

Modélisation et intégration de connaissances métier pour l'identification de défauts par règles linguistiques floues

Expert knowledge's modeling and expert knowledge's
integration for defect identification by fuzzy linguistic rules

Vincent Bombardier, Pascal Lhoste et Cyril Mazaud

Centre de Recherche en Automatique de Nancy, CNRS, UMR 7039, Faculté des Sciences
Bd des Aiguillettes – BP 239 – 54506 Vandoeuvre les Nancy Cedex
Vincent.Bombardier@cran.uhp-nancy.fr, Pascal.Lhoste@cran.uhp-nancy.fr,
Cyril.mazaud@cran.uhp-nancy.fr

Manuscrit reçu le 6 janvier 2004

Résumé et mots clés

Cet article propose une démarche d'intégration de connaissances pour l'amélioration d'un système de reconnaissance de défauts par vision sur des planches de bois.

Nous situons le problème de vision qui est à la base de cette étude, puis nous explicitons les connaissances métier nécessaires, aussi bien dans le domaine du métier du bois que dans le domaine de la vision.

Nous utilisons pour cela un modèle symbolique basé sur la méthode NIAM/ORM, formalisant ces connaissances métier à partir de leur expression en langage naturel.

Puis nous présentons la façon dont nous exploitons ces connaissances métier pour générer les nœuds d'une structure en arborescence pour l'identification des défauts des planches de bois. Chacun des nœuds consiste en un moteur d'inférence à base de règles linguistiques floues. Les résultats obtenus prouvent l'intérêt de cette démarche.

Identification de défauts, logique floue, règles linguistiques floues, méthode N.I.A.M., O.R.M., modélisation de connaissances métiers.

Abstract and key words

This article proposes a principle of knowledge's integration for the improvement of a system of defects recognition by vision on wooden boards.

We locate the problem of vision which is at the base of this study, then we clarify the expert knowledge, as well in the field of the wood profession as in the field of vision. We use for that a symbolic model based on the NIAM/ORM method, formalizing the expert knowledge expressed in natural language.

Then we present the way we exploit these expert knowledge to generate nodes of an arborescent structure for the defects identification of wooden boards. Each node represents an engine of inference based on fuzzy linguistic rules. The results obtained prove the interest of this principle.

Defect identification, fuzzy logic, fuzzy linguistic rules, N.I.A.M. method, O.R.M., modelling of expert knowledge.

1. Introduction

L'accroissement du niveau d'Information dont on peut disposer sur un système est essentiel pour l'amélioration de la maîtrise de ce système et des processus qui y sont intégrés (automatisation, maintenance, relations avec l'environnement, ...). Deux sources principales permettent l'accès à cette Information : d'une part, à partir de *connaissances* d'experts humains (experts en procédés, en produits, en normes, ...) qui donnent une information plutôt qualitative sur ce qu'est le système étudié, et, d'autre part, par acquisition de *données* directement sur le système, donnant plutôt une information quantitative sous la forme de mesures. Le rapprochement de ces sources d'informations aussi différentes que complémentaires doit alors contribuer à une modélisation de l'information plus complète et plus cohérente permettant une intégration plus forte des processus composant le système. En ce sens, l'étude proposée cherche à définir une méthodologie de modélisation et d'intégration permettant, d'une part, de transformer la Connaissance en Information, d'autre part, d'extraire l'Information à partir de Données Numériques prélevées sur le système et, enfin, d'intégrer ces Informations d'origines différentes en un seul référentiel d'Information Système. Cette méthodologie est appliquée au domaine de la reconnaissance de formes et, en particulier, à celui de l'identification de défauts sur des planches en bois par système de vision industriel.

1.1. Système de vision

Les travaux présentés se situent dans le cadre d'une collaboration université – entreprise entre le CRAN et la société luxembourgeoise Luxscan Technologies et concernent l'élaboration d'un système d'identification par vision de défauts sur des planches de bois. C'est sur la ligne de production que s'effectue, en temps réel, l'analyse des planches dont la vitesse de

déplacement peut atteindre 5 à 7 mètres par seconde en considérant un maximum de 200 défauts par mètre. La figure 1 résume le principe du système de vision.

L'acquisition de l'image de la planche de bois s'effectue par des capteurs linéaires qui peuvent être utilisés individuellement ou combinés. Ils sont principalement de deux types :

- couleur fournissant les canaux rouge, vert et bleu,
- lasers qui renvoient une mesure tridimensionnelle de la surface du bois, une mesure de densité et d'orientation des fibres du bois, et deux mesures de luminosité en rouge et infrarouge.

Les signaux, délivrés par ces capteurs, sont échantillonnés au rythme de 2000 lignes par seconde et quantifiés sur 1024 niveaux. Les images ainsi obtenues sont ensuite pré-traitées et « labellisées » pour fournir une information ne comportant plus que six classes de pixels. Ces classes définissent la classe de pixels « noir », « brun clair » (correspondant au bois sain), « brun foncé », « orange », « bleu », et « fentes ».

Cette étape de classification est réalisée soit par un classificateur bayésien, soit par un classificateur à seuil dont les paramètres sont réglés empiriquement. La segmentation, réalisée à partir de l'image labellisée, consiste en l'agrégation de pixels connexes n'appartenant pas à la classe de pixels « brun clair » définissant le bois sain. Elle permet donc d'extraire les régions dites « défectueuses » des régions dites « saines ».

La détection ou la non détection d'un défaut incombe entièrement à l'étape de segmentation. Un ensemble de paramètres (surface d'un défaut, orientation, couleur, ...) est calculé sur ces régions « défectueuses » pour fournir un vecteur caractéristique, utilisé par l'étape d'identification. Cette dernière a pour objectif de donner un nom au défaut : « nœud noir », « nœud sain », « nœud cerné », « poche de résine », « bleuissement », « fentes »...

À partir du moment où une région est extraite, on la considère alors comme un défaut. L'objectif de la partie haut niveau est donc d'identifier le défaut et non de détecter la présence ou non d'un défaut. La classe de rejet de notre étape d'identification est

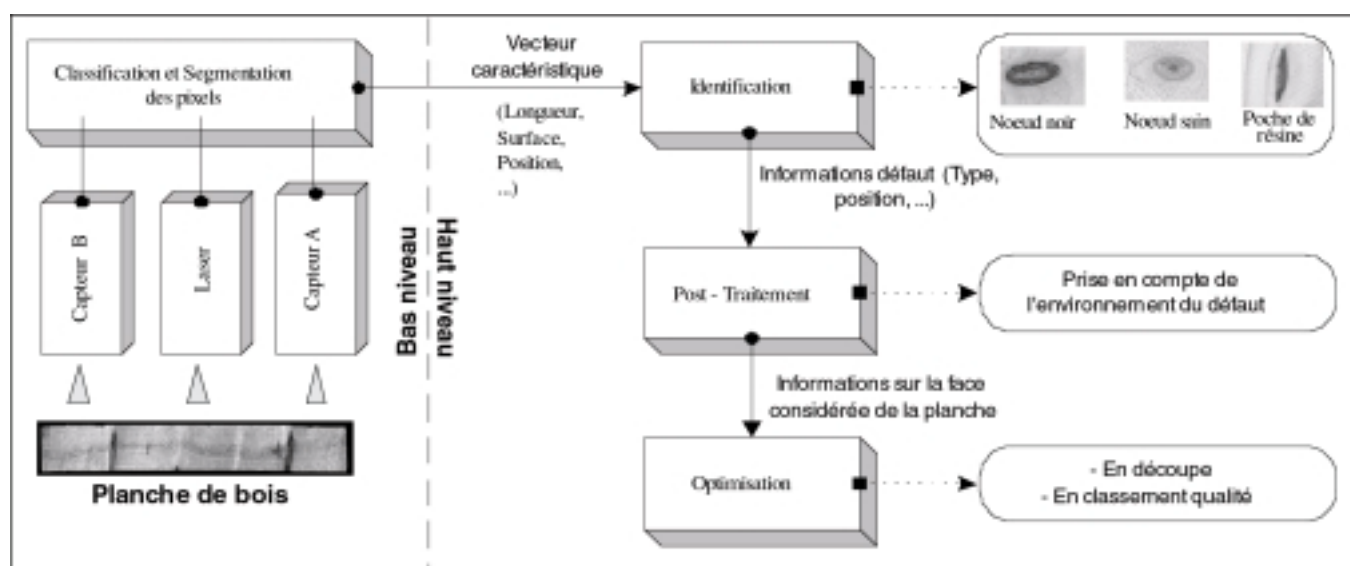


Figure 1. système de vision

alors « Défaut inconnu ». Il est à noter que l'étape d'identification qui ne prend en compte qu'un défaut à la fois n'est pas l'ultime étape du système. En effet, l'étape de post-traitement utilise l'information de chaque défaut, issue de l'étape d'identification pour évaluer l'environnement du défaut considéré (défauts voisins, défauts sur les autres faces, ...). Cette étape de post-traitement intègre des connaissances de plus haut niveau sur la mise en correspondance des défauts entre les différentes faces de la planche. Par exemple, au vu de l'épaisseur de la planche, tel type de défaut sur une face doit être de tel type sur l'autre, ou sur la possibilité de trouver tel défaut à côté de tel autre défaut... Enfin, ces informations cherchant à représenter un modèle « volumique » des défauts sont transmises au module de tri ou d'optimisation de coupe qui sert à piloter respectivement des classeurs à bois ou des scies en fonction de la ligne dans laquelle s'insère le système de vision. Les travaux que nous présentons ne concernent que le module d'« Identification » de la partie haut niveau présenté dans la [figure 1].

1.2. Méthodologie proposée

Dans le domaine de la reconnaissance de défauts sur des planches de bois, la tendance générale réside en l'utilisation d'un système de vision par caméra pour l'acquisition de l'image et d'un classificateur basé sur des réseaux de neurones supervisés ou non supervisés [Kauppinen] [Lampinen] [Pham]. Bien que les résultats cités dans les articles semblent satisfaire les auteurs, nous avons choisi de ne pas utiliser de méthode à base de réseaux de neurones pour notre application. En effet, des réseaux de neurones agissent comme une boîte noire dont il est difficile de connaître précisément l'organisation et le fonctionnement. De plus, avec ce type de méthode, nous ne pouvons pas vérifier si la structure des couches du réseau de neurones, créée lors de l'apprentissage, est en adéquation avec le modèle des connaissances expertes. De plus, ces méthodes neuronales nécessitent des lots d'apprentissage conséquents, ce dont nous ne disposons pas. Dans notre cas, nous désirons connaître le fonctionnement du système afin de pouvoir y intégrer l'information issue des connaissances expertes, et en extraire de l'information pour compléter nos modèles issus des connaissances expertes. C'est donc en partie pour cela que nous avons choisi de ne pas retenir les réseaux de neurones et de nous orienter vers des méthodes plus aisément paramétrables et interprétables comme les systèmes à base de règles.

À ce niveau, l'utilisation d'une méthode mettant en œuvre les outils de la théorie des ensembles flous et plus particulièrement de la logique floue [Bouchon-Meunier], se justifie par trois raisons. Premièrement, les défauts à détecter sont intrinsèquement flous, il n'existe pas toujours une frontière stricte entre un défaut et le bois sain, cette transition étant graduelle. La phase de segmentation qui ne prend pas en compte cette gradualité, fournit des « régions défectueuses » précises. Il en résulte que les caractéristiques calculées sont précises mais incertaines (si on traite

deux fois le même défaut, on n'est pas sûr d'obtenir deux fois le même vecteur caractéristique). L'utilisation de la logique floue vise à rendre le système d'identification moins sensible à ces incertitudes.

Deuxièmement, les classes de défauts en sortie ne sont pas toutes disjointes. En effet, la limite entre un « petit nœud noir » et un « grand nœud noir » n'est pas forcément stricte.

Troisièmement, les termes linguistiques utilisés par les experts pour définir les défauts (petit nœud, grand nœud, à peu près circulaire) sont souvent vagues et imprécis. Nous représentons ces termes sous la forme de termes flous caractérisés par leur fonction d'appartenance.

Dans le domaine des systèmes qui utilisent une expertise humaine et une expertise vision, la tendance est de construire un système de règles heuristiques [Foo] [Wang] [Wen] décrivant le fonctionnement du système décisionnel. Cependant, bien que les jeux de règles créés semblent fonctionner, aucune précision n'est donnée quant à leur validation par rapport aux connaissances émises par les experts du domaine concerné. Ce sont donc *a priori* des processus informels qui permettent cette validation.

Afin de pallier ce problème, nous proposons dans notre méthodologie une approche de validation formelle basée sur l'utilisation de la méthode NIAM [Habrias]. Celle-ci permet, en effet, de transcrire le modèle, traduisant l'information issue de la connaissance de l'expert, en langage naturel compréhensible par l'expert, qui peut ainsi, à travers cette transcription [Attiogbé], valider lui-même le modèle créé.

Pour nos travaux, deux orientations ont donc été retenues pour rapprocher les informations exploitables en identification de défauts sur des planches de bois par vision : celles provenant du système de vision et celles provenant de l'expert « métier du bois ».

D'une part, nous disposons de données numériques sous la forme d'un vecteur qui caractérise le défaut à identifier au travers d'une relation que nous appelons dans la suite de l'article « modèle numérique ». Nous avons choisi d'implémenter ce modèle numérique à l'aide d'un système de règles linguistiques floues [Dubois 3] afin de faciliter l'interprétation du mécanisme d'identification et de pouvoir en extraire l'information portée par le vecteur caractéristique et la formaliser. Pour obtenir le modèle numérique, nous utilisons la méthode proposée par [Nozaki] [Ishibuchi 1] que nous présentons dans le paragraphe 3.2.

D'autre part, nous avons la connaissance des défauts à identifier qui est exprimée en langage naturel en des termes métier. Nous avons entrepris une modélisation symbolique des informations portées par la connaissance métier en utilisant la méthode NIAM. Dans la suite de l'article, nous appelons « modèle symbolique » cette modélisation de l'information issue de l'expert « métier du bois ».

L'objectif de la méthodologie proposée, illustrée par la [figure 2], est alors de faire collaborer les trois modèles.

La première idée est d'obtenir un modèle numérique interprétable afin de pouvoir éventuellement le compléter grâce au

modèle symbolique par des informations non présentes dans les données et ainsi pouvoir, par exemple, modéliser des défauts dont il n'existe pas d'échantillons.

La deuxième idée est de valider le modèle numérique, en comparant les règles générées par l'apprentissage aux connaissances formalisées dans le modèle symbolique. Une différence peut signifier que le lot d'apprentissage n'est pas cohérent avec les défauts que l'on veut reconnaître.

Néanmoins, la mise en correspondance du modèle symbolique et du modèle numérique ne peut se faire directement car, d'un côté, nous avons des termes du métier du bois et, de l'autre, nous avons des termes issus du domaine du traitement de l'image. Il faut donc un modèle intermédiaire chargé de la « traduction » entre le domaine du bois et le domaine du traitement de l'image [Mazaud].

Dans la suite du document, nous utilisons un seul et même langage de formalisation, NIAM, pour présenter le modèle symbolique des défauts exprimés initialement par l'expert métier en langage naturel, ainsi que le modèle symbolique intermédiaire décrivant les défauts d'un point de vue «procédé de vision», et enfin, le modèle numérique qui représente l'identification des défauts.

2. Les modèles symboliques

Le modèle symbolique concerne la formalisation des connaissances expertes dans le domaine du bois ainsi que dans le domaine du procédé d'identification de défauts (système de vision). La méthode d'analyse retenue pour élaborer ce modèle est la méthode NIAM (Nijssen Information Analysis Method) [Habrias]. Nous utilisons un de ses formalismes dérivés, ORM (Object Role Modeling) [Halpin 1] [Halpin 2], pour représenter les résultats de cette analyse. De manière générale, la méthode NIAM distingue le *système objet* représenté par des *LOT* (*Lexical Object Type*), partie de la réalité observable pour laquelle on veut collecter de l'information, et le *système d'abs-*

traction représenté par des *NOLOT* (*NO Lexical Object Type*), modèle symbolique du *système objet*, constitué de classes d'objets, de classes de règles. Un LOT est représenté en formalisme NIAM/ORM par un cercle en trait pointillé alors qu'un NOLOT est représenté par un cercle en trait continu. Une notation particulière dite LOT/NOLOT montrant une superposition d'un cercle en trait continu et d'un cercle en trait pointillé correspond à l'association biunivoque d'un LOT identifiant (dont la désignation apparaît entre parenthèses) à un NOLOT.

La méthode d'analyse présente un intérêt significatif car elle repose directement sur une expression de faits en langage naturel. Le modèle d'informations résultant peut aussi être soumis à validation par l'expert sous une forme compréhensible, c'est-à-dire en Langage Naturel Binaire (LNB), transcription du modèle d'information appelée aussi « paraphrasage » [Attiogbé]. Il est possible de compléter progressivement le modèle en posant à l'expert des questions précises induites par la méthode (par exemple : un défaut possède-t-il une ou plusieurs couleurs) et pouvant ainsi introduire dans le modèle les notions de contraintes d'état ou de transition. Le formalisme NIAM/ORM permet de transcrire directement les phrases élémentaires tirées de l'analyse. Ce formalisme admet aussi des correspondances avec d'autres formalismes tels que les formalismes ensemblistes, entité-relation et U.M.L. (cf. annexe 1). Ce qui importe le plus ici est surtout la méthode permettant de créer le modèle plutôt que le formalisme utilisé pour le décrire. De plus, un modèle NIAM/ORM ainsi obtenu a la capacité d'être dérivé vers des applications informatiques (génération de structures de bases de données, prototypage d'interfaces, ...), et donne la possibilité de générer une base d'informations exploitable à destination des utilisateurs en conformité avec l'expression initiale en langage naturel des connaissances formalisées.

2.1. Le modèle symbolique de l'expert métier du bois

Afin de créer ce modèle, on demande à l'expert du domaine du bois de rédiger une liste, la plus complète possible, en langage naturel, des différents défauts que l'on rencontre sur le type

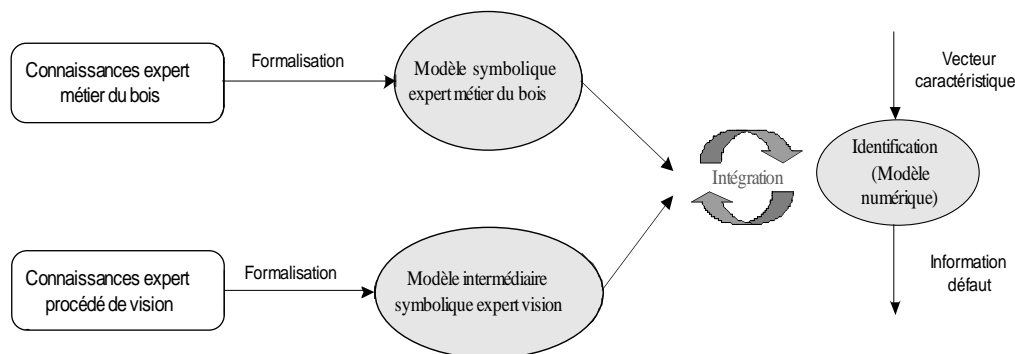


Figure 2. Méthodologie proposée

d'essence manipulée. Il intègre à cette liste des connaissances sur l'environnement et le contexte des défauts (par exemple, un défaut se trouve en général à proximité d'un autre). Cette expression est suffisante pour reconnaître, à l'œil nu, n'importe quel défaut énoncé sur un échantillon de bois.

Voici un exemple qui permettra d'illustrer ces propos. L'expert bois décrit un défaut de la manière suivante :

«La mælle est un défaut de forme allongée, généralement au centre de la planche. Elle suit la fibre du bois et est de couleur brune».

D'après la méthode NIAM, on décompose ces phrases brutes en phrases élémentaires non ambiguës. Une phrase élémentaire a une structure simple de type < sujet >, < verbe >, < complément >. L'ensemble des phrases élémentaires issues de la décomposition d'une phrase en langage naturel doit porter la même signification que la phrase initiale. Dans notre exemple, nous obtenons les phrases élémentaires qui suivent.

Le défaut désigné par le type {mælle} a une forme désignée par une valeur de forme {allongée}. Le défaut désigné par le type {mælle} a une position désignée par une valeur de position {centre de planche}. Le défaut désigné par le type {mælle} a une orientation désignée par une valeur d'orientation {sens de fibre du bois}. Le défaut désigné par le type {mælle} a une couleur désignée par une valeur de couleur {brune}.

La [figure 3] représente alors une formalisation partielle en NIAM/ORM de cet énoncé en phrases élémentaires.

Nous faisons de même pour un autre défaut dont la description par l'expert est la suivante :

«Un nœud noir est un défaut de forme plutôt ronde, qui peut être aussi ovale, généralement éloigné des bords de la planche. Sa couleur est noire par rapport à la couleur du bois naturel, mais peut ne pas être entièrement noire».

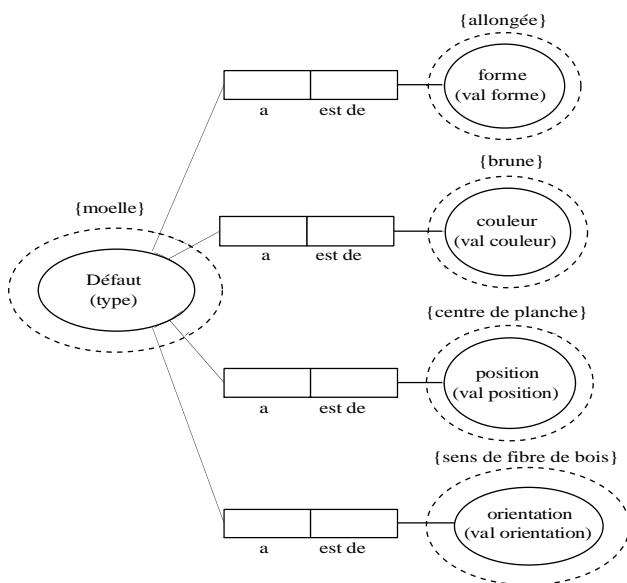


Figure 3. Modélisation partielle NIAM/ORM du défaut de type mælle

En appliquant le même principe de décomposition en phrases élémentaires, nous obtenons alors :

Le défaut désigné par le type {nœud noir} a une forme désignée par une valeur de forme {ronde, ovale}. Le défaut désigné par le type {nœud noir} a une position désignée par une valeur de position {éloignée des bords de la planche}. Le défaut désigné par le type {nœud noir} a une couleur désignée par une valeur de couleur {noire}.

La [figure 4] représente une formalisation partielle en NIAM/ORM de cet énoncé en phrases élémentaires.

Nous pouvons y remarquer que des invariants émergent de ces deux cas particuliers de défauts. Nous avons le nom du type de défaut, sa forme, sa couleur, sa position et son orientation. Bien que l'orientation ne soit pas commune à la définition des deux défauts, nous allons tout de même l'inclure au modèle générique en précisant qu'un défaut n'est pas obligatoirement caractérisé par une orientation. On peut même, à ce stade, se poser la question de savoir si cette notion d'orientation n'a pas de sens pour l'ensemble des défauts.

Nous allons donc créer un modèle générique issu de la généralisation de ces deux modèles particuliers.

Il faut maintenant introduire les notions de contraintes de totalité et d'unicité qui permettent de compléter l'information issue de la connaissance de l'expert. Sans entrer dans le détail, la notation graphique « • » indique une contrainte de totalité et la notation graphique « ↔ » une contrainte d'unicité. Le lecteur trouvera en Annexe 1 un exposé plus détaillé de cette syntaxe graphique. Pour illustrer cette prise en compte des contraintes, nous allons prendre l'exemple de la Position d'un Défaut.

Nous pouvons dire : Un défaut a une position, mais aussi qu'une position est de un défaut, ce qui a la même signification. Posons nous maintenant les questions sur l'existence de contraintes sur les deux formes de la phrase :

Totalité : Chaque défaut a t-il une position ? **Oui (1)**

Unicité : Un défaut a t-il une et une seule position ou une ou plusieurs ? **Une et une seule (2)**

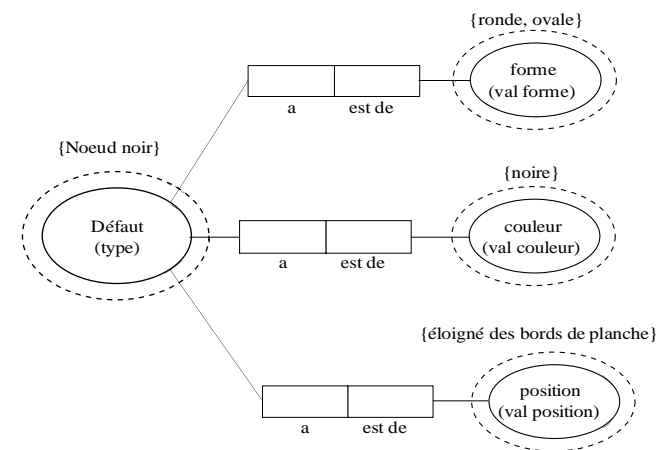


Figure 4. Modélisation partielle NIAM/ORM du défaut de type nœud noir

Totalité : *Chaque position est-elle d'un défaut ? Non (③)*

Unicité : *Une position est-elle de un et un seul défaut ou un ou plusieurs ? Un ou plusieurs (④)*

Nous pouvons ainsi en déduire la formalisation NIAM de cette connaissance sur la position du défaut illustrée par la [figure 5]. En faisant de même pour les autres phrases, nous pouvons en déduire un modèle générique contenant toutes les informations issues de la connaissance de l'expert.

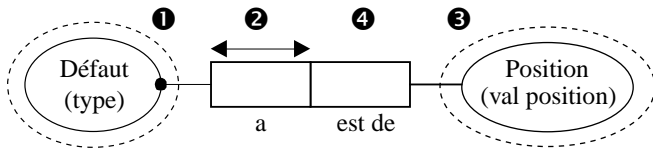


Figure 5. Modélisation NIAM/ORM du défaut et de sa position

La [figure 6] représente un extrait de la formalisation en NIAM/ORM de cette généralisation.

Après avoir fait ces opérations pour chaque défaut énoncé par l'expert bois, nous pouvons obtenir une modélisation informationnelle de sa connaissance, généralisée à n'importe quel type

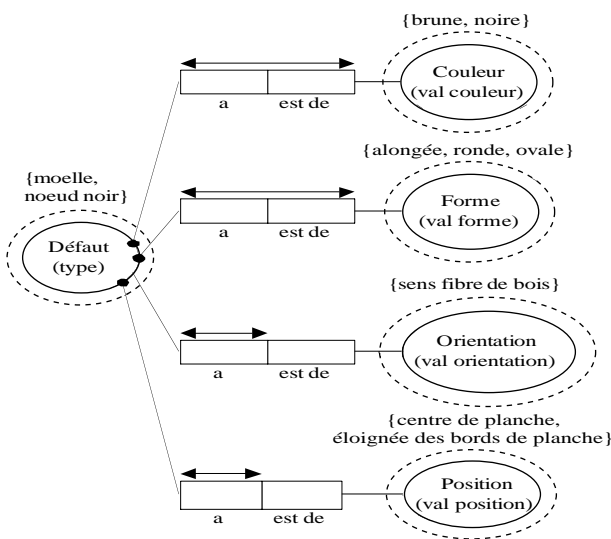


Figure 6. Extrait du modèle générique expert bois généralisant les défauts de type mælle et nœud noir

de défaut, par la construction progressive du modèle conceptuel associé.

Il est à noter que ce modèle a été validé sur la base de son équivalent en Langage Naturel Binaire par l'expert ayant lui-même initialement exprimé sa connaissance en langage naturel.

2.2. Le modèle symbolique expert du procédé de vision

De même que pour le modèle symbolique de l'expert « métier bois », on recense les connaissances de l'expert du procédé de vision. Ces connaissances portent sur le système d'identification de défauts et, plus particulièrement, sur les paramètres utiles à la quantification des caractéristiques du défaut à reconnaître. Nous appliquons le même principe que pour le modèle symbolique de l'expert métier et nous aboutissons à une caractérisation de tous les défauts rencontrés mais, cette fois, du point de vue des paramètres du système de vision. Nous avons interviewé les concepteurs du système de vision, et à partir de leur façon de caractériser chaque défaut, ou de leur connaissance sur les caractéristiques pertinentes pour identifier un défaut, nous avons construit un modèle général.

Voici ainsi un exemple qui permet de caractériser la couleur selon ce point de vue. L'expert procédé s'exprime de la manière suivante : « La couleur d'un défaut est mise en évidence par le paramètre renseignant sur le niveau de gris présent sur l'image du défaut ». Nous pouvons tirer les phrases élémentaires suivantes : *Une couleur désignée par une valeur de couleur {foncée} se caractérise par un niveau de gris désigné par une valeur de niveau de gris {élevé}. Un niveau de gris désigné par une valeur de niveau de gris {élevé} caractérise une couleur désignée par une valeur de couleur {foncée}.*

La [figure 7] représente une formalisation partielle en NIAM/ORM de cet énoncé.

Une fois tous les défauts caractérisés, en prenant en compte les contraintes de totalité et d'unicité, nous pouvons construire le modèle conceptuel associé permettant une généralisation de la connaissance de l'expert procédé. L'objectif de ce modèle est de relier les caractéristiques de vision aux caractéristiques métier. De même que pour le modèle symbolique de l'expert métier, nous ne présentons pas le modèle dans sa totalité. Néanmoins, précisons que le modèle obtenu comporte une dizaine d'idées (relation entre deux NOLOTs), et une dizaine de ponts de dénominations (relation entre un NOLOT et un LOT).

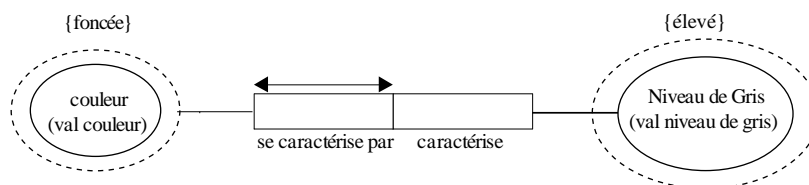


Figure 7. Modélisation partielle NIAM/ORM expert procédé

3. Le modèle numérique

Parallèlement aux connaissances exprimées par les modèles symboliques qui sont imprécises et incertaines car n'étant qu'une généralisation qualitative de la connaissance, nous disposons de mesures délivrées par la phase de segmentation qui elles sont entachées d'incertitude (cf. paragraphe 1.2.).

Afin de pouvoir faire un lien entre le domaine symbolique et le domaine numérique, nous utilisons une méthode s'appuyant sur la théorie des sous-ensembles flous [Kaufmann], permettant de supporter des raisonnements qualitatifs [Zadeh 3]. Ce processus est essentiel pour comparer les informations expertes (en langage naturel) avec la réponse du système numérique (données floues). Dans ce paragraphe, nous présentons le modèle numérique dont le principe général est illustré par la [figure 8].

Afin de détailler le principe du modèle numérique et l'utilisation des sous-ensembles flous, il est nécessaire de rappeler quelques notions sur la théorie des Ensembles Flous et plus particulièrement sur les mécanismes de raisonnement flou.

3.1. Raisonnement Flou

3.1.1. Mécanismes de base

Le raisonnement flou repose sur trois concepts liés entre eux : la proposition floue élémentaire, la proposition floue générale et les règles floues.

La proposition floue élémentaire [Bouchon-Meunier] permet de traduire dans le domaine symbolique des valeurs exprimées dans le domaine numérique. Pour pouvoir utiliser dans un même cadre connaissances numériques et symboliques, et tenir compte du fait que A peut subir des variations graduelles liées à un environnement, on utilise alors des expressions qui sont toujours de la forme « V est A », avec V associé à une variable linguistique [Zadeh 1]. On se limite alors à des descriptions de la forme « la taille est petite », « la couleur est foncée », ...

Une variable linguistique [Bouchon-Meunier] est défini par un triplet (V, X, T_V) où :

- V est une variable (surface, couleur, ...) définie sur un ensemble de référence X

- X l'univers de discours (domaine de variation de V)

- T_V le vocabulaire choisi pour décrire de manière symbolique les valeurs de V (petit, grand, foncé, clair, ...). L'ensemble $T_V = \{A_1, A_2, \dots\}$, fini ou infini, contient donc des sous-ensembles flous normalisés de X , utilisables pour caractériser V . Une proposition floue élémentaire est définie à partir d'une variable linguistique (V, X, T_V) de L par la qualification « V est A », où A est une caractérisation floue appartenant à T_V ou à $M(T_V)$ avec M un ensemble de modificateurs linguistiques. La valeur de vérité d'une proposition floue élémentaire est définie par la fonction d'appartenance μ_A de A .

Une proposition floue générale [Bouchon-Meunier] est la composition de propositions floues élémentaires « V est A », « W est B »... pour des variables V, W supposées indépendantes. La plus simple s'exprime comme la conjonction de propositions floues élémentaires « V est A et W est B » (par exemple, l'allongement est moyen et la compacité est faible) où V et W sont définis sur des ensembles de référence X et Y (où X est l'univers de discours de la variable linguistique floue V et Y l'univers de discours de la variable linguistique floue W). Elle est associée au produit cartésien $T(A, B)$ caractérisant la variable conjointe (V, W) sur l'ensemble $X \times Y$. La valeur de vérité est donc définie par $T(\mu_A(x), \mu_B(y))$.

Une telle proposition floue se retrouve habituellement dans les règles de systèmes à base de connaissances. La forme « V est A et W est B alors U est C » est appelée une règle floue [Kaufmann], « V est A et W est B » est la prémisse de la règle et « U est C » sa conclusion. Par exemple, « SI la couleur est foncée ET la forme est ronde ALORS le défaut est du type Nœud Noir ». Cette règle floue est une extension des règles d'inférence en logique propositionnelle (Si p alors q).

Une inférence floue associe à toute règle de la forme « si V est A alors W est B » construite à partir des variables linguistiques (V, X, T_V) et (W, Y, T_W) une relation floue R entre X et Y , de fonction d'appartenance telle que :

$$\forall (x, y) \in X \times Y, \mu_R(x, y) = I(\mu_A(x), \mu_B(y))$$

avec I un opérateur d'inférence floue.

Les règles mettant en œuvre ce principe peuvent être classées en deux catégories, suivant l'opérateur d'inférence choisie : les

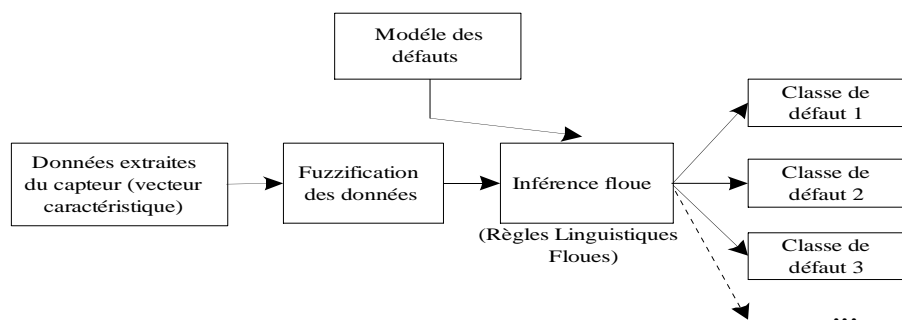


Figure 8. Schéma explicatif du modèle numérique

règles conjonctives et les règles implicatives qui regroupent respectivement d'une part les règles à possibilité et les règles anti-graduelles et d'autre part les règles à certitude et les règles graduelles [Dubois 3]. Cette dichotomie peut aussi s'expliquer d'un point de vue « culturel » dans le sens où les règles conjonctives sont issues du domaine de l'analyse de données où les mécanismes de raisonnement sont conduits par les données, alors que les règles implicatives sont plus utilisées dans les domaines des sciences cognitives où les raisonnements sont régis par les connaissances [Dubois 1]. Les règles peuvent ainsi être obtenues de deux façons, soit par un expert qui exprime la connaissance qu'il a du système grâce à l'expérience acquise, soit par apprentissage à partir d'un lot de données.

L'ensemble des règles forme une base de règles \mathcal{K} que le mécanisme d'inférence active en parallèle. Chaque règle délivre ainsi une conclusion partielle dont on doit faire l'union pour fournir la conclusion globale. La relation $R^{\mathcal{K}}$ correspondant à la base de règles est donc :

$$R^{\mathcal{K}} = \cup_{i=1, \dots, n} A_i \times B_i$$

L'opérateur d'union doit être un opérateur de conjonction (T-Norme : T) si on utilise des règles implicatives et un opérateur de disjonction (T-Conorme : \perp dans le cas de règles conjonctives [Ughetto].

3.1.2. Choix de règles conjonctives

On distingue deux grands types de raisonnement utilisant des règles floues, le raisonnement abductif où l'on cherche à obtenir des informations sur l'entrée X à partir de connaissances sur la variable de sortie Y , et le raisonnement déductif où l'on cherche à déduire Y à partir de valeurs de X . En ce qui nous concerne, nous nous plaçons dans le second cas qui correspond au Modus Ponens. Zadeh [Zadeh 2] en a proposé une généralisation adaptée à l'utilisation de règles floues [Bouchon-Meunier]. Elle se base sur un principe de combinaison-projection des deux relations floues représentant la règle et l'entrée :

$$\forall y \in Y, \mu_B(y) = \sup_{x \in X} (T(\mu_{A'}(x), \mu_R(x, y)))$$

Pour notre application qui est du domaine de l'analyse de données, nous avons logiquement choisi de retenir un mécanisme de règles conjonctives et plus particulièrement des règles à possibilité où l'inférence est une pseudo implication réalisée par une T-Norme. Chaque règle de l'ensemble est activée en parallèle et les résultats partiels sont ensuite combinés par un opérateur de disjonction (cf. paragraphe 3.1.1).

Ce mécanisme d'inférence fournit une interprétation et une sémantique différente des mécanismes utilisant une implication [Dubois 3] [Ruspini]. Il a l'avantage néanmoins, de ne pas poser de problème de cohérence et de redondance de la base de règles [Ughetto] [Dubois 2]. Il est à noter que le problème de cohérence, pour les règles conjonctives, se retrouve uniquement à la défuzzification, que nous n'utilisons pas dans notre cas. En cas d'absence d'information, c'est-à-dire dans le cas où la totalité

de l'espace d'entrée n'est pas couvert par l'ensemble de règles, la sortie donne une possibilité nulle, ce qui se traduit dans notre cas par l'affectation de la classe « défaut inconnu ». En outre, les opérateurs de conjonction utilisés autorisent la combinaison des variables d'entrée mixtes (précises, floues, nominales, ...) simplement normalisées. Enfin, ce type de règles permet de diminuer fortement les temps de calcul.

Les deux modèles les plus connus, implémentant ces règles, sont les modèles de Larsen et Mamdani [Bouchon-Meunier]. Il est à noter que nous n'avons pas retenu de modèle de Sugeno [Sugeno] car nous ne désirons pas obtenir de valeur de sortie numérique.

3.2. La méthode de Nozaki-Tanaka

Pour notre application, nous avons besoin d'une méthode qui nous permette de manipuler à la fois des informations numériques (données) et des informations symboliques (connaissances) dans un même référentiel. Concernant la partie numérique nous voulons manipuler des données précises mais incertaines, générer des règles automatiquement par apprentissage fonctionnant avec un nombre d'échantillons restreint, interpréter le système de règles obtenu pour le comparer au modèle symbolique. De plus, la méthode doit être simple à mettre en œuvre afin de permettre un temps de calcul compatible avec les contraintes « temps réel » du système industriel.

Il existe de nombreuses méthodes permettant d'obtenir automatiquement les règles floues à partir d'un lot de données [Cordon1] [Berthold]. Par rapport aux critères évoqués dans le paragraphe précédent, notre choix s'est porté sur la méthode de Nozaki-Tanaka qui met en œuvre un mécanisme d'inférence basé sur le modèle de Larsen. Nous avons choisi Larsen plutôt que Mamdani car la T-Norme Produit apparaît plus adaptée que le Min à la manipulation d'un grand nombre de prémisses. Elle permet en effet un découpage non linéaire de l'espace des variables d'entrée [Berthold].

La méthode de base [Ishibuchi 1] a fait l'objet de plusieurs évolutions. La version que nous utilisons est itérative [Nozaki] et favorise la règle de réponse maximale. Cette méthode fournit le meilleur résultat lorsque le découpage de l'espace des variables d'entrée est correctement effectué [Ishibuchi 2]. De plus, comme nous le verrons, nous n'utilisons pas l'étape d'affinage des frontières pour garder les sorties les plus graduelles possibles. Enfin, au vu des lots d'apprentissage dont nous disposons et qu'il est raisonnable d'envisager pour l'application, nous ne pouvons retenir de méthode basée sur l'utilisation d'algorithmes génétiques ou neuronaux pour en paramétrer les différentes étapes [Ishibuchi 3] [Cordon 2] [Shen].

3.2.1. Principe

Cette méthode de reconnaissance est une méthode itérative, mettant en œuvre un mécanisme d'apprentissage supervisé. Elle permet de générer des règles floues du type « Si... alors... » défi-

nissant les défauts. Par exemple : « SI surface (petite) ET SI couleur (foncée) ALORS défaut (nœud noir) ».

La [figure 9] montre l'algorithme de calcul de la méthode de Nozaki-Tanaka. Elle se décompose en 6 étapes principales que l'on peut regrouper, pour plus de simplicité, en trois grandes étapes : la fuzzification des entrées, la génération des règles floues et l'ajustement. Le jeu de règles obtenu est ensuite utilisé pour inférer la reconnaissance.

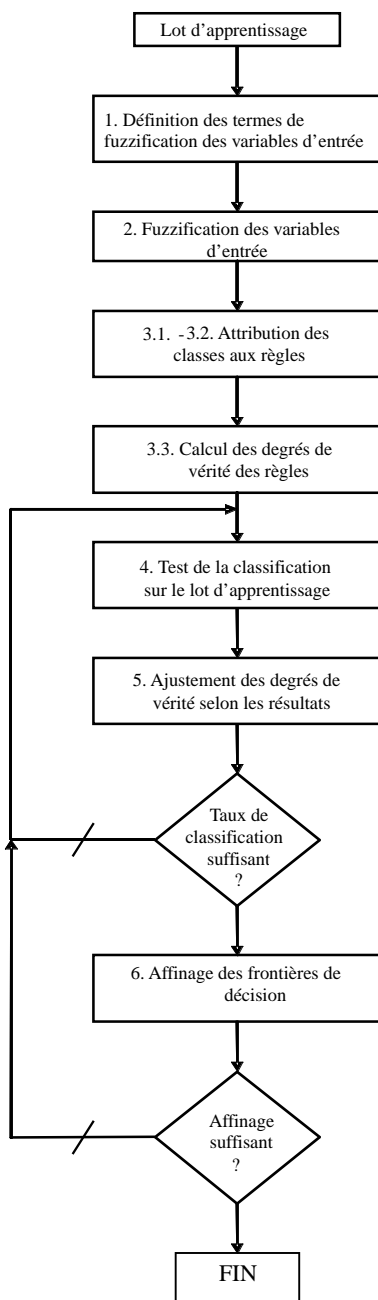


Figure 9. Algorithme de la méthode de Nozaki-Tanaka

3.2.2. Fuzzification des entrées

L'étape de fuzzification caractérise les variables linguistiques utilisées. Dans le triplet (V, X, T_V) définissant une variable linguistique floue, chaque terme de T_V est un sous-ensemble flou défini sur X et caractérisé par sa fonction d'appartenance. Cette étape définit le nombre de décompositions de la variable considérée. Les termes sont choisis en rapport avec le vocabulaire utilisé par l'expert. Par exemple, l'intensité peut prendre le qualificatif « sombre » ou « claire », la surface peut être « petite », « moyenne » ou « grande ». Le choix du nombre de termes pour qualifier une variable est généralement empirique. On utilise alors une équipartition de l'espace d'entrée avec un nombre de termes impair, relativement faible (trois ou cinq) pour ne pas avoir un nombre de règles trop important.

Dans notre cas, la décomposition est guidée par l'expertise qui définit le nombre de tendances (termes) pour une caractéristique (variable). Par exemple, il y a trois types de nœuds « petit, intermédiaire et grand » qui conduisent à une décomposition des paramètres de vision associés à la taille en trois termes, représentée par la [figure 10]. La position des fonctions d'appartenance est déterminée à partir d'une analyse du lot d'apprentissage. L'adaptation des termes aux données, qui permet un découpage de l'espace des entrées adéquat, n'améliore pas forcément le taux de reconnaissance, mais permet au moins de diminuer le nombre d'itérations de la méthode [Siaud].

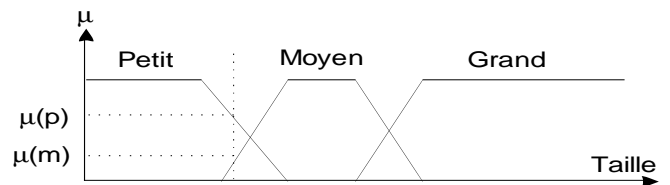


Figure 10. Exemple de fuzzification du paramètre de taille en 3 sous-ensembles

3.2.3. Génération des règles floues

Cette deuxième phase permet la génération des règles de type « Si ... alors ». Avec cette base de règles floues, l'algorithme de Nozaki-Tanaka décrit la perception que le système a des défauts qui lui sont présentés. En effet, un lot de défauts identifiés par un expert est pris comme paramètre par l'algorithme de génération de règles floues, c'est le lot d'apprentissage.

Si nous avons deux caractéristiques en entrée et une sortie, la forme générale de la règle floue est :

SI x_1 est A_i **ET** x_2 est A_j **ALORS** x_3 est dans la classe de défaut C_k

Avec :

x_1 et x_2 les données d'entrées (par exemple la couleur et la surface)

x_3 la donnée de sortie (nom du défaut)
 A_i et A_j les sous-ensembles flous (par exemple «petit» et «grand»)
 C_k la classe du $k^{\text{ème}}$ défaut
 Chaque règle délivre une conclusion partielle qui est ensuite agrégée aux autres pour fournir la conclusion finale par un opérateur flou de disjonction de Zadeh : $\perp(x, y) = \max(x, y)$.
 L'inférence repose sur le modèle de Larsen [Bouchon-Meunier] qui utilise comme opérateur de pseudo implication floue la T-Norme produit : $T(x, y) = x * y$.

3.2.4. Ajustement

L'ajustement représente la partie itérative de l'algorithme. Il consiste à tester chaque défaut à partir du même lot de données. Un mécanisme permet au système d'ajuster le découpage de l'espace de représentation en fonction des résultats obtenus [Nozaki].

Il est à noter que la méthode de Nozaki-Tanaka propose une phase supplémentaire d'affinage qui vise à augmenter le degré d'appartenance à la classe d'appartenance maximale en modifiant les pentes de la fonction d'appartenance. À l'extrême, cette phase permet d'avoir une représentation binaire. Nous n'utilisons pas cette étape car nous voulons conserver la gradualité des réponses. En effet, nous préférons ne pas classer un défaut dont le degré d'appartenance à la classe maximale est faible, plutôt que de faire une erreur de reconnaissance.

Dans ce sens, nous nous rapprochons de la notion de rejet en ambiguïté [Dubuisson] d'un classificateur bayésien. Néanmoins, l'idée est de lever ultérieurement cette ambiguïté dans l'étape de Post-Traitement, en exploitant par exemple, une vue plus générale de la planche (mise en correspondance des faces, proximité des défauts du même type, ...). Nous retenons les trois classes de défauts qui donnent les plus fortes réponses à la sortie du module d'« Identification », si celles-ci sont différentes de la classe « défaut Inconnu ». Le [tableau 1] montre un exemple des trois classes de sortie de défauts « Nœud Cerné » auxquelles est associé le degré d'appartenance correspondant.

Dans ce cas, nous avons en entrée trois défauts de la classe «Nœud Cerné ». Le premier a été reconnu avec un fort degré d'appartenance et donc sans ambiguïté, les classes suivantes n'ont pas à être considérées. Le deuxième a un degré d'appartenance à la classe « Nœud Cerné » plus faible en absolu, mais reste suffisamment supérieur au degré d'appartenance de la classe suivante, il pourra donc être identifié comme un « Nœud Cerné ». Dans le troisième cas, on peut remarquer que le degré associé à la classe maximale est relativement faible et que l'écart en terme de possibilité avec la classe qui arrive en deuxième position est relativement faible également. Dans ce cas, il serait judicieux de ne pas prendre de décision à ce niveau et de laisser l'étape de Post-Traitement lever l'ambiguïté : par exemple, si on trouve sur l'autre face, à cette position, un défaut identifié comme un « Nœud Cerné », on pourra favoriser la deuxième classe.

Tableau 1. Exemple de sortie du module d'Identification

Échantillons	Classes de sortie	Degré d'appartenance
Nœud Cerné	Nœud Cerné	0,84945
	Mœlle	0,00568
	Inconnu	0,00000
Nœud Cerné	Nœud cerné	0,30773
	Petit Nœud Sain	0,034569
	Nœud Sain	0,00973
Nœud Cerné	Nœud Sain	0,245132
	Nœud Cerné	0,121315
	Poche Résine Noire	0,00248

Il est à noter que pour les résultats donnés dans cet article (cf. paragraphe 5.1), nous avons retenu la classe de plus fort degré d'appartenance sans tenir compte des classes suivantes.

3.3. Inférence unique

Dans cette partie, les résultats sont obtenus à partir d'un moteur d'inférence, représenté par la [figure 11], comportant onze paramètres en entrée et quatorze classes de défauts en sortie. Sachant qu'en moyenne chaque variable V composant le vecteur caractéristique est décomposée en trois termes T_V (petit, moyen, grand), nous aboutissons à 311 règles linguistiques floues générées. Bien que les résultats obtenus soient très satisfaisants (cf. paragraphe 5.1), cette méthode a l'inconvénient majeur de ne pas respecter la contrainte de temps industrielle fixée à 2 ms pour l'identification d'un défaut puisque nous obtenons un temps de traitement d'environ 6 ms.

Un deuxième inconvénient rédhibitoire à nos yeux réside dans l'impossibilité d'interpréter l'ensemble des règles obtenues et ainsi de pouvoir les valider en les confrontant aux modèles symboliques.

C'est pourquoi nous proposons d'enchaîner hiérarchiquement plusieurs moteurs d'inférence plus simples afin de constituer une arborescence réduisant ainsi le nombre de règles.

3.4. Structure arborescente

3.4.1. Arborescence empirique

À partir de l'idée d'enchaîner hiérarchiquement plusieurs nœuds d'inférence plus simples, nous aboutissons à une arborescence représentée par la [figure 12] et composée de neuf nœuds décisionnels avec quatre niveaux de décision. Elle est construite empiriquement par une personne issue du domaine « vision ». De fait, elle utilise implicitement ses connaissances « *a priori* » pour construire le modèle numérique. Le principal objectif de cette arborescence est la réduction du temps de calcul pour respecter la contrainte de temps industrielle tout en essayant de maximiser le taux d'identification.

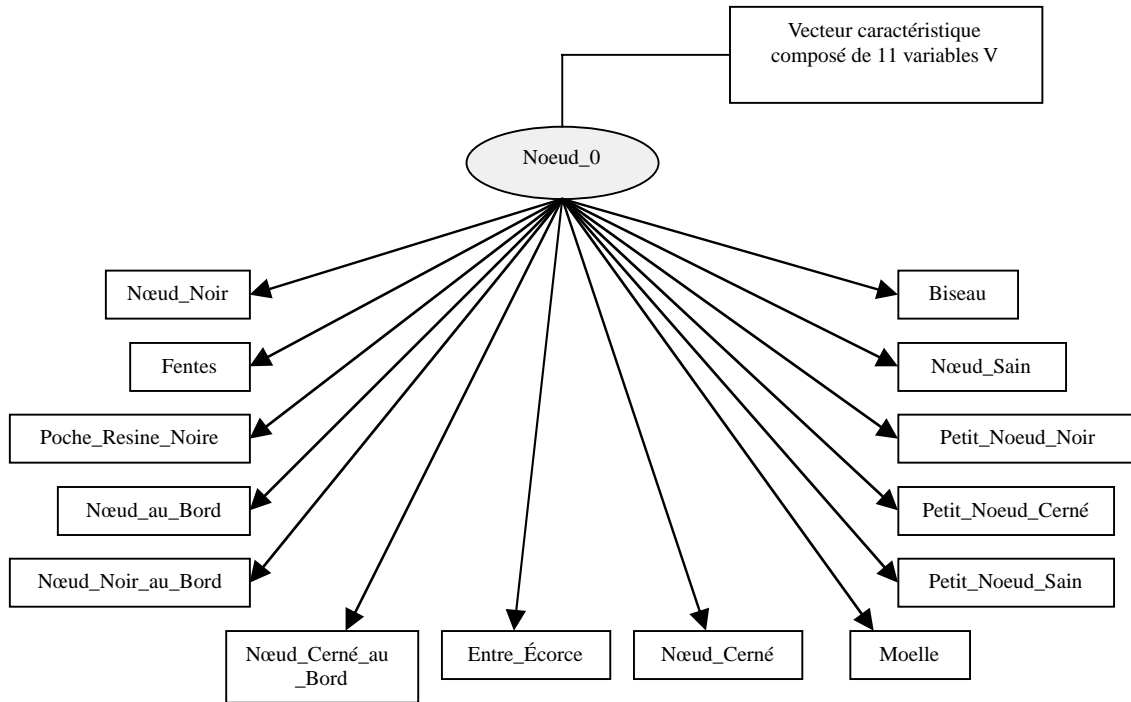


Figure 11. Schéma représentatif de l'inférence unique

Le gain de temps se situe autour de 4ms (soit environ 2ms pour identifier un défaut). Ceci est dû à la diminution du nombre de règles générées par le modèle numérique, passant de 3^{11} pour l'inférence unique à environ 500. Les résultats concernant le taux d'identification sont exposés dans le paragraphe 5.1.

3.4.2. Arborescence structurée

Néanmoins, l'arborescence, décrite précédemment, a été construite sans suivre une démarche méthodologique précise et les résultats obtenus dépendent assez fortement d'une structuration « judicieuse » de la décomposition. Nous pensons améliorer la démarche en essayant de trouver une décomposition logique, « explicable », voire justifiable.

Ainsi, la construction de cette arborescence est aujourd'hui basée sur l'utilisation de trois types de connaissances :

- la première prend en compte la hiérarchie entre les attributs, c'est-à-dire une importance relative des attributs entre eux. Ces attributs sont en fait les caractéristiques exprimées par l'expert du métier du bois (forme, couleur, position ...). Dans notre modèle, nous avons supposé que la forme est plus discriminante que la couleur. On distingue, par exemple, deux grandes sous-classes de défauts selon la forme : les défauts « plutôt allongés » et les défauts « plutôt compacts » ;

- la deuxième prend en compte la compatibilité des défauts, c'est-à-dire la possibilité de pouvoir confondre lors de l'identification deux types de défauts proches (par exemple, un petit nœud noir avec un nœud noir). Nous avons fait en sorte que cette confusion ne soit faite que dans les derniers niveaux de l'arborescence structurée ;

- enfin, le lien entre les attributs exprimés par l'expert métier du bois et les caractéristiques exprimées par l'expert du procédé de vision (lien entre modèle symbolique « expert métier du bois » et modèle symbolique « expert procédé de vision »).

Seule l'intégration du troisième type de connaissances est utilisée et formalisée dans la suite du document. En effet, l'exploitation des deux modèles de connaissances tient essentiellement dans la construction des nœuds de l'arborescence et non de l'arborescence en elle-même, les connaissances utilisées pour générer la structuration hiérarchique des nœuds n'étant pas encore entièrement formalisées.

Pour des raisons de confidentialité, nous ne pouvons montrer l'arborescence ainsi construite, mais le lecteur peut en trouver un aperçu en se référant à la [figure 16].

4. Prise en compte de connaissances pour la génération d'un nœud de l'arborescence structurée

4.1. Principe général

La démarche de génération des différents moteurs d'inférences de l'arborescence structurée repose sur une méthode descen-

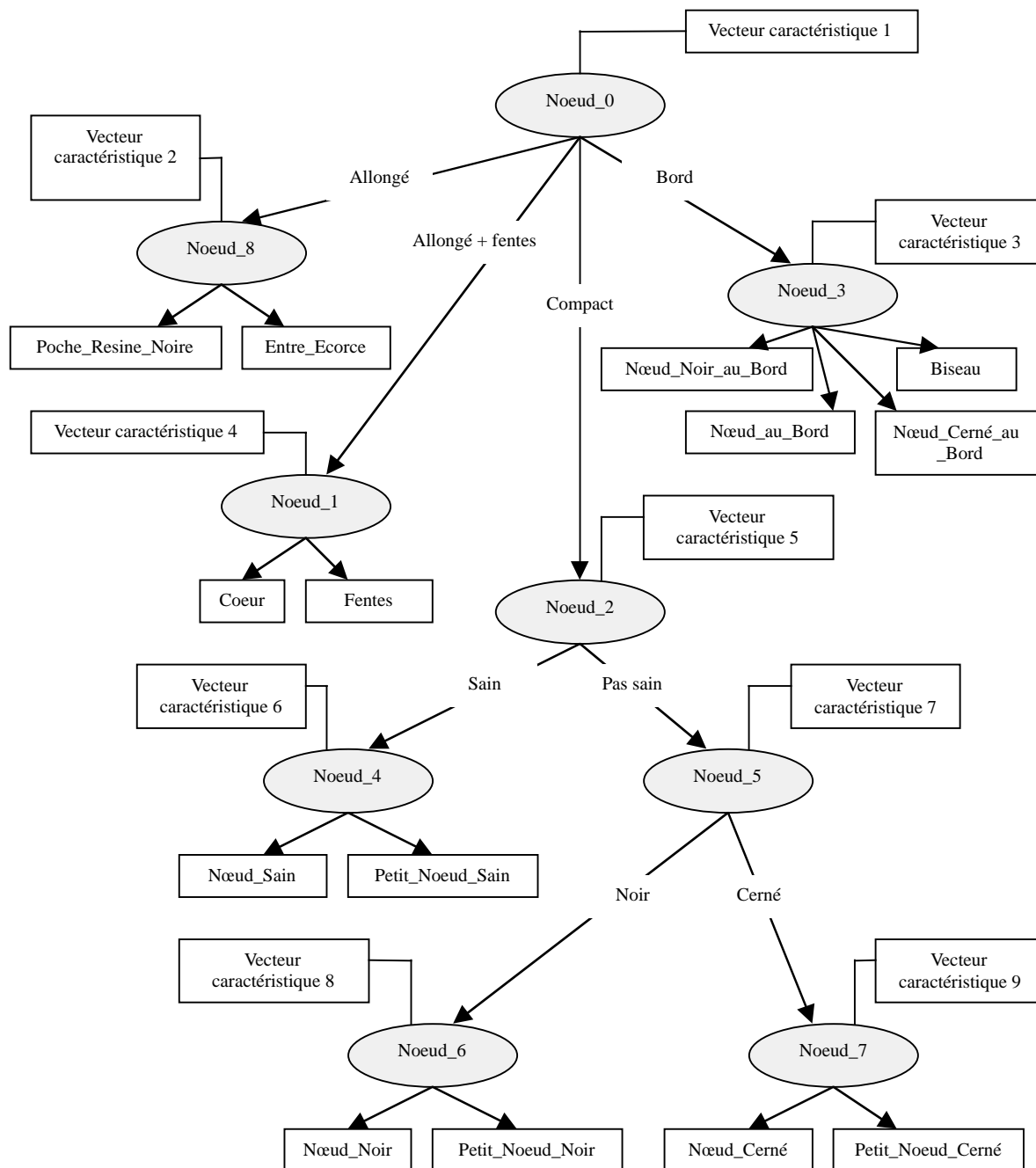


Figure 12. Schéma représentatif de l'arborescence empirique

dante, illustrée par la [figure 13], qui part des connaissances les plus complexes pour aboutir à la construction du modèle numérique. Par connaissances complexes, nous considérons les connaissances porteuses d'informations ambiguës, en l'occurrence, les connaissances de l'expert métier du bois et de l'expert procédé de vision, exprimées en langage naturel. Grâce à la méthode NIAM, nous avons pu lever toutes ces ambiguïtés par la construction de modèles symboliques associés aux connaissances des différents experts. Le but de cette arborescence est de remplacer la génération intuitive des nœuds du modèle numérique par une génération structurée, basée sur l'analyse et l'intégration des connaissances des différents experts.

4.2. Démarche appliquée à un nœud de l'arborescence structurée

Cette partie a pour but de présenter comment les deux modèles de connaissances sont utilisés pour construire de façon systématique un nœud de l'arborescence. L'exemple retenu concerne la différenciation des défauts de type «nœud noir», «petit nœud noir» et «nœud noir au bord». Par construction du nœud, on entend choisir les paramètres de vision les plus pertinents pour composer le vecteur caractéristique. Cette démarche, appliquée ici à un nœud, a été étendue à tous les nœuds de l'arborescence.

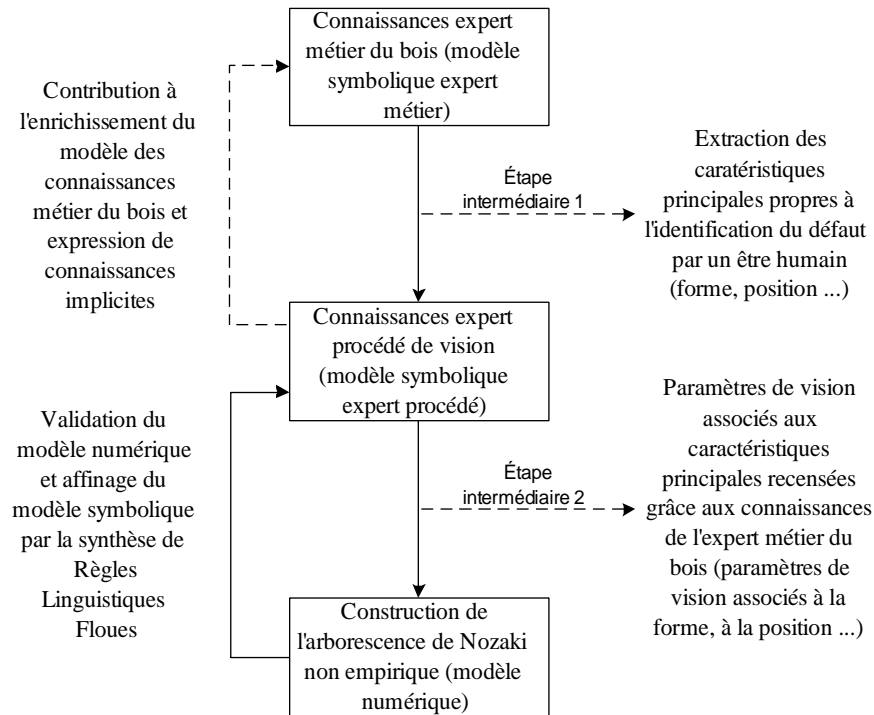


Figure 13. Schéma explicatif de la méthode descendante

Avant tout, afin de pouvoir utiliser correctement les connaissances émises par les experts du domaine du bois et de la vision, nous allons procéder à la formalisation de leurs connaissances. Pour ce faire, nous allons suivre la méthode représentée par la [figure 13]. Nous allons commencer par modéliser les connaissances de l'expert métier bois à partir de la définition qu'il nous donne des différents nœuds noirs.

Dans une interview l'expert métier du bois a défini ces défauts de la manière suivante :

« Un nœud noir est un défaut de forme plutôt ronde, mais il peut être légèrement ovale. Sa couleur est noire par rapport à la couleur du bois naturel, mais ne peut être entièrement noire. Un petit nœud noir possède les mêmes caractéristiques que le nœud noir mais il a une petite taille. Un nœud noir au bord possède les mêmes caractéristiques que le nœud noir mais il se situe en bord de planche. »

À partir de ces phrases exprimées en langage naturel, nous pouvons construire un modèle symbolique partiel représentant les connaissances de l'expert du bois, illustré par la [figure 14].

Ce modèle symbolique est en fait un modèle instancié du modèle générique formalisant les connaissances issues de l'expert métier du bois. En effet, cela revient à utiliser le modèle générique « expert métier du bois » et à l'instancier trois fois : une première pour définir le Nœud Noir, une deuxième pour le Petit Nœud Noir et enfin une troisième pour le Nœud Noir au Bord. Il est à noter qu'une partie de ces connaissances est également utilisée pour configurer l'étape de fuzzification du moteur d'inférence, notamment pour déterminer le nombre de terme (T_V)

des variables linguistiques associées à chaque caractéristique (cf. paragraphe 3.2.2.).

D'après cette modélisation, nous pouvons remarquer que la famille « nœud noir » est décrite par quatre caractéristiques qui sont la forme, la couleur, la position et la taille. Ces quatre caractéristiques correspondent à l'étape intermédiaire 1 de la [figure 13] visant à l'extraction des caractéristiques propres à l'identification des défauts de type « nœud noir » par un être humain.

Ensuite, en utilisant le modèle symbolique expert vision qui fait la liaison entre le domaine symbolique métier du bois et le domaine numérique des données, nous allons pouvoir déterminer les paramètres vision à associer aux attributs « métier » décrivant les défauts. Ceci correspond à l'étape intermédiaire 2. De cette mise en correspondance, nous pouvons en déduire le modèle symbolique regroupant les domaines métier du bois et procédé de vision. Ce modèle symbolique est représenté partiellement en [figure 15].

Nous savons donc maintenant qu'il faut n paramètres de vision pour pouvoir définir la famille « nœud noir ». Il est à noter que des paramètres de vision identiques peuvent être utilisés pour définir différentes caractéristiques du domaine métier du bois. Le nœud visant à identifier ces trois défauts est représenté sur la [figure 16] par le Nœud _{n} .

Pour ce Nœud _{n} , seuls cinq paramètres sont utilisés. Ils définissent les notions de Taille ($Larg_re$, $Long_re$, Pt_axe et $Surf$) et de Position ($C4$, qui est un paramètre propre à la société LuxScan Technologies). En effet, des paramètres vision définis-

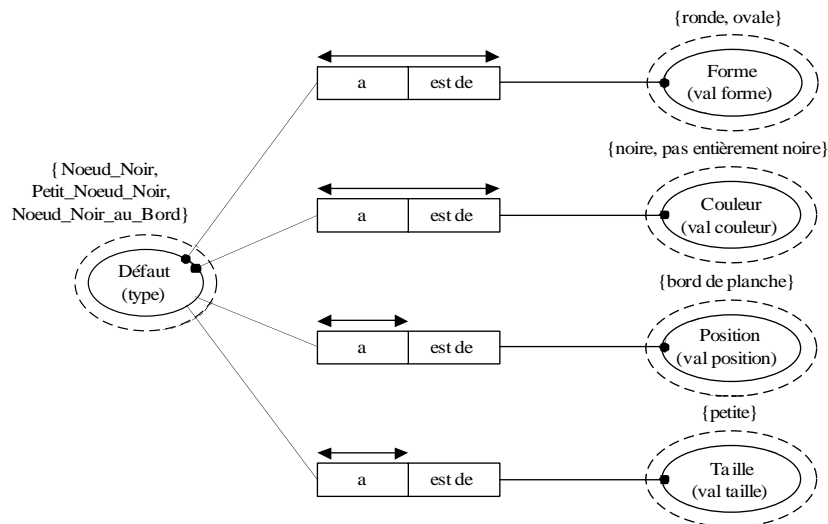


Figure 14. Modélisation partielle NIAM/ORM des connaissances exprimées par l'expert bu bois

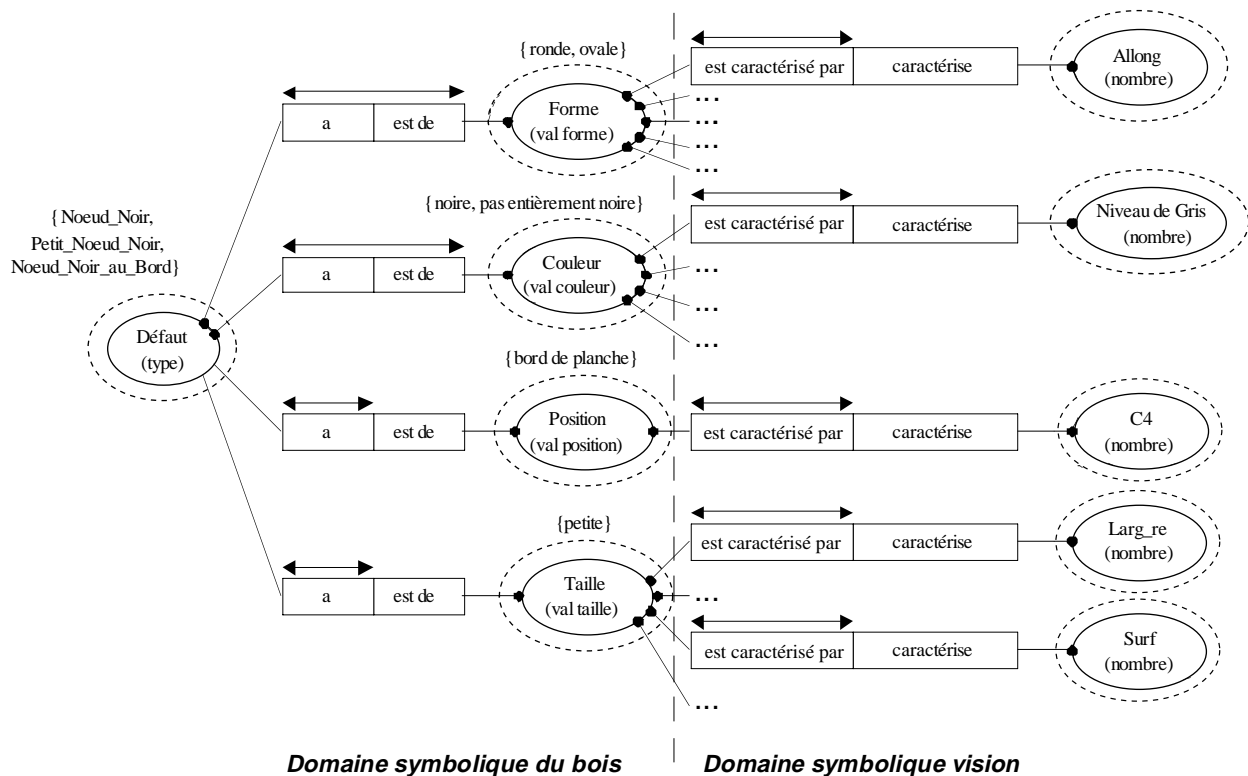


Figure 15. Modélisation NIAM/ORM des connaissances exprimées dans les deux domaines

sant les notions de Forme et de Couleur sont déjà utilisés à des niveaux décisionnels supérieurs, pour les nœuds Nœud_0 et Nœud_n - 1. Le Nœud_0 sert à la distinction de Forme et le Nœud_n - 1 à la distinction de couleur.

Les résultats de l'identification obtenus pour le Nœud_n sont résumés dans les [tableaux 2 et 3] qui suivent.

4.3. Analyse du modèle numérique obtenu

Cette étape consiste à interpréter la base de règles obtenue après apprentissage. Pour ce faire, nous proposons d'exploiter la matrice générée à partir des lots de données, afin d'interpréter le mécanisme d'identification. Cette matrice est la traduction

Tableau 2. Résultats de l'identification sur le lot d'apprentissage

Nombre d'échantillons	Pourcentage de bien classé	Nom du défaut
10	100,0	Nœud_Noir
4	100,0	Petit_Nœud_Noir
10	90,0	Nœud_Noir_au_Bord
Pourcentage global d'identification : 95,9 (23 sur 24)		

Tableau 3. Résultats de l'identification sur le lot de généralisation

Nombre d'échantillons	Pourcentage de bien classé	Nom du défaut
4	100,0	Nœud_Noir
20	100,0	Petit_Nœud_Noir
2	50,0	Nœud_Noir_au_Bord
Pourcentage global d'identification : 96,2 (25 sur 26)		

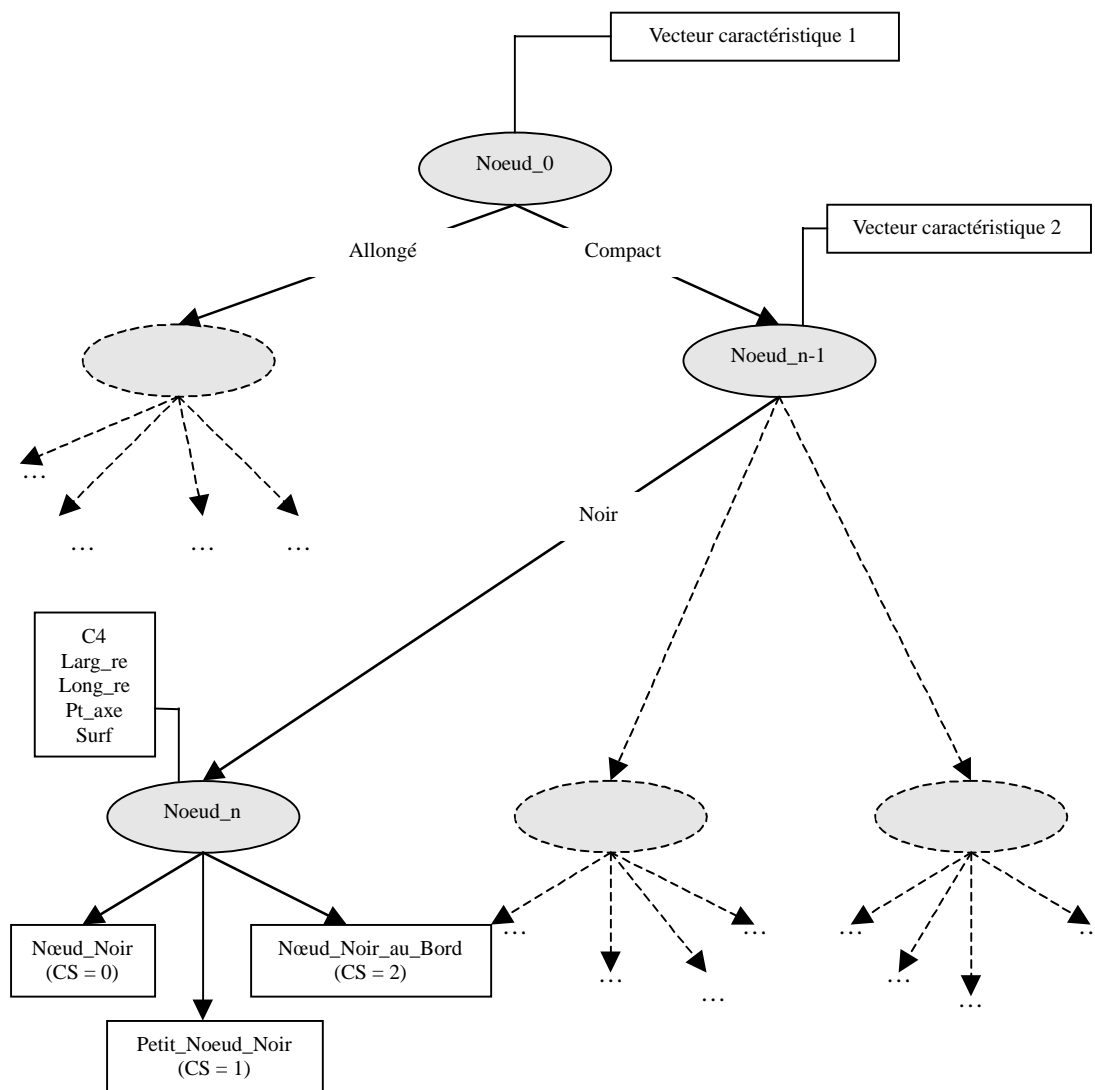


Figure 16. Schéma explicatif du nœud décisionnel visant à identifier les trois défauts

numérique des règles linguistiques floues, elle est formée de la manière suivante :

- les colonnes représentent les paramètres en entrée du nœud,
- les lignes représentent les règles.

À chaque prémisses est associé un numéro logique qui peut être 0, 1 ou 2 en fonction du nombre de termes (« petit », « moyen »

ou « grand ») utilisés pour la variable linguistique considérée. À chaque règle floue sont associés une classe de sortie (CS) de défaut et un coefficient de confiance (CF) que le système accorde à la règle.

Par exemple, dans la matrice générée pour le Nœud_n illustré par la [figure 16], nous obtenons 51 règles floues dont la forme est illustrée par le [tableau 4].

fuzzification adaptée aux variables caractéristiques puisque le taux de reconnaissance global atteint alors 93 %. Ces essais ont été menés sur un lot de 107 défauts pour l'apprentissage et un lot de 176 défauts pour la généralisation, mis à notre disposition par la société Luxscan Technologies.

Cependant, les principales conclusions que nous pouvons tirer de ces essais sont de trois types.

Premièrement, la structuration de l'arborescence apporte un gain logique en terme de taux de reconnaissance. Ce point nous conforte dans notre volonté de modéliser la démarche d'identification suivie par les experts afin de pouvoir formaliser et systématiser la construction de l'arborescence.

Deuxièmement, notre démarche d'intégration des connaissances au travers des modèles générés, nous a permis de choisir au mieux les différents vecteurs caractéristiques de chaque moteur d'inférence ainsi que le découpage le plus approprié des espaces d'entrée. Ceci nous a permis de diminuer le nombre total de règles sans dégrader les taux de reconnaissance. Ainsi, nous avons réussi à abaisser le temps moyen d'identification à 1,3 ms par défaut, respectant ainsi la contrainte de temps industrielle.

Enfin, indépendamment de la formule choisie, l'utilisation d'un mécanisme de raisonnement basé sur des règles floues, nous fournit un système au moins aussi performant que les méthodes utilisées par l'industriel jusqu'à présent, avec l'avantage d'être beaucoup plus simple à régler et à mettre au point en fonction des desiderata particuliers d'un client.

5.2. Méthode du « laissé pour compte »

Au vu du nombre relativement faible d'échantillons, nous utilisons la méthode du « laissé pour compte » [Dubuisson] pour calculer le taux de reconnaissance global. Cette méthode vise à

s'affranchir dans les résultats de l'influence du choix du lot d'apprentissage et du lot de test dans les taux d'identification. Ce taux représente l'association du taux de mémorisation et du taux de généralisation.

Pour ce faire, nous utilisons un lot unique de 283 défauts regroupant les 107 défauts du lot d'apprentissage et les 176 défauts du lot de généralisation. Les résultats obtenus sont résumés dans le [tableau 6].

Ces résultats montrent des taux d'identification sensiblement identiques à ceux obtenus à partir des deux lots distincts utilisés dans le paragraphe 5.1 indiquant que les lots utilisés sont représentatifs des données.

Ces résultats ont été obtenus en choisissant la classe de sortie où le degré d'appartenance est le plus fort car en réalité le système fournit trois classes de plus fort degré d'appartenance (cf. paragraphe 3.2.4.). Si nous tenons en compte les classes suivantes où la potentialité de confusion (degré d'appartenance maximum peu élevé) le taux d'identification peut être encore amélioré ([Tableau 1]).

6. Conclusion et perspectives

Nous avons proposé une démarche qui permet, par la mise en relation d'attributs issus d'un modèle symbolique formalisant les connaissances métier du bois et de caractéristiques visions issues d'un modèle symbolique formalisant les connaissances procédé de vision, de générer les nœuds décisionnels composant une arborescence structurée pour l'identification de défauts sur

Tableau 5. Résultats comparatifs des pourcentages globaux d'identification des trois arborescences

	Méthode point (*)	Un nœud	Empirique	Structurée
Apprentissage	72 %	99,1 %	94,4 %	93,5 %
Apprentissage + compatibilité		100 %	94,4 %	98,1 %
Généralisation		68,8 %	63,6 %	66,5 %
Généralisation + compatibilité		91,5 %	90,9 %	93,2 %

(*) : La méthode point est une méthode non itérative dont le taux global de reconnaissance est obtenu à partir d'un seul lot de données.

Tableau 6. Résultats des pourcentages d'identification avec la méthode du « laissé pour compte »

	Sans compatibilité des défauts	Avec compatibilité des défauts
Taux d'identification	84,1 %	95,1 %

des planches de bois. Cette mise en relation des deux modèles permet d'obtenir un vecteur caractéristique (pour chaque nœud) composé de paramètres cohérents par rapport aux connaissances émises par les experts et de découper de façon appropriée l'espace des variables d'entrée. Notre modèle numérique est constitué d'une base de règles floues conjonctives activées en parallèle et agrégées disjonctivement. Il repose sur la méthode de Nozaki-Tanaka qui répond en grande partie aux contraintes que nous fixe le système. Ensuite, grâce à l'interprétation des règles linguistiques floues, nous pouvons connaître le fonctionnement du modèle numérique, et valider le choix des paramètres composant le vecteur caractéristique et affiner les connaissances métiers associées, le cas échéant.

Cette démarche nous a permis d'obtenir un modèle numérique dont le nombre total de règles est le plus réduit possible, améliorant ainsi grandement les temps de calcul et permettant ainsi de respecter les contraintes de temps du système industriel. De plus, le modèle numérique généré est cohérent vis-à-vis des connaissances des experts. Ainsi, nous pouvons exprimer dans un même référentiel, les connaissances métiers et les informations issues des données. Les systèmes de règles synthétisés à partir du modèle numérique et ceux construits depuis le modèle symbolique, peuvent être ainsi comparés. Cette démarche montre ainsi un intérêt certain quant aux performances en terme de taux d'identification et en terme de performances de traitement de l'information.

Nous avons vu également que la mise en relation du modèle numérique et du modèle symbolique n'est possible que par la construction d'un modèle intermédiaire modélisant les connaissances de l'expert vision, lien entre le métier et l'application industrielle. Néanmoins, pour obtenir une méthodologie complète, il nous manque actuellement de pouvoir créer de manière systématique la structuration hiérarchique de l'arborescence des nœuds d'inférence. C'est pourquoi nos futurs travaux porteront sur la formalisation de la connaissance exprimant les relations hiérarchiques existant entre les attributs identifiant les défauts dans le métier du bois. De même, nous sommes amenés à considérer également le mécanisme d'analyse de l'expert pour l'identification (décomposition grossier-fin, sous-classes de défauts, ...). Cette étude peut nous amener à repenser le mécanisme d'inférence utilisé et à envisager la mise en place d'une base de règles implicatives.

Le modèle symbolique des connaissances métier peut être aussi un moyen de mettre en relation les différents industriels du

domaine du bois. En effet, il pourrait permettre l'utilisation d'un vocabulaire commun et ainsi faciliter la circulation de la connaissance [Blaise]. Par exemple, le terme « nœud cerné » de l'industriel 1 ne désigne-t-il pas le même défaut que le terme « nœud non adhérent » de l'industriel 2 ?

Quant au modèle symbolique du procédé de vision, il permet de capitaliser l'expérience des concepteurs de ce système. La base de données générée permet de recenser les caractéristiques utilisées et de les relier aux défauts pour lesquels elles sont pertinentes. Ainsi, en fonction des capteurs d'entrée, des essences de bois et d'autres paramètres d'environnement, l'expert vision peut s'appuyer sur ces connaissances pour concevoir ou configurer un nouveau système.

La méthode NIAM actuellement utilisée pour la formalisation des connaissances expertes montre cependant ses limites dès lors qu'il faut exprimer des contraintes autres qu'ensemblistes telles que des contraintes dites « procédurales ». Entrent dans ce cas les contraintes floues qui ne peuvent pas être directement décrites en NIAM. Cette lacune de NIAM ne nous permet donc pas d'intégrer complètement les modèles ainsi créés. Pour ce faire, nous envisageons de reprendre les travaux de Zgorzelski sur une extension floue de NIAM (« Fuzzy NIAM ») [Zgorzelski], de les appliquer à une modélisation floue des modèles symboliques expert métier et expert procédé vision et de les transposer en langage U.M.L. pour les faciliter d'intégration que ce langage permet (en particulier, entre attributs et méthodes/règles associés à une même classe).

Par ailleurs, ces travaux appliqués au domaine de l'identification de défauts par vision sur des planches de bois pourraient être généralisés à tout autre domaine d'application où il existe un cloisonnement entre une expertise métier (le qualitatif) et les informations issues de capteurs (le quantitatif).

7. Remerciements

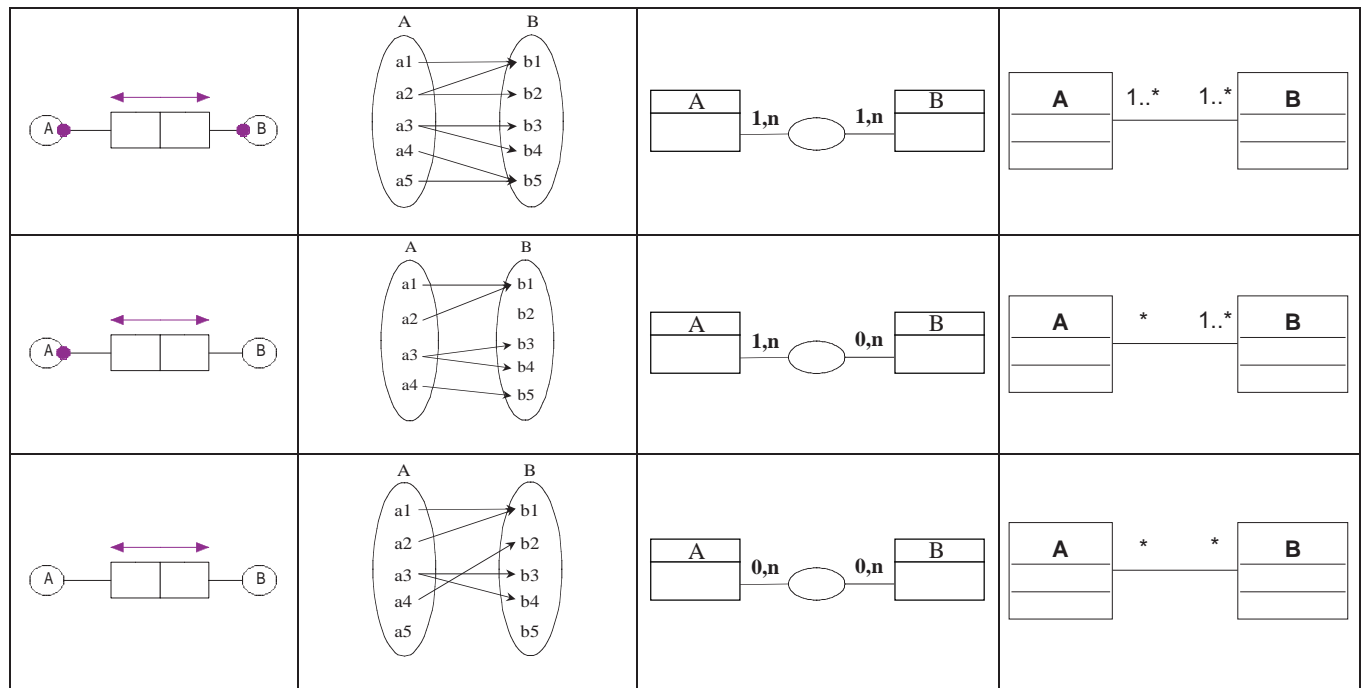
Nous tenons à remercier Monsieur Raphaël Vogrig, co-fondateur de la société LuxScan Technologies pour la mise à disposition des données techniques et pour sa collaboration dans l'établissement des différents modèles de connaissances.

Annexe 1

Tableau de correspondance entre le formalisme NIAM/ORM et les formalismes ensembliste, entité-relation et diagrammes de classes en U.M.L.

NIAM/ORM	ENSEMBLISTE	ENTITÉ-RELATION	UML
	<p>Bijection</p>		
	<p>Injection</p>		
	<p>Surjection</p>		
	<p>Application de A vers B</p>		

5



Références

[Attiogbé] C. ATTIIOGBE, H. HABRIAS, A. VAILLY, «Exercices de style», *rapport interne n°139 de l'Institut de Recherche Informatique de Nantes*, 1996.

[Berthold] M.R. BERTHOLD, « Mixed fuzzy rule formation » , *Int. Jour. of Approximate Reasoning*, vol. 32, 2003, p. 67-84.

[Blaise] J.C. BLAISE, « Apport d'une modélisation de l'information normative à l'intégration des règles de sécurité des machines en conception », *Thèse d'université*, Université Henri Poincaré, NANCY 1, 2000.

[Bouchon-Meunier] B. BOUCHON-MEUNIER, « La logique floue et ses applications », *Addison-Wesley*, 1995.

[Cordon 1] O. CORDON, M.J. DEL JESUS, F. HERRERA, « A proposal on reasoning methods in fuzzy rule-based classification systems », *Int. Jour. of Approximate reasoning*, 20, 1999, p. 21-45,

[Cordon 2] O. CORDON, F. GOMIDE, F. HERRERA, F. HOFFMANN, L. MAGDALENA, « Ten years of genetic fuzzy systems: current framework and new trend », *Fuzzy Sets and Systems*, Vol. 141, 2004, p 5-31.

[Dubuisson] B. DUBUISSON, «Diagnostic, intelligence artificielle et reconnaissance des formes», *édition Hermès*, 2001.

[Dubois 1] D. DUBOIS, H. PRADE, « What are Fuzzy rules and how to use them », *Fuzzy Sets and Systems*, Vol 84, 1996, p 169-185.

[Dubois 2] D. DUBOIS, H. PRADE, « On the validation of fuzzy knowledge bases », *Fuzzy reasoning in information, decision and control system*, Kluwer, Dordrecht, 1994, p. 31-49.

[Dubois 3] D. DUBOIS, H. PRADE, « Fuzzy rules in knowledge-based systems – Modelling gradedness, uncertainty and preference », *An introduction to fuzzy logic application in intelligent systems*, Kluwer, Dordrecht, 1992, p. 45-68.

[Foo] S.Y. FOO, « A rule-based machine vision for fire detection in aircraft dry bays and engine compartments », *Knowledge-Based Systems*, Vol. 9, n° 8, 1996, p. 531-540.

[Habrias] H. HABRIAS, «Le modèle relationnel binaire. Méthode I.A. (NIAM) », *éditions Eyrolles*, 1998.

[Halpin 1] T.A. HALPIN, «Object Role Modeling (ORM/NIAM)», *Handbook on Architectures of Information Systems*, 1998.

[Halpin 2] T.A. HALPIN, « Information Modelling and Relational Databases », published by *Morgan KAUFMAN edition*, 2001.

[Ishibuchi 1] H. ISHIBUCHI, K. NOZAKI, H. TANAKA, «Distributed representation of fuzzy rules and its application to pattern classification », *Fuzzy Sets and Systems*, Vol. 52, 1992, p 21-32.

[Ishibuchi 2] H. ISHIBUCHI, T. NAKASHIMA, T. MURATA, « Voting in fuzzy rule-based classification for pattern classification problems », *Fuzzy Sets and Systems*, Vol. 103, 1999, p 223-238.

[Ishibuchi 3] H. ISHIBUCHI, K. NOZAKI, N. YAMAMOTO, H. TANAKA, « Construction of fuzzy classification systems with rectangular fuzzy rules using genetic algorithms », *Fuzzy sets and systems*, Vol. 65, 1994, p. 237-253.

[Kaufmann] A. KAUFMANN, « Introduction à la théorie des sous-ensembles flous », *édition Masson*, 1975.

[Kauppinen] H. KAUPPINEN, H. RAUTIO, O. SILVEN, « Non-segmenting defect detection and SOM based classification for surface inspection using color vision », *Conf. on Polarization and Color Techniques in Industrial Inspection (SPIE 3826)*, Munich, Germany, 1999, p. 270-280.

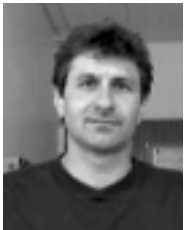
[Lampinen] J. LAMPINEN, S. SMOLANDER, M. KORHONEN, « Wood surface inspection system based on generic visual features », *International Conference on Artificial Neural Networks ICANN'95*, Paris, 1995.

[Mazaud] C. MAZAUD, V. BOMBARDIER, P. LHOSTE, «Modélisation de connaissances pour l'identification de défauts sur des planches de bois » , *ORASIS 2003*, Gérardmer (France), 19-23 mai 2003, pp 269-277.

[Nozaki] K. NOZAKI, H. ISHIBUCHI, H. TANAKA, « A Simple but powerful heuristic method for generating fuzzy rules from numerical data », *Fuzzy sets and systems*, Vol. 86, 1997, p. 251-270.

[Pham] D.T. PHAM, S. SAGIROGLU, «Training multilayered perceptrons for pattern recognition: a comparative study of four training algorithms », *International Journal of Machine Tools & Manufacture*, Vol. 41, n° 3, 2001, p. 419-430.

- [Ruspini] E.H. RUSPINI, «On the semantics of fuzzy logic», *Int. Jour. of Approximate reasoning*, Vol. 5, 1991, p. 45-88.
- [Ughetto] L. UGHETTO, «Les systèmes à base de règles floues. Vérification de la cohérence et méthodes d'inférences», *Doctorat de l'université Paul Sabatier*, INRIT, Toulouse, janv 1998.
- [Shen] Q. SHEN, A. CHOUCHOULAS, «A rough-fuzzy approach for generating classification rules», *Pattern Recognition*, Vol. 35, 2002, p. 2425-2438.
- [Siaud] STAUD O., «Contribution à l'intégration de connaissances expertes dans un système de reconnaissance de formes : application à l'identification de défauts dans le bois», *rapport de DEA*, UHP-Nancy 1, 2002.
- [Sugeno] M. SUGENO, « An introductory survey of fuzzy control », *Information Sciences*, Vol. 36, 1985, p. 59-83.
- [Wang] C.S. WANG, X.K. GONG, « A fuzzy approximate reasoning model for a rule-based system in laser threat recognition », *Fuzzy Sets and Systems*, Vol. 96, #2, 1998, p. 139-146.
- [Wen] Z. WEN, Y. TAO, «Building a rule-based machine-vision system for defect inspection on apple sorting and packing lines », *Expert Systems with Applications*, Vol. 16, 1999, p. 307-313.
- [Zadeh 1] L.A. ZADEH, «The concept of linguistic variable and its application to approximate reasoning», *Information sciences*, Vol. 8, 1975, p. 199-249 et p. 301-357.
- [Zadeh 2] L.A. ZADEH, «Outline of a new approach to analysis of complex systems and decision processes», *IEEE Trans. On Systems, Man and Cybernetics*, Vol. 3, 1973, p. 28-44.
- [Zadeh 3] L.A. ZADEH, «The calculus of fuzzy if/then rules», *AI Expert*, Vol. 7 (3), 1992, p. 23-27.
- [Zgorzelski] M. ZGORZELSKI, Z. ZALEWSKI, «Fuzzy NIAM for real world data modelling», *I.S.A.T.A. 29th International Symposium on Automotive Technology and Automation*, Florence, 1996.



Vincent **Bombardier**

Vincent BOMBARDIER est Maître de Conférences à l'Université Henri Poincaré Nancy 1, en poste à l'Ecole Supérieure d'Informatique et Applications de Lorraine. Ses travaux de recherche, effectués au sein du Centre de Recherche en Automatique de Nancy dans le thème SYStèmes de Production Ambiants, combinent Traitement d'Images et application de la Théorie des Ensembles Flous. Actuellement, il oriente ses travaux vers l'intégration de Systèmes de Vision dans les Systèmes Industriels avec notamment la modélisation des connaissances métiers (Produit, environnement,...) pour leurs prise en compte dans des problématiques de reconnaissance de formes



Cyril **Mazaud**

Cyril MAZAUD, diplômé de l'IUP de Génie Mécanique et Productique (UHP Nancy 1) en Juin 2002, a reçu son DEA en Production Automatisée (UHP Nancy 1 – ENS Cachan) en Septembre 2003. Il prépare actuellement son Doctorat dans le cadre d'une collaboration franco-luxembourgeoise entre le Centre de Recherche en Automatique de Nancy (CRAN), l'Université du Luxembourg (représenté par le LIASIT) et la société LuxScan Technologies. Les recherches actuelles portent sur la contribution à la modélisation de l'information et sur l'intégration d'une méthode floue dans un système temps réel de reconnaissance des défauts du bois.



Pascal **Lhoste**

Pascal LHOSTE est Maître de Conférences habilité à diriger des recherches à l'Université Henri Poincaré Nancy 1. Il anime au sein du Centre de Recherche en Automatique de Nancy une thématique centrée sur la Modélisation du Processus d'Ingénierie des Systèmes Automatisés de Production. Après avoir porté sur les modèles utilisés dans ce type de processus, ses travaux se sont orientés vers des modèles aptes à formaliser des connaissances métiers (modèles de référence, patterns, normes, ...) et à les intégrer dans les différentes phases de conception des systèmes automatisés de production.