An anthropocentric tool for decision making support

Elisabeth Le Saux
Philippe Lenca
Philippe Picouet
Jean-Pierre Barthelemy
LIASC - ENST Bretagne
BP 832
F-29285 BREST CEDEX

Abstract

Nowadays, firms, formerly considering the human operator as the main error source in process control, bend their efforts towards anthropocentric approaches to (re)integrate the human factor, especially the knowledge he/she has been developping, as the essential resource for a high quality decision process.

As the expert operator remains a rare resource and in order to capitalize his/her knowledge and know-how, the development, of tools integrating this new dimension has become an important challenge.

This paper deals with a tool for knowledge acquisition under cognitive constraints, assuming that cognitive principles could be sometimes useful to improve machine learning tools results. Additionally, we have to cope with the difficulty linked to the fact that the acquired strategies have to be adapted on-line.

After describing the underlying cognitive principles, we will introduce the decision representation space and its related notations. We will then show the difficulties linked to the search of an optimal representation of the expert strategies set and how the heuristics used by the algorithm studied avoid these NP-complete problems. Finally, the current results and our work perspectives are stated.

1 Introduction

In a context of short time, if not on-line, decision making for the control of industrial processes, we've been led to defining a new method for process control experts decision support. As the classical machine learning and total quality management tools are not fully convenient here because they Inav not fully account for flexibility and reuse, we have adopted an anthropocentric approach, putting back the expert, operator in the center of the decision process so that the algorithm can turn his/her capacities as advantage.

This on-going work has been initialised in the frame of the European Brite-Eurarn project C0MAPS (Cognitive Management of Anthropocentric Production Systems - BE 9G-3941).

2 Cognitive model

We distinguish 3 phases in the life cycle of a process (see figure 1 and [Barthelemy *et a*/., 1995]):

- 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,
- a maintenance phase: the expert operator applies his/her know-how and adapts the process control rules,
- a reinitialisation phase (breaking/revision phase): the structural changes are so important that a simple adaptation is not, enough anymore. A learning phase must bo initialised once more.

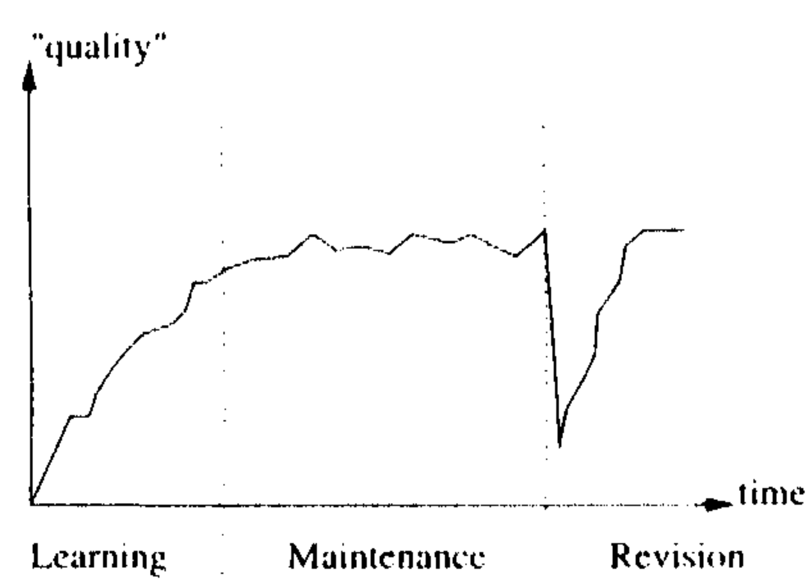

Figure 1: Life cycle of a process

In the frame of Anthropocentric Production Systems, which are forms of advanced manufacturing dependent upon a balanced integration between human skill, collaborative work organisation and adapted technologies, the human being is always *inside the loop* of the industrial process. Consequently, we have to adapt classical machine learning techniques to give back to the expert user the possibility to intervene in the decision process.

The underlying cognitive models we've used in this aim suppose the operator is an expert, applying stabilised process control rules for the usual maintenance of the production process. Once this restriction is cleared, we can apply the two following cognitive models:

- Bounded Rationality as described by Simon in [Simon, 1979],
- Moving Basis Heuristic from Barthelemy and Mullet ([Barthelemy and Mullet, 1992]).

2.1 Bounded Rationality

It 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 by his/her cognitive abilities (of stocking and computing in short-term memory) and his/her satisfaction.

It also supposes he/she uses a not too large set of stable strategies involving a small number of attributes. These strategies, constructed by his/her experience, are assumed to be stored in his/her long-term memory and they may be rather complex.

In addition, it supposes that some combinations of attributes are used more frequently than others.

As defined, this bounded rationality can be seen as a constraint for the expert to search among aspects (subset of attribute values) for a subcollection that should be short but large enough, with regards to its size and/or to its quality, to achieve decision.

2.2 Moving Basis Heuristic

The Moving Basis Heuristic (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 - strategies are computed in short-term memory -) - see [Asehenbrenner and Kasubek, 1978] and [Johnson and Payne, 1985] -,
- *reliability/wairantahility:* the chosen sub-collection of aspects has to be large enough (size and/or quality) for individual or social justification - see [Adelbratt and Montgomery, 1980], [de Hoog and van der Wittenboer, 1986], [Montgomery, 1983] and [Ranyard and Crozier, 1983] -,
- *decidability /flexibility:* the decision maker must effect a choice by appropriate changes in the subcollection until a decision is taken (he/she has to achieve decision quicklv in almost all cases) - see [Huber, 1986], [Montgomery, 1983], [Svenson, 1979]. Consequently, his/her strategies must be stable.

These principles, together with Bounded Rationality, show that the expert operator uses, at the same time, only a small amount of information.

We are not working in the classical expert system frame because:

- the operator modelling and the extraction strategies are following these cognitive principles. The human expert uses complex strategies on a limited knowledge amount whereas expert systems use well-known algorithms on a huge amount of information;
- the operator is always "in the loop", even for the strategies convergence to a pertinent set of process control rules;
- the protocol and the algorithmic techniques are specific to the incremental and iterative aspect underlying the maintenance phase of a process life cycle.

To represent the expert's decision space, we have chosen to use a geometrical paradigm which is now described.

3 Representation space

Experts are taking their decisions among a. set X of n parameters A', together with their related value x_i. We call the tuple $x = (x_1, \ldots, x_i, \ldots, x_n)$ a *parameters setting.* Each parameter X_i has its values domain V_i which can correspond to nominal as well as numerical, discrete or continuous, values set.

On the base of this parameters set ting, an expert has to take a decision, that is to say he/she has to answer the question: "What do I have to modify on the process parameters setting to obtain a good quality for my product (which is defined by the product parameters setting) ?". His/her answer consists of assigning a value d to the control parameter D, called *decision outcome* and the couple *rs — (x,d)* is called a *control situation.*

We work in an n-dimensional space, each point of which corresponds to a parameters setting. The control situations, i.e. parameters settings which have already been examined by the expert, are labelled with the value d of the decision outcome D.

As experts[1] decisions are taken on a restricted subset of the available parameters that describe the process and/or the product to be manufactured, a process control rule R will have, as premises, a conjunction of a few number p of attributes aspects A_j, i.e. subsets of parameters values domain ($A_j \subset V_j$). For example a rule R can be:

$$R: \text{ if } (x_\alpha \in A_\alpha) \wedge (x_\beta \in A_\beta) \text{ then } (D = d)$$

This means that the attributes not appearing in these preconditions can take any value without influencing the current decision process. In an n-dimensional space, a rule R can then be seen as a *cylinder W* for which the *dimensions* are all free except those corresponding to the restricted aspects of R's preconditions. This cylinder is labelled with the value of R's decision outcome. Figure 2 shows, in the 3-dimensional space $X = X_1 \times X_2 \times X_3$, the cylinder corresponding to the rule

$$R: \text{ if } (x_1 \in A_1) \text{ and } (x_3 \in A_3) \text{ then}(D = d)$$

The dimension corresponding to X_2 is free, every value of X_2 in V_2 is valid.

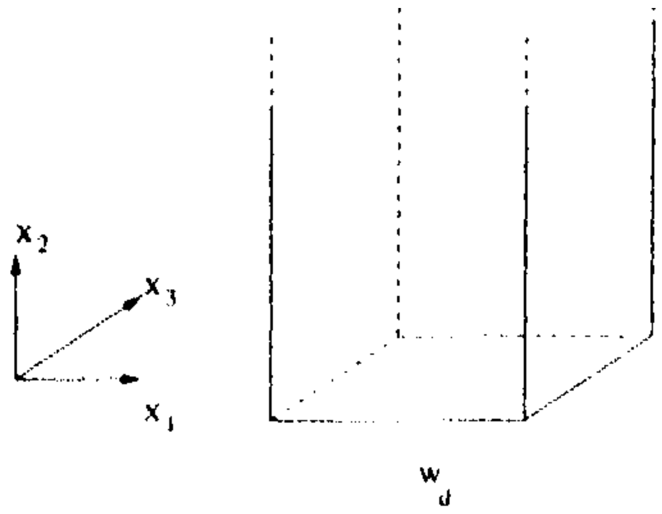

Figure 2: Representation of a cylinder

The *cylinder base D(W)* is the hypercube defined by the rule *R* preconditions:

$$B(W) = \{(X_j, A_j)/A_j \subset V_j\}$$

It has a *dimension d(W)* corresponding to the number of preconditions $d(W) — |B(W)|$. A set of cylinders is called a *paving P.*

4 COMAPS algorithm

4.1 Three distinct phases

The COMAPS tool has been divided into 3 distinct parts which correspond more or less to the 3 phases of the process life cycle:

- the *learning phase* allows the extraction of an initial cylinders set P_0 from a learning data set - a set of control situations that have already been examined by the expert - , called history and denoted H, using more standard but adapted machine learning - and especially decision tree* learning - techniques;
- the *maintenance phase* takes the result 'P_0 of the learning phase as well as // as inputs and it consists of an on-line update of the paving *P* according to new incoming control situations. This phase is the one we will further develop in this paper;
- a *conflict solving phase* is called when no acceptable - for the expert or for the maintenance phase algorithm - modification is found to correctly update the current paving. This kind of conflict mostly appears when important technological changes are[1] observed on the process. For more details concerning this phase, see [Saunier, 1998 .

Whereas the learning phase consists of an off-line process, the maintenance phase has to work at least at the pace the expert does: the algorithm is a kind of machine learning algorithm under cognitive constraints. But maintaining a paving consistent with the history to which one adds new control situations leads to face several NP-complete problems.

4.2 NP-cornplete problems encountered

P_1: Search for maximal cylinders

Let V1, ..., V_n *n* finite sets and $V = V_1 \times \ldots \times V_n$, D a finite set, $H \subseteq V m\times D$, an integer and $j \in D$.

Is there a cylinder *W,* corresponding to the decision outcome *j* for // (i.e. *W H*-compatible with j), of dimension d, so that $|W| \geq m$?

Searching for such a cylinder *W,* i.e. trying to find a cylinder covering as much control situations as possible, becomes NP-complete since $d \geq 2$.

P_2 Covering by a minimal number of cylinders having at most *d* dimensions

Let V_1, ..., V_n *n* finite sets and $V = V\backslash \times \ldots \times V_n$, *D* a finite set, $H \subseteq V \times D$, *s* an integer, $j \in D$ and $H^J = \{cs = (x,y) \in H/x \in V$ and $y \in D$ with $y = j\}$.

Does it exist a set of *s* cylinders $W\backslash$, ..., W_8 so that $\forall i$, W_i is H-compatible with j, $d(W_i) \leq d$ and $\bigcup_{1 \leq i \leq s} W_i \supseteq W$?

The covering problem by a minimal number of cylinders which dimension does not exceed a given integer *d* is NP-complete since $d \geq 2$.

P_3. Consistency checking problem

Let *d* an integer, $\{W_1, \ldots, W_m)$ a set of cylinders differently labelled and so that $d(W_i) \geq d$.

Do we have $\bigcap_{1 \leq i \leq m} W_i \neq \emptyset$?

If we have cylinders corresponding to different decision outcome values, we will have to verify the consistency of the paving, checking their intersection is empty and this becomes a NP-complete problem since $d \geq 3$.

Most of the work to tune the maintenance phase consists then in finding heuristics to bypass these NP-complete problems.

4.3 Main notions

Covering notion

Because:

- not every point of the space can be a real parameters setting, some values combinations being impossible or known as giving a far too bad result according to the product quality,
- we are searching for maximal cylinders but their base is bounded according to the extreme parameters values of the control situations in //,

the cylinders space paving is not complete and thus, a control situation - i.e. a labelled point - can have several status:

- not covered: the parameters setting doesn't fit with any of the cylinders bases. In any other case, the control situation is covered,
- bad covered: the control situation is covered but the labelling of the covering cylinder(s) is not the same as its own decision outcome value,
- well covered: the control situation is covered and the covering cylinder(s) has (have) the same decision outcome value.

Priority between cylinders

As P3 is NP-complete, we have decided not to guarantee a paving without any intersection. To cope with potential intersection, we have introduced a notion of priority between cylinders. We say that a cylinder W\ has the priority over a cylinder W_2, noted $W_1 W2$, if

$$|cs = (x,d)/cs \in W_1 \cap W_2| > \sigma$$

and if

$$\frac{|cs = (x,d)/cs \in W_1 \cap W_2 \wedge d = label(W_1)|}{|cs = (x,d)/cs \in W_1 \cap W_2 \wedge d = label(W_2)|} > \tau$$

where *a* and r are thresholds either fixed with the tool configuration or adapted dynamically according to the size of *H* and to the proportion of each label in *If.*

Two cylinders W\ and W_2 of different labels can thus intersect without having neither $W_1 \mathcal{P}_r W_2$ nor $W_2 \mathcal{P}_r W_1$ and this can happen if:

- their intersection is empty, i.e. no control situation has already been seen in this area of the space*,
- there are not enough control situations in the intersection according to the threshold a,
- none of the two labels in competition is related to a. number of control situations significantly higher according to the threshold T.

Evaluation of a cylinder accuracy

As we cannot guarantee to generate a paving with cylinders only covering control situations labelled in the same way and without any intersection of differently labelled cylinders, we need to be able to measure the "quality" of the paving and thus the "quality" of a cylinder with regards to H.

The quality of a cylinder, which we call *CF(W),* is estimated by the ratio between the number of control situations it covers well and the number of control situations it covers:

$$CF(W) = \frac{|cs = (x,d)/cs \in W \wedge d = label(W)|}{|cs/cs \in W|}$$

Of course, a cylinder *W* is too weak if *CF(W)* is not at least equal to the proportion of control situations having *d(cs) = label (W).* In the same way as for the priority computation, we have defined a threshold *ft* under which a new cylinder to be integrated to the paving can not be accepted.

Furthermore, the global quality of the paving is defined as the average of the *n* constituting cylinders qualities:

$$CF(\mathcal{P}) = \frac{1}{n}\sum_{i=1}^{n} CF(W_i)$$

As the aim of maintenance phase is to update the current paving, this quality evaluation is also used to determine if a possible modification has to be applied or not. The criterion is then that the quality of the rule after modification is at least as good as the old rule one.

"Wait and see" policy

As the expert's rules are supposed to be stabilised, the algorithm is designed in a way that no action on the current paving is immediately tried. It waits until a sufficient number of control situations have confirmed something has to be modified. The arrival of an inconsistent (i.e. bad or not covered) control situation implies a modification only if there are *enough* control situations with the same decision outcome value in its neighbourhood. This neighbourhood can be defined in several ways according to the inconsistent type. For example, for a not covered control situation *cs* = $((x_1, \dots, x_i, \dots, x_n), d)$, we define it is an hypercube which aspects are $[x_i - \alpha(V_i), x_i + \alpha(V_i)]$, for numerical attributes and *\xi}* for nominal attributes. If the control situation examined appears at the intersection of at least two rules, this neighbourhood is the intersection itself.

5 The maintenance phase

We remind that its aim is to update continuously the cylinders paving, starting with an initial paving P_o which is consistent with the history *H* of the process and taking into account, as they arrive, new control situations the expert user has just added, with regards to the following constraints, highly linked to the cognitive principles which have to be followed:

- remaining consistent with *H* at each step, i.e. modifying the cylinders so that *H* is still correctly described by the updated rules set underlying the cylinders paving,
- keeping a small number of dimensions (at most 4 to be compatible with the expert's short-term memory abilities) in the cylinders base - a cylinder can be based on more than 4 dimensions but it should not be more than a temporary state, corresponding for example to a less common situation for the expert, during which he has to adapt a bit further his/her own rules -,
- keeping a small number of cylinders in the paving.

5.1 Cylinders modification functionalities

Different modification functions have been implemented that can be applied to one or several cylinders of the current paving, according to the status of the control situation to be integrated to *If:*

1. generalisation of an existing cylinder by suppressing one constrained dimension: $d(W^t) = d(W^{t-1}) - 1 \quad - \quad W^{t-1} \subset W^t$
2. generalisation of an existing cylinder by extending the possible values over one dimension. For one *i* so that $A_i \subset V_i$, $A_i^{t-1} \subset A_i^t \quad - \quad W^{t-1} \subset W^t$
3. creation a cylinder by dividing a cylinder into two: $W^{t-1} \rightarrow W_1^t \oplus W_2^t$ with $label(W_1^t) = label(W^{t-1})$ and $label(W_2^t) \neq label(W^{t-1})$

4. creation a new cylinder, according to the current control situation *cs ~ (x,d),* by creating aspects for all the attributes[1] and taking the conjunction of at most four of them to define the new cylinder base, which will have *d* as a label.

5. restriction of a cylinder that is unnecessarily large by adding some constraints on one dimension:
$d(W^t) = d(W^{t-1}) + 1 \quad - \quad W^t \subset W^{t-1}$

6. restriction of a cylinder by restricting the aspect extension on one of its dimension. For one *i* so that
$A_i \subset V_i,$
$A_i^t \subset A_i^{t-1} \quad - \quad W^t \subset W^{t-1}$

7. modification of the priority between the cylinders covering the current control situation by recomputing it.

5.2 Functionalities use

Of course, these modifications are not used in any case.

If the new control situation to be treated is already well covered, the only thing we might try is the first function: as the cylinder is confirmed by the arrival of a well classified example, we can try to see if all its dimensions are needed or if it can be extended without loosing its quality (using function (I.).

For a control situation currently not covered, the algorithm will first try to integrate it to an existing cylinder of the same label using function (2.) and, if it doesn't work, to create a new cylinder to cover it using function (4.).

For a bad covered control situation cs, and according to the inconsistency type with regards to the cylinder(s) involved - i.e. considering the "environment": number of cylinders covering *cs,* number of different labels, priority value(s), ...- several of these seven modifications can be tried to improve the situation. But, as the expert rules are supposed to be stabilised, the order in which these functions are applied is strongly dependent on how deep the resulting modification is for the paving.

Indeed, according to the cognitive model, cylinders must stay large - i.e. with not too constrained bases, which corresponds to short rule premises conjunction - and not too numerous.

One of the criteria involved while sorting the functions is that they imply local or more global modifications: function (7.) - priority modification - is clearly the one that leads to the most, local modification whereas function (4.) - cylinder creation - leads to far deeper changes so it is the last one tested.

Function (1.) is not sorted with the other ones: it, is a bit different, because it is applied only in case the current control situation is well covered by a too restricted, and thus not satisfying, cylinder - i.e. corresponding to a rule for which there are more than four aspects in the premise -.

[1]We don't describe hero the way these aspects are computed.

As the modifications implied by function (2.) - extension of an aspect values - remain local, it is the second the algorithm tests. Then, the third alternative is function (3.) - cylinder division -: of course it creates a new cylinder but in an area that is already covered by a more general cylinder.

Finally, functions (5.) and (6.) are tried, in this order, because they both imply smaller cylinders - i.e. more constrained premises for the rules -, which contradicts one of the cognitive principles.

6 Comparison with previous works in this domain

This kind of approach, using Bounded Rationality and MBH principles as underlying cognitive principles to an anthropocentric algorithm, has already been led and validated in [Barthelemy *et* a/., 1995] and [Laurent *et* a/., 1994], The improvements we've brought for COM APS are of three kinds:

- we don't assume anymore any hypothesis on the order of the attributes modalities: the previous works were turning this order, which corresponds to the decision maker's attractiveness scales, to advantage because the representation space was derived from Galois' lattices, but these scales are difficult to obtain without using verbalisation techniques and they are different from an expert to another;
- the formerly defined methods were rapidly limited by the number of attributes taken into account for the computation. As for real industrial processes we often have to deal with more than 20 parameters, it was a real need to find a way to push back this limit, especially because we can not afford long computation time for an on-line control;
- the expert is observed in his/her real decision making process: after the initial set of rules has been extracted off-line, every new control situation is given a decision outcome value on-line by the expert and is immediately treated by the algorithm. The former developments were using dynamically computed questionnaires to ask the expert its decision in a given situation but, this was not his/her normal task, and if something was changing in his/her decision rules, it could be modified in the computed rules only by starting again to answer a new set of questions.

7 Current results and work perspectives

In the frame of this european project, we have the possibility to test our program on three real industrial processes, in the fields of copper foil production, printed circuit boards manufacturing and brake pads production.

7.1 Current results

Real data being confidential, we've been testing the algorithm through a mock up with coded data. The first tests have been led without the expert, dividing the coded data set into a training set for the learning phase and an incoming situations set for the maintenance phase.

With one of the pilot sites data, we had the possibility to test the behaviour of the maintenance phase facing a, known evolution of the process control. Starting from a set of 6 institutional rules and an history of 1086 *cs,* the rules set has been updating according to 556 new e\s.

The results according to the quality of the rules are summarized in Figure 3, the last column corresponding to the type of modification applied according to the list presented in 5.1. All the results have been shown to the expert and they were validated.

Rule	CF Before	CF After	Modification
1	0.41	1	
2	0.84	0.87	
3	0.56	0.87	5.
4	0.57	0.69	6.
5	0.89	0.87	5.
6	0.70	0.88	5. and 6.
7		0.84	4.
8		0.86	4.

Figure 3: Quality of the rules

7.2 Work perspectives

We still have to improve the internal model of priority between cylinders: at this time, it is represented by a boolean matrix where $P(i,j)$ is 1 if $W_i\mathcal{P}_rW_j$ and is 0 otherwise. This doesn't give any information concerning the priority status for an intersection of more than two cylinders and it's not possible to deduce it looking at the priority between every pair of cylinders involved. On the opposite, this kind of deduction could lead to a non-sense: if we have, for example, three cylinders W_1, W_2 and W_3 w $W_1\mathcal{P}_rW_2$, $W_2\mathcal{P}_rW_3$ d $W_3\mathcal{P}_rW_1$, which of them should have the priority ?

In addition, we wait for a prototype, now under development, to be installed (benchmarking in April this year) to be able to make some more tests, but this time in a real decision context and not only with a *posteriori* validation.

Some comparisons between classical machine learning tools, and especially decision tree learning tools like 1D3 [Quinlan, 1986] and C4.5 [Quinlan, 1993] are also under way.

References

[Adelbratt and Montgomery, 1980] T. Adelbratt and H. Montgomery. Attractiveness of decision rules. *Acta Psychologies* (45):177-185, 1980.

[Aschenbrenner and Kasubek, 1978] K.M. Aschenbrenner and W. Kasubek. Challenging the rusting syndrome: multi-attribute evaluation of cortison drugs. *Organizational Behaviour and Human Performances,* (22):216 234, 1978.

[Barthelemy and Mullet, 1992] J.-P. Barthelemy and E. Mullet. A model of selection by aspect. *Acta Psychologies,* (79):1 19, 1992.

[Barthelemy *et al,* 1995] J.-P. Barthelemy, G. Coppin, and F. Guillet. Smelting process control: from experimental design to acquisition of expertise. In *International Conference on Industrial Engineering and Production Management,* volume 2, pages 2 11, March 1995.

[de Hoog and van der Wittenboer, 1986] R. de Hoog and G. van der Wittenboer. Decision justification, information structure and the choice of decision rules. In B. Brehiner, H. Jungermann, P. Lourens, and G. Sevon, editors, *New direction in research on decision making,* North Holland, 1986.

[Huber, 1986] 0. Huber. Decision making as a problem solving process. In B. Brehmcr, H. Jungermann, P. Lourens, and G. Sevon, editors, *New directions in research on decision making,* 1986.

Johnson and Payne, 1985] E.J. Johnson and J.W. Payne. Effort and accuracy in choice. *Management Science,* (31):295 314, 1985.

Laurent *et al,* 1994] S. Laurent, E. Pichon, and R. Bisdorff. Solving by resolving: cognitive solving approach to finite decision problems in chip. *First European Conference on Cognitive Science in Industry,* pages 53 66, 1994.

[Montgomery, 1983] 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,* North Holland, 1983.

[Quinlan, 1986] J. R. Quinlan. Induction of decision trees. *Machine Learning,* 1(1):81 106, 1986.

[Quinlan, 1993] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA, 1993.

Ranyard and Crozier, 1983]
R. Ranyard and 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,* North Holland, 1983.'

[Saunier, P. Saunier. Comaps: check as you decide paradigm. In *Seventeenth European Conference on Human Decision Making and Manual Control,* December 1998.

[Simon, 1979] H.A. Simon. *Models of thought.* Yale University Press, New Haven, 1979.

[Svenson, 1979] O. Svenson. Process description of decision making. *Organizational behaviour and human performance,* (23):86-112, 1979.