



HAL
open science

Updating a rule basis under cognitive constraints: the COMAPS tool

Elisabeth Le Saux, Philippe Lenca, Jean-Pierre Barthélemy, Philippe Picouet

► **To cite this version:**

Elisabeth Le Saux, Philippe Lenca, Jean-Pierre Barthélemy, Philippe Picouet. Updating a rule basis under cognitive constraints: the COMAPS tool. Seventeenth European Annual Conference on Human Decision Making and Manual Control, Dec 1999, Valenciennes, France. pp.3 - 9. hal-02148228

HAL Id: hal-02148228

<https://hal.science/hal-02148228v1>

Submitted on 4 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UPDATING A RULES BASE UNDER COGNITIVE CONSTRAINTS: THE COMAPS TOOL

Elisabeth Le Saux
Philippe Lenca
Jean-Pierre Barthélemy
Philippe Picouet

*Département d'Intelligence Artificielle et Sciences Cognitives
ENST Bretagne - BP832
29285 Brest Cedex
FirstName.LastName@enst-bretagne.fr*

ABSTRACT

To be able to cope with the needs of flexibility and robustness, and after not fully satisfying trials with classical methods, the firms bend one's efforts towards anthropocentric approaches to (re)integrate the human factor in their process control as the essential resource.

The work we are leading, in the frame of an European project, is aiming at creating a process control experts decision support that takes into account his/her capacities and limits. We will first see what these limits are. Then, we will describe the model applied and we will finish with some remarks on the current state of tests and validations on pilot sites with real data.

KEYWORDS

bounded rationality, anthropocentric production systems, decision support, moving basis heuristic, rule basis.

CONTEXT: PROCESS CONTROL EXPERTS DECISION SUPPORT

In the frame of an European project COMAPS (COgnitive Management of Anthropocentric Production Systems - BRITE EURAM BE 96-3941), our work is aiming at helping the control expert of an industrial process to maintain his/her expertise for a longer time and at improving his/her knowledge of the process, revealing him/her the strategies s/he uses.

Nowadays, firms need to cope with an environment in perpetual evolution. To be able to cope with this need of flexibility and robustness, they apply methods allowing modelling, and thus prediction. In this case, these methods, often referred to as total quality management methods, coming from statistical modelling, operational research or artificial intelligence, do not fully account for flexibility and reuse. That is why an anthropocentric approach, where the human being is inside the loop of the industrial process, constitutes a meaningful alternative to full automation. Firms must rely on their operator expertise: the human being can no longer be considered as the origin of the errors on the system but as the safest and most reliable mean to guarantee flexibility and adaptability.

Thus, expert operators need to be identified, valorised and strengthened in their functions, because they are the guardians of the firm know-how.

We distinguish three phases (see [3]) in the life cycle of a process and, consequently, in the evolution of an expert's know-how:

1. a learning phase: the operator comes from the «novice» state to the «expert» state. During this phase, the operator daily makes trials on the process,
2. a maintenance phase: the expert operator applies his/her know-how and adapts the process control rules,
3. a reinitialisation phase (breaking phase): the structural changes are so important that a

simple adaptation is not enough anymore. A learning phase must be initialised once more.

Anthropocentric Production Systems (see [16]) are forms of advanced manufacturing which are dependent upon a balanced integration between human skills, collaborative work organisation and adapted technologies. This kind of approach is also a mean for knowledge capitalisation and thus for expertise transfer.

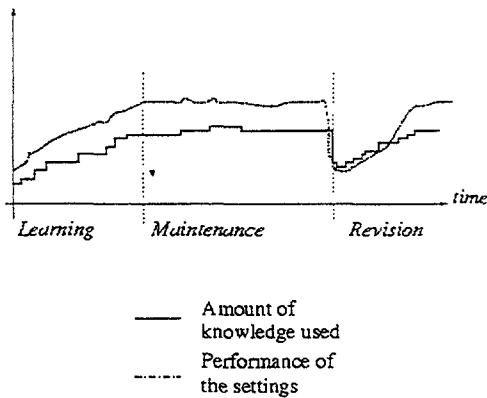


Figure 1 : evolution of an expert knowledge according to the process's

COGNITIVE MODEL

Our methodology is based on an on-line, non-intrusive acquisition of the operator's behavioural strategies and on a dialogue allowing to validate (on-line or off-line) their relevance.

COMAPS is not in the classical expert system frame because :

1. the operator modelling and the strategies extraction techniques are following cognitive principles (bounded rationality, parsimony, ...), that are, in general, relevant for decision support. We consider each choice made by the operator as an elementary decision,
2. the operator is always «in the loop», even for the strategies convergence to a pertinent set of rules,
3. the protocol and the algorithmic techniques are specific to the incremental and iterative aspect underlying the maintenance phase.

In COMAPS frame, the expert is observed directly in decision making situation, on the basis

of real cases called control situations (CS). A CS is a $(p+1)$ -tuple of p attributes values (related to the parameters of the process, the production and the environment) and one decision outcome. This decision outcome (or label) is the expert's decision. It can correspond to an effective modification of a certain physical setting, to an evaluation of the risk to have an insufficient quality on the product if it is produced without any modification of the current settings, ...

In order to learn decision maker strategies, we follow (as we already said before) cognitive principles which are mainly based on bounded rationality (as it was defined by Simon, see [14]) and on the Moving Basis Heuristic (MBH; see [4]).

The cognitive model assumes that the decision maker shows rationality (for situations in his/her usual domain of expertise) in the way that *something* is optimised. But this rationality is bounded ([14]) by his/her cognitive abilities (stocking and computing in short-term memory) and his/her satisfaction (shortcuts).

It also supposes s/he uses a not too large set of stable strategies involving a small number of attributes. These strategies are assumed to be stored in his/her long-term memory and may be rather complex. They have been constructed by his/her experience. In addition, it supposes that the expert uses some combinations of attributes more frequently than others.

This bounded rationality constrains the expert to search among aspects (attribute values) for a short sub-collection (limited but large enough to achieve decisions ; the data are processed in the short-term memory).

The MBH involves three cognitive principles:

- parsimony: the decision maker manipulates a short subset of aspects due to his/her short-term memory capacity (storage capacity - there is no intermediate storage in the long-term memory - and computational abilities) - see [2] and [8] - ,
- reliability/warrantability: the chosen sub-collection of aspects has to be large enough for individual or social justification - see [1], [5], [11] and [12] - ,
- decidability/flexibility: the decision maker

must effect a choice by appropriate changes in the sub-collection until a decision is taken (he/she has to achieve decision quickly in almost all cases) - see [7], [11] and [15].

We present now the algorithmic consequences of the cognitive model to build a rules basis consistent with expert decision.

ALGORITHMIC CONSEQUENCES

Three distinct phases

The COMAPS tool is articulated into 3 distinct phases, following the same kind of evolution has the process:

1. a first phase - learning phase - extracts a set of initial rules from a learning data set - called history - with an algorithm based on decision trees techniques,
2. a maintenance phase updates these rules according to new arriving CSs,
3. a conflict solving phase (see [13]) is called in case that no acceptable modification has been recommended by the maintenance phase. This part of the algorithm is the one that needs to interact a lot with the expert user and so all the man-machine interface is designed conjointly with it.

In case that the conflict solving phase fails in its job, there is a reinitialisation phase that can be a learning phase once more. This paper is dedicated to the maintenance phase which is the one we have been working on.

The maintenance phase

The maintenance phase can start either with the out come of the learning phase - a set of initial rules and the history - or with an empty rule set. This phase mainly controls the consistency of the rules base according to the arrival of new CSs, with regards to the following constraints, linked to the cognitive principles we want to follow:

1. remaining consistent with the previous CSs , i.e. modifying the rules so that the history is still described correctly with the updated rules set,

2. keeping a small number of attributes (at most 4 to be compatible with the expert's short-term memory abilities) involved in each rule - a rule can be based on more than 4 attributes but then it is just a temporary state and these rules are not shown to the expert -,
3. keeping a small number of rules.

When a new CS is to be processed, the expert gives the corresponding decision outcome, according to the parameters values, through the man-machine interface. The hand is then given to the maintenance phase that updates the rules set according to this new incoming CS.

Definitions

A control situation is said to be *covered* if at least one rule is containing it, even if the label of the rule is different from the label of the control situation. A control situation is said to be *not covered* if it does not exist any rule of any label containing it.

A control situation is said to be *well covered* if it is covered by at least one rule with the same label, such that :

- this rule has an higher priority than other covering rules or,
- there is no priority between the covering rules.

A control situation is said to be *badly covered* if:

- it is covered but only with rules having a label different than the control situation's one or,
- the rules of the same label as the control situation's one are dominated (according to the priorities between these rules) by a differently labelled rule.

To evaluate the quality of a rule modification, we have defined an indicator, called *certainty factor* (CF), which is the ratio of the number of well covered CSs on the number of covered CSs, for this rule. As it is shown in *Figure 2*, $CF(R_1)$ is $30/36 = 0.83$. It may be evaluated for a whole rule, for a specified neighbourhood of a control situation or for a potential extension of a rule. A threshold σ_{CF} is used to take a decision

concerning for example the creation of a new rule (its *CF* must be greater than σ_{CF}). This constant function σ_{CF} is not completely satisfactory because it does not take the context into account (it could be a function of the number of control situations, for example). Thus in the future it could take into consideration the labels distribution.

To deal with the intersection of rules of different labels, we have introduced the notion of *priority* between rules: a rule R_1 has the priority on a rule R_2 if, at their intersection, there is a majority (according to a threshold σ) of *CSs* having R_1 label: in $R_1 \cap R_2, |\text{label}(R_1)| > \sigma |\text{label}(R_2)|$.

For example, in *Figure 2*, if σ is equal to 1.5, R_1 has priority on R_2 .

Note that even with non-empty intersections, it can happen that no rule has the priority, if the threshold σ is not reached of course, but also if there are two few *CSs* to take any decision. Finally, we have to improve the priority model, which is now just a binary table, to avoid cycles in the priorities, which are just artifacts due to this matrix model.

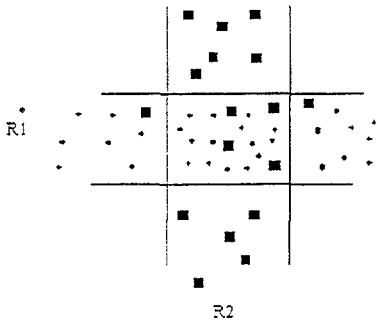


Figure 2 : priority between rules

Another thing that has to be mentioned is that the algorithm doesn't react immediately and waits until a sufficient number of *CSs* have confirmed that something has to be modified. The arrival of a conflicting (i.e. not or badly covered) control situation implies a modification try only if there are *enough* (according to a certain threshold) other control situations with the same label in its neighbourhood. The minimum number of *CSs* of the same label that a rule has to cover correctly to be retained is also a parameter of the algorithm.

The neighbourhood of a control situation can be defined in several ways, according to the conflict type: for a not covered control situation, it would be an extension of its values, on the different attributes scales, and, for a badly covered control situation, it would be the restriction of the covering rules to their intersection. This neighbourhood, used together with this previously mentioned threshold on the number of *CSs*, is not only used to know if we try or not to do something but also during the modifications themselves, to see if it can be interesting or not to apply them.

The last important thing is that a modification is accepted only if it improves, at least locally, the situation. In fact, no modification implying a former existing rule is accepted if it makes its certainty factor decrease and no new rule can be added to the rules base if its certainty factor is not *sufficient* (according to σ_{CF}).

All the thresholds that have been mentioned have been kept tunable because their value can be very different from a process to another, especially because the frequency of new control situations arrival can vary a lot.

Adapting the rules base

Three different cases can occur concerning the covering of a new incoming *CS*:

1. this *CS* is well covered,
2. there is no covering rule for this *CS*,
3. this *CS* is covered by at least rule having a different label.

The new CS is well covered

This first situation is of course the more comfortable. When the *CS* is correctly covered, the only thing that has to be done is to update the history to take this new confirming case into account for the following computations.

The new CS is not or is badly covered

We have implemented six different functions to adapt rules to *CSs* that are not or badly covered.

Five of them consist in modifying the rules base and the last one deals with priorities between intersecting rules:

- generalise an existing rule to cover a set of control situations that are not or badly covered by:
 1. suppressing one dimension over a rule to extend it,
 2. extending the possible values over one dimension of a rule.
- create a new rule to cover a set of control situations that are not or badly covered:
 3. from a bad-covering rule, the algorithm tries to create a sub-rule on the same dimensions than the initial rule ones but that would correctly cover the control situation,
 4. the algorithm tries to create a new rule independently of the dimensions of bad-covering rules.
- 5. restrict a rule that would be unnecessarily large: here, we try to restrict a rule by adding some constraints on one of the dimension that are not yet implied in the rule definition. We add a dimension to the rule to eliminate the badly covered CSs if the rule is too weak (i.e. it has a bad *CF*),
- 6. change the priorities between the rules involved: priorities are recomputed between the covering rules when these rules have a non-empty intersection. Let remember that even with non-empty intersections, there could be no priority between such rules, depending on the ratio of control situations per label.

According now to the situation for the current CS, we have the possibility to try some of these functionalities to adapt the rules base to this new case.

For a not covered CS, the only modifications we can make are to extend the possible values over one dimension of an existing rule (2.) or to create a completely new rule (4.). These two modifications are tried in this order because we always keep in mind to modify the rules base as few as possible and to keep a small number of rules. One of the criteria used to order the possible functions is the fact that the modifications they imply are local or not: extending the values domain of an attribute is a

local modification with regards to the addition of a new rule in the rules set, so we try it first.

For a badly covered CS, and according to the conflict type, several of these modifications can be tried, always starting with the one that implies the fewer modifications and ending with the really modifying one.

According to the cognitive model, we have to keep a small number of rules and not to have more than 4 attributes describing a rule. As some of these functions lead to increase the number of rules and others to increase the number of dimensions of a rule, we had to decide which of these criteria had to be first taken into account. Changing the priorities (6.) is clearly the function that brings the less modification and creating a new rule (4.) the one that leads to the biggest changing. The suppression of a dimension (1.) is a functions that is really different from the others: it is applied, whenever a well covered control situation arrives, for all the covering rules having more than 4 dimensions - we remind that these rules are not satisfying rules for us so if something confirms them, we try to find if they still explain the same control situations with a dimension less - . The extension of the values of a rule dimension (2.) is the second applied function: the modification is local. Then, the third alternative is the function (3.) because, of course it adds a new rule, but in a part of the space that was already covered by another existing rule. Then comes the addition of a dimension (5.) that is applied as late as possible because we want to keep small rules.

As the expert is the only one that decides in the end, it can occur that s/he doesn't accept the modification proposed. In this case, the hand is given to the conflict solving phase, which starts an interactive dialogue with the expert in order to understand the nature of the problem (which can be linked, for example, with a technology change).

This kind of approach has already been studied and validated in [3], [6], [9] and [10]. It is here improved by the fact that:

1. we don't assume anymore any hypothesis on the order (which should correspond to the decision maker attractiveness scales; these scales are difficult to obtain and are individual and so should be determined for each decision maker) of the attributes modalities,

2. there is no limitation on the number of attributes that can be taken into account (for real process, we have to deal often with more than twenty parameters),
3. the expert is observed in his/her real decision making process. It is a good way for an ecological validation of our methodology. COMAPS tool could provide the expert an on-line help.

CURRENT RESULTS

At this time, a mock-up has been implemented and tested on 2 industrial pilot sites (a copper foils manufacturing process and the pressing and drilling phases of a printed circuit boards manufacturing process). For each process, we have to deal with more than 20 attributes of any type (discrete or continuous, numerical as well as nominal).

Starting with an empty rules set and running the algorithm with approximately 1600 control situations, we have obtained sets of at most 10 rules, none of them relying on more than 4 attributes (only one rule with 4 attributes, 4 with two and 5 with 3), and all having certainty factors greater than 0.75.

For one of the processes, the expert has indicated a significant technological change, leading to a change in the *institutional rules* applied. To test the behaviour of the algorithm, we have run it with the old institutional rules and the underlying control situations as input, and the new control situations were corresponding to the production after this change. The rules obtained have been shown to the expert and he validated them, even though they were quite far from the new institutional ones: the difference between both rules sets is that the set issued from the algorithm aggregates rules that are kept different in the other set.

WORK PERSPECTIVES

The next step in this project is to install a prototype in each pilot site so that we can make the tests in a real decision context. Other work perspectives concern the algorithm itself: we want the algorithm to use the most frequent combinations of attributes (given by the expert or computed) in new heuristics, especially for the optimisation of the creation of new rules. As it has already been mentioned, we also want to

improve the certainty factor model, for example by comparing this ratio with a function of the labels frequency in the history. And the priority model has to be improved as well, by adding some constraints to avoid the cycles or by changing the matrix representation.

REFERENCES

- [1] T. Aldelbratt, H. Montgomery, « Attractiveness of decision rules », in *Acta Psychologica*, 45, 177-185, 1980.
- [2] K.M. Aschenbrenner, W. Kasubek, « Challenging the custing syndrome: Multi-attribute evaluation of cortison drugs », in *Organizational Behaviour and Human Performances*, 22, 216-234, 1978.
- [3] J.P. Barthélemy, G. Coppin, F. Guillet, « Smelting process control: from experimental design to acquisition of expertise », in *International Conference on Industrial Engineering and Production Management*, (2), 2-11, 1995.
- [4] J.P. Barthélemy, E. Mullet, « Expert individual decision: the moving basis heuristics », in *Collection des rapports de recherche de Télécom Bretagne*, RR-94005-IASC, 1994.
- [5] R. de Hoog, G. van der Wittenboer, « Decision justification, information structure and the choice of decision rules », in B. Brehmer, H. Jungermann, P. Lourens, and G. Sevon (editors), *New directions in research on decision making*, North-Holland, 1986.
- [6] F. Guillet, « Contributions à la maîtrise de qualité d'un processus industriel par apprentissage des stratégies de contrôle de l'opérateur », PhD. Thesis, Université de Rennes I, 1995.
- [7] O. Huber, « Decision making as a problem solving process », in B. Brehmer, H. Jungermann, P. Lourens, and G. Sevon (editors), *New directions in research on decision making*, North-Holland, 1986.
- [8] E.J. Johnson, J.W. Payne, « Effort and accuracy in choice », in *Management science*, 31, 295-314, 1985.

- [9] S. Laurent, E. Pichon, R. Bisdorff, « Solving by resolving: cognitive solving approach to finite decision problems in CHIP », in First European Conference on Cognitive Science in Industry, 53-66, 1994.
- [10] P. Lenca, « Acquisition automatique et analyse de processus de décision, application au domaine bancaire », PhD thesis, Université de Rennes I, 1997.
- [11] H. Montgomery, « Decision rules and the search for a dominance structure: toward a process model of decision-making », in P.C. Humphrey, O. Svenson, and A. Vari (editors), *Analyzing and Aiding Decision Process*, 343-369. Amsterdam North-Holland, 1983.
- [12] R. Ranyard, R. Crozier, « Reasons given for risky judgment and choice: a comparison of three tasks, in P.C. Humphrey, O. Svenson, and A. Vari (editors), *Analyzing and Aiding Decision Process*, Amsterdam North-Holland, 1983.
- [13] P. Saunier, « COMPAS : check as you decide paradigm », in *Seventeenth European Annual Conference on Human Decision Making and Manual Control*, 1998.
- [14] H. Simon, « Models of thought », Yale University Press, New Haven, 1979.
- [15] O. Svenson, « Process description of decision making », in *Organizational behavior and human performance*, 23, 86-112, 1979.
- [16] W. Wobbe, « Anthropocentric Production Systems: A Strategic Issue for Europe », volume I of APS Research Paper Series (FAST), European Communities, Brussels.