

## SELECTION DE CAPTEURS POUR SYSTEMES DE DETECTION DECENTRALISEE PAR UN ALGORITHME D'APPRENTISSAGE BASE SUR L'ENTROPIE

Marie-Paule CARTON(\*), Denis POMORSKI(\*\*), Marcel STAROSWIECKI(\*\*)

\* ISEN, 41Bd Vauban 59046 Lille Cédex, France

\*\* LAIL - URA CNRS 1440D, Cité Scientifique, Bât. P2  
59655 Villeneuve d'Ascq Cédex, France

### RÉSUMÉ

Le rôle d'un système de détection décentralisée est de faire collaborer un ensemble de capteurs dans un but commun de détection. La mise en oeuvre de cette collaboration se fait le plus souvent suivant une architecture imposée. L'objet de ce travail est de doter les systèmes de détection décentralisée de la capacité de sélectionner au préalable les capteurs apportant réellement une information dans le processus de décision. Nous introduisons ainsi une phase d'apprentissage basée sur un critère entropique.

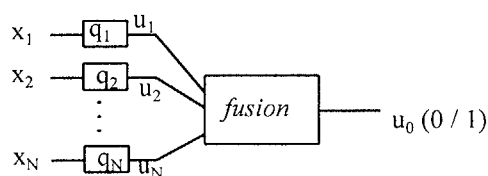
### 1. INTRODUCTION

Tout système de détection décentralisée comprend un ensemble de  $N$  capteurs ainsi qu'un ensemble de  $N$  opérateurs de traitement local transmettant chacun un résumé de leur observation à un opérateur central de fusion (*fig. 1*). Ce dernier élabore la décision d'ensemble. Le problème général de la détection décentralisée concerne l'optimisation de ces systèmes, qui mettent en commun des données collectées par plusieurs capteurs. Un grand nombre de travaux ont été publiés sur ce sujet, s'attachant à déterminer les opérateurs de traitement qui optimisent les performances de détection par rapport à un critère donné [1], [2], [3]. La plupart de ces études ont été développées dans le cadre d'une structure imposée, où les capteurs considérés sont déterminés à l'avance. Une telle structure signifie implicitement qu'un choix des sources d'information à fusionner a été établi au préalable et qu'il ne peut être remis en cause. Or cette démarche est parfois mal adaptée aux conditions pratiques de fonctionnement des systèmes. Considérons par exemple le contexte de la surveillance industrielle, où un grand nombre de capteurs observant des grandeurs physiques différentes sont disponibles. Il est possible qu'il soit plus intéressant, en termes de volumes de calculs et de tests, de ne considérer qu'un sous-ensemble de capteurs, plutôt que l'ensemble des informations disponibles, qui peuvent présenter des redondances, dont certaines peuvent dégrader les performances de l'ensemble, ou noyer le système sous un flot d'informations

### ABSTRACT

The aim of decentralized detection system is to make collaborate a set of sensors. This collaboration is often used with an imposed architecture. We propose to selection the sensors which are important for the detection problem. We introduce thus a learning procedure based on an entropy criterion.

trop coûteux à gérer. C'est pour cette raison que nous proposons d'introduire dans le fonctionnement des systèmes de détection décentralisée une étape de configuration. En d'autres termes nous proposons de doter les systèmes de détection décentralisée de la capacité de sélectionner, parmi tous les capteurs disponibles, ceux qu'il est intéressant de faire intervenir dans le processus de décision (les capteurs apportant beaucoup d'information). Nous introduisons une phase d'apprentissage dans laquelle nous utilisons un critère entropique tout à fait adapté à ce problème de sélection.



*fig. 1*

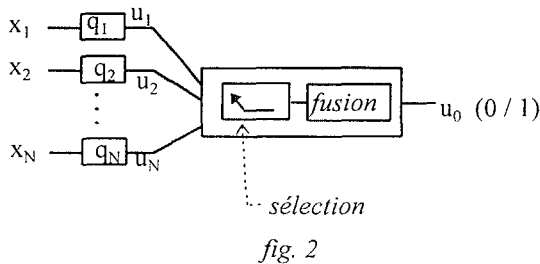
### 2. DESCRIPTION DU PROBLEME

Les systèmes que nous étudions ont pour but de résoudre un problème de détection, qui se traduit par le choix entre deux hypothèses  $H_0$  et  $H_1$  à partir d'observations collectées sur un ensemble de  $N$  capteurs. Chaque source délivre un vecteur  $x_i$   $i=1, \dots, N$ . L'observation globale  $X$  est la concaténation de tous ces vecteurs, les lois de probabilité conditionnelles de la variable aléatoire  $X$ ,  $P_0(X)=P(X/H_0)$  et  $P_1(X)=P(X/H_1)$  sont



connues, ainsi que les probabilités a priori  $P(H_0)$  et  $P(H_1)$ . Les systèmes comprennent des processeurs locaux notés  $q_i$  (fig.1) qui élaborent chacun, à partir d'une observation  $x_i$ , un résumé local  $u_i=q_i(x_i)$  prenant ses valeurs dans un alphabet fini. Les processeurs  $q_i$  seront donc des quantificateurs vectoriels. L'opérateur de fusion  $F$  construit la décision globale  $u_0=0,1$  en combinant les résumés locaux  $u_0=F(u_1, \dots, u_N)$ . L'ensemble regroupant la loi de fusion et les opérateurs de traitement local est noté  $G=\{q_1, \dots, q_N, F\}$ . La démarche habituelle de conception d'un système de détection décentralisée consiste à optimiser les éléments de  $G$ . Il s'agit donc d'un problème d'optimisation d'une structure déterminée à l'avance et nécessairement considérée dans son ensemble.

Notre approche consiste à déterminer le sous-ensemble de capteurs, issu de l'ensemble de tous les capteurs disponibles, qu'il est intéressant de prendre en compte dans le processus de décision. En d'autres termes, nous considérons qu'un ensemble de capteurs est disponible, que les quantificateurs vectoriels associés à chacun de ces capteurs sont déterminés. Nous remplaçons l'opérateur central de fusion par un opérateur de « sélection-fusion » (fig.2) qui agit en deux étapes. La première étape dite « de sélection » identifie le sous-ensemble de variables à fusionner. La sélection étant faite, l'opérateur central reprend un fonctionnement de fusion-décision au sens habituel.



Notre approche est donc différente de la démarche habituelle :

- nous n'imposons pas la structure du système ; nous nous attachons à déterminer certains de ses constituants.
- les capteurs étant sélectionnés, nous ne cherchons pas à optimiser les opérateurs (quantificateurs locaux ou fusion centrale) qui traiteront les données.

Cependant il est intéressant de rappeler les enseignements mis à jour par les études menées sur l'optimisation des opérateurs de traitement d'une structure décentralisée, qui sont des guides dans le choix de ces opérateurs. Tout d'abord, que la démarche d'optimisation porte sur les opérateurs locaux indépendamment de l'opérateur de fusion, ou à l'inverse uniquement sur l'opérateur de fusion indépendamment des traitements locaux, ou enfin sur les deux niveaux simultanément, il a été montré que le rapport de vraisemblance joue un rôle prépondérant. Il en découle en particulier que la structure optimale de quantification locale est la quantification par rapport de vraisemblance. Enfin il est à noter que l'optimisation des systèmes de détection décentralisée n'est accessible qu'au travers d'hypothèses

simplificatrices, telle que l'indépendance des observations. En conséquence si cette dernière hypothèse n'est pas valable, il est tout à fait souhaitable de ne retenir que l'information réellement utile pour notre problème de décision. Cette dernière remarque montre l'importance que peut avoir une démarche d'apprentissage comme la nôtre.

### 3. SELECTION DE CAPTEURS

#### 3.1. LES DONNEES

La phase d'apprentissage que nous nous proposons de définir repose sur la connaissance des variables  $u_i$  ( $i=1, \dots, N$ ), et de la conclusion correspondante  $H_v$  (hypothèse vraie), constituée de deux modalités  $H_0$  et  $H_1$ .  $P(X/H_j)$ ,  $P(H_j)$  et  $u_i=q_i(x_i)$  ( $j=0,1 ; i=1, \dots, N$ ) étant connues, le tableau de contingence suivant peut être défini :

	$M_{Hv}$	$H_0$	$H_1$	
$M_U$				
$\alpha_1$		$P_{i0}$	$P_{i1}$	$P_{i.}$
$\alpha_2$				
...				
$\alpha_i$				
...				
$\alpha_n$				
		$P_{.0}$	$P_{.1}$	

fig.3 : Tableau de contingence  $[P_{ij}]$

où :

- $P_{ij} := \{p_{ij} ; i=1, \dots, n ; j=0,1\}$
- $U = \{u_1, u_2, \dots, u_N\}$
- $M_U$  est l'ensemble des modalités observées de l'ensemble  $U$  ( $n = \text{card}(M_U)$ ).
- $M_{Hv}$  représente les conclusions possibles :  $M_{Hv} = \{H_0, H_1\}$
- $p_{ij}$  est la probabilité d'occurrence conjointe des modalités  $\alpha_i$  et  $H_j$ .

$$- p_{i.} = p_{i0} + p_{i1} \text{ et } p_{.j} = \sum_{i=1}^n p_{ij}$$

#### 3.2. APPRENTISSAGE ET ENTROPIE

Le rôle de l'opérateur de sélection est de déterminer la configuration qui permettra de tirer le meilleur parti de toutes les informations dont dispose le système. L'approche que nous avons suivie pour concevoir un tel opérateur se place dans le cadre de l'apprentissage. A la mise en route du système, l'algorithme de sélection dispose d'un modèle probabiliste des variables  $u_i$ . Le critère retenu pour guider cet apprentissage est tout naturellement la minimisation de l'entropie conditionnelle ([4], [5]), mesurant l'intensité de liaison entre les deux variables  $U$  et  $H_v$  :

$$H(H_v / U) = H(U, H_v) - H(U)$$

$$= - \sum_{i=1}^n (p_{i0} \cdot \log p_{i0} / i + p_{i1} \cdot \log p_{i1} / i)$$

Pourtant, des modèles plus simples peuvent être obtenus en ne considérant qu'un sous-ensemble  $S$  des variables  $U$ . Basés sur  $H(H_v/U)$  et  $H(H_v/S)$ , deux indices de « modélisabilité » sont utilisés :

\*  $m(H_v / U) = 1 - \frac{H(H_v / U)}{H(H_v)}$  mesure la modélisabilité de  $H_v$  par  $U$ . [4]

\*  $q(H_v / S) = \frac{H(H_v) - H(H_v / S)}{H(H_v) - H(H_v / U)}$  mesure la

qualité du modèle plus simple  $H_v = \tilde{f}(S)$  par rapport à la qualité du modèle  $H_v = f(U)$ .

### 3.3. UNE METHODE D'INDUCTION PAR ARBRE DE DECISION

Le critère étant établi (minimisation de l'entropie conditionnelle), il nous reste à déterminer une méthode d'apprentissage nous permettant de déterminer  $S$ , le sous-ensemble optimal de  $U$ .

Une méthode exhaustive de recherche de  $S$  n'est pas envisageable lorsque le nombre de capteurs et le nombre de modalités des paramètres  $u_i$  ( $i=1, \dots, N$ ) sont importants. C'est pour cette raison que des heuristiques ont été développés, notamment des méthodes d'induction par arbre de décision, dont la plus connue est certainement *ID3* [7].

Cet algorithme calcule les entropies conditionnelles  $H(H_v/u_i)$  (ou  $\bar{I}(u_i;H_v)$  [4]) et choisit la variable qui, indépendamment des autres, minimise cet indice (ou maximise  $\bar{I}(u_i;H_v)$ ). Cette variable est ainsi celle qui apporte le plus d'information sur  $H_v$ . *ID3* construit ainsi un arbre pour lequel chaque branche représente un test sur une modalité de la variable en question.

Puis, si la variable  $H_v$  n'est pas discriminée, alors, pour chaque noeud non discriminant, on recherche à nouveau la variable, parmi celles restantes, qui minimise le critère... et ainsi de suite ; jusqu'à ce que tous les noeuds soient discriminés, ou jusqu'à ce qu'il n'y ait plus de variables disponibles.

Cependant, on sait que  $gain(u_1, u_2, \dots, u_N) \neq \sum_{i=1}^N gain(u_i)$

(où la fonction « gain » est la quantité d'information apportée par la connaissance de la variable en question). Dans ce sens, il peut arriver que le pouvoir explicatif de chaque variable d'un ensemble  $S$  soit faible (elles ne seront alors pas retenues comme variables pertinentes pour notre problème de détection) alors que le pouvoir explicatif de l'ensemble  $S$  considéré globalement est important.

C'est pour cette raison qu'à chaque niveau de l'arbre, nous traitons le problème en tenant compte des variables déjà testées précédemment. La variable retenue est alors celle qui, ajoutée

aux variables déjà testées, minimise le critère choisi (et ceci de façon multidimensionnelle) [6].

L'approche agrégative est la démarche la plus *naturelle* pour résoudre ce problème, car intuitivement on commence par chercher la variable que l'on va tester en premier : on cherchera la variable qui *discrimine* le plus la variable  $H_v$ . Puis, on cherche la variable, parmi celles restantes, qui discrimine le plus les noeuds ainsi créés, tout en tenant compte des tests précédents. Et ainsi de suite, jusqu'à ce qu'il n'y ait plus de variable à tester, ou que la variable  $H_v$  soit parfaitement discriminée.

A la différence d'*ID3* (qui traite le problème variable par variable), nous traitons le problème de façon multidimensionnelle.

Nous proposons ainsi l'algorithme suivant :

; APPROCHE AGREGATIVE GLOBALE

; initialisation

; boucle

1. pour  $i = 1$  à  $N$  ( $N =$  nombre de variables explicatives)
2. calculer les  $H(H_v/u_i)$
- fin pour
3. retenir  $u_i$  qui minimise  $H(H_v/u_i)$
4.  $S \leftarrow (u_i)$

; boucle

5. répéter
6. calculer les  $H(H_v/S \wedge u_i)$  avec  $u_i \notin S$
7. retenir  $u_i$  qui minimise  $H(H_v/S \wedge u_i)$
8.  $S \leftarrow S \times (u_i)$
- jusqu'à (discrimination totale) ou ( $\text{card } S = N$ )
9. tracer l'arbre de décision correspondant aux questions dans l'ordre de  $S$

; FIN de l'algorithme.

avec  $u_i \in U$  et  $S \in \mathcal{G}(U)$  (où  $\mathcal{G}(U)$  est l'ensemble des sous-ensembles de  $U$ ).

### 4. UNE APPLICATION

A titre d'exemple, nous avons modélisé un système de suspension représenté figure 4 :

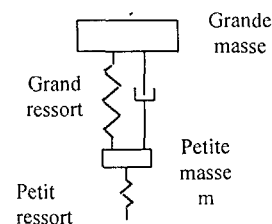


fig. 4

Ce système est soumis à une série de créneaux (simulation de bordures). Les grandeurs relevées sur ce système sont les vitesses des différents éléments :

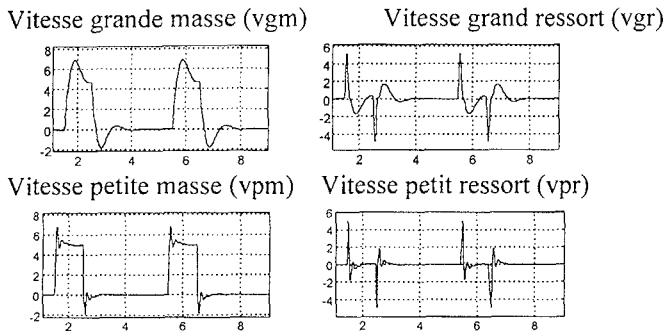


fig. 5

Nous quantifions celles-ci en nous limitant à dix modalités pour chacune d'elles, et nous cherchons à expliquer la variable « bordure ».

L'algorithme proposé nous indique que la variable apportant le plus d'information sur la variable « bordure » est *vpm* (entropie conditionnelle de 0.242 bits avec un pourcentage d'explication de 74.2%), puis *vgm*, avec 52.2% (les autres variables n'apportant que très peu d'information). La variable retenue est donc *vpm*. La courbe COR (Caractéristique Opérationnelle du Récepteur) ci-dessous nous confirme bien ce résultat.

Courbes COR locales

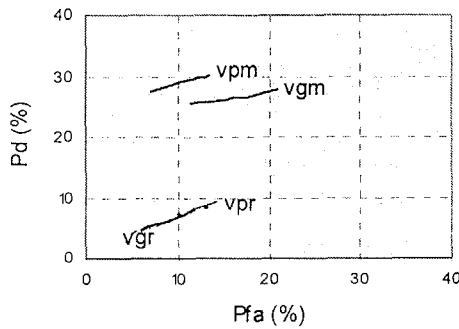


fig. 6

A la deuxième itération de l'algorithme, la variable qui, associée à *vpm*, apporte le plus d'information est *vpr* (explication de 93.9%). On préférera alors le couple (*vpm,vpr*) à tout autre couple, moins explicatif. Les courbes COR nous confirment encore ce résultat.

Courbes COR locales

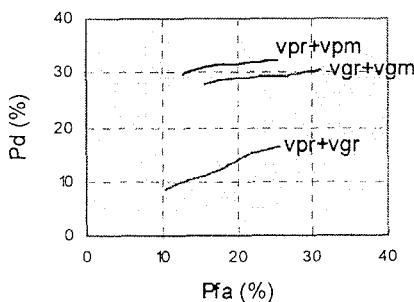


fig.7

A la troisième itération, le triplet retenu est (*vpm,vpr,vgr*) avec 100% d'explication, c'est-à-dire qu'avec trois variables (en omettant *vgm* qui est une variable redondante), on explique parfaitement la variable « bordure ».

5. CONCLUSION ET PERSPECTIVES

Le fait de doter les systèmes de détection décentralisée de la capacité de sélectionner au préalable les capteurs nous permet de ne conserver que l'information utile dans une optique de décision ultérieure. Un algorithme a été proposé dans cette contribution ; d'autres algorithmes ont été développés [6]. Des exemples plus complexes sont actuellement en cours de traitement.

De plus, nous pouvons remarquer qu'une variable apportant beaucoup d'information est prépondérante pour la décision ; il faudra par conséquent la prendre en compte très fortement lors de la fusion. Par contre, une variable secondaire devra être prise en compte de façon moins significative. Cette remarque nous permettrait de mettre en place des contraintes relatives à la loi de fusion.

6. REFERENCES

[1] R. Tenney and J. Sandell, « Detection with distributed sensors », *IEEE Transactions on Aerospace and Electronic Systems*, AES-17, pp. 501-510,1981.  
 [2] R. Reibman and L. Nolte, « Optimal detection and performance of distributed sensor systems », *IEEE Transactions on Aerospace and Electronic Systems*, AES-23, pp. 24-30, 1987.  
 [3] M.P. Carton-Boyer, « Détection et estimation réparties, approches par la quantification et le filtrage de Volterra », Thèse de doctorat, Université de Paris Sud, 1992.  
 [4] V.M. Toro Cordoba, « Contribution à l'analyse structurale des systèmes complexes à l'aide de l'entropie et ses généralisations », Thèse de 3ème cycle, USTLFA, 8 mars 1982.  
 [5] D. Pomorski, « Apprentissage automatique symbolique/numérique - Construction et évaluation d'un ensemble de règles à partir des données », Thèse de doctorat, USTLFA, 6 décembre 1991.  
 [6] P.B. Perche, D. Pomorski, M. Staroswiecki, « Méthodes d'induction par arbre de décision utilisant un critère entropique », Deuxième Conférence Internationale sur l'Automatisation Industrielle, 7-9 juin 1995, CRAN, Nancy.  
 [7] J.R. Quinlan, « Induction of decision trees » ; *Machine Learning Journal*, 1, 1986, pp.81-106.