning*. When changing from *training mode* to *classification mode*, we just need to suppress that additional data field for classification under the classes previously trained.

Whenever it finds a "problem case" that has not matched a classification pattern at the tree set, AdapTree calls an statistical mechanism based on relative observed frequency up to the moment and the string sequence already read, concluding the most probable class for that input data.

Although it seems too simple, AdapTree has been well positioned on benchmarking tests(Pistori and Neto, 2003) among several well-known algorithms, showing a great benefit/cost relation for a large number of projects.

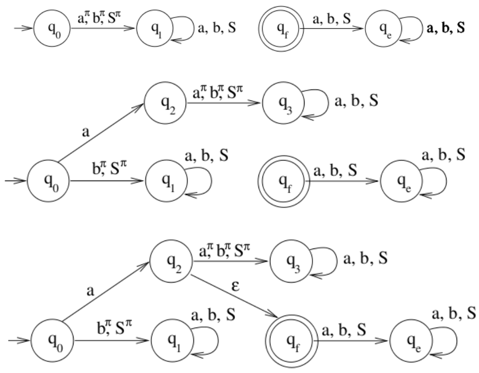

Figure 2: Three stages of AdapTree: the beginning, after a string read *a*, after a string read *aS*, respectively.(Pistori and Neto, 2003).

5 PROPOSAL

Once given the entire theoretical basis for understanding the problem we aimed, this section formalizes a proposal for the low-amplitude pulse problem (that we will refer as *low-pulse*).

Low-pulse is a condition of extremely difficult detection and though many A/D transducers have the capability to detect that amplitude, it is not safe to keep that sensibility activated because many environmental interferences (like patient's movement) may be erroneously translated as pulse signal. But some classes of special patients commonly presents this condition, when they are under drug or pathological effects, or even naturally, possibly introducing complications for their correct treatment and diagnosis. Analyzing some NIBP measures bank[1], we were able to design a simple class-feature classification that keeps cause-consequence relationship with low-pulse as shown below:

Table 1: Classes related with low-pulse.

Class	Feature
Patient	Neonate
Patient	Elderly
Place	ICU
Place	Surgical Ward
External Modules	Ventilation
External Modules	Anesthesia

It is important to note that the above information are available on ICU monitor once they're used to configure alarms, cuff pressures and other safety resources. Hence, the classification is appropriate to use on (almost) every ICU monitor without restrictions.

Although table 1 seems too simplified for some significant improvement, it's enough for problem definition without negative side effects on quality pulse detection; doubtful datum won't be necessary, in this research scope, for an appropriate solution definition.

Once we have the classes of probably low-pulse occurrence cases, the expected step is to realize some kind of algorithmic classification of them making possible a correct low-pulse classification based on those input datum. Previously, on section 4, we have seen that *decision trees* are the natural mechanism for knowledge classification; ID3 or C4.5 algorithms could be used but, as discussed early, biomedical equipments need some personalization characteristics for each new patient, hence, given this requirement, AdapTree fits as the best solution.

Using AdapTree, we should train the algorithm with real and classified data group (training body) for some rule determination before the classifier (decision tree) usage. In the same way it would be done if we were using some other conventional decision tree algorithm. With a large enough database, we could use it as a test body and reach a significant advantage once the (statistically) majority of cases would be covered with this kind of training.

If we cover the majority number of cases, why do we need AdapTree for? We need it for those cases in which common decision tree algorithms are unable to classify the complementary group, converging to the *dynamic learning* region of AdapTree; this kind of learning is usually made by statistical inference, based on counting.

Even thou literature benchmark (Pistori and Neto, 2002) shows its great performance, we must not use the counting mechanism because it is not based on medical knowledge for classifying data. We will use instead a probabilistic technique already used on medical decision auxiliary software, the *certainty factor* — which may be referenced as *inference machine* —, developing a hermetic rules set based on specialist knowledge (hypothesis relations) and statistical information (hypothesis weights).

Therefore, we propose the set of hypothesis of the inference machine as the classes presented on 1 and the development of the relations as showed on figure 3.

5.1 Connecting the Devices "Low-pulse Detector" and "Pulse Detector"

Once we have defined above the *low-pulse detector* device, now it must be connected with the traditional

[1]This NIBP bank of measures is private and it is exclusive of an ICU monitor manufacturer.

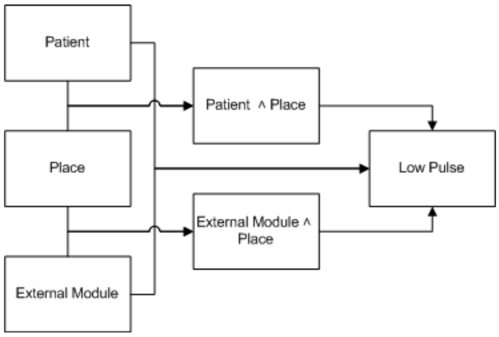

Figure 3: Hypothesis-inference machine relationship.

pulse detector device already implemented and in use on the ICU monitor.

Focusing on minimum system interference, we propose an event-driven (Ferg, 2006) mechanisms connection: after a specific no pulse timeout, the main module (which contains the pulse detector) requests, through an event, that the low-pulse identifier (or inference machine) evaluates and returns its *certainty factor* as other data, for example new pulse amplitude levels (for detection). Figure 4 shows this connection.

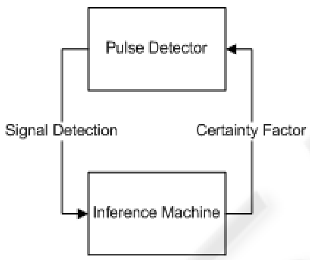

Figure 4: Communication interface between pulse identifier and inference machine.

A great advantage of this kind of communication between the modules is that it keeps the information segregated in each module, isolating possible software *bugs* from one module to affect the whole NIBP system and exposes this new module (low-pulse detector) as a system improvement, either.

6 CONCLUSIONS AND FURTHER WORK

This work proposes an evolution on apparently stagnated research field of NIBP (Rolfe, 1979), dealing with risk low-pulse cases through the use adaptive technology and probabilistic methods. The addition of these two features on NIBP measurements brings personalized diagnosis to automatic measure biomedical devices.

A numerical validation is planned to verify the methodology on the "field work" of the system and explore the opportunity to look for improvements. These data will be provided by a brazilian ICU monitor manufacturer, based on their large real measurements database and their certified and validate pulse detector algorithm.

Another step to be taken is to complete the solution, so we will also propose the design of methods for recalculating the (maximum and minimum) cuff pressure limits providing, this way, a complete solution for low-pulse detection.

REFERENCES

Ferg, S. (2006). Event-driven programming: Introduction, tutorial, history. Online.

Frankenreiter, M. (1990). Method for measuring blood pressure and apparatus for automated blood pressure measurements.

Lin, C.-T., Liu, S.-H., Wang, J.-J., and Wen, Z.-C. (2003). Reduction of interference in oscillometric arterial blood pressure measurement using fuzzy logic. *Transaction on Biomedical Engineering*, 50(4):432–441.

Neto, J. J. (1993). *Contributions to Compiler Design Methology*. PhD thesis, Escola Politécnica da Universidade de São Paulo. In Portuguese.

Neto, J. J., de Almeira Jr., J. R., and dos Santos, J. M. N. (1998). Synchronized statecharts for reactive systems. *Proceedings of the IASTED International Conference on Applied Modelling and Simulation*. Proceedings of the IASTED International Conference on Applied Modelling and Simulation.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kauffman.

Pedrazzi, T., Tchemra, A., and de Azevedo da Rocha, R. L. (2005). Adaptive decision tables - a case study of their application to decision-taking problems. *Proceedings of International Conference on Adaptive and Natural Computing Algorithms - ICANNGA 2005*. Proceedings of International Conference on Adaptive and Natural Computing Algorithms - ICANNGA 2005.

Pistori, H. and Neto, J. J. (2002). Adaptree - proposal of an algorithm for decision tree induction based on adaptive technology. *Anais Conferłncia Latino Americana de Informtica - CLEI 2002*. In Portuguese.

Pistori, H. and Neto, J. J. (2003). Decision tree induction using adaptive fsa. *CLEI Eletronic Journal*, 6.

Quinlan, J. R. (1996). Learning decision tree classifiers. *ACM Computing Surveys*.

Rolfe, P., editor (1979). *Non-Invasive Physiological Measurements*. Academic Press.

Shutt, J. N. (1995). Self-modifying finite automata: Power and limitations. *Technical Report*, (WPI-CS-TR-95-4).