



HAL
open science

Premiers travaux relatifs au concept de matière communicante : Processus de dissémination des informations relatives au produit

Sylvain Kubler

► **To cite this version:**

Sylvain Kubler. Premiers travaux relatifs au concept de matière communicante : Processus de dissémination des informations relatives au produit. Informatique ubiquitaire. Université Henri Poincaré - Nancy I, 2012. Français. NNT: . tel-00759600

HAL Id: tel-00759600

<https://theses.hal.science/tel-00759600v1>

Submitted on 1 Dec 2012

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

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



**UNIVERSITÉ
DE LORRAINE**

FACULTE DES SCIENCES & TECHNOLOGIES

U.F.R. : Sciences et Techniques Mathématiques, Informatique et Automatique

Ecole Doctorale : IAEM Lorraine

Département de Formation Doctorale : Automatique

CENTRE DE RECHERCHE
EN AUTOMATIQUE DE
NANCY



CNRS UMR 7039

Thèse

présentée pour l'obtention du titre de

Docteur de l'Université de Lorraine

en Sciences, spécialité Automatique,
Traitement du Signal et Génie Informatique

par **Sylvain KUBLER**

Premiers travaux relatifs au concept de matière communicante : Processus de dissémination des informations relatives au produit

Soutenue publiquement le 7 Novembre 2012

Membres du jury :

<i>Rapporteurs :</i>	Pr. Kary FRÄMLING	Professeur, Université de Technologie de Helsinki, AALTO
	Pr. Damien TRENTESAUX	Professeur, Université de Valenciennes et du Hainaut-Cambrésis, TEMPO
	Pr. Daniel JOLLY	Professeur, Université de Lille Nord, LGI2A
<i>Examineurs :</i>	Pr. Éric RONDEAU	Professeur, Université de Lorraine, CRAN (directeur de thèse)
	Pr. André THOMAS	Professeur, Université de Lorraine, CRAN (directeur de thèse)
	Dr. William DERIGENT	Maître de Conférences, Université de Lorraine, CRAN (co-directeur de thèse)
	M. Joseph RABITA	Ingénieur, Directeur de l'Ingénierie Stockage et Réseaux, Orange

Centre de Recherche en Automatique de Nancy (CRAN)

Campus Sciences, B.P. 70239, 54506 Vandœuvre-lès-Nancy

Remerciements

Je tiens à remercier l'ensemble des personnes qui ont contribué à ces années d'études et particulièrement, aux travaux de cette thèse. Tout d'abord, Éric Rondeau, professeur au CRAN, qui m'a fait découvrir les différentes facettes du travail de chercheur. Je le remercie pour ses qualités humaines et professionnelles mais surtout pour la confiance qu'il m'a accordée et qui a fortement contribué à mon épanouissement durant cette thèse. Je souhaite également remercier André Thomas, professeur au CRAN, pour le temps qu'il m'a accordé, les discussions scientifiques qui m'ont poussées et ont fait avancer cette thèse, les discussions moins professionnelles mais tout aussi importantes à mes yeux. Je voulais également le remercier pour ses qualités d'encadrement qui m'ont permises d'acquérir de nouvelles connaissances dans bons nombres de domaines. Enfin, je tenais à remercier William Derigent, Maître de conférence au CRAN, qui a été pour moi une personne clé de la réussite de cette thèse, de part sa motivation, sa joie de vivre, son soutien inconditionnel et pour avoir accepté de m'épauler, quelle que soit l'heure et le jour. Je m'estime heureux et chanceux d'avoir eu ce trio d'encadrants.

Je souhaite remercier M. Damien Trentesaux, Professeur à l'Université de Valenciennes et du Hainaut-Cambrésis, M. Daniel Jolly, Professeur à l'Université de Lille Nord, M. Kary Främling, Professeur à l'Université de Technologie de Helsinki pour avoir accepté d'être rapporteur de ce travail. Je remercie M. Joseph Rabita, directeur de l'Ingénierie stockage et réseaux chez Orange qui a accepté d'examiner les travaux. Je remercie particulièrement M. Kary Främling qui me fait l'honneur et le privilège de me recevoir pour une durée d'un an dans son laboratoire, prolongeant ainsi mon rêve de chercheur.

Je voulais également exprimer mes sincères remerciements à Alexandre Voisin pour l'aide si précieuse qu'il a su m'apporter durant cette dernière année de thèse. J'en profiterai, de fait, pour remercier mon encadrant de Master, Jean-Philippe Georges, qui est aujourd'hui un ami que j'estime énormément et qui a également su me donner goût à la recherche (bien qu'il soit têtu comme un Vosgien...). Un grand merci à mon collègue de bureau, Jérémy Robert, avec qui j'ai tissé une amitié toute particulière. Je n'oublie bien évidemment pas les personnes et amis que j'ai côtoyés durant cette thèse : Thierry Divoux, Francis Lepage, Nicolas Krommenacker, Vincent Lecuire, Michael David ; les personnes ayant écouté ou supporté les histoires sur Saint-Quirin (un des plus beaux villages de France) : Gabriela, Pascale, Pierre, Mario, Leila, Arnould, Xavier, Damien, Nicolas, Fabien, Alexandre, Thomas, Idriss, Carlos, Jérémy J., Benjamin, Baptiste,...

Ces remerciements ne seraient pas complets s'il n'y avait pas quelques lignes pour ma famille qui m'a suivi tout au long de ces huit années d'études supérieures : mes parents pour leur amour et leur soutien inconditionnels, mes grands frères qui ont toujours été pour moi des modèles.

Je ne peux terminer ces remerciements sans quelques mots pour celle qui, certes involontairement, m'a permis de mettre un pied dans la recherche. Pour ses encouragements, son aide, sa passion pour les langues étrangères, si importantes pour un chercheur, qu'elle a su me communiquer. Peu de mots peuvent traduire ma reconnaissance pour ce qu'elle a fait, alors je n'en n'utiliserai qu'un seul : merci !

Table des matières

Introduction	vii
1 Les objets ou produits intelligents - communicants	1
1.1 Concepts et champs d'applications	2
1.1.1 Vers une société d'objets communicants	2
1.1.2 Panel d'applications	3
1.2 Cycle de vie des produits et gestion associée	5
1.2.1 Historique du concept PLC	5
1.2.2 Système PLM	6
1.2.3 Produits intelligents dans la littérature	10
1.2.4 Bénéfices générés par les produits intelligents	20
1.2.5 Vers un nouveau paradigme de « matière communicante »	21
1.3 Proposition du concept de « matière communicante »	23
1.3.1 Axes de recherche émergents	23
1.4 Positionnement	24
2 Dissémination de l'information : principes et fondements	27
2.1 Architecture et distribution des informations relatives aux produits	28
2.2 Introduction aux systèmes de bases de données distribuées	30
2.3 Étapes clés de la distribution de l'information	31
2.3.1 Étape de fragmentation	31
2.3.2 Étape d'allocation	33
2.3.3 Conclusion	37
2.4 Dissémination des données en environnement mobile	37
2.4.1 Architectures, principes et mécanismes	37
2.4.2 Méthodes traitant de la réduction des données sur les équipements mobiles	41
2.5 Apports de la littérature dans un contexte « système matière communicante »	42
2.6 Proposition d'un processus de dissémination des informations relatives au produit	43
2.6.1 Entrées du processus de dissémination des informations	45
2.6.2 Mise en œuvre de processus de dissémination des informations dans le PLC	47

3	Étape 2 : Sélection des informations pertinentes à stocker sur le produit	49
3.1	Algorithme d'identification des « informations produit »	50
3.1.1	Description de l'algorithme	50
3.1.2	Développement de l'algorithme RetrievalData	52
3.1.3	Algorithme détaillé par l'exemple	56
3.2	Méthode d'évaluation de la pertinence des informations : Chan-Roddick	58
3.2.1	Introduction	58
3.2.2	Calcul de la priorité relative ρ_x	60
3.2.3	Calcul de la priorité assignée $\phi_x(l)$	62
3.2.4	Synthèse du modèle d'évaluation proposé par Chan et Roddick	66
3.3	Méthode d'évaluation de la pertinence des informations : AHP-Floue	67
3.3.1	Introduction	68
3.3.2	Point 1 : structure AHP de l'étude	76
3.3.3	Point 2 : collecte et agrégation des informations	76
3.3.4	Point 3 : calcul de la matrice de jugement flou \tilde{A}	86
3.3.5	Point 4 : calcul de la matrice de performance floue \tilde{H}	86
3.3.6	Point 5 : classification des alternatives/data items	87
3.4	Conclusion de l'étape 2	91
4	Étape 3 : Stockage et relecture des informations sur la matière communicante	93
4.1	Conception de la matière communicante	94
4.2	Conception de l'architecture de communication	95
4.2.1	Deux types d'architecture pour lire/écrire des données sur la matière	95
4.2.2	Dissémination des informations au sein de la matière communicante	97
4.3	Outillage de l'expérimentation	100
4.4	Conclusion de l'étape 3	102
5	Application du processus de dissémination des informations au cas d'un textile	105
5.1	Cycle de vie du textile communicant	106
5.1.1	Définition des points d'écriture et de lecture dans le PLC	107
5.2	Exécution des phases d'écriture et de lecture	109
5.2.1	Principe de comparaison des approches Chan-Roddick / AHP-Floue	109
5.2.2	Point d'écriture n°1	110
5.2.3	Point de lecture n°1 : activité « Découpe »	123
5.2.4	Point d'écriture n°2	126
5.3	Synthèse et conclusion de l'étude	131
5.3.1	Conclusions des études comparatives de l'Annexe VII	131
5.3.2	Tableau de synthèse	133

Conclusion générale et perspectives	135
Annexe I : Déclinaison des problématiques scientifiques et technologiques	139
Annexe II : Analyse détaillée des approches de bases de données distribuées	149
Annexe III : Détail des Algorithmes	159
Annexe IV : Détail des calculs liés à l'approche AHP-Floue	165
Annexe V : Prototypage d'un textile communicant	179
Annexe VI : Localisation des tags RFID basée sur la méthode de la corde	183
Annexe VII : Comparaisons supplémentaires des approches de Chan-Roddick et AHP-Floue	185
Annexe VIII : Taux de contribution au projet	199
Glossaire	201
Tables des figures	202
Liste des publications	209
Notice bibliographique	211

Introduction

Cette thèse aborde le problème de la gestion des informations liées au produit tout au long de son cycle de vie, par l'exploitation du concept de produit intelligent. Depuis de nombreuses années, plusieurs communautés telles que IMS (*Intelligent Manufacturing Systems*), HMS (*Holonic Manufacturing System*) ou encore PDS (*Product Driven Systems*) ont suggéré l'utilisation de produits intelligents pour rendre les systèmes adaptables et adaptatifs (sensibles au contexte). De nombreux travaux dans la littérature ont montré les bénéfices pouvant être réalisés avec de tels systèmes/produits, au niveau économique (Kärkkäinen, Holmström, Främling and Arto, 2003), au niveau de la traçabilité des produits (Johnson, 2002), au niveau du partage des informations ou encore au niveau de l'optimisation des procédés de fabrication (Tajima, 2007). Néanmoins, certains auteurs tels que (McFarlane et al., 2012) soulignent qu'il y a encore un grand nombre de questions ouvertes comme la collecte des informations liées au produit, leur stockage ou encore la dissémination et la gestion de ces informations tout au long de leur cycle de vie. On peut encore s'interroger sur les manières de coupler les systèmes de contrôle des ressources physiques avec les systèmes d'informations et de prises de décisions nécessaires aux systèmes industriels et logistiques d'aujourd'hui. De nouvelles architectures sont alors à concevoir, en prenant en compte l'évolution des technologies Auto-ID en particulier. Les produits dits « intelligents » réalisent majoritairement leurs fonctions de manière « discrète ». Le terme « discret » signifie que les produits embarquent un composant unique pour assurer la (les) fonction(s) embarquée(s). Le plus souvent, le produit intègre une puce RFID unique servant de pointeur pour accéder à l'information déportée sur le réseau (p. ex. les plateformes EPCglobal, Dialog, WWAI ou encore les plateformes agents) (Meyer et al., 2009). Ce type de produit a cependant des limites : i) le flux d'informations relatif au produit est rompu si la puce RFID est endommagée, ii) la transformation du produit peut générer des trous dans le système d'information (p. ex. si le produit est transformé/découpé, certaines pièces résultantes seront vierges d'informations), iii) les technologies RFID actuelles sont limitées (capacité mémoire, débit, accessibilité aux informations, etc.). Après avoir vu les produits intelligents comme des produits physiques associés à leur information via un agent ou un produit virtuel, (Thomas, 2009) a proposé un nouveau paradigme qui change radicalement la manière de voir le produit et la matière. Ce nouveau concept consiste à donner la faculté au produit d'être intrinsèquement et intégralement « communicant ». « Intrinsèquement » signifie que le produit est « par essence » communicant ; d'un point de vue purement conceptuel, toute la matière est communicante. L'idée forte et futuriste de ce concept est d'imaginer que la matière soit com-

municante au niveau moléculaire. Même si cette approche est encore loin d'être réaliste (au regard des limitations technologiques), certaines recherches semblent prometteuses, comme les travaux de (Kubler et al., 2011b) ou encore (Brun et al., 2009)¹. Prendre comme postulat de rendre la matière communicante ouvre un champ de recherche considérable, tant d'un point de vue technologique que scientifique. C'est dans ce contexte que cette thèse a été initiée.

Le Chapitre 1 retrace l'évolution des produits/objets intelligents et communicants dans notre société, ainsi que les travaux majeurs menés autour de ce concept et particulièrement, ceux liés à la gestion du cycle de vie du produit. Ce chapitre pose ensuite les limites actuelles des produits en mode « discret » et introduit le concept de « matière communicante ». Ce changement de paradigme débouche sur un champ de recherche considérable et nécessite une méthode rigoureuse d'analyse des exigences, en amont, afin de positionner de manière précise et concise les travaux de recherche entrepris dans cette thèse (tout en identifiant les problématiques à investiguer dans l'avenir). Les résultats issus de l'analyse ont ainsi permis de limiter le périmètre de la thèse à la gestion des informations liées au produit durant l'ensemble de son cycle de vie.

Dans le cadre de nos travaux, la matière constituant le produit est considérée comme un vecteur d'informations mobile, ayant des capacités similaires à celle d'une base de données. De fait, concevoir un système d'information comprenant des produits composés de matière communicante revient à étudier un système de bases de données distribuées et mobiles. En ce sens, le Chapitre 2 présente un état de l'art sur les systèmes de bases de données distribuées en abordant notamment les mécanismes de mise à jour des informations, les mécanismes de répllication et les bases de données mobiles. Au vu de l'état de l'art, des conclusions quant aux architectures et aux mécanismes de maintien de la cohérence dans les bases de données « matière communicante » sont esquissées. De plus, cette étude permet de faire émerger la notion d'informations sensibles au contexte d'usage. En effet, les informations embarquées sur des bases de données mobiles sont fortement dépendantes de leur utilisation future. Il en est de même pour les produits portant leurs propres informations. Cette notion de sélection d'informations basée sur le contexte est le pilier central de nos travaux, et justifie les deux questions majeures auxquelles cette thèse apporte des réponses, à savoir :

- quelles informations doivent être stockées sur le produit durant son cycle de vie et où doivent-elles être localisées sur la matière ?
- quelle est la pertinence à les stocker sur le produit et selon quels critères ?

En ce sens, l'approche proposée par (Chan and Roddick, 2003) est intéressante puisqu'elle tient compte de la sémantique de la donnée qui est trop souvent négligée dans les approches de réduction des données. Parallèlement, les méthodes de décisions multi-critères classiques (AHP, agrégation floue,...) sont potentiellement exploitables moyennant des adaptations à notre contexte, car elles permettent très simplement de réaliser des choix dans un ensemble d'alternatives. À partir de plusieurs

1. Les travaux menés dans (Brun et al., 2009) ne traitent pas du concept de « matière communicante », mais développent des technologies de miniaturisation des puces RFID.

de ces approches (Chan et Roddick, AHP-floue), nous proposons un processus visant à disséminer les informations entre/dans les « produits communicants » et les « bases de données traditionnelles ». Ce processus est composé de trois étapes clés :

- l'étape 1 correspond à la mise en place de l'architecture de bases de données propres à chaque acteur du cycle de vie du produit. Cette étape fait principalement appel aux mécanismes et travaux présentés dans les états de l'art du Chapitre 2,
- l'étape 2 vise à identifier au sein de ces bases de données les informations qui sont liées au produit, puis à évaluer leur pertinence à être (ou ne pas être) répliquées/embarquées sur le produit communicant. Cette évaluation s'appuie dans un premier temps sur l'approche de (Chan and Roddick, 2003). Cependant, cette approche est difficile à mettre en œuvre lorsque les informations présentent une certaine « complexité » relative, par exemple, à la multiplicité des points de vue sur le produit. Par conséquent, une technique basée sur la logique *AHP-Floue* est développée pour pallier ce problème. Cette étape fait l'objet du Chapitre 3,
- l'étape 3 intègre les outils logiciels et matériels permettant de stocker physiquement les informations sur la matière communicante, puis de les reconstruire (lors de la relecture de la matière). La particularité du concept de matière communicante telle que nous l'avons concrétisée dans un premier prototype de textile communicant, nécessite de spécifier une architecture de communication adéquate et des mécanismes de localisation et de répartition des données sur la matière qui sont décrits dans le Chapitre 4.

Le Chapitre 5 présente un cas d'étude dans lequel le prototype de textile communicant conçu est utilisé pour valider le processus de dissémination des informations. Nous identifions pour différentes étapes du cycle de vie du textile, les informations jugées pertinentes à y être stockées, et nous procédons physiquement au stockage et à la relecture de ces informations via les outils logiciels et matériels présentés dans le Chapitre 4. Ce cas d'étude est également l'opportunité d'étudier et de comparer le modèle d'évaluation proposé par (Chan and Roddick, 2003) avec l'approche AHP-Floue.

La conclusion présente un bilan du processus de dissémination des informations, quantifie la contribution de cette thèse au projet global du concept de « matière communicante » et développe ainsi les perspectives futures de recherche et d'action.

Chapitre 1

Les objets ou produits intelligents - communicants

Sommaire

1.1 Concepts et champs d'applications	2
1.1.1 Vers une société d'objets communicants	2
1.1.2 Panel d'applications	3
1.2 Cycle de vie des produits et gestion associée	5
1.2.1 Historique du concept PLC	5
1.2.2 Système PLM	6
1.2.3 Produits intelligents dans la littérature	10
1.2.4 Bénéfices générés par les produits intelligents	20
1.2.5 Vers un nouveau paradigme de « matière communicante »	21
1.3 Proposition du concept de « matière communicante »	23
1.3.1 Axes de recherche émergents	23
1.4 Positionnement	24

La première section de ce chapitre offre une vision de l'univers des objets communicants, leur évolution à travers le temps, ainsi que les concepts sous-jacents sur lesquels ils reposent.

Tout objet ou produit est confectionné, manipulé et utilisé par de nombreux acteurs tout au long de sa vie. Les produits peuvent être complexes et les informations qu'ils génèrent durant leur vie (p. ex. durant leur conception, leur utilisation, etc.) sont souvent éparpillées dans divers systèmes et entreprises. Bien évidemment, il est nécessaire de gérer ces informations, ces systèmes dans le but de prendre les meilleures décisions durant tout le cycle de vie du produit (PLC - *Product Life Cycle*). Cette évidence a conduit à plusieurs concepts dont le PLM (*Product Lifecycle Management*). L'histoire de ces concepts est retracée dans la section 1.2 où nous discutons de la nécessité de gérer les informations des produits durant leur vie et des problèmes que cela engendre. Cette section fait également un point sur les solutions et plates-formes existantes qui utilisent des produits intelligents pour répondre à ces problèmes. Certaines limites de ces produits intelligents sont toutefois soulignées et nous conduisent à introduire un concept innovant proposé par le laboratoire en 2009 appelé « matière communicante » (Thomas, 2009).

La section 1.3 pose les bases d'un tel concept, puis propose une démarche rigoureuse d'analyse dans le but de hiérarchiser les objectifs à atteindre et de les décliner en diverses problématiques technologiques et de recherches. Cette hiérarchisation nous permet ainsi de positionner les travaux entrepris dans cette thèse et de les formuler.

1.1 Concepts et champs d'applications

1.1.1 Vers une société d'objets communicants

Au fil du temps, la notion de « système informatique » a évolué. Il y a eu tout d'abord l'époque où plusieurs personnes partageaient un même ordinateur, puis l'époque où chaque personne disposait de son propre ordinateur et enfin, est arrivée l'époque où chaque personne dispose d'une panoplie d'équipements informatiques. Cela peut aller du simple objet muni d'une puce RFID (identifiant renvoyant vers une base de données ou un agent logiciel) jusqu'au PDA, PC portable, en passant par le Nabaztag¹. La FIGURE 1.1(a) illustre l'évolution de notre société vers cette société d'objets communicants. Les avancées technologiques ont en effet favorisé la miniaturisation des composants électroniques et ont permis d'embarquer de l'intelligence dans tout type d'objet. Dès lors, (van Wesemael, 2005) pose la question suivante : ne pourrait-on pas imaginer un réseau reliant en permanence tous nos appareils pour qu'ils puissent collaborer, parler spontanément ensemble et avec le reste du monde, constituer collectivement un seul ordinateur virtuel qui serait la somme de leurs intelligences, de leurs connaissances et de leurs savoir-faire respectifs ? Ce type d'interrogation a donné lieu, à ce jour, à de nombreux concepts dont les plus connus sont rappelés ci-après.

Citons en premier lieu, le concept d'informatique ubiquitaire/pervasive (*Ubiquitous/Pervasive Computing*) rendant l'informatique invasive dans chaque objet, leur permettant ainsi d'entrer en relation de façon intelligente avec leur environnement et les autres objets. Le terme *Ubiquitous Computing* a été introduit par Marc Weiser avec cette définition largement reprise dans la littérature :

« *the most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.* », (Weiser, 1991)

Ubiquitous/Pervasive Computing a été traduit par « informatique diffuse », « informatique omniprésente » ou « informatique ubiquitaire » (Banâtre et al., 2007). Cette traduction a par la suite évolué vers celle plus générique d'Intelligence ambiante, où il est courant de parler de communication ambiante, de calcul ambiant ou bien de système diffus. L'objectif est de réaliser un espace communicant, sans contrainte de structure, permettant à des utilisateurs et à des équipements d'interagir sans connaissance préalable du contexte. Une caractéristique forte de ces environnements est de permettre à ces technologies de se « fondre » dans l'environnement, jusqu'à se confondre avec lui. On parle dès lors de *disappearing computing* (Streitz and Nixon, 2005).

1. « Nabaztag » signifie « lapin intelligent » en arménien. Créé par la société française *Violet* (Peters, 2006), le Nabaztag est un lapin communicant qui émet des messages vocaux ou lumineux et peut également remuer les oreilles. Il se connecte à internet par Wi-Fi et offre une multitude de services (voir : <http://fr.wikipedia.org/wiki/Nabaztag>).

Plus récemment, le concept de l'Internet des objets (*Internet of Things* - IoT) a vu le jour avec les avancées du Web. Ce terme a été introduit par Kevin Ashton en 1998 (Santucci, 2009). Une formulation (consensuelle) de l'IoT a été proposée par (Tan and Wang, 2010), le définissant comme :

« des objets ayant des identités et des personnalités virtuelles, opérant dans des espaces intelligents et utilisant des interfaces intelligentes pour se connecter et communiquer au sein de contextes d'usages variés ».

IoT est en quelque sorte un monde où aussi bien les objets physiques que les données et environnements virtuels interagissent entre eux dans le même espace-temps (Sundmaecker et al., 2010). Il est difficile de définir une frontière claire entre tous ces concepts (IoT, système pervasif, système ambiant, ...), car ils tendent tous vers un objectif commun, celui de donner « vie » aux objets, de les rendre interactifs et collaboratifs au sein de l'environnement, créant ainsi des environnements artificiels qui nous aident de manière proactive et réactive.

On parle également d'intelligence artificielle comme introduit pour la première fois en 1955 par (McCarthy et al., 2006). Le cadre de l'intelligence artificielle peut se définir comme un environnement composé de réseaux informatiques, d'interfaces Hommes-Machines (IHM) et Machines-Machines (IMM), de capteurs/actionneurs/contrôleurs, le tout associé à des concepts tels que précédemment introduits (*Ubiquitous computing*, etc.). La FIGURE 1.1(b), extraite de (Augusto, 2007), illustre ces propos. La finalité de tous ces concepts est l'augmentation de l'espace physique au moyen de services numériques formant un tout adapté à l'homme, c'est-à-dire adaptable et adaptatif.

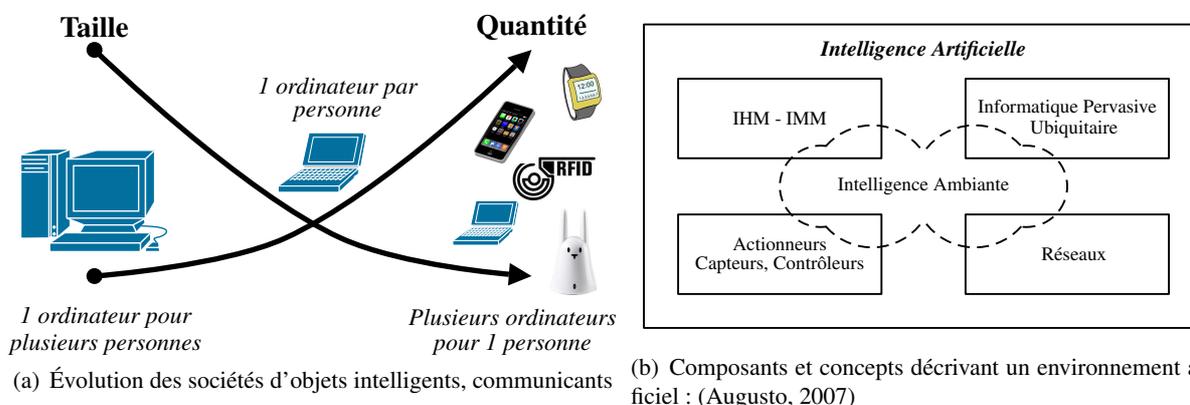


FIGURE 1.1: Évolution et environnement des objets intelligents - communicants

1.1.2 Panel d'applications

Les concepts précédemment énumérés mettent ainsi en œuvre des structures et des sociétés d'objets communicants/intelligents. La notion « d'intelligence » varie de la simple étiquette RFID associée au produit, à des unités de traitement embarquées beaucoup plus performantes. La littérature décrit de nombreuses applications et systèmes intelligents dans des domaines divers et variés. Citons :

- *le secteur médical* : de nombreuses applications utilisent les *e-textiles* (pour *electronic textiles* ou *smart textiles*) dans le but de percevoir les signaux vitaux du corps humain, puis d'offrir à la personne des services en conséquence (Bargagli et al., 1997; Schwarz and Van Langenhove, 2006),
- *les applications militaires et métiers à risques* : les *e-textiles* aident à lutter contre des dangers tels que le feu, les produits chimiques ou encore à offrir des services (p. ex. alerte en cas de problème, régulation de la température). Le *LifeShirt*, développé par l'Institut technologique de Géorgie (Park et al., 2002), permet par exemple d'avoir une connaissance de différents paramètres tels que la consommation calorique du soldat, sa température ou encore obtenir des informations concernant ses blessures (p. ex., localisation de l'impact de la balle).
- *le secteur de la domotique* : l'application la plus représentative est la *Smart Home* (Jiang et al., 2004) mettant en œuvre une multitude de capteurs, contrôleurs et actionneurs destinés à améliorer le confort des usagers ou encore d'assurer leur sécurité. (Bohn et al., 2005) disent « *the walls could have ears and these walls could talk* », laissant ainsi transparaître des perspectives de maintenance prédictive afin de prévenir d'éventuels problèmes (fissures, pertes de chaleur, ...). Citons encore l'exemple donné par (Ley, 2007) de la machine à laver capable de communiquer avec les vêtements et par ce biais, capable d'ajuster son cycle de lavage en fonction des informations échangées,
- *le secteur animalier* : il arrive qu'au-delà des objets, les animaux soient parfois instrumentés, le but étant le suivi des déplacements (d'une manière générale, la traçabilité des troupeaux/meutes) ou encore la gestion de cheptels, le suivi des gestations (à des fins d'accouplement) (Eradus and Jansen, 1999),
- *le secteur industriel* : l'instrumentation des matières premières, des produits semi-finis ou finis est également d'actualité en milieu industriel. Le but est principalement de faciliter l'accès à l'information, d'assurer une traçabilité de ces produits tout au long de leur processus de production ou de logistique et d'aider à l'interopérabilité² au sein de la chaîne d'approvisionnement (McFarlane et al., 2003; Kärkkäinen, Holmström, Främling and Arto, 2003). Cela concerne aussi bien le secteur automobile, que le secteur du bois, la chaîne du froid, le secteur agroalimentaire ou encore l'aéronautique.

Les objets/produits gravitant dans les champs d'applications précédemment cités sont souvent de natures différentes. En effet, un produit dans son état initial (p. ex. grume de bois), aussi bien que dans son état final (p. ex. un meuble) peut être instrumenté et utilisé pour diverses finalités. Il est cependant difficile d'évaluer le « niveau d'intelligence » d'un produit³ sachant qu'il existe des relations entre le niveau d'intelligence d'un environnement et le niveau d'intelligence des objets le composant. Néanmoins, une première différenciation peut être établie au regard des matières composant les produits. En effet, deux classes de matériaux intelligents existent. La première concerne les matériaux dépourvus d'électronique, considérés comme intelligents au vu de leur composition chimique. Ces matériaux réagissent

2. aptitude de deux partenaires à interagir efficacement et sans effort, que ce soit au plan organisationnel ou technique.

3. certains auteurs tels que (Meyer et al., 2009; McFarlane and Sheffi, 2003) ont proposé des modèles permettant de classer les produits intelligents. Ces modèles d'évaluation sont présentés dans la section 1.2.3.

aux stimuli de l'environnement à travers des lois physiques. On retrouve les matériaux piézoélectriques, électrostrictifs, alliages à mémoire de forme, les fibres optiques ou encore les polymères intelligents. La seconde classe concerne les matériaux initialement vierges d'électronique, sur lesquels on vient ajouter les fonctions attendues, à l'aide de composants électroniques tels que des capteurs, des puces RFID, des unités de traitement ou de stockage.

Cette première recherche bibliographique montre que notre société évolue vers une société d'objets communicants où tous interagissent, collaborent et agissent dans un même but : nous aider de manière proactive et réactive à une multitude de tâches.

Aujourd'hui, avec la mondialisation et l'ouverture des marchés, tout objet ou produit est confectionné, manipulé et utilisé par de nombreux acteurs tout au long de sa vie. De ce fait, les produits deviennent complexes (Kärkkäinen, Holmström, Främling and Artto, 2003) et les informations relatives à sa vie (sa conception, son utilisation, sa maintenance, etc.) sont souvent éparpillées dans une multitude d'entreprises et de systèmes. Tout cela n'est pas sans conséquence puisqu'il est nécessaire de gérer ces informations, ces systèmes dans le but de prendre les meilleures décisions quant à la vie du produit. La prochaine section introduit la notion de « cycle de vie » des produits ainsi que la gestion qui est menée autour de ces produits. Puis, en nous appuyant sur une revue de la littérature, nous montrons que les produits intelligents peuvent être utilisés et contribuer à la gestion des informations (p. ex. des produits capables de porter leur propres informations, d'influer sur leur cycle de vie, etc.).

1.2 Cycle de vie des produits et gestion associée

Les sections 1.2.1 et 1.2.2 présentent respectivement l'évolution jusqu'à aujourd'hui des concepts PLC (*Product Life Cycle*) et PLM (*Product Lifecycle Management*) où nous discutons de la nécessité de gérer les informations des produits durant leur vie et des problèmes que cela engendre. La section 1.2.3 fait un point sur les solutions et plates-formes existantes qui utilisent des produits intelligents pour répondre à ces problèmes. La section 1.2.4 expose les bénéfices pouvant être ou ayant été réalisés par des projets/systèmes utilisant des produits intelligents. Enfin, nous relevons dans la section 1.2.5 les limites actuelles de ces produits, ce qui nous conduit à la proposition d'un nouveau type de produit communicant, appelé « matière communicante ».

1.2.1 Historique du concept PLC

Le terme « cycle de vie du produit » ou PLC (*Product Life Cycle*) est aujourd'hui largement employé dans la littérature. Ce concept fut pour la première fois introduit, d'après (Muhs, 1985), par Conrad Jones qui a déclaré lors de la conférence AMA (*American Medical Association*) en 1957 :

« There are compelling forces behind this drive for new products. There is a life cycle that is characteristic of many - if not most - products. Since all products are "new" at their outset, we can call it the basic life cycle for new products. »

L'idée centrale du concept est qu'un produit conçu par une compagnie a une vie limitée de manière similaire aux êtres vivants. Le cycle de vie basique présenté par Jones et qui est, aujourd'hui, largement repris à travers la littérature, se compose de 5 phases : introduction, croissance, maturité, saturation et déclin. Graphiquement, ces phases peuvent être représentées en fonction du temps par une courbe en S dont l'ordonnée correspond aux bénéfices engendrés, comme l'illustre la FIGURE 1.2. Cette figure montre, d'une part, que le produit évolue à des vitesses différentes dans les 5 phases du cycle de vie et, d'autre part, que les bénéfices augmentent considérablement dans la phase de croissance du PLC et commencent à baisser dans la phase de maturité (cette baisse est liée à la compétitivité du marché).

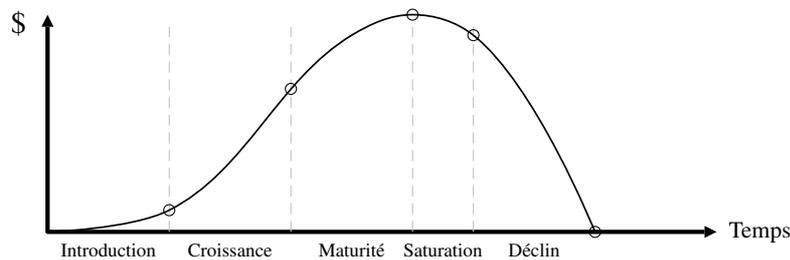


FIGURE 1.2: Phases du PLC : (Rink and Swan, 1979)

Par la suite, d'autres auteurs comme (Levitt, 1965; Polli and Cook, 1969) ont exprimé la pensée que connaître le lieu dans lequel le produit se situe à un instant donné de son cycle de vie permettrait de prendre des décisions marketing. Les dix années qui suivirent cette pensée firent l'objet de débats houleux entre les pro-PLC (Anderson and Zeithaml, 1984) et les anti-PLC comme le mythique article *Forget the PLC concept* écrit par (Dhalla and Yuspeh, 1976). Ces auteurs expliquent que croire au fait que tous les produits ont nécessairement une phase de déclin conduit souvent à les retirer de manière prématurée du marché, alors que certaines solutions de marketing permettent d'avertir et d'éviter cette phase de déclin. D'autres auteurs expliquent que la forme en S qui décrit le cycle de vie du produit est, dans certains cas, une approximation non appropriée (p. ex. (Rink and Swan, 1979) qui identifient 11 cycles de vie différents des produits), ou encore qu'il n'est pas simple de savoir dans quelle phase du cycle de vie le produit se situe à un instant t (Day, 1981). L'ouvrage de (Gardner, 1986) est une mine d'informations sur l'histoire du cycle de vie du produit, sur les problèmes qui ont été soulevés dans la littérature avant 1986, ainsi que sur les premiers travaux qui ont tenté de résoudre ces problèmes.

Aujourd'hui, étudier les cycles de vie des produits fait partie intégrante des stratégies des entreprises pour concevoir, gérer et planifier au mieux leur vie. Des systèmes de gestion ont depuis vu le jour, principalement du fait de l'expansion sans précédent des systèmes et outils informatique/d'information. Ces systèmes de gestion se regroupent sous l'acronyme PLM (*Product Lifecycle Management*). La section qui suit décrit les différents aspects (concepts, méthodes, outils, etc.) qui constitue un système PLM.

1.2.2 Système PLM

Introduction

L'acronyme PLM a émergé dans les années 1990 et s'est formé autour de deux courants (Lee et al., 2008), à savoir :

- la gestion des entreprises : elle se compose de systèmes de planification des ressources matérielles et de l'entreprise (ERP - *Entreprise Ressource Planning*), de systèmes de gestion des relations clients (CRM - *Customer Relationship Management*) et de gestion de la chaîne logistique (SCM - *Supply Chain Management*). Puisque l'on dispose d'une vision complète du cycle de vie des produits, les risques peuvent être estimés et en ce sens, le système PLM offre un ensemble d'outils, de bases de données pour gérer les données,
- la gestion des « informations produit » : de nombreux aspects/secteurs de l'entreprise sont connectés à la vie du produit, ce qui génère de grandes quantités d'informations à stocker, à échanger et à gérer. Dans ce contexte, l'intégration des systèmes tels que CAD, CAM, CAE (*Computer Aided - Design, Manufacturing and Engineering*) ou encore PDM (*Product Data Management*) à travers le système PLM facilite la collaboration entre les entreprises virtuelles (Ameri and Dutta, 2005).

Le cycle de vie du produit que Jones décrit en 5 phases, peut également être découpé suivant les processus dans lesquels le produit transite durant sa vie. La FIGURE 1.3 présente un découpage du cycle de vie du produit suivant 7 processus⁴ (conception, fabrication, . . . , recyclage). Chaque processus est constitué, à son tour, de sous-processus ou d'activités qui sont nécessaires, par exemple, à la transformation et/ou l'utilisation du produit. Le PLM est un concept par lequel « collaboration » et « coordination » des activités des processus nécessaires à l'évolution du produit sont obtenus par intégration à l'intérieur d'un même système de gestion (Garetti et al., 2003), comme cela est illustré dans la FIGURE 1.3. Notons qu'un produit dans un système PLM n'est pas nécessairement un produit physique, cela peut être un service ou tout autre produit incorporel (p. ex. un logiciel).

Applications et systèmes d'informations relatifs au système PLM

Nous avons expliqué précédemment que le système PLM gère l'intégration d'une multitude de systèmes, d'applications, d'outils comme ERP, CRM, CAD, CAM, CAE ou encore PDM (énoncé dans la précédente section). Toutefois, il existe bien d'autres systèmes/applications en raison de la complexité du cycle de vie des produits et des acteurs qui y prennent place, citons :

- les applications utilisées pour gérer les informations techniques comme les systèmes TDM (*Technical Data Management*),
- les applications gérant les informations d'exécutions de fabrication comme les systèmes MES (*Manufacturing Execution System*),

4. le nombre de phases peut varier en fonction du niveau de détails considéré/souhaité.

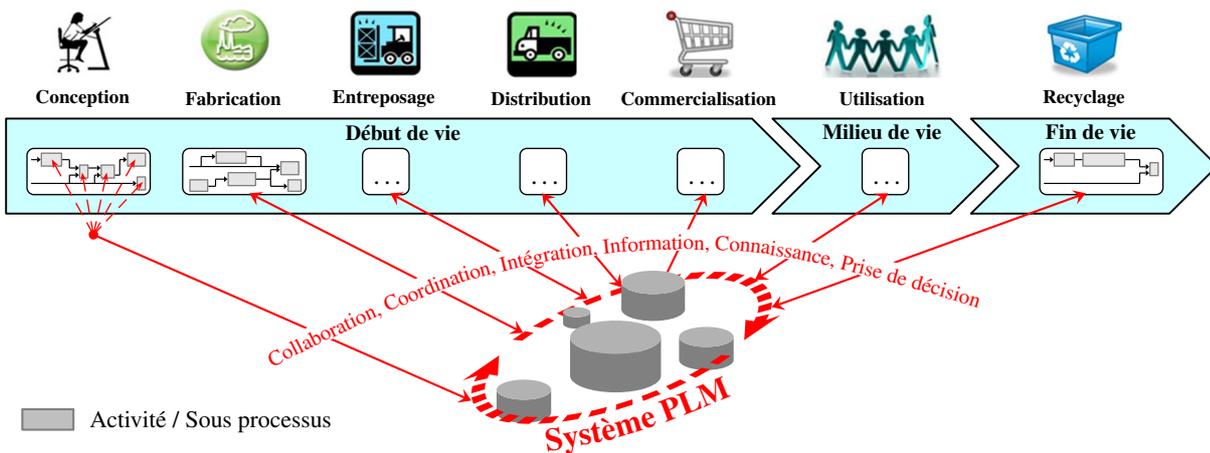


FIGURE 1.3: Vue générale du système PLM et de ses objectifs

- les applications gérant les informations relatives aux ordres de maintenance comme les systèmes CMMS (*Computerized Maintenance Management System*),
- les outils permettant d'optimiser les activités de la chaîne logistique et des entreprises à un instant donné, ou à travers le temps comme les systèmes APS (*Advanced Planning Systems*),
- les outils supportant les opérations de fabrication, de planification des processus et des méthodes de conception, tels que les systèmes CAPP (*Computer Aided Process Planning*) ou ceux précédemment cités, CAD, CAM, CAE,
- les outils permettant d'évaluer l'impact d'un produit sur l'environnement depuis sa conception jusqu'à son démantèlement, comme les outils LCA (*Life Cycle Assessment*).

Tous ces systèmes fournissent des fonctions dédiées au domaine d'application, ainsi que les bases de données qui leur sont associées (contenant les informations propres à l'entreprise) (Jaber, 2009). Des problématiques d'interopérabilité entre systèmes doivent également être gérées par le système PLM puisque chaque groupe d'acteurs/compagnies met en place son propre système d'information et plus exactement, son propre système de bases de données. Divers standards d'interopérabilité ont vu le jour comme IEC 62264, B2MML, ISA-88, ISA-95 (Koronios et al., 2007; Tursi et al., 2009). Toutefois, ces standards n'assurent actuellement qu'une partie de l'intégration de l'ensemble des systèmes d'information.

Nécessité de gérer les informations dans le cycle de vie du produit

La nécessité de gérer les informations du produit durant l'ensemble de son cycle de vie est devenue incontournable, d'une part, en raison des techniques de sophistication des produits et, d'autre part, en raison des réglementations gouvernementales strictes concernant la gestion du cycle de vie des produits (problèmes de traçabilité par exemple). Aujourd'hui, les produits sont souvent fabriqués avec des pièces provenant de sociétés distinctes. Un des challenges particuliers, par exemple d'une chaîne logistique, est de traiter ce grand nombre de produits individuels de la source jusqu'à la fin de la chaîne, sans perdre d'informations (Kärkkäinen, Ala-Risku and Främling, 2003). En effet, il y a souvent plusieurs ac-

teurs/fournisseurs au sein d'un même projet, ce qui génère inévitablement des pertes d'informations. Ces pertes sont liées au fait que toutes les entreprises impliquées dans le PLC stockent les informations des produits individuels de manière locale et ce, avec des références produits différentes. Lorsque les acteurs, membres du réseau, s'échangent des informations, il y a souvent un problème de synchronisation entre les flux d'informations et les flux physiques (Kärkkäinen, 2003). La FIGURE 1.4, extraite de (Kiritsis et al., 2003), donne un aperçu du cheminement des flux physiques et des flux d'informations à travers le cycle de vie du produit. Durant l'ensemble du cycle de vie, nous constatons que ces deux flux bouclent de multiples fois entre les différentes étapes des processus concernés (conception, fabrication, utilisation et recyclage). En effet, en fin de vie, le produit peut être réutilisé, ré-usiné ou encore recyclé en matière nouvelle, au sens de l'économie circulaire (McDonough and Braungart, 2002). Une maîtrise optimale de ces deux flux rend d'autant plus crucial le challenge souligné précédemment (c.-à-d. éviter la perte et les débordements d'informations), cependant cela aurait plusieurs avantages comme le défendent (Kiritsis et al., 2003; Kiritsis, 2011), à savoir :

- les membres de la chaîne logistique (producteurs) seraient livrés avec les informations exhaustives relatives aux modes d'utilisation, aux conditions de stockage, de recyclage, etc.,
- les experts des services de maintenance, de recyclage et autres pourraient être assistés dans leurs tâches, en disposant par exemple d'informations actualisées par rapport à l'état courant du produit et pourraient fournir une assistance en temps-réel au client (via Internet ou autre),
- le bouclage d'informations pourrait être optimisé (facilité/exactitude accrue des retours d'expertises concernant le produit), permettant ainsi d'améliorer les processus de conception, de fabrication, etc.,
- les informations utiles pour les phases de recyclage du produit seraient plus pérennes (routes possibles pour le bouclage des flux physiques, cf. FIGURE 1.4).

Une solution pour pallier le problème de synchronisation entre ces deux flux est de faire en sorte qu'ils n'en fassent plus qu'un : l'instrumentation du flux physique via l'implémentation de technologies et concepts tels que présentés dans la section 1.1 est une solution, rendant le produit actif, intelligent ou communicant au sein de son environnement. La technologie des codes à barres est un exemple. Elle fournit un identifiant pour accéder à des informations stockées sur des bases de données déportées (McFarlane and Sheffi, 2003). Cependant, malgré son efficacité et usage très important, cette technologie montre encore certaines limites. En effet, les codes à barres nécessitent une intervention manuelle (Boxall, 2000) et perdent souvent leur lisibilité par le fait de détériorations dues aux manipulations successives qu'ils subissent (Ollivier, 1995). De plus, il n'est pas possible de changer leur contenu après les avoir imprimés, ce qui rend impossible la mise à jour dynamique des informations. Les messages EDI (*Electronic Data Interchange*) sont une autre solution permettant de transmettre des informations dans un format électronique aux partenaires en aval de la chaîne logistique. Cependant, de nombreux auteurs tels que (McFarlane and Sheffi, 2003; Kärkkäinen, 2003; Främling et al., 2006; Parlikad and McFarlane, 2007) ont démontré que la prise en main des « informations produit » avec des messages électroniques ou papier

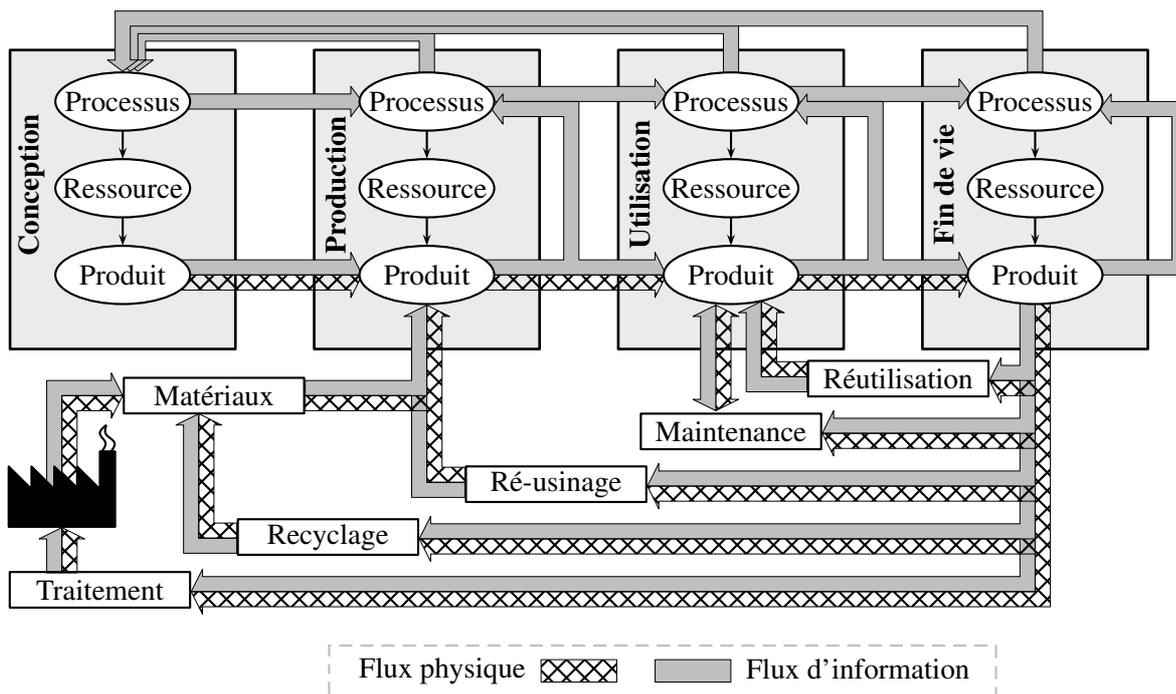


FIGURE 1.4: Bouclage des flux informationnel et physique au sein du PLC : (Kiritsis et al., 2003)

se révèle être problématique à plusieurs niveaux :

- toutes les compagnies doivent être capables de communiquer les unes avec les autres. Si l'une des compagnies n'est pas en mesure de recevoir ou transmettre des données, alors le flux informationnel est interrompu,
- il est difficile de maintenir à jour des informations (relatives aux produits) qui sont envoyées et stockées en aval de la chaîne. De plus, le producteur d'une nouvelle information peut être incapable de savoir quels sont les partenaires qui doivent être informés d'une telle modification. Par conséquent, les compagnies qui accèdent à des informations obsolètes risquent de prendre les mauvaises décisions (Kärkkäinen, Ala-Risku and Främling, 2003).
- transmettre toutes les informations en aval peut causer des engorgements dans le flux informationnel tout au long du PLC (Beulens et al., 2000) (spécialement vrai pour des produits complexes).
- l'EDI est une solution qui peut être coûteuse à mettre en place entre deux partenaires, elle s'avère dès lors inappropriée dans le cadre de petites entreprises (Timm et al., 2001), ce qui limite les échanges d'informations au sein du cycle de vie (Kärkkäinen, Ala-Risku and Främling, 2003).

Ainsi se sont créées depuis plusieurs années des communautés de chercheurs travaillant sur le paradigme du produit intelligent, tel que IMS (*Intelligent Manufacturing Systems*), HMS (*Holonic Manufacturing System*), PDS (*Product Driven Systems*) ou encore IFAC *Coordinating Committee on Manufacturing and Logistics Systems* (Ollero et al., 2002; Nof et al., 2006). L'état de l'art proposé par (Meyer et al., 2009) témoigne de la richesse des travaux dans le domaine et des nombreuses perspectives d'ap-

plications. Ces produits intelligents permettent, entre autres, d'améliorer les performances de la chaîne logistique (sécurité, gestion d'actifs, efficacité, re-routage des produits, traçabilité) mais également du système de fabrication (gestion de la personnalisation retardée, contrôle et planification de la production).

1.2.3 Produits intelligents dans la littérature

Apparition

Le terme « produit intelligent » a été évoqué pour la première fois en 1988 par (Ives and Vitale, 1988) dans un contexte de service après-vente. Dans la communauté IMS, (Bussmann, 1998) est le premier à s'intéresser au produit intelligent. Par la suite, (McFarlane et al., 2003; Kärkkäinen, 2003) définissent un produit intelligent comme étant un produit physique auquel on a adjoint une représentation informationnelle.

Au cours des vingt dernières années, la communauté IMS a développé des programmes visant à mettre en pratique des systèmes agiles B2M (*Business to Manufacturing*), basés sur un réseau hétéroarchique d'unités autonomes. Ce type d'organisation à vocation d'être un moyen plus approprié pour faire face à des perturbations et pour être adaptable plus rapidement (Morel and Grabot, 2003). Un problème majeur est de combiner l'agilité (autonomie) et la capacité à communiquer (coopération) de ces unités autonomes afin qu'elles puissent interagir en créant, générant et en exécutant des plans de production (Valckenaers, 1998). Un moyen pour répondre à ce défi est de combiner le paradigme MAS (*Multi-Agents System*), provenant du paradigme d'intelligence artificielle distribuée, avec le paradigme HMS (*Holonic Manufacturing Systems*), provenant de la communauté IMS (Valckenaers, 1998). Le consortium HMS⁵ a proposé les systèmes holoniques de production. Ces systèmes sont basés sur le concept de holon qui est une entité pouvant être contenue dans d'autres holons (Van Brussel et al., 1998). Les holons sont organisés dans des holarchies et montrent des propriétés comme l'autonomie et la coopération. L'intérêt principal de ce paradigme est de préserver la hiérarchie de la structure du système et de conserver une flexibilité dynamique au niveau hétéroarchique.

Ce paradigme a, par conséquent, nécessité de donner aux produits des capacités de stockage et de traitement des données pour interagir avec leur environnement. Ces systèmes, basés sur des produits intelligents, offrent ainsi de nouvelles opportunités comme par exemple, la liberté pour ces produits d'assurer eux-mêmes leur routage/re-routage dans la chaîne⁶ (Sallez et al., 2009). Dès lors, comme le soulignent (Banaszak and Zaremba, 2003), le produit intelligent nécessite des technologies lui permettant non seulement de porter des informations mais également de la connaissance, ceci faisant nécessairement appel aux concepts introduits en section 1.1.1 (*Ubiquitous Computing, Internet of Things*). Cette notion de produit intelligent a également donné lieu à la notion de systèmes pilotés par les produits (Morel

5. les idées partagées par le consortium HMS proviennent des concepts exprimés par (Koestler, 1967) dans son livre *The ghost in the Machine*.

6. le concept de re-routage de produit est particulièrement important lorsque ces produits sont confrontés à des applications dites « temps-critique » (Meyer et al., 2009).

et al., 2007) ou encore de méthodes orientées agents (Jennings and Wooldridge, 1998; Fox et al., 2000) pour la prise en main de l'information dans une chaîne logistique dynamique. Il n'y a pas de définition consensuelle sur le concept d'agent, mais ses capacités intrinsèques sont toujours exprimées de la façon suivante : il doit être autonome, sociable, réactif, proactif et multifonctionnel (Wooldridge and Jennings, 1995; Shen and Norrie, 1999) :

- autonome : l'agent opère seul, sans intervention humaine ou autre,
- sociable : l'agent interagit avec d'autres agents via un langage de communication,
- réactif : l'agent perçoit son environnement et répond dans un temps opportun pour modifier le cours des événements si nécessaire,
- proactif : l'agent n'agit pas uniquement en réponse à son environnement, il peut également prendre des initiatives,
- multifonctionnel : l'agent est utilisé pour couvrir plusieurs fonctions au sein du PLC, il peut prendre le rôle d'agent logistique, d'agent d'acquisition, d'agent de transport, d'agent de planification, etc.

Dans la suite de cette section, nous décrivons les modèles d'évaluation et de classification des produits dits « intelligents », puis nous présentons les solutions existantes à ce jour (technologies, standards, plates-formes, etc.), basées en partie sur l'état de l'art réalisé par (Meyer et al., 2009).

Définitions et modèles de classification des produits intelligents

Plusieurs auteurs ont proposé des définitions concernant le produit intelligent, qui peuvent couvrir plusieurs aspects. Comme nous l'avons dit auparavant, McFarlane définit le produit intelligent comme :

« *a physical and information-based representation of a product.* » (McFarlane et al., 2003)

Ce produit intelligent, selon McFarlane, dispose des caractéristiques suivantes :

1. il possède une identification unique,
2. il est capable de communiquer de manière efficace avec son environnement,
3. il peut stocker ou conserver les données le concernant,
4. il peut déployer un langage pour décrire ses caractéristiques, ses contraintes de production,
5. il est capable de participer à ou prendre des décisions importantes pour son devenir.

Basés sur cette définition, deux niveaux d'intelligence sont alors définis par (Wong et al., 2002). Le premier couvre essentiellement les points 1 à 3, où le produit est capable de communiquer son état (forme, localisation, composition, etc.) et le second couvre les points 1 à 5, permettant au produit d'évaluer et d'influencer le cours de sa vie. Il devient ainsi proactif en complément du fait qu'il soit capable de communiquer ses états. Les auteurs expliquent qu'établir une connectivité des produits avec des informations temps-réel peut, d'une part fournir des informations précises, et d'autre part permettre aux produits d'évaluer et d'influencer leur propre destin. L'exemple de la boîte de conserve de spaghetti proposé dans

(Wong et al., 2002), rappelé sur la FIGURE 1.5, illustre ce concept dans lequel la boîte de conserve est un objet physique dont les informations sont stockées dans une base de données distante, et l'intelligence est fournie par l'agent de prise de décision. La connexion entre le produit physique et la représentation de l'information appropriée se fait via un lecteur de tag RFID.

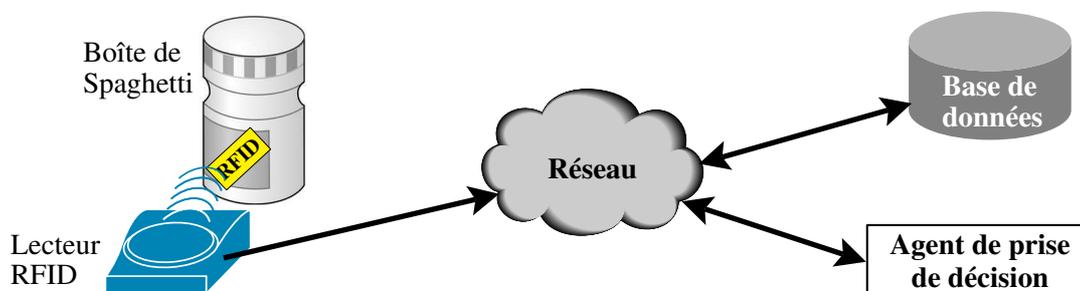


FIGURE 1.5: Boîte de conserve de spaghetti : (Wong et al., 2002)

(Kärkkäinen, Holmström, Främling and Arto, 2003) défendent l'idée que le produit intelligent est un moyen de contrôle d'entrée-sortie des produits sur la chaîne logistique. En d'autres termes, les produits individuels dans la chaîne logistique contrôlent leur destination ainsi que leur itinéraire. Pour cela, les auteurs expliquent que le produit doit posséder les caractéristiques suivantes :

1. il possède une identification unique,
2. il est un moyen pour accéder aux informations qui le concernent dans l'organisation, que ce soit grâce au code d'identification ou par d'autres mécanismes,
3. il peut communiquer.

(Ventä, 2007) fournit une autre définition de l'intelligence d'un produit ou d'un système, laquelle se rapporte essentiellement à des produits capables de prendre des décisions, c'est-à-dire une extension du point 5 de la première définition et du point 3 de la seconde. L'intelligence est définie comme :

1. une surveillance continue de son statut et de son environnement,
2. une réactivité et une adaptation aux conditions opérationnelles et environnementales,
3. un maintien des performances optimales, quelles que soient les circonstances (même dans des cas non prévus),
4. une communication active avec l'utilisateur, l'environnement ou autres systèmes/produits.

Cette définition est majoritairement axée sur les produits embarquant des capacités de traitement et de communication suffisantes pour dialoguer avec des systèmes externes. Ce point est la différence notable avec les définitions précédentes, dans lesquelles les produits possèdent uniquement des identifiants utilisés pour accéder aux informations et aux agents de décisions déportés sur le réseau.

(Meyer et al., 2009) montrent néanmoins que ces définitions ne couvrent pas l'ensemble des caractéristiques des produits intelligents. Par conséquent, les auteurs proposent une nouvelle classification à trois dimensions, reprenant les aspects introduits dans les définitions précédentes, en ajoutant une dimension liée au niveau d'agrégation de l'intelligence (décrite ci-après). Cette classification est illustrée sur la FIGURE 1.6. Nous faisons également apparaître sur cette figure les différents critères relatifs aux définitions des précédents auteurs afin de mettre en évidence les points de recouvrement. Les trois dimensions du modèle de classification sont :

- Degré d'intelligence comprenant trois niveaux :
 - traitement des informations : le produit doit être au minimum capable de gérer ses propres informations (provenant des capteurs, des lecteurs RFID, etc.). Sans cette capacité, il serait difficile de parler de produit intelligent.
 - notification de problème : en complément de la capacité à porter de l'information, le produit peut aussi notifier/avertir son propriétaire lorsqu'une anomalie ou tout autre événement se produit. À ce stade, le produit n'est pas encore capable de gérer son comportement, ni de modifier son devenir.
 - prise de décision : le produit le plus intelligent est celui qui est complètement autonome, capable de prendre des décisions quant à son devenir et ce, sans intervention extérieure.
- Localisation de l'intelligence comprenant deux niveaux :
 - intelligence à travers le réseau : l'intelligence du produit est déportée sur des bases de données distantes. (Ramparany and Boissier, 2002) nomment cela des plates-formes portails (*portal platforms*). Néanmoins, comme cela est souligné dans (Främling et al., 2006), le portail est uniquement une interface graphique présentant les informations pour l'abonné. Les questions liées à la gestion de l'information, comme par exemple relier des informations à des éléments spécifiques du produit et de la prise en compte de la place du produit dans la nomenclature, demandent des solutions supplémentaires. Lier les informations pertinentes résidant dans plusieurs systèmes de données différents, tout en évitant l'engorgement du flux informationnel, reste un défi sans réponse.
 - intelligence embarquée dans l'objet : toute l'intelligence prend place dans l'objet. (Ramparany and Boissier, 2002) nomment cela des plates-formes embarquées (*embedded platforms*).
- Forme d'agrégation de l'intelligence comprenant deux niveaux : nous remarquons sur la FIGURE 1.6 que cette dimension nouvelle n'a pas été prise en considération par les précédents auteurs. (Meyer et al., 2009) soulignent néanmoins que cette dimension est importante et qu'elle ne doit pas être négligée. En effet, comme cela a été expliqué auparavant, un produit est une composition hiérarchisée de composants plus élémentaires. L'analyse de cette dimension amène à la structuration suivante :
 - article intelligent : l'objet ne gère que les informations, les notifications et/ou décisions sur lui-même. S'il est composé de plusieurs composants (parties), ils ne pourront pas être distingués

comme des objets individuels.

- conteneur intelligent : le conteneur intelligent ne gère pas uniquement ses propres informations, notifications et/ou prises de décisions, il est aussi conscient des composants qui le forment. Si le conteneur intelligent est démonté ou des pièces sont supprimées, les parties sont capables de continuer à fonctionner comme des articles intelligents (point précédent). Un exemple serait celui d'un moteur de voiture que l'on enlève, rénove, dont on ajoute de nouveaux composants tels qu'un alternateur, radiateur.

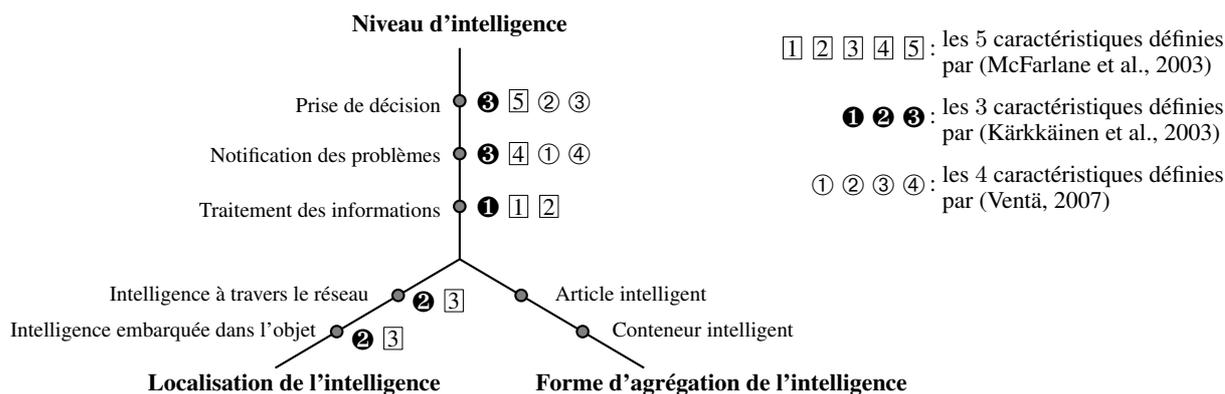


FIGURE 1.6: Modèle de classification des produits intelligents : (Meyer et al., 2009)

Solutions et plates-formes

Depuis la naissance des systèmes de fabrication intelligents et le vif intérêt porté aux produits intelligents/communicants, de nombreuses solutions et plates-formes ont vu le jour. Cette partie donne un aperçu et résume brièvement les solutions les plus connues à ce jour.

Les approches basées « agent » pour la gestion de l'information produit a reçu une attention grandissante au fil des années (Holmström et al., 2002; McFarlane et al., 2002; Kärkkäinen, Ala-Risku and Främling, 2003; Valckenaers et al., 2004). Néanmoins, la question fondamentale dans cette approche est : comment associer les agents logiciels avec leur partie physique ? Pour assurer cette connexion, deux solutions sont envisageables :

- une plate-forme portail : il s'agit d'une interface graphique permettant aux abonnés d'accéder aux informations produit qui leur sont nécessaires/destinées. On peut donc dire que l'agent du produit est déporté à travers un réseau (sur un serveur ou autre) comme l'illustre la FIGURE 1.7(a). Le système MASCOT (*Multi-Agent Supply Chain cOordination Tool*) est un exemple de plate-forme multi-agents, assurant des fonctions telles que la planification et la reconfiguration de la chaîne logistique dans le cadre d'applications critiques (Sadeh et al., 2003). Un autre exemple est le système AMCC (*Agent-based Manufacturing Control and Coordination*) mettant en œuvre (en partie) des agents produits à l'aide de la technologie RFID, des agents de passerelles et des ressources réseaux afin de superviser et contrôler les flux de production (Chen and Tu, 2009),

- une plate-forme embarquée : il s'agit ici d'embarquer l'intelligence dans l'objet (*Intelligence at object*) comme illustré sur la FIGURE 1.7(b). Plusieurs exemples de plates-formes développées pour supporter des agents embarqués sur des équipements mobiles existent, citons ASDK (*Aglets Software Development Kit*) conçue pour sécuriser et fiabiliser les exécutions des agents dans un système ouvert (Karnik and Tripathi, 2001) ; la plate-forme LEAP (*Lightweight Extensible Agent Platform*) qui est une extension de JADE⁷, qui peut être embarquée et distribuée sur plusieurs équipements ; ou encore les plates-formes FIPA-OS⁸ (*Foundation for Intelligent Physical Agents*) et Grasshopper (Baumer et al., 1999). Notons que ces plates-formes sont généralement développées en JAVA et que les équipements sur lesquels sont embarqués les agents ont suffisamment de ressources (PDA, téléphones mobiles).

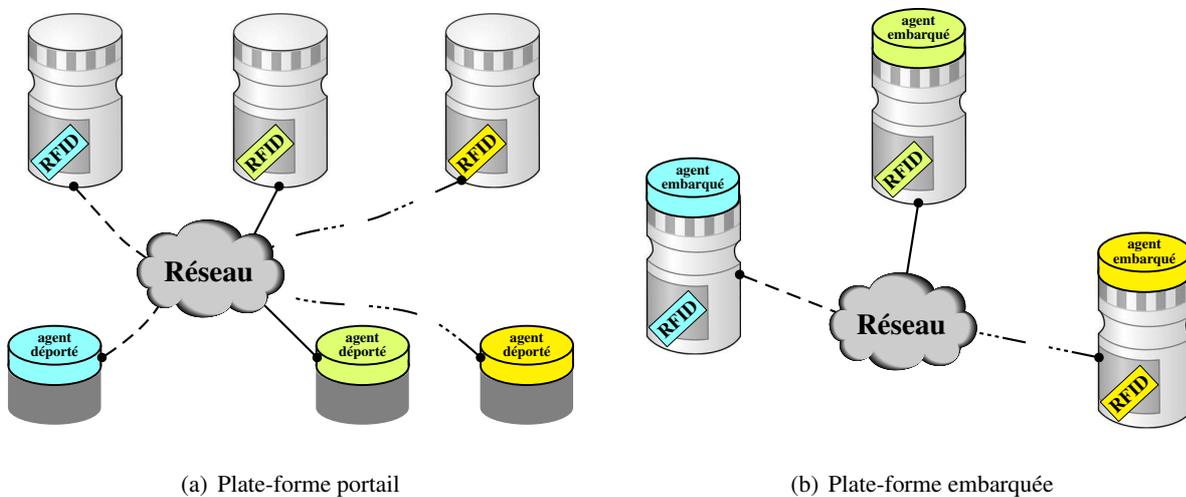


FIGURE 1.7: Types de plates-formes basées « agent »

Plusieurs solutions et plates-formes ont été conçues à travers le temps afin d'assurer le lien entre l'agent et sa partie physique, citons les systèmes SPHINX (Lee, 1989), STOCKWAY connus à ce jour sous l'appellation WWAI (*World Wide Article Information*), ID@URI (Huvio et al., 2002), EPCglobal (*Electronic Product Code global*), ESDS (*Extensible Supply chain Discovery Service*) (Young, 2008), etc. Les trois plus connues à ce jour sont EPCglobal, ID@URI et WWAI comme le soulignent (Meyer et al., 2009). Le concept-clé de l'architecture EPC, qui a d'ailleurs émergé en même temps que le concept de l'IoT, consiste à séparer l'information d'un objet de l'objet lui-même (Brock, 2001). L'intérêt majeur de cette solution est de permettre l'utilisation de codages standardisés/répandus de type GS1 (GTIN, SSCC...), lesquels sont embarqués dans EPC et évitent ainsi une migration vers un nouveau codage. Les éléments dont il faut disposer pour mettre en œuvre la solution EPC sont synthétisés dans la FIGURE 1.8 et détaillés ci-dessous :

7. JADE : <http://jade.cselt.it/>

8. Micro-FIPA-OS : <http://fipa-os.sourceforge.net/index.htm>

- une référence unique : identifie un produit ou objet spécifique sur le PLC (cf. trame EPC dans la FIGURE 1.8),
- un tag RFID : porte le code EPC et est attaché au produit,
- un lecteur RFID : permet de lire les codes EPC des produits et de les transmettre sur le réseau afin d'identifier les agents (informations) associé(e)s,
- un logiciel « savant » (logiciel EPC) : assure le bon déroulement de la lecture du tag et de l'association avec l'agent sur le réseau (gère les mauvaises lectures, filtre certaines données, envoie des requêtes sur un ONS - *Objet Naming Service*, etc.). Ces logiciels peuvent être localisés dans des magasins, des centres de distribution, ainsi que dans des centres de données nationaux,
- un serveur ONS (*Objet Naming Service*) : service automatisé permettant de connecter chaque EPC aux informations associées sur le réseau. Un serveur ONS est comparable à un serveur DNS nécessaire à la localisation de ressources sur Internet. Des propositions ont été étudiées pour remplacer ou compléter le service ONS telles que ESDS (*Extensible Supply Chain Discovery Service*),
- des services : permettent aux utilisateurs d'échanger des données avec des partenaires basés sur les EPCs. Toute l'information d'un objet est spécifiée en utilisant un dérivé d'XML appelé PML (*Physical Markup Language*).

La FIGURE 1.8 décrit le séquençage des interactions entre le lecteur RFID, le code EPC, le service de nommage (ONS) et le serveur PML.

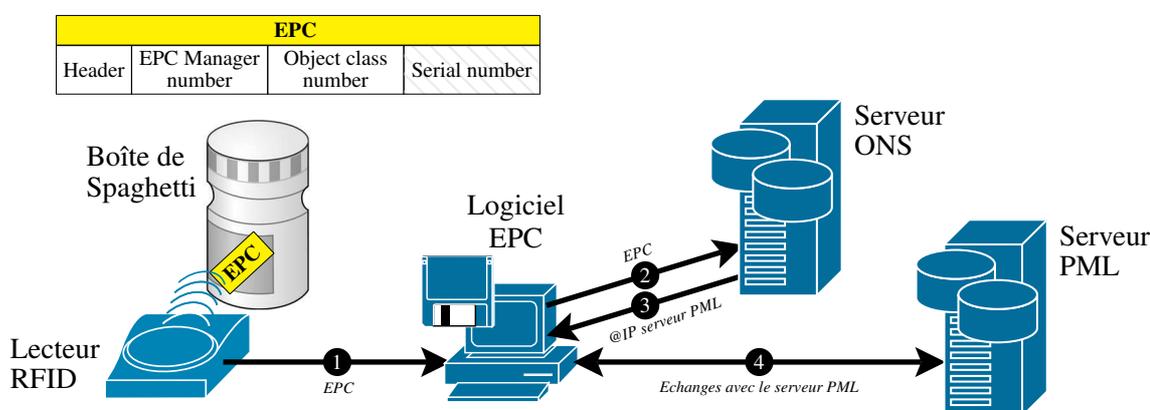


FIGURE 1.8: EPCglobal : une requête ONS typique avec l'utilisation d'un serveur PML

La seconde proposition notable est ID@URI, qui est une solution développée par l'université de Technologie d'Helsinki (Huvio et al., 2002; Främpling et al., 2003). Dans cette approche, le produit porte son propre identifiant (ID), ainsi qu'une adresse appelée URI (*Uniform Resource Identifier*). L'adresse URI correspond généralement à une adresse URL permettant de localiser l'agent sur Internet. Puisque cette adresse est directement embarquée dans l'identifiant du produit, l'infrastructure DNS (*Domain Name Service*) existante est suffisante. Tout comme EPCglobal, cette approche permet d'utiliser des codages industriels standardisés tels que GTIN, SSCC, GRAI, incluant le codage EPC. Une plate-forme appelée *Dialog*, basée sur la solution ID@URI, a été développée et mise en œuvre dans divers projets

comme le projet PROMISE (*PROduct lifecycle Management and Information tracking using Smart Embedded systems*) (Kärkkäinen et al., 2004). Les éléments dont il faut disposer pour mettre en œuvre la plate-forme *Dialog* sont synthétisés dans la FIGURE 1.9 et détaillés ci-dessous :

- un code ID@URI : un code unique par produit, se composant d'un nombre unique qui identifie le produit (ID) concaténé avec l'adresse de localisation de l'agent produit (URI), cf. FIGURE 1.9,
- un vecteur d'informations (p. ex., tag RFID, code à barres ou autre) : porte le code ID@URI et est attaché au produit,
- un lecteur d'informations (p. ex., lecteur RFID, lecteur code à barres ou autre) : permet de lire les codes ID@URI des produits et de les transmettre sur le réseau dans le but d'identifier les agents de décisions et d'informations associés,
- une application cliente (p. ex. le logiciel *Dialog*) : assure le bon déroulement de la lecture du tag et la mise en relation avec le serveur distant via l'adresse URI,
- services : ensemble de services permettant aux utilisateurs d'échanger des informations avec des partenaires basés sur la technologie ID@URI. Ces informations sont accessibles par Internet grâce à l'adresse URI ou plus exactement l'URL.

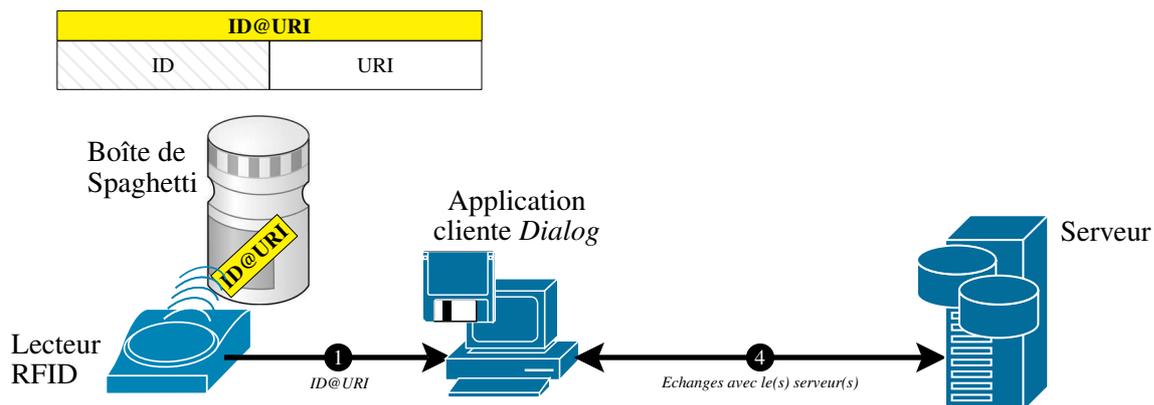


FIGURE 1.9: ID@URI : une requête classique sur la plate-forme *Dialog*

La dernière solution, WWAI, repose sur le principe du *Peer to Peer* (P2P) qui est essentiellement destiné au partage de fichiers multimédias. L'avantage d'un tel protocole est qu'il permet à tous les nœuds de maintenir un contrôle sur la distribution des informations (c.-à-d. de connaître quel système possède telle ou telle information), d'assurer un bon niveau de tolérance aux fautes (une défaillance d'un nœud n'affecte quasiment pas le réseau global) et la possibilité d'effectuer de l'équilibrage de charge à travers le réseau (plusieurs routes possibles entre les serveurs). WWAI utilise un identifiant produit qui combine les identifiants existants dans l'organisation de type GS1, avec le type du produit et un numéro de série. Cet identifiant est alors utilisé comme une clé de recherche afin de récupérer les informations sources liées au produit. Que ce soit pour récupérer les informations sources ou pour les échanges d'informations, tous deux exploitent le principe du P2P. WWAI permet aux entreprises de partager les informations liées aux produits en temps-réel sur Internet, quelle que soit la technologie Auto-ID implémentée. En effet,

l'approche WWAI donne la liberté à l'organisation d'embarquer n'importe quel type d'identifiant via le champ « Prefix » (cf. FIGURE 1.10). Cette solution permet une distribution des informations entre les systèmes d'informations (c.-à-d. les bases de données) des différentes entreprises concernées par le cycle de vie du produit. Chaque participant gère sa propre information et décide des privilèges accordés aux utilisateurs y accédant (public ou privé). Cette technique de distribution des informations où chaque entreprise gère celles qui lui sont propres, rend les réseaux WWAI flexibles et évolutifs suivant le nombre de nœuds fournisseurs d'informations et/ou le nombre de produits. Les éléments dont il faut disposer pour mettre en œuvre un réseau WWAI sont synthétisés dans la FIGURE 1.10 et détaillés ci-dessous :

- un code WWAI : un code unique par produit, se composant d'un « Prefix » dans lequel l'organisation décide du type d'identifiant (EPCglobal, GS1,...) et d'un « Suffix » permettant d'identifier le produit de manière unique (équivalent aux champs *Serial number* et *ID* des solutions ERP et *ID@URI*),
- un vecteur d'informations (p. ex., tag RFID, code à barres ou autre) : porte le code WWAI et est attaché au produit,
- un lecteur d'informations (p. ex., lecteur RFID, lecteur code à barres ou autre) : permet de lire les codes WWAI des produits et de les transmettre à l'application cliente,
- une application cliente : assure le bon déroulement de l'opération de lecture et permet de se connecter à un point d'entrée du réseau WWAI (cf. FIGURE 1.10) qui se charge de retrouver l'ensemble des informations liées au produit,
- des services : permettent aux utilisateurs d'échanger des informations avec les différentes entreprises à travers le réseau WWAI. Ce réseau est composé d'un certain nombre de boucles de nœuds réseau.

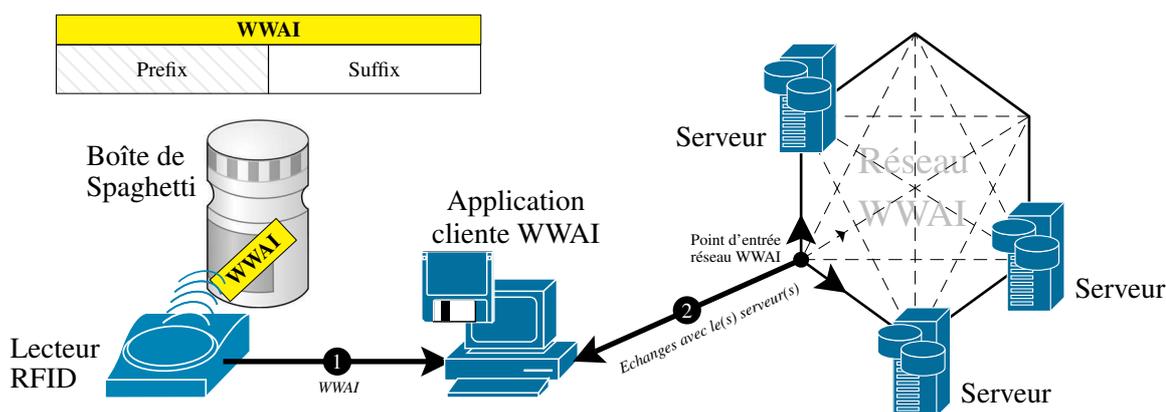


FIGURE 1.10: WWAI : Vue globale de l'architecture WWAI

Une boucle de nœuds WWAI est généralement nécessaire lorsque l'identifiant du produit pour une entreprise donnée est vu pour la première fois. Après cela, les adresses des nœuds connus sont mémorisées de telle sorte à éviter que de nouvelles boucles de nœuds soient créées (excepté si ces adresses ne fonctionnent plus). Pour accéder à d'autres nœuds disposant d'informations au sujet d'un produit ou

d'éléments particuliers (document, clip vidéo, etc.), il est uniquement nécessaire d'avoir un point d'entrée au réseau WWAI, qui se charge automatiquement d'envoyer et d'acheminer les requêtes à tous les fournisseurs d'informations (serveurs des entreprises concernées). Ainsi le demandeur reçoit les « informations produit » en fonction de ses droits d'accès chez les partenaires du PLC. Le modèle de distribution des informations est souvent basé sur le mode asynchrone (aspect détaillé dans le Chapitre 2), laissant ainsi la responsabilité au demandeur de gérer la cohérence des données collectées. Néanmoins, le mode synchrone peut être envisagé et configuré dans certains cas et sur certaines portions du réseau WWAI.

L'étude comparative menée par (Främling et al., 2007) sur ces trois solutions est très intéressante. Ils concluent, au regard de la gestion du cycle de vie d'un produit, outre que la solution EPC a l'avantage d'être standardisée, que cette solution est très flexible car l'identifiant est toujours dissocié de l'adresse réseau et ainsi, le service de résolution peut supporter de nouveaux protocoles. Néanmoins, les auteurs relèvent quelques inconvénients comme le fait que cette solution ne soit pas compatible avec les petites entreprises et pas adaptable à 100% à de nouveaux codes identifiants, bien qu'elle en supporte un grand nombre⁹. Rajoutons que comparativement aux solutions WWAI et ID@URI, l'approche mise en œuvre pour convertir les codes EPC en adresses réseaux est quelque peu complexe. Les solutions WWAI et ID@URI, contrairement à EPCglobal, peuvent fonctionner avec des technologies autres que la RFID (p. ex. codes à barres), élargissant ainsi le champ d'intégration dans le système réseau, indépendamment de la technologie, de l'industrie et des infrastructures informatiques. L'inconvénient le plus notoire concernant ID@URI est la pérennité/longévité des URLs embarquées sur les produits. En effet, un produit peut avoir une durée de vie significative, des dizaines d'années, alors que les URLs changent fréquemment sur de telles périodes, ce qui engendre une rupture entre le flux informationnel et le flux physique. La solution WWAI semble être techniquement plus sophistiquée que les deux autres, cependant trop peu d'industriels supportent à ce jour cette solution pour permettre sa pérennité.

1.2.4 Bénéfices générés par les produits intelligents

La mise en place de systèmes basés sur le concept de produit intelligent est essentiellement liée aux enjeux formulés précédemment, à savoir : gérer les informations produit durant l'ensemble du cycle de vie ; participer à la synchronisation du flux physique et informationnel et assurer le bouclage optimal de ces deux flux ; fournir en temps réel une visibilité de la chaîne d'approvisionnement ; améliorer les processus logistiques (fabrication, routage des produits, maintenance, etc.) ; assurer la pérennité des informations (p. ex. lorsque celles-ci sont adjointes au produit via un agent embarqué) ; assurer un niveau de sécurité (p. ex. éviter des erreurs liées à l'intégrité physique ou encore gérer des droits d'accès aux informations). Certaines études ont rapporté les bénéfices réalisés (de manière quantifiable ou non) après mise en place de systèmes mettant en œuvre des produits intelligents ou communicants.

Les bénéfices économiques proviennent d'une part, d'une prise en main des informations produit

9. la liste des solutions supportées est disponible à : <http://www.epcglobalinc.org/standards>

plus rapide et plus sûre. En effet, l'automatisation des données d'entrée réduit le taux d'erreurs. La saisie manuelle des informations est généralement considérée comme ayant un taux erreur de $\frac{1}{300}$ (Kärkkäinen, Holmström, Främpling and Artto, 2003). Dans un environnement difficile pour un projet de chaîne de livraison, la lecture peut être bien plus précise et rapide en utilisant une technologie du type RFID (comparativement à la solution des codes à barres). Les étiquettes électroniques de type RFID permettent une lecture rapide (en règle générale, elle augmente avec la fréquence), une lecture de plusieurs étiquettes de manière quasi simultanée, et des lectures distantes (Thomas, 2009). Par exemple, *British Airways* a pratiquement éliminé les erreurs liées au tri des bagages avec un système basé sur la technologie RFID. Ils ont estimé, en 2000, une économie à travers le monde comprise entre 6 et 12 millions de dollars (US\$) contre une dépense de 1.2 millions de dollars avec cette solution (Kärkkäinen, 2003). La société *LynxExpress* a également, en 1999, investi dans une plate-forme appelée « superhub » dans laquelle les palettes sont munies d'une puce RFID. Ainsi, lorsqu'un agent passe avec la palette sous un portique muni de lecteurs RFID, ce dernier indique à l'agent (à travers un jeu de lumière) si cette palette est destinée ou non à ce quai d'embarquement. Ce système « superhub » est 70% plus rapide qu'un système conventionnel, plus flexible et exempt d'erreur (Anon, 2000). La société automobile *Ford* a également amélioré ses coûts de production avec l'utilisation de la technologie RFID (Johnson, 2002). En effet, le système de codage manuel de l'information utilisé sur les différentes lignes de production, était sujet à de nombreuses erreurs et par conséquent entraînait de fortes pertes financières. De nombreux exemples de ce genre peuvent être cités. (Chao et al., 2007) énumèrent une multitude de cas industriels, hospitaliers, pharmaceutiques, militaires, environnementaux et mettent en avant les bénéfices atteints pour chacun d'eux. (Tajima, 2007) explique que les gains possibles liés au déploiement de produits intelligents (munis de tag RFID) se situent à plusieurs niveaux sur la chaîne d'approvisionnement et se répercutent sur chacun des acteurs, comme l'illustre la FIGURE 1.11 (bénéfices du point de vue des acteurs et de la chaîne logistique).

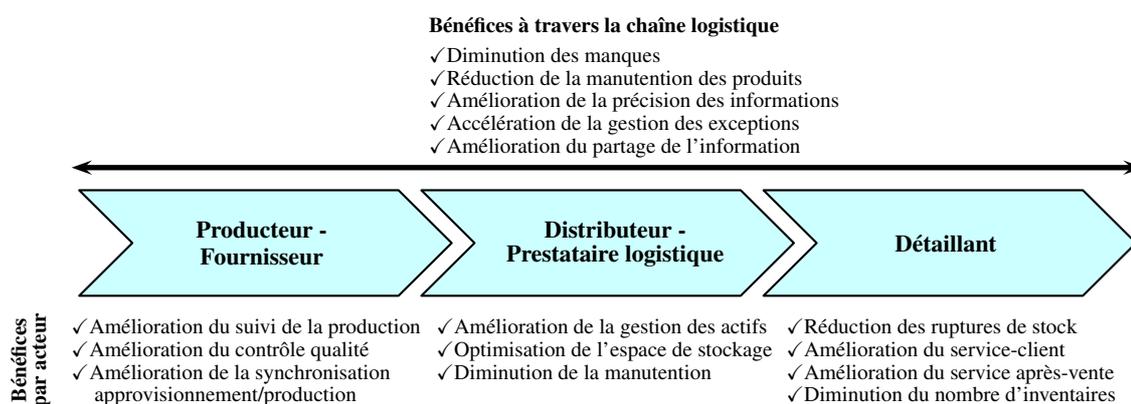


FIGURE 1.11: Bénéfices des produits communicants à travers la chaîne logistique : (Tajima, 2007)

1.2.5 Vers un nouveau paradigme de « matière communicante »

Limites actuelles

La section précédente a mis l'accent sur les bénéfices pouvant être engendrés par la mise en œuvre de produits intelligents/communicants tout au long leur cycle de vie. Néanmoins, un certain nombre de problématiques concernant les différentes solutions, plates-formes et technologies décrites précédemment subsistent encore comme le soulignent (McFarlane et al., 2012; Thomas, 2009). Ces derniers déclarent qu'il y a encore un grand nombre de questions ouvertes comme la collecte des informations liées au produit, leur stockage à travers la chaîne logistique, ou encore la dissémination et la gestion de ces informations tout au long de leur cycle de vie. De nouvelles architectures sont alors à concevoir pour prendre en compte l'évolution des technologies Auto-ID en particulier. En effet, nous avons pu constater que les produits dits « intelligents » dans la littérature sont dans la majorité des cas, des produits « discrets ». Par exemple, un produit embarque le plus souvent une puce RFID unique afin d'accéder à l'information déportée sur le réseau. Il peut aussi embarquer un capteur afin de réaliser des mesures sur son environnement, etc. Par conséquent, cette notion d'intelligence discrète est limitée, à savoir :

- lecture discrète : il est parfois nécessaire de « lire le produit » suivant un positionnement particulier, bien que la technologie RFID ait résolu en grande partie ce problème principalement lié à la technologie des codes à barres. Ce qui ne diffère cependant pas entre ces deux technologies, c'est le fait que l'étiquette RFID ou papier est localisée à un endroit particulier du produit, rendant de fait l'information discrète (l'information ne peut pas être attachée à des morceaux/zones différent(e)s de matière durant le PLC),
- risque d'endommagement des étiquettes : si les étiquettes (RFID ou codes à barres) sont illisibles (p. ex. tag endommagé, code à barres souillé), le flux d'informations relatif au produit sera dès lors rompu. (Tajima, 2007) souligne la grande proportion de tags défailants et de lecture impossible dans des projets pilotes, pouvant atteindre 20 à 50%,
- problème de transfert de l'information : lorsqu'un produit subit une transformation (p. ex., le produit est coupé en deux sous-ensembles ou pièces), une des parties résultantes de l'opération est alors vierge d'informations (excepté si l'on ajoute une étiquette à cette nouvelle pièce),
- faible capacité mémoire : la capacité des tags RFID actuelle est limitée et souvent corrélée à la dimension du tag (plus le tag est petit, plus la capacité mémoire décroît). Pour exemple, la plus petite puce RFID actuellement sur le marché (μ tag Hitachi) a une dimension de $0,4 \times 0,4 \times 0,1 \text{ mm}$ mais ne dispose pas de mémoire outre celle stockant l'identifiant du tag (Usami et al., 2007),
- niveau d'agrégation de l'intelligence : la plupart des produits discrets n'agissent pas comme un conteneur intelligent (cf. FIGURE 1.6).

Le paradigme de « matière communicante »

Depuis de nombreuses années et après avoir vu les produits intelligents ou holons comme des produits physiques associés à leur information via un agent ou un produit virtuel (grâce aux technologies Auto-ID), nous avons proposé un nouveau paradigme qui change radicalement la manière de voir le produit et la matière (Thomas, 2009). Ce nouveau concept consiste à donner la faculté au produit d'être intrinsèquement et intégralement « communicant ». « Intrinsèquement » signifie que le produit est « par essence » communicant ; d'un point de vue purement conceptuel, toute la matière est communicante ! Les travaux qui apportent des premières réponses à ce challenge sont ceux de (Kubler et al., 2010; Kubler et al., 2011a; Kubler et al., 2011b; Kubler, Derigent, Thomas and Rondeau, 2012a; Kubler, Derigent, Thomas and Rondeau, 2012b). L'idée forte et futuriste de ce concept est d'imaginer que la matière soit communicante au niveau cellulaire/moléculaire. Par conséquent, on ne se soucie pas de la localisation des composants de communication (p. ex. les tags RFID) dans la matière puisque toute la matière est communicante. Faisons l'illustration suivante :

Si on prend un morceau d'acier, lequel est composé de fer et de carbone dans un type de structure donné et qu'on le divise en morceaux de plus en plus petits, nul besoin de se soucier de savoir où sont les atomes de carbone, il y en a partout ! Si ces atomes de carbone étaient les instruments de communication, nous dirions que la matière est intrinsèquement communicante !

Évidemment, cette vision est loin d'être réaliste à l'heure actuelle, surtout à cause des limitations technologiques. Cependant, certaines recherches semblent être prometteuses, tel que le projet *Diabolo* (Brun et al., 2009) ou encore le prototype de textile communicant développé dans notre laboratoire (Kubler et al., 2011b), dans lequel des μ tags RFID ont été disséminés en masse. Bien que les technologies actuelles ne nous permettent pas de rendre l'« atome matière » communicant, la conception de nouvelles solutions, méthodes fournissant de nouveaux services aux utilisateurs peut être cependant engagée pour le futur. Par exemple, un produit fait de matière communicante pourrait servir à mettre en œuvre des solutions de sauvegarde, de copie et de redondance d'informations. Nous pourrions imaginer lui donner des facultés de prises de décisions ou de contrôles sur certaines de ses parties uniquement. Mais pour autant, l'ensemble des outils et méthodes développés par le concept de « matière communicante » devra pouvoir être facilement instanciables sur d'autres types de produits intelligents, dont ceux existants.

Le concept de « matière communicante » esquissé ci-dessus ouvre sur un champ de recherche considérable, tant d'un point de vue technologique que scientifique. Enumérer l'ensemble des problématiques liées à ce concept sans une démarche d'analyse rigoureuse serait difficile. Par conséquent, une technique adaptant des outils d'ingénieries existants a été développée et proposée dans (Kubler et al., 2010). Cette étude a permis de positionner de manière précise et concise les travaux de recherche entrepris dans cette thèse et a permis de dégager des problématiques à investiguer dans l'avenir.

1.3 Proposition du concept de « matière communicante »

L'étude proposée dans (Kubler et al., 2010) est basée sur une méthodologie de spécification des exigences et de définition des champs de recherches, tant technologiques que scientifiques, inhérents au nouveau concept de « matière communicante ». Cette approche est résumée dans la section qui suit et les problématiques résultantes sont détaillées.

1.3.1 Axes de recherche émergents

Dans cette étude, nous développons une approche d'ingénierie système que nous concevons à partir d'une analyse fonctionnelle (AFNOR, 1996) qui nous a permis de lister les fonctions que peut réaliser le produit. Pour ce faire, nous mettons en place une approche par matrices de hiérarchisation s'inspirant du QFD - *Quality Function Deployment* (Zaïdi, 1990). Rappelons que l'approche QFD permet aux entreprises de répondre au mieux aux attentes clients et conduit à déployer celles-ci tout au long du processus de conception, industrialisation par l'usage de matrices de corrélation et de pondération. Finalement, l'approche QFD permet de cibler les paramètres fondamentaux, ou justes nécessaires, pour satisfaire les attentes des clients. Il s'agit de travailler sur la qualité perçue dès la conception, alors qu'il est encore possible d'explorer un large éventail de solutions. Dans notre étude, cette approche permet, d'une part, d'identifier les problèmes technologiques et scientifiques relatifs à la matière communicante et, d'autre part, de déterminer les corrélations entre ces problèmes. Enfin, à l'aide d'une approche basée sur les algorithmes spectraux (Fiedler, 1975b; Fiedler, 1975a), ces différents problèmes sont regroupés par familles de problèmes à fortes corrélations, devant être traitées au sein d'un même groupe de travail et suivant un ordre de priorité. Cette approche est détaillée dans l'Annexe I. Au final, quatre groupes de problèmes sont identifiés :

1. **Conception de la matière** : ce groupe est essentiellement composé de problèmes liés aux contraintes technologiques relatives à la réalisation de la matière communicante. Cela concerne le choix de la technologie à mettre en œuvre (p. ex. RFID), l'intégration/la fixation des composants électroniques (p. ex. les systèmes tags) dans un composite fibreux ou dans un textile,
2. **Architecture de communication** : ce groupe traite des problématiques liées à la conception de l'architecture de communication, laquelle permet à la matière d'interagir avec son environnement. Il s'agit de définir le protocole de communication, tant au niveau de la couche physique (couche constituée par les tags et les lecteurs) qu'au niveau de la couche application. Nous traitons ici des problèmes tels que l'accessibilité à la matière, l'ordre des tags à lire et à écrire, la manière de décomposer une information entre plusieurs systèmes tags, etc.,
3. **Exploitation des données** : ce groupe est le « cœur » du projet, de par sa taille (*cf.* FIGURE 27 dans l'Annexe I), son positionnement (en interaction avec les trois autres groupes) et des problématiques le composant (leur nature « scientifique » et leur poids d'importance). Il s'agit ici de développer un système d'information permettant de stocker des informations dans la matière et

fournir aux utilisateurs des services précis tout au long de son cycle de vie. Ce système d'information nécessite de définir les spécifications conceptuelles des modèles de données représentant les informations échangées durant son cycle de vie, et requiert des mécanismes de dissémination des informations entre « matière communicante » et ses environnements. Les réponses aux spécifications seront suivies d'une étude de validation du système d'information en termes notamment de niveau de confiance offert (cohérence, disponibilité des informations distribuées et pertinence des informations stockées sur la matière) et d'une validation des protocoles et services concernant la fiabilité des communications et la facilité d'accès à la matière,

4. **Sécurité & Exhaustivité** : durant son cycle de vie, la matière est amenée à évoluer en raison de transformations physiques (découpe, assemblage,...), informationnelles (consultation, ajout/suppression d'informations) et de contraintes d'accessibilité (lecture/écriture partielle de la matière, ...). Ceci engendre une multitude de problèmes tels que le choix des informations à conserver, la modification des informations portées par la matière, la réduction du volume de données ou encore des problèmes de sécurité de données (cryptage, identification de l'acteur).

Le taux de contribution au projet final pour chaque groupe a également été calculé via les matrices de hiérarchisation¹⁰ (cf. FIGURE 27 dans l'Annexe I). Il en résulte que le groupe « Exploitation des données » a le taux de contribution le plus élevé avec 58%, contre 25%, 40% et 30% concernant respectivement les groupes « Conception de la matière », « Architecture de communication » et « Sécurité & Exhaustivité ». Le groupe « Exploitation des données » s'impose comme le cœur du projet. On évolue d'une vue « système physique » (groupes 1-2) vers une vue « système informationnel » (groupes 3-4) comme l'illustre la FIGURE 27 dans l'Annexe I. Sur la base de cette analyse, nous avons pu spécifier le cadre général des travaux de cette thèse. La prochaine section présente, positionne et décrit les problématiques/spécifications qui vont être traitées dans cette thèse.

1.4 Positionnement

La méthodologie mise en œuvre précédemment a permis de hiérarchiser les diverses problématiques scientifiques et technologiques. Nous avons constaté que les problématiques soulevées par les groupes 1, 2 et une partie du groupe 3 couvraient essentiellement les aspects de conception de l'architecture de « terrain ». Nous y retrouvons à la fois les problèmes de conception de la matière communicante (p. ex. le procédé industriel permettant de concevoir la matière communicante), mais également de conception de l'architecture de communication nécessaire pour lire/écrire la matière (outils matériels et logiciels). La seconde classe de problématiques (une partie du groupe 3 et le groupe 4) s'intéresse principalement à l'information dans le système : comment l'information est-elle représentée ? Comment est-elle disséminée dans le système (sur la matière, dans le réseau) ? Comment est-elle mise à jour ? Comment fournir

10. des exigences ont été formulées à l'aide de plusieurs analyses fonctionnelles. Le taux de contribution d'un groupe i indique, en pourcentage, le nombre d'exigences qui sont satisfaites si nous répondant à l'ensemble des problématiques contenues dans le groupe i .

l'information utile à l'utilisateur au moment opportun ? etc.

Dans la suite de ce document, nous allons donner des éléments de réponses pertinents et fondés à ces deux vues (systèmes physique et informationnel). Nous traitons en priorité la vue informationnelle et plus exactement, la manière de disséminer et gérer l'information dans le « système matière communicante ». Comme nous l'avons souligné auparavant, une multitude d'acteurs interviennent dans le PLC. Ces acteurs sont soit des bénéficiaires d'un service fourni par le produit (p. ex. le produit contrôle son environnement), soit des prestataires de service (p. ex. service de maintenance sur le produit). Aussi, il est nécessaire de gérer l'information dans l'espace (dissémination de l'information) et dans le temps (pérennité et cohérence de l'information). Notre objectif est de proposer une architecture globale du système d'information dédiée à la gestion du PLC, avec comme spécificité que les produits jouent le rôle de vecteurs d'informations. En d'autres termes, grâce au concept de « matière communicante », ces produits sont capables d'embarquer une quantité de données plus ou moins conséquentes, qu'ils véhiculent entre les différents acteurs de leur PLC. Il est alors nécessaire de fournir aux utilisateurs les outils permettant d'identifier les informations pertinentes devant être embarquées sur le produit, à un instant donné du PLC. Cette pertinence doit être évaluée selon plusieurs critères : attentes des utilisateurs, contraintes liées à l'environnement du produit, prise en compte des performances du système, etc. Les différentes plateformes introduites dans la section 1.2.3 (plates-formes agents, EPC global, ID@URI, WWAI) offrent un moyen de lier les informations des produits aux produits eux-mêmes, mais elles ne spécifient pas quelles informations doivent être déportées ou embarquées sur le produit et quelles sont leur pertinence au regard du cycle de vie du produit.

Puisque notre objectif est de proposer un cadre global de dissémination des informations avec des moyens de stockage fixe (p. ex. les bases de données des fournisseurs) et mobile (matière communicante), il est donc nécessaire de comprendre les principes et fondements de la distribution des informations au sein de tels environnements. C'est en ce sens que nous proposons dans le Chapitre 2, un état de l'art sur la distribution des informations en environnements fixes et mobiles dans le but d'identifier les solutions et mécanismes approprié(e)s à notre système d'information lié au « système matière communicante ».

Chapitre 2

Dissémination de l'information : principes et fondements

Sommaire

2.1	Architecture et distribution des informations relatives aux produits	28
2.2	Introduction aux systèmes de bases de données distribuées	30
2.3	Étapes clés de la distribution de l'information	31
2.3.1	Étape de fragmentation	31
2.3.2	Étape d'allocation	33
2.3.3	Conclusion	37
2.4	Dissémination des données en environnement mobile	37
2.4.1	Architectures, principes et mécanismes	37
2.4.2	Méthodes traitant de la réduction des données sur les équipements mobiles	41
2.5	Apports de la littérature dans un contexte « système matière communicante »	42
2.6	Proposition d'un processus de dissémination des informations relatives au produit	43
2.6.1	Entrées du processus de dissémination des informations	45
2.6.2	Mise en œuvre de processus de dissémination des informations dans le PLC	47

S'appuyant sur une revue de la littérature, le Chapitre 1 a mis en lumière que le partage de l'information est crucial pour une saine gestion du cycle de vie du produit. Cependant, le grand nombre d'acteurs présents tout au long de son cycle de vie et les nombreuses familles d'informations qui sont gérées par le système PLM ne facilitent pas un tel partage (problème de droit d'accès aux informations, de transfert des informations, de mise à jour des informations, d'interopérabilité entre systèmes, etc.).

La section 2.1 montre tout d'abord que l'architecture du système d'information liée au « système matière communicante » peut être décomposée en deux niveaux distincts. Le premier niveau fait référence aux systèmes de bases de données propres à chaque acteur (ou compagnie). Le second niveau vise à identifier les informations qui nécessitent d'être stockées/répliquées sur les produits¹ et à étudier leur implémentation dans la matière.

Notons que la littérature est riche de solutions permettant de distribuer de manière optimale des systèmes de bases de données. Par conséquent, afin de spécifier les fondements pour gérer au mieux la

1. désormais, l'emploi du mot « produit » dans le document fait référence à un objet fabriqué en matière communicante.

distribution des informations du premier niveau de l'architecture du système, nous proposons une taxonomie générale des principes et des mécanismes liés à la distribution des informations dans la section 2.2.

Sur la base de cette taxonomie, des états de l'art successifs sont présentés dans les sections 2.3 et 2.4. Trois objectifs sont visés : i) permettre au lecteur une compréhension des mécanismes déployés pour distribuer des informations, ii) présenter les travaux de référence qui gèrent la distribution des informations dans des systèmes de bases de données, iii) identifier les mécanismes qui devront être déployés au second niveau de l'architecture du système d'information. Par exemple, lorsque des informations sont stockées/répliquées sur les produits, il est nécessaire de mettre en œuvre des mécanismes particuliers afin de garantir une disponibilité et une cohérence des données.

Dans la section 2.5, nous identifions des solutions appropriées pour développer notre « système matière communicante » et gérer l'information au sein de celui-ci. Ceci pose certaines questions qui peuvent être formulées comme suit :

- sous quelle forme les informations doivent-elles être représentées/stockées dans le système ?
- comment les informations doivent-elles être disséminées dans le système ? Quels sont les critères pris en compte dans la communauté scientifique pour établir le plan de dissémination optimal ? Ces critères conviennent-ils à nos attentes et à notre contexte d'étude ?
- quels mécanismes permettent de gérer l'accès aux informations (disponibilité, cohérence, mise à jour des données) ? Quels sont ceux appropriés à notre environnement d'étude ?
- comment fournir les informations utiles à l'utilisateur, au moment opportun ?

Enfin, dans la section 2.6, nous proposons et décrivons un processus de dissémination des informations relatif au « système matière communicante ».

2.1 Architecture et distribution des informations relatives aux produits

Les informations générées par les applications relatives au système PLM (*cf.* section 1.2.2) sont principalement stockées dans des bases de données. Ces bases de données sont formalisées par des modèles de type entités-relations et leur mise en place nécessite des techniques particulières, permettant d'optimiser les performances du système (gain de temps pour accéder aux informations, respect des exigences relatives à la cohérence et à la disponibilité des informations, etc.). La littérature est riche de modèles et techniques permettant de spécifier la base de données adaptée au contexte d'étude, celle-ci pouvant être centralisée ou distribuée entre plusieurs sites (Ozsu and Valduriez, 1991). Cette étape correspond dans notre étude au premier niveau de l'architecture du système d'information global, comme illustré sur la FIGURE 2.1.

Le second niveau de l'architecture concerne la dissémination des informations sur les produits communicants. Notre approche consiste à mettre en place dans un premier temps l'architecture de niveau 1 puis, à répliquer (si souhaité/nécessaire) directement sur les produits communicants les informations pertinentes aux acteurs de son cycle de vie, comme illustré sur la FIGURE 2.1 (voir flèches vertes en poin-

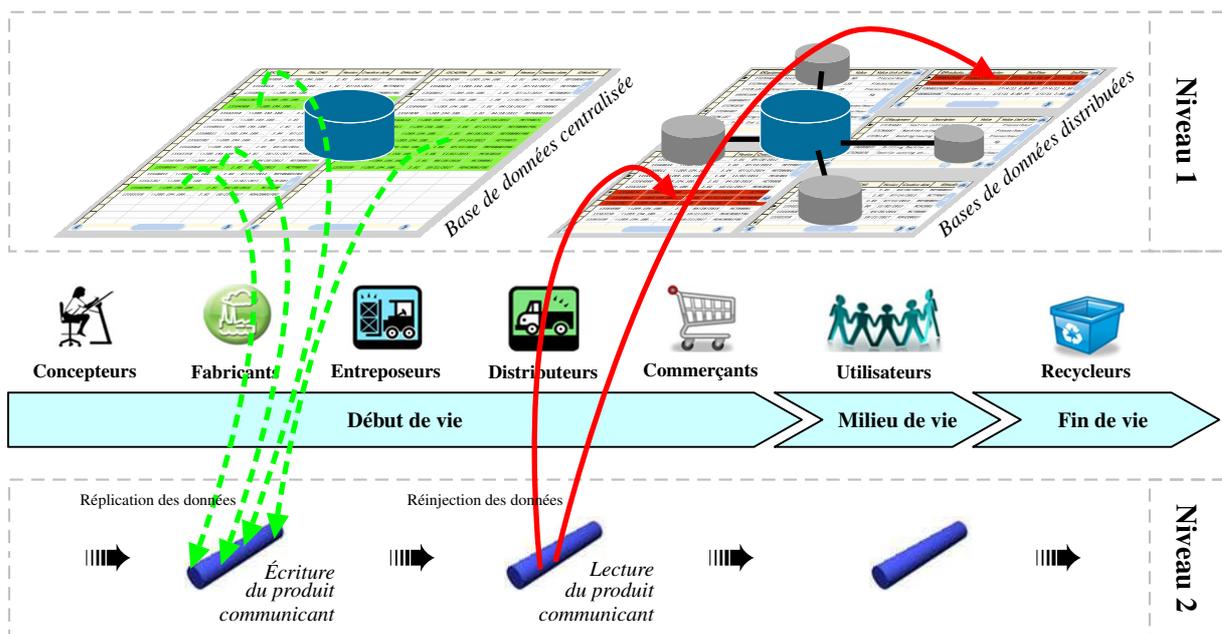


FIGURE 2.1: Système d'information à deux niveaux lié au « système matière communicante »

tillés). Les informations sont répliquées pour la simple raison qu'une matière communicante ne dispose pas des fonctionnalités intrinsèques d'une base de données². Cette matière joue uniquement le rôle de vecteur d'informations, capable de véhiculer durant son cycle de vie les informations entre acteurs et systèmes d'information. La matière communicante véhicule exclusivement les données qui lui sont propres, qui auraient été, par exemple, trop coûteuse et/ou impossibles de véhiculer d'une autre manière (p. ex., mise en place des accès entre fournisseurs trop coûteux ou impossibles), des informations de traçabilité, ou encore des informations destinées à offrir de nouveaux services aux utilisateurs (exemple des habits capables de communiquer avec la machine à laver afin de régler automatiquement le cycle de lavage (Ley, 2007)). Ces données sont par la suite lues et extraites de la matière communicante (à un point de lecture donné) et peuvent être directement exploitées, réinjectées dans le système de bases de données de l'utilisateur. Cette étape de « réinjection » nous ramène au niveau 1 de l'architecture comme le montrent les flèches rouge en trait plein dans la FIGURE 2.1, où des mécanismes traditionnels de mises à jour des informations sont employés. Il est donc nécessaire de comprendre en détail ces mécanismes et d'étudier les travaux du domaine dans le but de converger vers des mécanismes appropriés (objet des sections 2.2, 2.3 et 2.4). Puisqu'il est nécessaire à un instant donné du cycle de vie du produit de sélectionner les informations pertinentes (présentes dans les bases de données), qui doivent être embarquées/répliquées sur les produits communicants et ce, au regard d'une multitude de critères (attentes clients, contraintes applicatives, etc.), nous présentons dans la section 2.4 les travaux visant à sélectionner et embarquer sur des équipements mobiles des informations liées au contexte d'usage.

2. les informations portées par le produit ne peuvent être accessibles qu'à deux conditions i) disposer physiquement du produit ii) mettre en place un point de lecture (p. ex. via des lecteurs RFID).

2.2 Introduction aux systèmes de bases de données distribuées

Cette section résume les principes et les fondements de la distribution de l'information à l'aide d'une taxonomie générale présentée ci-après. Cette taxonomie sert de base pour classer les travaux de la littérature. Ces classifications vont permettre de dégager les meilleures approches pour gérer la distribution des informations au niveau 1 du système d'information global (cf. FIGURE 2.1).

Trois termes sont souvent utilisés dans les travaux relatifs à la distribution de l'information : bases de données réparties, fédérées et parallèles. Ce sont trois domaines distincts ayant une problématique commune, la distribution des données ou des traitements sur plusieurs unités de traitement qui coopèrent pour gérer les informations. La conception d'une base de données répartie peut s'appuyer sur deux approches :

- conception descendante (*Top-down design*) : à partir d'un schéma global, on construit les schémas locaux. Cette conception est intéressante lors de la mise en place de nouvelles bases de données. Dans ce cas, on parle de bases de données réparties/distribuées,
- conception ascendante (*Bottom-up design*) : elle permet l'intégration de bases de données locales existantes dans une base de données distribuées. Il s'agit cette fois de construire un schéma global à partir de schémas locaux existants. On parle alors de fédération de bases de données.

Le principe de distribution des informations s'effectue en deux étapes :

- la fragmentation : première étape dont l'objet est de subdiviser les tables du modèle physique de données en éléments simples appelés « fragments »,
- l'allocation : dernière étape qui consiste à allouer les fragments de données résultants à l'ensemble des sites composant le système d'information.

Une taxonomie résumant l'ensemble des techniques et concepts relatifs à la distribution de l'information est proposée dans la FIGURE 2.2. Cette taxonomie se compose de deux branches principales liées respectivement à la phase de fragmentation et à la phase d'allocation des données. Ces deux phases peuvent être soit « statiques » soit « dynamiques ». Une fragmentation ou une allocation statique signifie que le schéma de découpage des données n'est jamais amené à évoluer au cours du temps. En effet, le résultat de ces opérations n'est jamais remis en cause, ce qui s'avère problématique lorsque les caractéristiques de l'application évoluent fortement (p. ex. nombre de requêtes exécutées). Les approches dynamiques permettent de solutionner ce problème en réalisant une refragmentation ou une réallocation des fragments de données lorsque cela est nécessaire. La taxonomie montre que trois types de fragmentation existent : *verticale* (V), *horizontale* (H) et *hybride/mixte* (M), et qu'il est possible d'effectuer un chevauchement d'attributs ou de tuples, appelé *overlapping*³. Les trois types de fragmentation et la notion d'*overlapping* sont introduites par la suite. On remarque également que l'allocation de fragments de données peut se faire avec ou sans répliquions. Dans le cas où une répliquion est mise en œuvre, plusieurs mécanismes peuvent être implémentés pour assurer la cohérence et la disponibilité des données. Il

3. nous conservons le terme anglo-saxon dans la suite du document en raison de son usage courant dans la communauté francophone.

s'agit des mécanismes *Synchrone (S)*, *Asynchrone (As)*, *Primary Copy (Pc)* et *Update Everywhere (Ue)*.

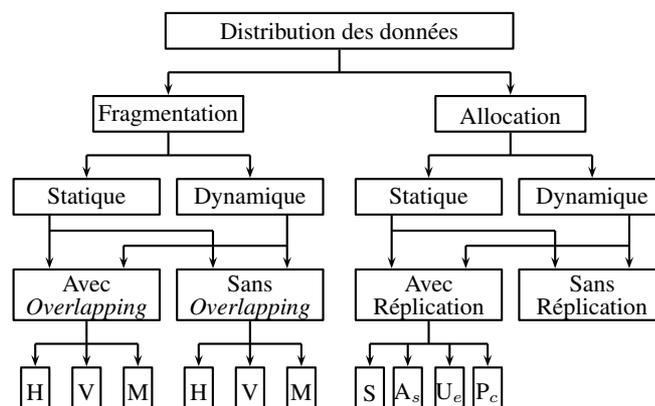


FIGURE 2.2: Taxonomie générale de distribution de l'information

Les prochaines sections ont pour objectif de détailler chacun des points présentés sur la FIGURE 2.2. Les mécanismes de fragmentation et d'allocation sont analysés dans la section 2.3. Les spécificités de ces méthodes en environnements mobiles sont exposées dans la section 2.4. Notons que chaque article inclus dans ces classifications a fait l'objet d'une analyse, où nous avons fait émerger ses points forts et ses points faibles. De manière à rester concis, ces analyses sont fournies dans l'Annexe II.

2.3 Étapes clés de la distribution de l'information

Comme cela a été souligné dans la section précédente, la distribution de l'information dans les systèmes de bases de données s'effectue en deux étapes : la fragmentation du schéma relationnel suivie de l'allocation des fragments de données résultants. Ces deux étapes font respectivement l'objet des sections 2.3.1 et 2.3.2.

2.3.1 Étape de fragmentation

L'étape de fragmentation des données consiste à diviser puis regrouper des données issues d'une table relationnelle. Avant de détailler les mécanismes de fragmentation existants, intéressons-nous à la terminologie utilisée. La FIGURE 2.3 illustre une table relationnelle nommée « voiture ». Le nom de la table équivaut également au nom de la relation. Les attributs de la table correspondent aux colonnes, les lignes sont appelées des « tuples » ou « instances » de la relation et l'intersection entre une ligne et une colonne correspond à une cellule de la table, appelée également « data item » en termes anglo-saxon. Nous conservons le terme anglo-saxon dans la suite du document. Le contenu d'un data item correspond le plus souvent à une chaîne de caractères comme illustré avec le data item localisé à la ligne 3, colonne 2 de la table voiture.

Il existe trois types de fragmentation : V, H, M. La fragmentation verticale (V) consiste à fragmenter les attributs d'une relation. La fragmentation horizontale (H) consiste à fragmenter un grand nombre

Nom de la relation/table ↙	Attribut 1	Attribut 2	Attribut 3
VOITURE	Marque	Couleur	N° d'immatriculation
tuple/instance 1 →	Audi	Argent	AC 369 FF
tuple/instance 2 →	Renault	Rouge	BE 934 YY
tuple/instance 3 →	Volvo	Blanc	DD 101 AC
tuple/instance 4 →	Toyota	Noir	RU 888 CY

┌ Data item de l'intersection du tuple 3 et de l'attribut 2, dont le contenu est « Blanc »

FIGURE 2.3: Terminologie relative à une table relationnelle

d'instances en sous-ensembles disjoints (c.-à-d., à grouper les tuples d'une relation). Enfin, la fragmentation hybride/mixte (M) divise dans un premier temps les relations horizontalement, puis chaque fragment obtenu est divisé à son tour de manière verticale ou vice-versa. La FIGURE 2.4(a) résume chacune de ces techniques. La fragmentation verticale est généralement utilisée durant la phase de conception de la base de données (Navathe et al., 1984) et par conséquent, est plus appropriée aux phases dites « statiques ». À l'inverse, la fragmentation horizontale est plus appropriée pour les mécanismes dynamiques. Dans les deux types de fragmentation, il est possible de réaliser de l'*overlapping* : *overlapping* d'attributs dans le cadre de la fragmentation verticale et *overlapping* de tuples dans le cadre de la fragmentation horizontale. Par définition, on dit qu'il y a *overlapping* lorsque l'intersection de deux fragments n'est pas vide, comme illustré sur la FIGURE 2.4(b).

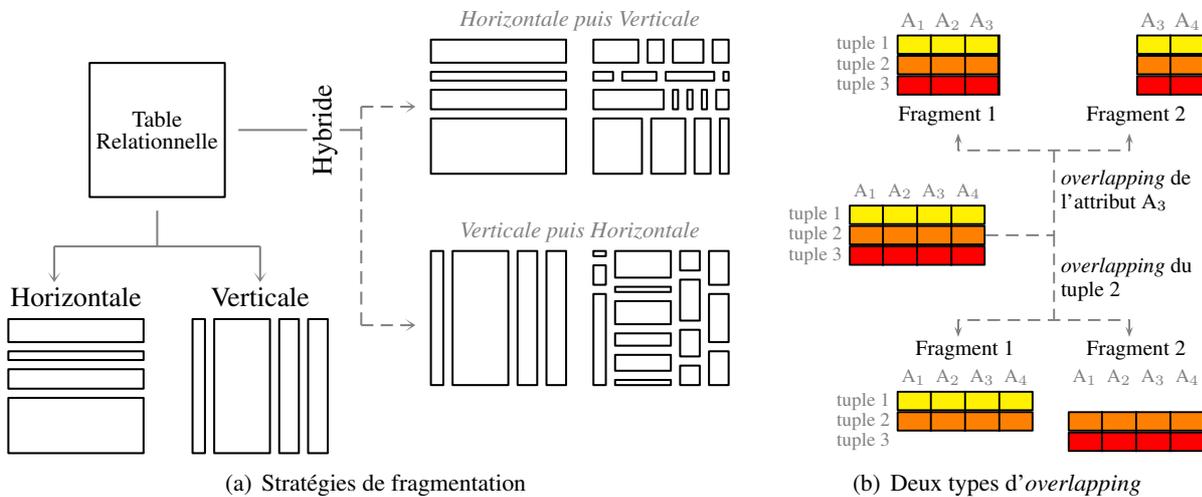


FIGURE 2.4: Stratégies de fragmentation et principe d'overlapping

La TABLE 2.1 positionne les travaux significatifs selon la taxonomie générale présentée dans la FIGURE 2.2. L'intitulé de chaque colonne correspond à un aspect précis de la taxonomie (branche dédiée à la fragmentation). Les deux dernières colonnes renseignent respectivement le nombre de citations du papier dans la littérature⁴ et le ratio « nombre de citations/nombre d'années depuis parution ». La table

4. sur la base du site web Google Scholar.

TABLE 2.1: Classification : méthodes de fragmentation

	Fragmentation	Overlapping	Dynamique	Statique	Nombre de citations	Nombre de citations/an
(Navathe et al., 1984)	V	✓		✓	308	308/28 = 11.0
(Apers, 1988)	M		✓	✓	247	247/24 = 10.3
(Navathe and Ra, 1989)	V	✓		✓	172	172/23 = 7.48
(Navathe et al., 1995)	M	✓		✓	83	83/17 = 4.88
(Son and Kim, 2004)	V			✓	29	29/8 = 3.62
(Tamhankar and Ram, 1998)	M			✓	47	47/14 = 3.35
(Baião and Mattoso, 1998; Baião et al., 2002)	M			✓	38	38/12 = 3.17
(Ezeife and Barker, 1998)	V			✓	43	43/14 = 3.07
(Bellatreche et al., 1997)	H			✓	41	41/15 = 2.73
(Chakravarthy et al., 1994)	V			✓	47	47/18 = 2.61
(Hauglid et al., 2010)	H		✓		5	5/2 = 2.50
(Abuelyaman, 2008)	V			✓	9	9/4 = 2.25
(Sleit et al., 2007)	M		✓		8	8/5 = 1.60
(Khan and Hoque, 2010)	H			✓	3	3/2 = 1.50
(Fung et al., 2002)	V			✓	12	12/10 = 1.20
(Malinowski and Chakravarthy, 1997)	V			✓	12	12/15 = 0.80
(Darabant et al., 2004)	H			✓	6	6/8 = 0.75
(Ezeife and Zheng, 1999a)	H		✓		1	1/13 = 0.08
(Pinto and Torres, 2002)	H	✓	✓		0	0/10 = 0.00
(Adrian Runceanu, 2007)	M			✓	0	0/5 = 0.00
	20	4	5	16		

est ordonnée suivant cette dernière colonne. Nous constatons que la majorité des travaux considérés dans cette classification repose sur des fragmentations statiques (16 contre 5 dynamiques) et par conséquent, sur la fragmentation verticale (10 contre 6 horizontales et 7 hybrides). Ceci montre que la fragmentation dans la littérature est plus étudiée pour le déploiement de systèmes d'information, et non pas pour maintenir les performances des systèmes dans le temps (uniquement 5 méthodes dynamiques).

2.3.2 Étape d'allocation

La phase d'allocation intervient à la suite de la fragmentation. Elle a pour but d'établir l'assignation optimale des fragments de données sur les différents sites composant l'architecture informatique. La FIGURE 2.5 illustre l'assignation de 4 fragments de données verticaux sur différents sites. Certains sites disposent de fragments (PC portable, Serveur 1, PC2, Serveur 2) et d'autres non (PDA, PC1).

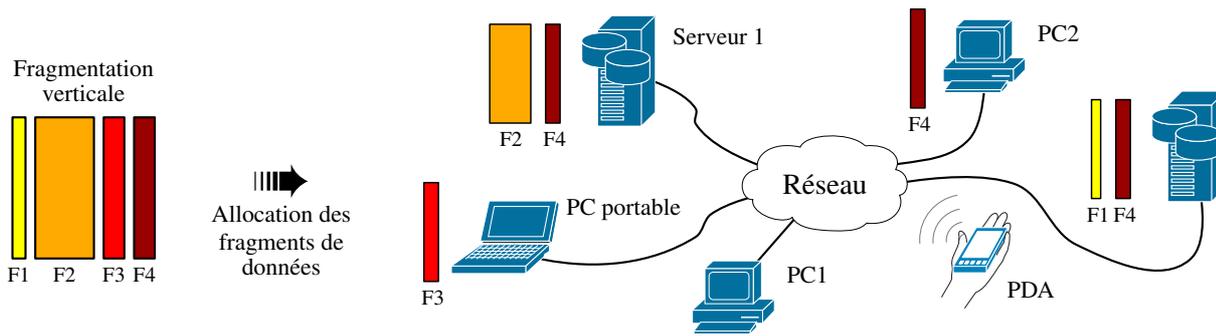


FIGURE 2.5: Terminologie relative à une table relationnelle

Habituellement, les méthodes tendent à assigner les fragments aux sites y accédant le plus afin de réduire les coûts de communication. À noter que l'opération de réplication des données est possible. Il s'agit de répliquer un même fragment sur plusieurs sites comme illustré sur la FIGURE 2.5 (le fragment F4 est répliqué sur Serveur 1, PC2 et Serveur 2). Ceci permet de garantir la fiabilité du système, mais également d'optimiser ses performances (Padmanabhan et al., 2008) (p. ex., réduction du trafic sur le réseau, gain de temps pour les utilisateurs). Cependant, la réplication des données nécessite des mécanismes gérant à la fois la diffusion des modifications (c.-à-d. les mises à jour des répliques) et les règles d'accès aux informations (autoriser un site ou un groupe de sites à modifier des répliques). Les informations et les attentes applicatives influent fortement sur les mécanismes à mettre en œuvre. Pour identifier le mécanisme de réplication le plus approprié, nous devons répondre à deux questions : Quand et Où ?

- Quand est-ce que les mises à jour doivent être propagées ? Deux modes existent :
 - mode *Synchrone* (S) : il est nécessaire d'avertir et de recevoir l'acquittement de l'ensemble des sites avant d'effectuer une modification d'un fragment de données.
 - mode *Asynchrone* (As) : il permet de réaliser des modifications localement, sans en avertir ses pairs.
- Où est-ce que les mises à jour doivent être effectuées ? Deux principes existent :
 - principe de la Copie Primaire (*Primary copy* - Pc) : il n'autorise qu'un seul site à effectuer la modification du fragment,
 - principe de la mise à jour par « tout le monde » (*Update everywhere* - Ue) : il autorise un groupe de sites à effectuer des mises à jour sur le fragment.

Dès lors, quatre types de répliquions peuvent être envisagés, à savoir :

1. *Update everywhere* - *Synchrone* (Ue-S)
2. *Update everywhere* - *Asynchrone* (Ue-As)
3. *Primary copy* - *Synchrone* (Pc-S)
4. *Primary copy* - *Asynchrone* (Pc-As).

La FIGURE 2.6 illustre le comportement des quatre types de répliquions⁵ à travers les échanges

5. chaque combinaison dispose d'avantages et d'inconvénients qui sont détaillés dans (Wiesmann et al., 2000).

réseaux résultant d'une transaction⁶. Considérons quatre sites [Site 1, Site 2, Site 3, Site 4] sur lesquels un fragment 1 est alloué/répliqué. Dans un premier temps, on autorise tous les sites à mettre à jour ce fragment basé sur le mode *synchrone* (principe du Ue-S). Le site 1 désire mettre à jour ce fragment, il effectue une demande aux sites 2, 3, 4 (sites disposant d'une réplique) et attend l'acquittement de ces derniers afin de procéder à sa transaction. Nous remarquons que dans le cadre d'une réplication Ue-As, des conflits peuvent survenir puisque l'on autorise tous les sites à modifier leurs données, sans se soucier des répliques. Néanmoins, des mécanismes de réconciliation existent pour solutionner ce type de problème (Gray et al., 1996; Petersen et al., 1997). À l'opposé, le principe du *Primary Copy* (Pc) oblige les sites désirant faire une mise à jour sur un fragment à faire une demande à un site particulier, appelé « Primary Copy », qui lui seul décide de quand et où réaliser la transaction et répliquer les mises à jour (selon que l'on est en mode S ou As). Nous pouvons voir sur la FIGURE 2.6 que le site 3 désire faire une mise à jour du fragment, par conséquent, il doit transmettre sa demande au site dit Pc, c'est-à-dire le site 1.

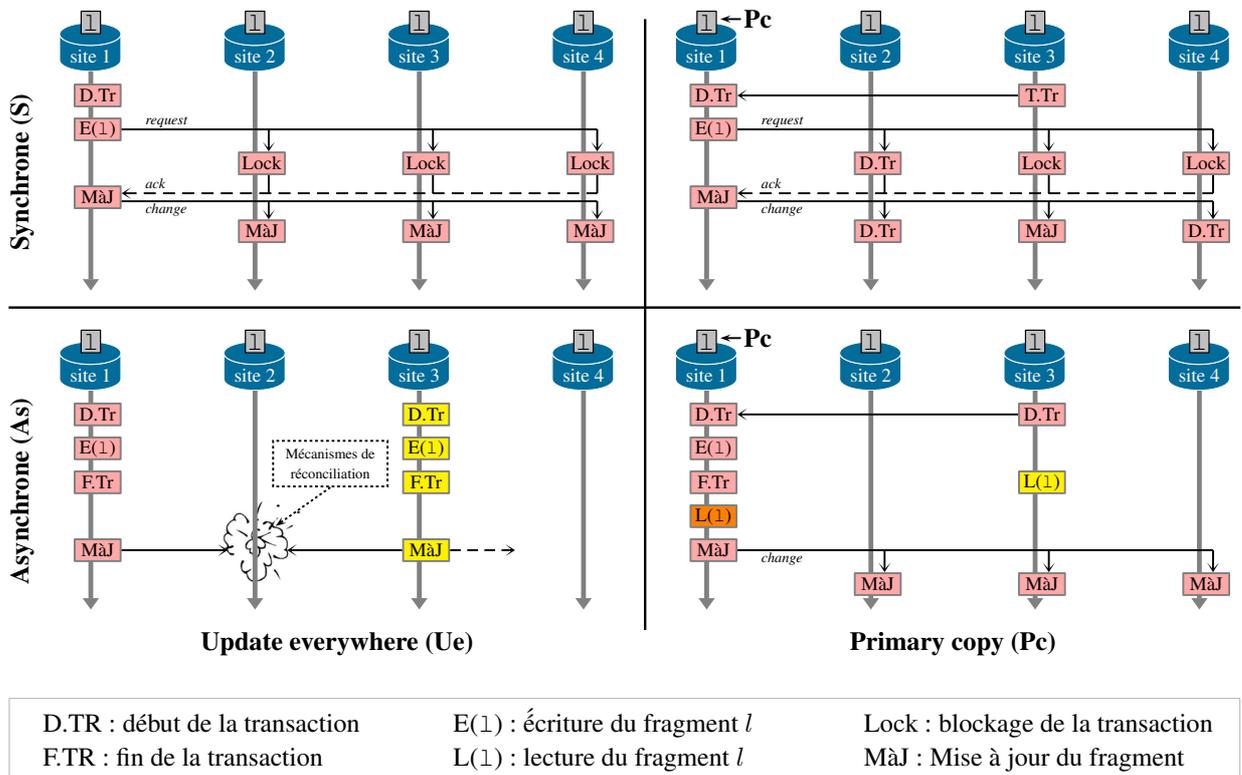


FIGURE 2.6: Techniques de réplication : Asynchrone, Synchrone, Update everywhere, Primary copy

La TABLE 2.2 présente une classification des travaux traitant des mécanismes d'allocation. Chaque intitulé de colonne correspond à un aspect particulier de la taxonomie et a été décrit dans la section 2.2. Notons que dans les colonnes intitulées « Allocation » et « Réplication », nous spécifions si la méthode

6. l'utilisateur interagit avec la base de données via des opérations de lecture/écriture, lesquelles sont appelées des « transactions ».

d'allocation requiert une fragmentation particulière (H, V, M) et si la réplication repose sur un mode particulier (P_c, U_e, S, A_s). Si uniquement le symbole « ✓ » est précisé, cela signifie qu'aucune information n'a pu être extraite à ce sujet. Comme précédemment, les travaux listés dans cette classification sont détaillés dans l'Annexe II. Nous remarquons globalement que la majorité des approches d'allocation est de type dynamique (14 contre 7 statiques), contrairement à la fragmentation. Cela montre que les performances des systèmes sur la durée sont principalement remises en cause à travers les méthodes de réallocation des données, et non à travers celles de refragmentation. Nous pouvons également noter que la moitié des travaux de la TABLE 2.2 gère la réplication de fragments de données. Ceci montre que le principe de réplication des données est un moyen convoité pour améliorer les performances des systèmes d'information distribués.

TABLE 2.2: Classification : méthodes d'allocation

	Fragmentation	Allocation	Réplication	Dynamique	Statique	Nombre de citations	Nombre de citations/an
(Apers, 1988)	M	M	✓	✓	✓	247	247/24 = 10.3
(Loukopoulos and Ahmad, 2004)		✓	P _c	✓	✓	59	59/8 = 7.37
(Huang and Chen, 2001)		M	U _e	✓		70	70/11 = 6.36
(Ahmad et al., 2002)		✓			✓	42	42/10 = 4.20
(Tamhankar and Ram, 1998)	M	M	P _c		✓	47	47/14 = 3.36
(Ulus and Uysal, 2003)		M		✓		25	25/9 = 2.78
(Brunstrom et al., 1995)		H		✓		43	43/17 = 2.53
(Hauglid et al., 2010)	H	H	P _c	✓		5	5/2 = 2.50
(Singh and Kahlon, 2009)		✓		✓		7	7/3 = 2.33
(Chin, 2002)		✓		✓		18	18/10 = 1.80
(Hababeh et al., 2005)		✓	✓		✓	12	12/7 = 1.71
(Sleit et al., 2007)	M	M	✓	✓		8	8/5 = 1.60
(Blankinship et al., 1991)		✓			✓	19	19/21 = 0.90
(Chaturvedi et al., 1994)		M	✓	✓		16	16/18 = 0.89
(Uysal and Ulus, 2007)		M		✓		4	4/5 = 0.80
(Petrescu, 2010)		✓	P _c -A _c	✓		1	1/2 = 0.50
(Grebla et al., 2004)		H		✓		2	2/8 = 0.25
(Tasharofi and Basseda, 2006)		✓		✓		1	1/6 = 0.17
(Hamideh Sabouri, 2008)		✓	U _e		✓	0	0/4 = 0.00
	4	19	10	14	7		

2.3.3 Conclusion

En résumé, des méthodes et mécanismes dédiés à la fragmentation des modèles relationnels, d'une part, et à l'allocation des fragments de données résultants, d'autre part, ont respectivement été exposés dans les sections 2.3.1 et 2.3.2. Ces méthodes sont principalement dédiées à des architectures fixes et sont utilisées pour la distribution de « grands » systèmes d'information, ce qui correspond au niveau 1 décrit dans la FIGURE 2.1. Les travaux précédemment exposés ne considèrent généralement pas la présence de vecteurs d'informations mobiles tels que des ordinateurs portables, des PDAs ou encore la présence de produits/matières communicant(e)s. Par conséquent, la prochaine section s'intéresse aux solutions et mécanismes utilisés dans le cadre d'environnements mobiles. En effet, notre matière communicante soulève des problèmes particuliers comme sa capacité mémoire limitée, ou le nombre de déconnexions au réseau qu'elle occasionne du fait de sa mobilité.

2.4 Dissémination des données en environnement mobile

Cette section a pour objectif de comprendre la gestion et la distribution des données au sein d'environnements mobiles. Cette analyse vise principalement à identifier les méthodes susceptibles d'être implémentées ou adaptées au niveau 2 de notre système d'information global pour stocker/répliquer les informations sur le produit.

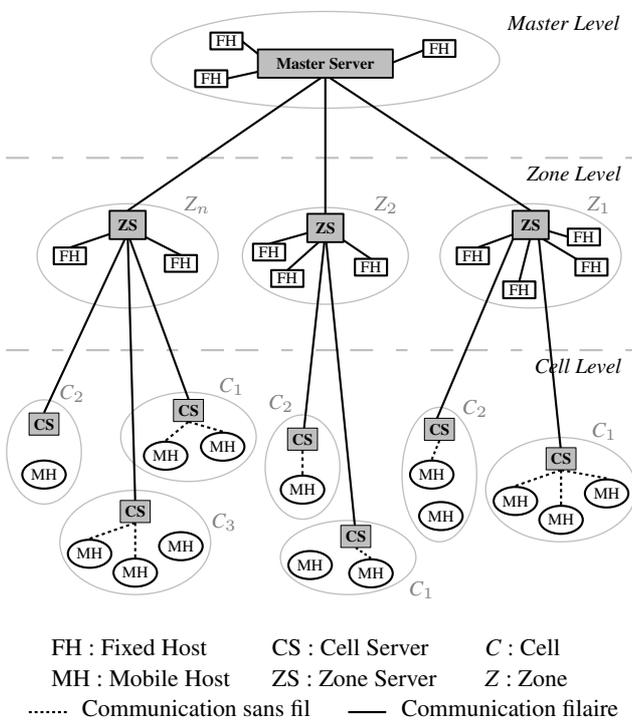
2.4.1 Architectures, principes et mécanismes

Les équipements mobiles ont des ressources limitées comparativement aux équipements fixes (batterie limitée, puissance de calcul limitée, faible bande passante). De plus, les communications entre nœuds⁷ sont souvent interrompues du fait de leur mobilité. Ces caractéristiques doivent être prises en compte dans la définition des méthodes de fragmentation et d'allocation des données. Par exemple, la réplication des données est primordiale au sein d'environnements mobiles afin de maintenir un niveau de disponibilité de l'information acceptable. (Padmanabhan et al., 2008) dressent un état de l'art des différentes techniques de réplication et indiquent la technique la plus appropriée au contexte applicatif.

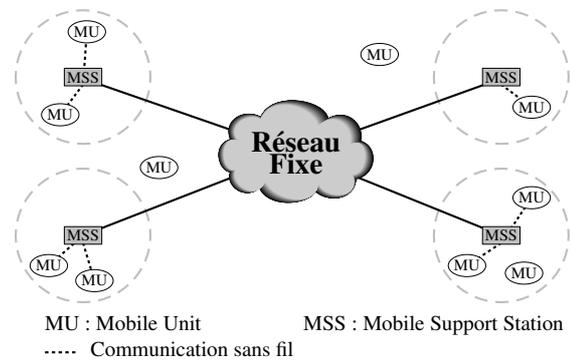
Plusieurs types d'architectures de réseaux mobiles ont été conçus et sont aujourd'hui utilisés pour la dissémination des données. (Chan and Roddick, 2003; Madria et al., 1998; Moiz and Rajamani, 2009; Barbará, 1999) proposent des environnements informatiques mobiles classiques qui reposent sur une architecture cellulaire similaire à celle utilisée dans les réseaux GSM (cf. FIGURE 2.7(b)). Elle se compose de stations de supports mobiles (MSS - *Mobile Support Station*) et d'unités mobiles (MU - *Mobile Unit*). Les MSS sont des équipements stationnaires responsables d'une aire géographique appelée cellule et sont connectés entre eux via un réseau filaire. Le MU est un équipement mobile qui peut passer d'une cellule à une autre et qui communique sans fil avec les MSS. On retrouve également, comme type d'architecture, les LMDDBS (*Large Scale Mobile Distributed Database System*) qui se caractérisent par un

7. le terme « nœud » est employé dans la littérature pour désigner un équipement mobile.

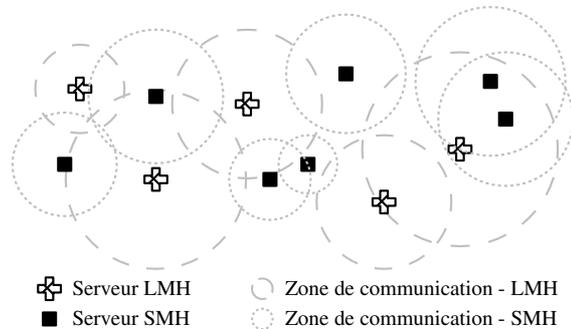
grand nombre de répliques de fragments de données (jusqu'à des milliers de répliques) et par un grand nombre de mises à jour. Ces répliques sont distribuées à la fois sur des zones mobiles et fixes. (Ashraf et al., 2010) proposent une nouvelle stratégie de réplication des données basée sur une architecture réseau hiérarchisée, comme illustrée sur la FIGURE 2.7(a). Ils divisent les répliques en trois niveaux (*Master level, Zone level, Cell level*) afin de fournir une plus grande flexibilité de distribution des répliques sur de vastes zones et afin d'augmenter la disponibilité des informations. Le dernier type d'architecture mobile connu est un MANET (*Mobile Area NETWORK*) qui est une collection de serveurs et de clients mobiles. Tous les nœuds sont sans fil, mobiles, à autonomie limitée et peuvent être classés en fonction de leur capacité comme expliqué dans (Fife and Gruenwald, 2003). La FIGURE 2.7(c) illustre un MANET dans lequel un client ou un *Small Mobile Host* (SMH) est un nœud muni de ressources de traitement, de stockage, de communication et de puissance limitées comparativement à un serveur ou à un *Large Mobile Host* (LMH). Cette notion de puissance est représentée graphiquement sur la FIGURE 2.7(c) par une zone circulaire autour de chaque équipement (zone dans laquelle sa transmission peut s'effectuer), laquelle varie en fonction de sa puissance. Habituellement, les serveurs LMH contiennent le système complet de gestion de la base de données et gèrent les requêtes clients. Les clients SMH n'ont pas suffisamment de ressources pour stocker le système complet, mais stockent généralement une partie de la base de données ainsi que certains modules de traitements (Gruenwald et al., 2002).



(a) Architecture mobile d'un LMDDBS



(b) Environnement informatique mobile classique



(c) Mobile Ad-hoc Network (MANET)

FIGURE 2.7: Types d'architectures mobiles

La TABLE 2.3 dénombre une quinzaine de travaux menée sur les bases de données distribuées mobiles. Trois nouveaux critères sont introduits dans cette classification en complément de ceux définis dans les TABLE 2.1 et TABLE 2.2, à savoir :

- le type de nœud : on spécifie si la méthode considère un environnement exclusivement mobile (M) ou hybride (c.-à-d., Mobile et Fixe : M-F),
- la capacité de stockage : on spécifie si la méthode tient compte de l'espace mémoire des équipements et/ou si une technique de réduction des données y est développée/utilisée (p. ex., compression, agrégation des données, etc.),
- la pertinence de l'information : elle traduit l'aspect lié à la sémantique de l'information, à savoir si la méthode intègre l'objectif de l'usage de l'information dans la sélection/dissémination des données. Ce critère est particulièrement important dans le cadre d'applications où la capacité⁸ est critique.

TABLE 2.3: Classification : Méthodes destinées aux environnements mobiles

	Fragmentation	Allocation/ Réplication	Dynamique	Statique	Capacité de stockage	Pertinence de l'info.	Type de nœud	Nombre de citations	Nombre de citations/an
(Demers et al., 2002)		P _c -A _s	✓		✓		M	377	377/10 = 37.7
(Huang et al., 1994)		✓	✓				M-F	201	201/18 = 11.2
(Hara and Madria, 2006)		P _c -A _s	✓		✓		M	104	104/6 = 17.3
(Bobineau et al., 2000)			✓	✓	✓	✓	M-F	105	105/12 = 8.75
(Phatak and Badrinath, 1999)	H			✓			M-F	21	21/13 = 1.62
(Chan and Roddick, 2003)		✓	✓	✓	✓	✓	M-F	14	14/9 = 1.56
(Lubinski and Heuer, 2000)		✓	✓		✓	✓	M-F	18	18/12 = 1.50
(Madria et al., 1998)				✓	✓		M-F	17	17/14 = 1.21
(Abawajy et al., 2006)		A _s -S					M-F	7	7/6 = 1.17
(Pabmanabhan and Gruenwald, 2006)		P _c -A _s	✓		✓		M	7	7/6 = 1.17
(Zábovský, 2008)		✓	✓				M	4	4/4 = 1.00
(Lubinski, 2000)		✓	✓	✓	✓		M-F	8	8/12 = 0.67
(Ashraf et al., 2010)		✓	✓				M-F	1	1/2 = 0.50
(Moiz and Rajamani, 2009)		P _c -A _s		✓			M-F	1	1/3 = 0.33
		1	11	10	6	8	3		

Nous constatons à partir de la TABLE 2.3 que les travaux menés dans le cadre des MANETs (*cf.* type de nœud=M) sont majoritairement basés sur une réplication de type P_c-A_s, puisque les nœuds sont conti-

8. au sens : quantité de données à stocker vs. espace mémoire disponible.

nuellement en mouvement. Notons également qu'en environnement mobile, le problème d'allocation des données est largement plus étudié que celui de la fragmentation. Cela s'explique principalement par le fait que la réplication des données dans de tels environnements permet de garantir une haute disponibilité des informations. Comme nous l'avons décrite dans la section 2.1 et illustrée sur la FIGURE 2.1, notre étude repose sur une architecture hiérarchique où le niveau 1 se compose principalement de bases de données fixes et le niveau 2 se compose de matières communicantes (jouant le rôle de vecteurs d'informations mobiles). Par conséquent, nous nous intéressons dans cette section aux environnements mixtes (c.-à-d. combinant des équipements mobiles et fixes) et non pas au MANETs (bien qu'une analyse de ces travaux soit proposée dans l'Annexe II).

Une stratégie de réplication est présentée dans (Abawajy et al., 2006), celle-ci met en œuvre différentes méthodes pour répliquer et gérer les données dans des environnements hybrides. Dans la partie fixe du réseau, les données sont répliquées de manière synchrone sur l'ensemble des sites, alors que dans la partie mobile, les données sont répliquées de manière asynchrone sur un et un seul site (principe As-Pc). À noter qu'une réplication synchrone empêche le réseau fixe d'être étendu sur de larges zones. (Huang et al., 1994) élaborent une technique où seuls les serveurs fixes sont en droit d'effectuer des mises à jour, les clients mobiles supportent uniquement des requêtes en lecture. Une nouvelle stratégie de réplication dite « optimiste » est proposée par (Ashraf et al., 2010) afin de maintenir une cohérence et une disponibilité des informations dans les LMDDBS. Leur stratégie repose sur trois niveaux hiérarchiques tels que nous les avons introduits précédemment dans la FIGURE 2.7(a) : *Master level*, *Zone level* et *Cell level*. Initialement dans cette architecture, la base de données complète est stockée sur le *Master server*. Lorsque le niveau *master* est divisée en une multitude de zones (c.-à-d., *Zone level*), une réplique de cette base de données est distribuée à chaque serveur ZS. De manière similaire, lorsque ces zones sont à leur tour divisées en une multitude de cellules (c.-à-d., *Cell level*), de nouvelles répliques sont alors générées sur les serveurs CS. Ensuite, une structure logique assurant l'échange des mises à jour entre l'ensemble des serveurs est proposée. Enfin, une méthode de réplication propre à leur stratégie est développée dans le but de gérer les répliques à un même niveau. (Lubinski and Heuer, 2000) offrent un cadre pour le choix du protocole de réplication à mettre en œuvre. Ils introduisent les stratégies de réplication « pessimiste » et « optimiste », ainsi qu'un modèle triparti reflétant les critères incontournables pour le choix du protocole, à savoir : la cohérence des informations, la disponibilité des informations et la minimisation des coûts de communication. Ils expliquent qu'il est impossible de satisfaire les trois objectifs simultanément comme illustré sur la FIGURE 2.8. En effet, un compromis doit être établi. Ils soulignent également l'influence de la « pertinence de l'information » sur le choix du mécanisme de réplication à mettre en œuvre. Cette pertinence est définie à partir des besoins des utilisateurs. Deux exemples sont fournis par (Lubinski and Heuer, 2000) :

- application santé (cas ① FIGURE 2.8) : en milieu hospitalier, les docteurs et infirmières traitent un grand nombre de patients et les informations médicales sont de grandes importances. Par conséquent, la cohérence des informations est de priorité majeure. Ceci peut être respecté uniquement en

réduisant la disponibilité (peu de répliques engendre moins d'incohérences) et/ou en s'assurant que les propagations des modifications soient effectuées de manière immédiate, ce qui augmente les coûts de communication. Ce compromis est illustré par la position ① dans le triangle FIGURE 2.8. Bien entendu, cette position varie en fonction des niveaux de disponibilité et de cohérence attendus,

- guide touristique (cas ② FIGURE 2.8) : considérons un guide touristique disponible sur PDA. Les touristes requièrent essentiellement des informations liées aux sites géographiques. De telles informations ne sont pas amenées à changer fréquemment, mais lorsqu'elles le sont, les coûts de communications sont significatifs. Bien entendu, la disponibilité est essentielle comme le traduit la position ② dans la FIGURE 2.8.

Après avoir convenu du meilleur compromis entre ces trois critères, les auteurs expliquent comment choisir le mécanisme de réplication .

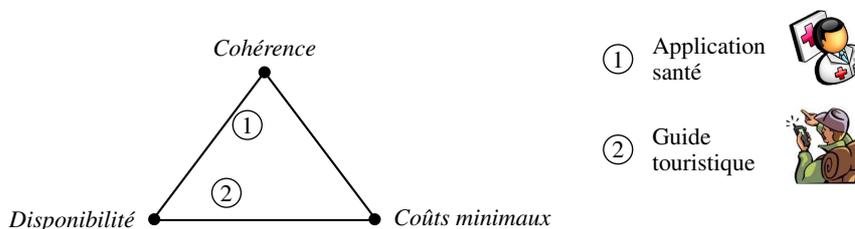


FIGURE 2.8: Triangle illustrant l'impossibilité de remplir les trois critères simultanément

2.4.2 Méthodes traitant de la réduction des données sur les équipements mobiles

La limitation des espaces mémoires des équipements mobiles est un problème fréquemment relevé dans la littérature (Lubinski and Heuer, 2000; Bobineau et al., 2000; Madria et al., 1998). En effet, les disques durs multiples et de grandes capacités présents dans les serveurs fixes ne peuvent pas être embarqués dans les équipements mobiles. Par conséquent, certains auteurs s'intéressent à la réduction des données, également appelé « *data summarization* » en termes anglo-saxon (sous-classe du *data mining* (Dunham, 2006)). (Chan and Roddick, 2005) répertorient les différentes techniques de *data summarization* dans le cadre des bases de données mobiles (agrégation des données, substitution de l'information, compression des données, ...) et évoquent au passage, la méthode de fragmentation/allocation comme une des techniques de réduction des données. En effet, cette technique permet d'embarquer sur l'équipement mobile qu'une partie de la base de données. De plus, les auteurs spécifient que cette technique peut être utilisée conjointement à d'autres mécanismes, citant l'exemple du couplage : fragmentation/concept hiérarchique. À propos du concept hiérarchique, un modèle de traitement des requêtes utilisant ce concept est présenté par (Madria et al., 1998), qui a pour but de gérer les différentes contraintes (déconnexions, faible bande passante, ...). Les auteurs fournissent une analyse sur les bénéfices de leur approche en fonction des coûts de stockage, des coûts de transmission et des coûts de traitement des

requêtes. (Lubinski, 2000) propose une méthode de réduction graduelle des données qui tient compte de la localisation des données (proches ou lointaines de l'utilisateur). En fonction de cette localisation, la méthode décide de transmettre des informations plus ou moins complètes à l'utilisateur. Dans la même direction, (Bobineau et al., 2000) développent une solution appelée *PicoDBMS* dédiée au domaine médical, laquelle permet de compacter fortement les données et d'exécuter les requêtes sans mémoire RAM. La méthode est basée sur un modèle de stockage mettant en œuvre des pointeurs. Elle garantit à la fois la compression des données, ainsi que des index. (Chan and Roddick, 2003) considèrent trois modes de communication : i) mode « connexion complète », ii) mode « déconnecté » iii) mode « connexion partielle ou faible ». Pour chaque mode, ils développent un processus de réduction des données qui dépend de la localisation des équipements mobiles (noté MU dans la FIGURE 2.7(b)), ainsi que des informations pertinentes pour l'utilisateur. Cette approche est intéressante dans le sens où elle tente de faire correspondre le contexte avec les données et ce, afin de sélectionner des informations sensibles au contexte (*context-sensitive information*). Cette approche permet de tenir compte d'une multitude de critères tels que les contraintes liées à l'environnement de la base de données mobile, des utilisateurs, des attentes des acteurs du cycle de vie et permet ainsi de sélectionner des informations sensibles et pertinentes au contexte d'usage, qui sont ensuite répliquées sur la base de données mobile.

2.5 Apports de la littérature dans un contexte « système matière communicante »

Au cours de ce chapitre, une classification des travaux sur la distribution de l'information a été décrite en considérant des environnements divers et variés (fixes, mobiles et hybrides). Ceci a été mené dans le but d'identifier les mécanismes appropriés au « système matière communicante » que nous désirons mettre en place. L'architecture globale du système d'information a été brièvement esquissée dans la FIGURE 2.1 et se compose de deux niveaux distincts :

1. les bases de données : il s'agit de mettre en place les systèmes de bases de données propres à chaque compagnie/acteur. Les sections 2.2 et 2.3 ont détaillé les travaux de la littérature permettant d'assurer la distribution optimale d'un système de base de données en fonction des attentes et contraintes applicatives. Dans notre « système matière communicante » tout type de mécanisme peut être envisagé et déployé. Plus concrètement, le système de bases de données peut être centralisé, distribué, peut implémenter des mécanismes de refragmentation ou réallocation des fragments de données, ou encore peut implémenter des mécanismes de réplication des données (de type Ue-S, Ue-A, Pc-S et Pc-As, cf. section 2.3.2). Le lecteur peut s'appuyer sur ces classifications pour converger vers les solutions appropriées à son cadre d'application,
2. la partie dédiée aux produits communicants : autour de ces systèmes de bases de données gravitent dans l'espace et dans le temps des produits communicants capables de véhiculer entre acteurs et systèmes des quantités d'informations plus ou moins conséquentes (en fonction de leur capacité

de stockage et des méthodes de codage de l'information). Dans notre cadre de dissémination, les informations pertinentes à véhiculer et à stocker sur la matière sont des répliques d'informations déjà stockées dans les bases de données. Les informations sont répliquées pour la simple raison qu'une matière communicante ne dispose pas des fonctionnalités intrinsèques d'un système de gestion de base de données traditionnel. En effet, les informations portées par le produit sont uniquement accessibles si l'on dispose physiquement du produit et de l'architecture permettant de lire celui-ci. Rappelons que la matière communicante vise à véhiculer des informations qui lui sont propres. Ces informations sont par la suite lues et extraites de la matière communicante (à un point de lecture donné), puis peuvent être directement exploitées (p. ex. via un logiciel dédié) ou, si souhaité, réinjectées dans les systèmes de bases de données des utilisateurs. Dès lors, cette étape de « réinjection » nous ramène aux problématiques de gestion des informations de niveau 1 et fait donc appel aux différents mécanismes introduits dans le précédent point. Il est important de noter que l'utilisateur pourra reconstruire les informations sans connaissance du MLD utilisé pour les stocker. Néanmoins, la réinjection de ces informations dans la base de données nécessite de partager un MLD commun. La prochaine étape vise, à un instant donné du cycle de vie du produit, à sélectionner les informations pertinentes présentes dans les bases de données qui doivent être embarquées/répliquées sur les produits communicants. Pour établir cette sélection, nous nous appuyerons sur l'approche développée par (Chan and Roddick, 2003), introduite précédemment, qui vise à sélectionner au sein d'une base de données des informations sensibles au contexte.

Ceci nous amène à la proposition d'un processus de dissémination des informations relatif au « système matière communicante ».

2.6 Proposition d'un processus de dissémination des informations relatives au produit

Le processus de dissémination des informations que nous proposons se compose de cinq étapes comme l'illustre la FIGURE 2.9, à savoir :

- étape *a* : mise en place du système de bases de données, faisant référence à l'ensemble des travaux présentés au cours du Chapitre 2. Cette première étape est bien entendu réalisée avant que le produit débute sa vie réelle/physique,
- étape *b* : sélection des informations pertinentes à stocker et plus exactement, à répliquer sur les produits communicants,
- étape *c* : stockage des informations pertinentes sur les produits,
- étape *d* : relecture des informations portées par le produit et reconstruction de celles-ci,
- étape *e*₁ : cette étape consiste à se connecter à la base de données pour soit réinjecter dans cette base les informations extraites du produit, soit compléter les informations extraites du produit (p. ex. pour répondre à une requête qui requière des informations non présentes sur le produit),

- étape e_2 : affichage des informations à l'utilisateur. Cette étape ne nécessite pas d'accès à la base de données (au réseau).

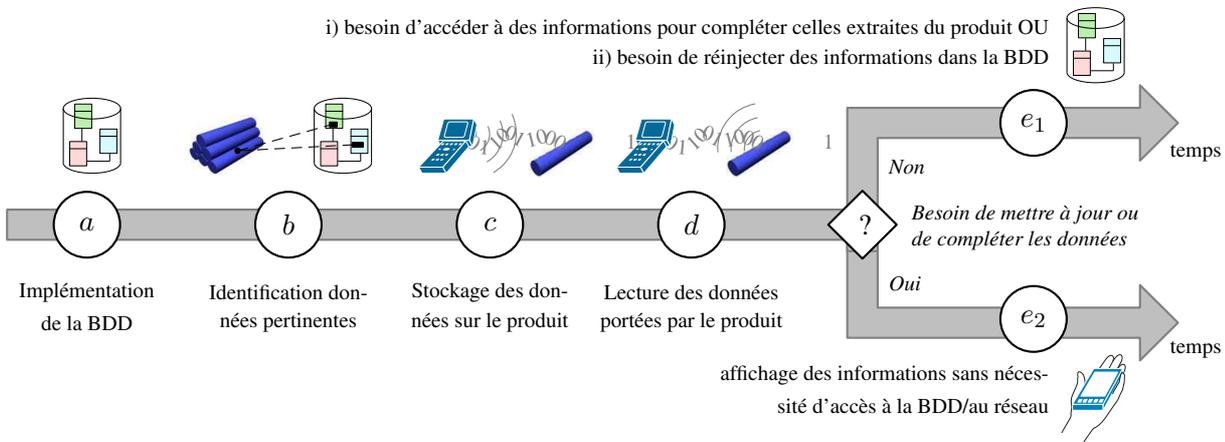
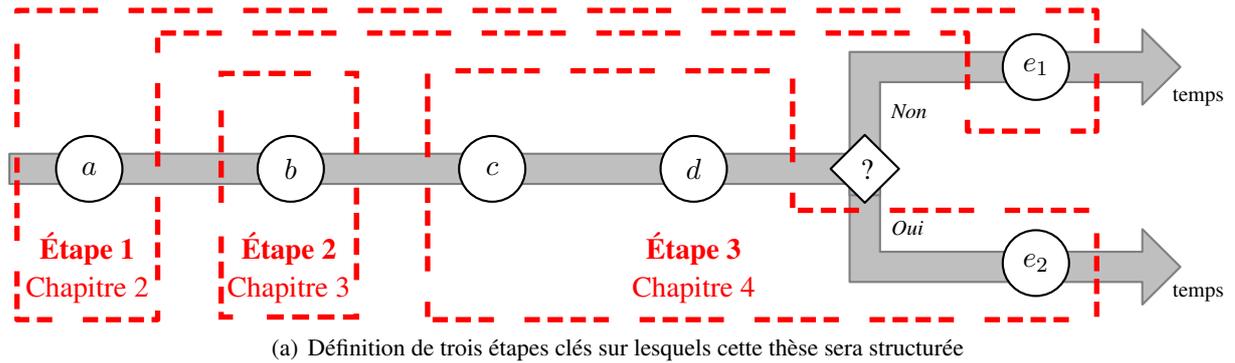


FIGURE 2.9: Vue temporelle du processus de dissémination des informations relatives au produit

Il est possible de structurer la suite de cette thèse autour de ce processus. Une possibilité est de détailler les étapes une à une. Cependant, certaines d'entre elles reposent ou utilisent les mêmes outils, à savoir :

- les étapes a et e_1 : l'étape e_1 correspond soit, à la réinjection dans la base de données des informations extraites du produits ou soit, à un accès aux informations présentes dans la base. Dans les deux cas, cela fait appel aux mécanismes initiaux mis en œuvre lors de l'étape a (c.-à-d. lors de la mise en place de la base de données),
- les étapes c , d et e_2 : l'étape c consiste à écrire sur le produit les informations qui ont été jugées pertinentes par l'étape b . À l'inverse, les étapes d et e_2 consistent respectivement à relire les informations qui ont été stockées sur le produit et à les reconstruire, les afficher. Ces trois étapes réalisent des fonctions différentes mais reposent sur un même protocole pour y répondre (protocole décrit dans la suite du document).

Dans cette thèse, nous souhaitons d'une part que la structuration des chapitres respecte l'aspect temporel du processus (ordre des étapes) et, d'autre part, que les chapitres soient homogènes (les étapes qui reposent sur les mêmes outils doivent être décrites dans un même chapitre) dans le but de faciliter la compréhension et la lecture. Par conséquent, les étapes a et e_1 sont traitées ensemble et forment désormais qu'une seule et même étape dans notre processus, appelée « Étape 1 », comme indiqué dans la FIGURE 2.10(a). L'étape b est traitée de manière individuelle et prend désormais la dénomination « Étape 2 » dans notre processus comme indiqué dans la FIGURE 2.10(a). Cette étape fait l'objet du Chapitre 3. Les trois étapes restantes (c , d et e_2) forment finalement la dernière étape du processus, appelée « Étape 3 », décrite dans le Chapitre 4. À noter que cette structuration engendre un déséquilibre en termes de nombre de pages entre les chapitres, particulièrement entre le Chapitre 3 et 4.



... indique une tâche informatique réalisée *online*
... indique une tâche informatique réalisée *offline*
... indique une entrée requise *online*
... indique une entrée requise *offline*

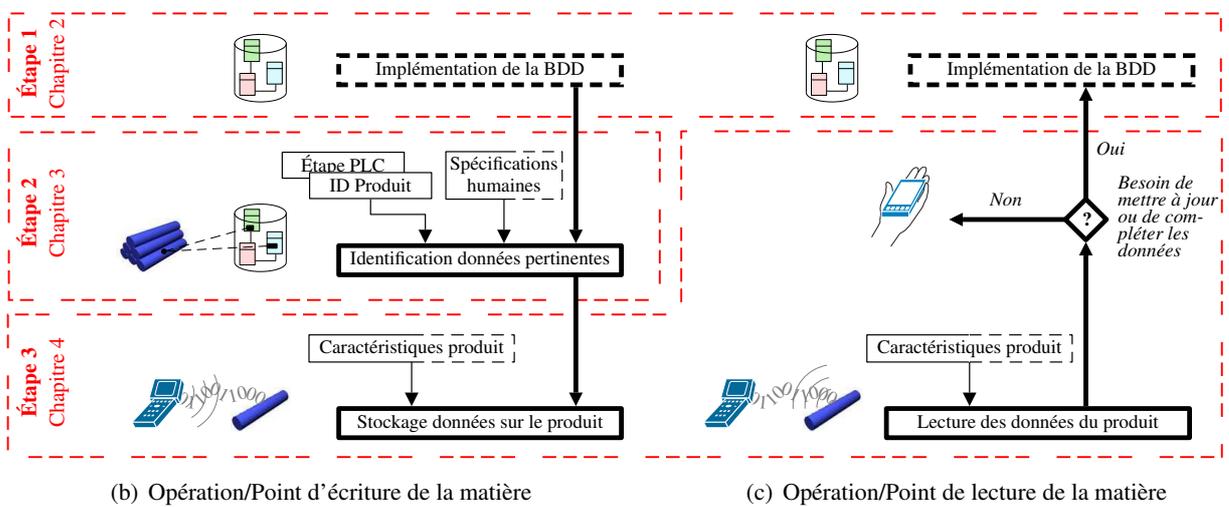


FIGURE 2.10: Processus de dissémination dans le cadre du « système matière communicante »

Sur la base de ces trois étapes clés, il est possible de représenter le processus de dissémination des informations au cours d'un « point d'écriture du produit » ou d'un « point de lecture du produit », comme l'illustrent respectivement les FIGURES 2.10(b) et 2.10(c). La section qui suit détaille les entrées requises par les trois étapes clés du processus.

2.6.1 Entrées du processus de dissémination des informations

Nous spécifions dans les FIGURES 2.10(b) et 2.10(c) les entrées requises par chaque étape et nous précisons si ces entrées et ces étapes sont générées/exécutées *online* ou *offline*. Une entrée générée/fournie *online* ou une étape exécutée *online* signifie que l'ensemble des opérations et calculs est effectué lorsque le produit est physiquement présent. Une entrée générée/fournie *offline* ou une étape exécutée *offline* signifie que l'ensemble des opérations et calculs est effectué avant que le produit ne soit présent.

Les entrées nécessaires à l'étape 1 (assurant la mise en place du système de bases de données) ne sont pas détaillées dans la suite du document puisque cette étape fait appel à des notions de bases de

données centralisées ou distribuées, en partie exposées dans le Chapitre 2 et dans l'Annexe II. Rappelons néanmoins que dans le cadre des bases de données distribuées, les méthodes qui permettent d'établir le schéma de distribution optimal des données (entre plusieurs sites) requièrent des informations d'entrée précises telles que le nombre d'accès sur les tables, le nombre d'accès sur les attributs des tables, les coûts des communications entre sites, les exigences et contraintes applicatives (p. ex. réplication ou non des informations), etc.

L'étape 2 qui consiste à identifier les « informations produit » issues de la base de données dont nous avons à évaluer la pertinence à être embarquées/répliquées sur le produit, nécessite :

- un accès aux informations contenues dans la (les) base(s) de données à chaque fois que le produit se présente à un point d'écriture à un moment donné de son cycle de vie (flèche entre l'étape 1 et 2 sur la FIGURE 2.10(b)). Cette entrée est par conséquent de nature *online*,
- la date/l'étape courante du cycle de vie du produit, notée « Étape PLC » dans la FIGURE 2.10(b). Cette date/étape doit être fournie à chaque fois que le produit se présente à un point d'écriture. Il s'agit une nouvelle fois d'une entrée *online*,
- l'identifiant porté par le produit, noté « ID produit » dans la FIGURE 2.10(b), qui sert de pointeur vers une ou plusieurs information(s) au sein de la base de données. Il s'agit d'une entrée *online* puisque cet identifiant est fourni lorsque le produit est physiquement présent au point d'écriture,
- des spécifications formulées par des experts. Ces spécifications sont nécessaires lors de l'évaluation de la pertinence des informations. Elles peuvent être, selon l'application d'usage, fournies de manière *online* ou *offline*. Par exemple, si nous considérons une tâche industrielle ayant un grand nombre de produits à traiter, il est préférable de définir l'ensemble des spécifications pour le type de produit concerné de manière *offline*. À l'inverse, si un utilisateur désire stocker une information particulière sur son produit (p. ex. les informations de lavage de son T-shirt), il peut effectuer les spécifications au moment venu, donc *online*. L'ensemble des critères de sélection est détaillé dans le Chapitre 3 avec les méthodes d'évaluation.

L'étape 3 qui gère le stockage des données sur le produit ainsi que l'extraction des données portées par le produit, requiert :

- des fragments de données identifiés dans l'étape 2 lorsqu'on est dans une opération d'écriture (flèche entre l'étape 2 et 3 sur la FIGURE 2.10(b)). Comme l'étape 2 est exécutée *online*, cette entrée est également de nature *online*,
- des caractéristiques liées au produit, notées « caractéristiques produit » dans la FIGURE 2.10(b), comme le type de technologie RFID embarquée, l'espace mémoire disponible, etc. Certaines de ces caractéristiques peuvent être fournies *offline* (p. ex. la technologie RFID embarquée dans la matière communicante) et d'autres *online* (p. ex., l'espace mémoire disponible sur la matière qui est fonction de la surface de la matière, de la quantité de tags au m^2 , de la mémoire déjà occupée).

Les outils et méthodes développés pour remplir les fonctions des étapes 2 et 3 du processus de dissémination des informations font respectivement l'objet des Chapitres 3 et 4. Pour conclure ce chapitre,

la prochaine section offre un bref aperçu de la mise en place de ce processus (opération d'écriture et de lecture du produit) dans le PLC.

2.6.2 Mise en œuvre de processus de dissémination des informations dans le PLC

La FIGURE 2.11 met en évidence que dans certaines activités du PLC, l'acteur peut vouloir lire/écrire des informations pertinentes sur le produit communicant. Il est alors amené à mettre en place l'opération d'écriture du produit décrite dans la FIGURE 2.10(b) (c.-à-d. le déploiement des étapes 2 et 3, l'étape 1 étant déjà effectuée) ou l'opération de lecture du produit décrite dans la FIGURE 2.10(c) (c.-à-d. le déploiement de l'étape 3 et, facultativement, de l'étape 1).

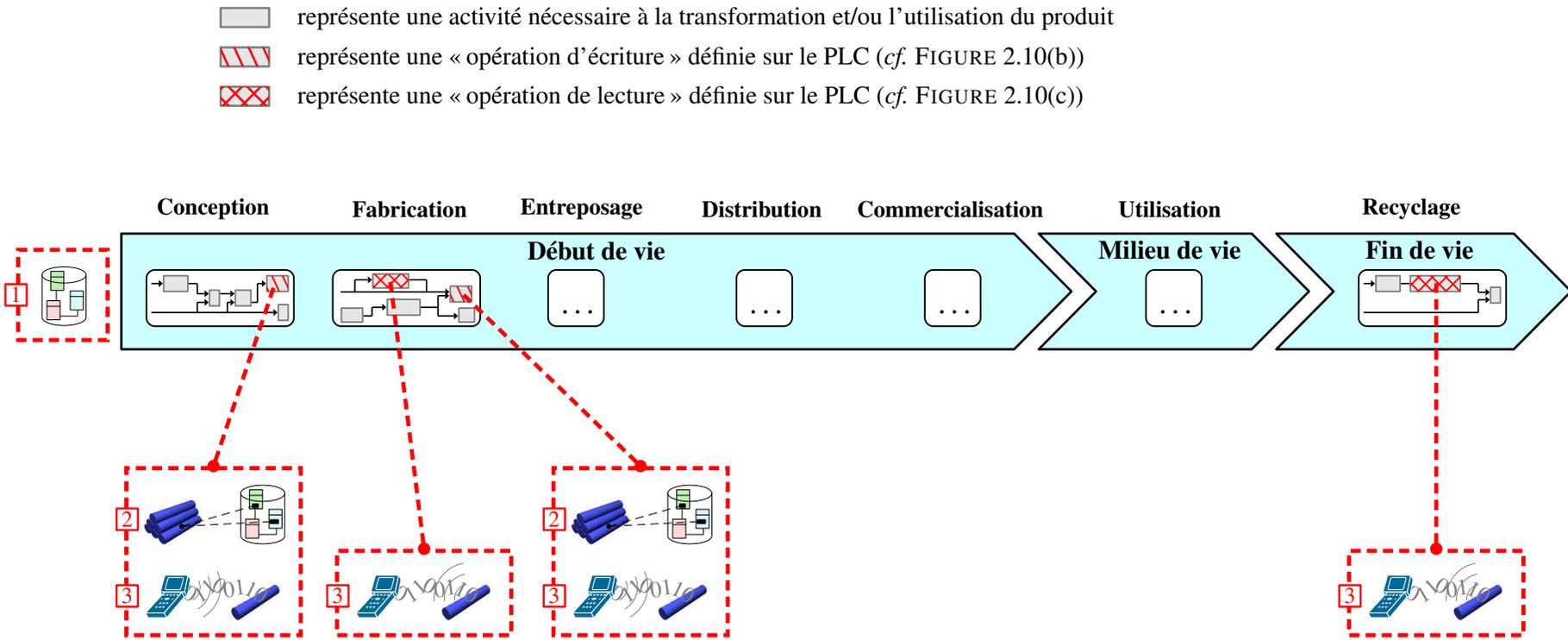


FIGURE 2.11: Illustration de plusieurs points d'écriture/de lecture du produit dans le PLC

Chapitre 3

Étape 2 : Sélection des informations pertinentes à stocker sur le produit

Sommaire

3.1	Algorithme d'identification des « informations produit »	50
3.1.1	Description de l'algorithme	50
3.1.2	Développement de l'algorithme RetrievalData	52
3.1.3	Algorithme détaillé par l'exemple	56
3.2	Méthode d'évaluation de la pertinence des informations : Chan-Roddick	58
3.2.1	Introduction	58
3.2.2	Calcul de la priorité relative ρ_x	60
3.2.3	Calcul de la priorité assignée $\phi_x(l)$	62
3.2.4	Synthèse du modèle d'évaluation proposé par Chan et Roddick	66
3.3	Méthode d'évaluation de la pertinence des informations : AHP-Floue	67
3.3.1	Introduction	68
3.3.2	Point 1 : structure AHP de l'étude	76
3.3.3	Point 2 : collecte et agrégation des informations	76
3.3.4	Point 3 : calcul de la matrice de jugement flou \tilde{A}	86
3.3.5	Point 4 : calcul de la matrice de performance floue \tilde{H}	86
3.3.6	Point 5 : classification des alternatives/data items	87
3.4	Conclusion de l'étape 2	91

Le processus général visant à disséminer les informations entre « produits communicants » et « bases de données traditionnelles » a été introduit dans le Chapitre 2 (synthétisé par la FIGURE 2.10). Le Chapitre 3 a pour objectif de fournir les méthodes et outils permettant de sélectionner les informations qui doivent être stockées sur le produit. Concrètement, lorsque le produit se présente à un point d'écriture durant son cycle de vie, cette étape vise à identifier au sein des bases de données toutes les informations qui sont liées au produit, puis à évaluer leur pertinence à être (ou ne pas être) répliquées/embarquées sur le produit communicant.

La phase d'identification des « informations produit » repose sur un algorithme qui est présenté dans la section 3.1.

La phase d'évaluation s'appuie dans un premier temps sur l'approche proposée par (Chan and Roddick, 2003) que nous présentons dans la section 3.2 et où nous soulignons ses limites. Aussi, nous proposons une nouvelle approche dans la section 3.3, qui utilise la méthode AHP-floue (*Fuzzy-Analytical Hierarchy Process* en termes anglo-saxon).

3.1 Algorithme d'identification des « informations produit » au sein de la base de données

Lorsque le produit se présente à un point d'écriture, il est nécessaire d'identifier les informations au sein de la base de données qui sont liées à ce produit. Il s'agit de réaliser un filtrage des informations afin d'éliminer celles n'ayant pas d'intérêt à être stockées/répliquées sur le produit et par conséquent, pas d'intérêt à être évaluées.

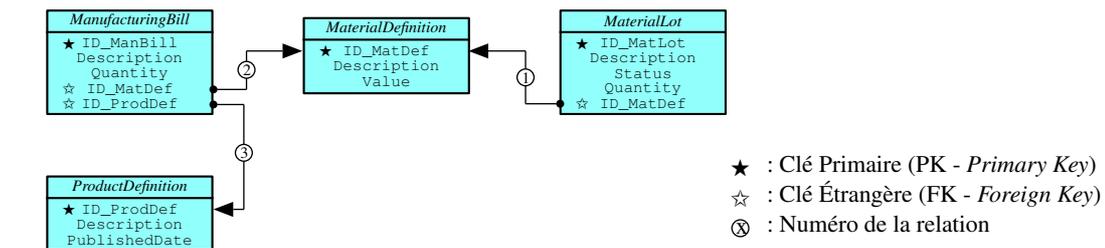
L'identification des « informations produit » est effectuée à l'aide d'un algorithme qui prend en entrée la structure du Modèle Logique de données (MLD) ainsi que les données des tables. La FIGURE 3.1(a) offre un aperçu d'une partie d'un MLD et des relations clés primaires (PK - *Primary Key*) - clés étrangères (FK - *Foreign Key*)¹. Une fois implémenté, ce MLD conduit aux tables de la base de données comme illustré dans la FIGURE 3.1(b) avec la table `MaterialDefinition`. Sur l'exemple de la FIGURE 3.1(b), seul le tuple 3 est identifié via l'algorithme comme étant un « tuple produit » (les tuples 1, 2 et 4 ne sont aucunement liés au produit).

La section 3.1.1 discute des différentes manières d'identifier ces « tuples produit », où nous montrons que les approches existantes sont limitées ou complexes. Une proposition est alors formulée dans la section 3.1.2, s'appuyant sur un algorithme spécifique. Enfin, de manière à faciliter la compréhension de cet algorithme, nous l'appliquons sur un exemple dans la section 3.1.3.

3.1.1 Description de l'algorithme

Au cours de cette section, les différentes manières permettant d'identifier les « tuples produit » présents dans les tables de la base de données sont étudiées. En effet, les « informations produit » sont liées entre tables par les clés (primaires, étrangères) et il est nécessaire de rechercher de manière hiérarchique ces informations. Par exemple, il faut retrouver des informations dans une nomenclature de produits, dans laquelle les produits sont composés d'un ou plusieurs sous-produits, qui à leur tour, sont composés d'un ou plusieurs sous-composants, etc. Dans un premier temps, les approches utilisant des requêtes traditionnelles sont présentées, lesquelles s'avèrent limitées et complexes au vu de notre objectif. Par conséquent, une nouvelle approche est introduite. Cette approche prend en entrée la référence portée par le produit qui est utilisée comme pointeur vers une (ou plusieurs) information(s) au sein de la base de données. Cette référence produit est notée d_p (abréviation pour Data item Produit).

1. une clé étrangère dans une table A est forcément clé primaire dans une table B ($B \neq A$). C'est ce qui permet de mettre en relation ces deux tables.



(a) Exemple d'une partie d'un Modèle Logique de Données (MLD)

	1 (★)	2	3
	ID_Material	Description	Value
1	MD041...	Wooden plank with a nominal...	4m of...
2	MD993...	Textile with a high developed pol...	3mm...
3	MD060...	Textile which is provided with...	15°C
4	MD770...	Vehicle headrests that conform...	2 × ...

 Indique un « tuple produit » (lié à d_p)

(b) Table MaterialDefinition et les « tuples produit » identifiés

FIGURE 3.1: Vue d'une partie du MLD, de la table MaterialDefinition et des « tuples produit »

L'utilisation de requêtes traditionnelles nécessite une requête initiatrice permettant d'identifier le tuple relatif à d_p dans la base de données. Ensuite, le résultat obtenu est utilisé pour construire dynamiquement les requêtes suivantes dans le but d'explorer les tables voisines, donnant lieu à de nouvelles requêtes de manière à explorer l'ensemble des tables. En langage SQL (*Structured Query Language*), la construction de requêtes faisant référence à d'autres requêtes peut être menée via deux mécanismes qui sont, les « sous-requêtes » et les « requêtes récursives ». Les sous-requêtes (Garcia-Molina et al., 2008) sont des requêtes SQL imbriquées les unes dans les autres et qui permettent d'exécuter une série de requêtes liées dans un code unique. Cependant, ce mécanisme n'est pas adapté à notre objectif car plus le nombre de tables est élevé, plus le nombre de sous-requêtes à définir est conséquent. L'utilisation d'une telle approche rend alors la conception du processus d'extraction des données très complexe et très coûteuse en temps. De plus, tout changement au sein du MLD (p. ex., ajout de nouvelles tables ou d'attributs) rendrait impossible la construction d'un algorithme générique basé sur les sous-requêtes. Le second mécanisme utilise les requêtes récursives définies dans le standard SQL :1999 (Melton et al., 2001). À titre d'exemple, dans Microsoft Server 2005 (Otey, 2005), les requêtes récursives sont formées en utilisant une table commune appelée CTE (*Common Table Expression*), qui est une table temporaire pouvant se référencer elle-même et créer ainsi une table récursive. Une telle table permet donc de retourner des sous-ensembles de données de manière récursive, jusqu'à ce que l'ensemble complet des résultats soit atteint. Une utilisation commune des requêtes récursives est de rechercher de manière hiérarchique des informations comme l'exemple donné concernant la nomenclature des produits ou encore, retrouver les employés dans un organigramme. Le principe de notre processus d'extraction des données est récursif puisqu'il répète la même procédure jusqu'à identifier l'ensemble des tuples liés à d_p . Cependant, ces requêtes ne sont pas exécutées de manière récursive sur une même table, ce qui rend impossible l'utili-

sation d'un tel mécanisme. La dernière possibilité est d'utiliser des langages d'automatisation de bases de données tel que le langage JDBC (*Java DataBase Connectivity*). Cette dernière solution permet le développement d'algorithmes capables d'utiliser le résultat de chaque requête dans le but d'explorer dynamiquement l'ensemble des tables. Dans notre approche, nous proposons, lorsque le produit se présente à un point d'écriture, d'extraire l'ensemble des tables \mathcal{T} et de les parcourir à l'aide d'un algorithme particulier, nommé `RetrievalData`.

3.1.2 Développement de l'algorithme `RetrievalData`

L'ALGORITHME 1 permet de parcourir les tables de la base de données et d'identifier les « tuples produit ». Cet algorithme s'appuie sur deux sous-fonctions : `ExplorePK` et `ExploreFK`, qui sont décrites ci-après. L'ensemble des variables utilisées est listé dans la TABLE 3.1.

TABLE 3.1: Définition des variables utilisées dans les ALGORITHMES 1, 2 et 3

Variabiles	Description
$\mathcal{T} = [T_1..T_t..T_z]$	Ensemble de tables composant le MLD, où T_t représente la table t et z indique le nombre de tables extraites du MLD
$T_t\{i, j\}$	Réprésente le data item localisé à la ligne i et à la colonne j de la table t , avec $i \in \{1, m_t\}$ et $j \in \{1, n_t\}$, où m_t indique le nombre de tuples et n_t le nombre d'attributs de la table t
$\text{cont}(T_t\{i, j\})$	Indique le contenu du data item (<i>cf.</i> terminologie présentée dans la FIGURE 2.3). $\text{cont}(T_t\{i_1, j_1\}) = \text{cont}(T_{t'}\{i_2, j_2\})$ signifie que le contenu du data item localisé à la ligne i_1 , colonne j_1 de la table T_t est identique au contenu du data item localisé à la ligne i_2 , colonne j_2 de la table $T_{t'}$
$\text{value}(T_t\{j\})$	Indique le nom de la colonne j de la table T_t . $\text{value}(T_t\{j_1\}) = \text{value}(T_{t'}\{j_2\})$ signifie que le nom des colonnes j_1 et j_2 des tables T_t et $T_{t'}$ respectivement, sont identiques ²
$\mathcal{K} = [K_1..K_t..K_z]$	Ensemble des clés liées au modèle de données, où K_t est un vecteur renseignant le type de clé de chaque attribut de T_t . Un élément de K_t est noté $K_t\{j\} \in \{0, 1, 2\}$, où 0 signifie que l'attribut j de la table t n'est ni une clé primaire (PK), ni une clé étrangère (FK), 1 signifie que c'est une PK et 2 signifie que c'est une FK
$d_p = T_p\{i_p, j_p\}$	Point de départ de l'algorithme tel que $K_{T_p}\{i_p, j_p\} = 1$, où T_p est la table référencée par le produit, i_p est l'instance et j_p l'attribut. d_p est fourni par l'appareil qui lit ou écrit le produit.
$\mathcal{R} = [R_1..R_t..R_z]$	Ensemble des matrices où R_t représente une image de la table t , indiquant si les data items sont liés au produit (c.-à-d., s'ils sont liés ou non à d_p). Un data item dans R_t est noté $R_t\{i, j\} = \text{true}$ s'il s'agit d'un « data item produit » et $R_t\{i, j\} = \text{false}$ sinon
D	Matrice renseignant les distances entre les tables composant le MLD, où $D_{T_t, T_{t'}}$ correspond à la distance entre les tables T_t et $T_{t'}$ (nombre de tables les séparant)
\mathcal{E}	Ensemble des tables déjà explorées
\mathcal{N}	Ensemble des tables qui n'ont pas encore été explorées et qui sont directement en relation avec des tables déjà explorées (c.-à-d. des tables $\in \mathcal{E}$)
\mathcal{I}	Ensemble des data items qui restent à explorer dans les tables, où I_l représente le $l^{\text{ème}}$ élément de I

L'algorithme `RetrievalData` identifie tous les tuples qui sont liés d'une manière ou d'une autre à la référence d_p , en parcourant une à une l'ensemble des tables \mathcal{T} . Le principe de l'algorithme est de parcourir les tables en s'appuyant sur la connaissance du modèle, c'est-à-dire sur la connaissance des clés primaires (PK) et clés étrangères (FK). Cette fonctionnalité est fournie par la sous-fonction `ExplorePK` (voir ALGORITHME 2). Cette sous-fonction identifie les clés étrangères d'une table donnée (celle en cours d'exploration), puis poursuit son exploration dans les tables correspondant à ces clés étrangères (où

2. par exemple, lorsque l'attribut clé primaire j_1 de la table T_t correspond à un attribut clé étrangère j_2 de la table $T_{t'}$

FK est PK). Ainsi, au cours de cette exploration, de nouveaux « tuples produit » sont identifiés. Notons que la fonction `ExplorePK` peut être insuffisante pour explorer toutes les tables de \mathcal{T} car certaines relations peuvent ne pas être atteintes. En effet, lorsqu'une table ne possède pas de clé étrangère, il est alors nécessaire de procéder à l'inverse. En d'autres termes, rechercher de nouvelles relations dans des tables non explorées et qui sont en relation directe avec les tables déjà explorées³. Ce procédé inverse est réalisé par la sous-fonction `ExploreFK` qui est détaillée dans l'ALGORITHME 3. Les différentes étapes de l'algorithme `RetrievalData` et de ses sous-fonctions `ExplorePK`, `ExploreFK` sont détaillées sur un exemple dans la prochaine section.

Algorithme 1: `RetrievalData(\mathcal{T}, d_p)`

output : \mathcal{R}

```

1 begin
2    $\mathcal{I} \leftarrow d_p$ ; // Initialisat° de la liste  $\mathcal{I}$  avec le data item référencé par le produit
3    $\mathcal{E} \leftarrow \emptyset$ ; // Initialisation à vide de la liste des tables parcourues
4   while  $\mathcal{I} \neq \emptyset$  do // On poursuit l'exploration des tables tant que  $\mathcal{I}$  n'est pas vide,
    après exploration basée sur les PK et FK (i.e. ExplorePK et ExploreFK)
5     while  $\mathcal{I} \neq \emptyset$  do // On poursuit l'exploration des tables tant que  $\mathcal{I}$  n'est pas vide,
      après exploration basée sur les PK (i.e. ExplorePK)
6       [ $\mathcal{I}'$ ;  $\mathcal{R}$ ]  $\leftarrow$  ExplorePK( $\mathcal{T}, \mathcal{R}, \mathcal{I}_1$ ); // On identifie les « data items
          produit » (mis à true dans  $\mathcal{R}$ , cf. TABLE 3.1) et on relève les data items qui
          restent à être explorés (ces data items composent la variable  $\mathcal{I}'$ )
7        $\mathcal{I} \leftarrow \mathcal{I} \cup \mathcal{I}'$ ; // On ajoute les data items devant être explorés à ceux déjà
          présents dans  $\mathcal{I}$ 
8        $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{I}_1$ ; // On mémorise le fait que la table contenant  $\mathcal{I}_1$  a été explorée
9        $\mathcal{I} \leftarrow \mathcal{I} - \mathcal{I}_1$ ; // On supprime le data item qui a été utilisé pour
          l'exploration des tables, afin de poursuivre avec le suivant
10       $\mathcal{I} \leftarrow$  ExploreFK( $\mathcal{E}, \mathcal{T}$ ) // On recherche de nouvelles relations (de nouveaux data
          items) dans des tables non explorées jusqu'alors et qui sont liées d'une manière
          ou d'une autre à  $d_p$ . Ces nouveaux data items composent la variable  $\mathcal{I}$ 

```

3. nous recherchons en réalité des PK de tables déjà explorées qui sont des FK dans des tables non explorées. Alors qu'avec la précédente sous-fonction `ExplorePK`, nous recherchons à partir des FK, les tables dans lesquelles ces clés sont PK.

Algorithme 2: ExplorePK($\mathcal{T}, \mathcal{R}, T_t\{i, j\}$)output : $\mathcal{I}', \mathcal{R}$

```

1 begin
2    $\mathcal{I}' \leftarrow \emptyset;$  // On initialise la liste des data items restant à parcourir, à vide
3   forall the  $j_1 \in \{1, n_t\}$  do // pour tous attributs  $j_1$  de la table  $T_t$  (i.e. la table qui
   contient le data item  $T_t\{i, j\}$ , qui est un paramètre d'entrée de la fonction ExplorePK)
4      $\mathcal{R}_t\{i, j_1\} \leftarrow true;$  // Mettre à true les « data items produit »
5     if  $\mathcal{K}_t\{j_1\} = 2$  then // Si l'attribut  $j_1$  de la table  $T_t$  est une FK alors
6       forall the  $t' \in \mathcal{T}, j_2 \in \{1, n_{t'}\} | value(T_t\{j_1\}) = value(T_{t'}\{j_2\}) \ \& \ \mathcal{K}_{t'}\{j_2\} = 1$  do
       // Nous parcourons toutes les tables  $\in \mathcal{T}$  et nous cherchons à identifier dans
       quelle table  $T_{t'}$  et pour quel attribut  $j_2$ , la clé étrangère  $j_1$  de  $T_t$  est PK
7       forall the  $i_2 \in \{1, m_{t'}\} | cont(T_t\{i, j_1\}) = cont(T_{t'}\{i_2, j_2\})$  do // Si  $j_2$  est
       identifié, nous parcourons les tuples de la table  $T_{t'}$  et nous cherchons à
       identifier si le contenu d'un data item de la colonne  $j_2$  (colonne PK de  $T_{t'}$ )
       est égal à celui de la table initiale (data item noté  $T_t\{i, j_1\}$ )
8        $\mathcal{I}' \leftarrow \mathcal{I}' \cup T_{t'}\{i_2, j_2\};$  // Si la condition est validée, il s'agit alors
       d'un data item à parcourir, qui est donc ajouter à la liste  $\mathcal{I}'$ 

```

Algorithme 3: ExploreFK($\mathcal{E}, \mathcal{T}, \mathcal{R}$)output : \mathcal{I}

```

1 begin
2    $\mathcal{I} \leftarrow \emptyset;$  // On initialise la liste des data items restant à parcourir à vide
3    $\mathcal{N} \leftarrow \emptyset;$  // On initialise la liste des tables non explorées qui sont directement en
   relation avec des tables déjà explorées, à vide
4   forall the  $T_t \in \mathcal{E}, T_{t'} \in \mathcal{T} | D_{T_t, T_{t'}} = 1 \ \& \ T_{t'} \notin \mathcal{E}$  do // Nous recherchons les tables  $T_{t'}$ 
   qui n'ont pas encore été explorées (i.e.  $T_{t'} \notin \mathcal{E}$ ) et qui sont distantes d'une table des
   tables  $T_t$  qui ont déjà été explorées (i.e.  $T_t \in \mathcal{E}$ ), ce qui est noté  $D_{T_t, T_{t'}} = 1$ 
5      $\mathcal{N} \leftarrow \mathcal{N} \cup T_{t'};$  // Si la condition est validée, on ajoute alors  $T_{t'}$  à  $\mathcal{N}$ 
6   forall the  $T_t \in \mathcal{N}, j_1 \in \{1, n_t\} | \mathcal{K}_t\{j_1\} = 2$  do // Pour tous les attributs  $j_1$  des tables
    $T_t$  composant  $\mathcal{N}$ , on recherche les attributs  $j_1$  qui sont des FK (i.e.  $\mathcal{K}_t\{j_1\} = 2$ )
7     forall the  $T_{t'} \in \mathcal{E}, j_2 \in \{1, n_{t'}\} | \text{value}(T_t\{j_1\}) = \text{value}(T_{t'}\{j_2\}), \mathcal{K}_{t'}\{j_2\} = 1$  do
   // Si  $j_1$  et  $T_t$  sont identifiés, on cherche à identifier dans quelles tables déjà
   explorées  $T_{t'}$  ( $T_{t'} \in \mathcal{E}$ ) et pour quel attribut  $j_2$ , la clé étrangère  $j_1$  de  $T_t$  est PK
8       forall the  $i_1 \in \{1, n_{t'}\} | \mathcal{R}_{t'}\{i_1, j_2\} = \text{true}$  do // Si  $T_{t'}$  et  $j_2$  sont identifiés,
   nous parcourons tous les tuples de la table  $T_{t'}$  et nous cherchons à identifier
   si le data item localisé à la ligne  $i_1$ , colonne  $j_2$  de  $T_{t'}$  est un « data item
   produit » (i.e. si  $\mathcal{R}_{t'}\{i_1, j_2\} = \text{true}$ )
9         forall the  $i_2 \in \{1, m_t\} | \text{cont}(T_t\{i_2, j_1\}) = \text{cont}(T_{t'}\{i_1, j_2\})$  do // Si
    $\mathcal{R}_{t'}\{i_1, j_2\}$  est un data item produit, on regarde si le contenu de  $T_{t'}\{i_1, j_2\}$ 
   (i.e. la PK  $T_{t'}\{i_1, j_2\}$ ) est présente dans la colonne  $j_1$  (FK) de la table  $T_t$ 
10         $\mathcal{I} \leftarrow \mathcal{I} \cup T_t\{i_2, j_1\};$  // Si condition validée, on ajoute  $T_t\{i_2, j_1\}$  à  $\mathcal{I}$ 

```

3.1.3 Algorithme détaillé par l'exemple

Les différentes étapes de calcul de l'algorithme `RetrievalData` sont détaillées sur un exemple mettant en œuvre trois tables : `MaterialDefinition`, `MaterialLot` et `ManufacturingBill`, comme illustré dans la FIGURE 3.2(a). Les abréviations « MatDef », « MatLot » et « ManBill » sont également employées dans notre exemple concernant ces trois tables.

L'algorithme nécessite la référence du produit d_p afin de commencer l'exploration des tables. Nous supposons que cette référence correspond au data item $T_{\text{MatLot}}\{2, 1\} = \text{LPB61}$, comme pointé sur la FIGURE 3.2(a). L'exploration des tables débute alors par la fonction `RetrievalData` avec $d_p = T_{\text{MatLot}}\{2, 1\}$. La FIGURE 3.2(b) présente les étapes de `RetrievalData` sous forme d'algorithme où la syntaxe $\#l$ fait référence à la ligne l de l'ALGORITHME 1. $T_{\text{MatLot}}\{2, 1\}$ est donc ajouté à \mathcal{I} (#2) et \mathcal{E} est mis à vide. Puisque $\mathcal{I} \neq \emptyset$ (#4,#5), l'algorithme appelle la sous-fonction `ExplorePK`($\mathcal{T}, \mathcal{R}, T_{\text{MatLot}}\{2, 1\}$) (#2)⁴. Cette sous-fonction met, d'une part, à *true* dans \mathcal{R} ⁵ le tuple contenant $T_{\text{MatLot}}\{2, 1\}$ (c.-à-d la ligne 2 de `MaterialLot` comme illustré dans la FIGURE 3.2(a)) et, d'autre part, cherche s'il y a une ou des clé(s) étrangère(s) liée(s) à ce « tuple produit ». Ceci est vrai avec MD06 (c.-à-d. $T_{\text{MatLot}}\{2, 5\}$) qui est une clé primaire dans `MaterialDefinition` ($T_{\text{MatDef}}\{3, 1\}$) (flèche bleue en trait plein). $T_{\text{MatDef}}\{3, 1\}$ est alors ajouté dans \mathcal{I} (#7) et l'exploration se poursuit (#8, #9, #5⁶). Le data item $T_{\text{MatDef}}\{3, 1\}$ est ainsi utilisé pour explorer `MaterialDefinition` via la sous-fonction `ExplorePK`, comme indiqué par `ExplorePK`($\mathcal{T}, \mathcal{R}, T_{\text{MatDef}}\{3, 1\}$) (#6)⁷. Cette sous-fonction met, d'une part, à « *true* » dans \mathcal{R} le tuple contenant $T_{\text{MatDef}}\{3, 1\}$ (c.-à-d la ligne 3 de `MaterialDefinition`) et, d'autre part, cherche s'il y a une ou des clé(s) étrangère(s) liée(s) à ce « tuple produit ». Cette fois-ci, il n'y en a pas (il n'y a pas de symbole \star au-dessus des attributs de `MaterialDefinition`), par conséquent $\mathcal{I}' = \emptyset$, et \mathcal{I} devient également vide (#7,#8,#9,#5⁸). On fait donc appel à la seconde sous-fonction `ExploreFK`($\mathcal{E}, \mathcal{T}, \mathcal{R}$) (#10), qui recherche d'éventuelles relations où les clés primaires des tables déjà parcourues (c.-à-d de tables $\in \mathcal{E}$) sont des clés étrangères dans des tables qui n'ont pas encore été parcourues (c.-à-d de tables $\in \mathcal{T}$ et $\notin \mathcal{E}$)⁹. Dans notre MLD, la table `ManufacturingBill` est une voisine de la table `MaterialDefinition` (l'attribut 1 de `MaterialDefinition` est clé étrangère dans `ManufacturingBill` : attribut 4) et n'a pas encore été explorée. Par conséquent, la fonction `ExploreFK` regarde s'il y a une clé primaire de `MaterialDefinition` qui est une clé étrangère dans la colonne 4 de `ManufacturingBill`. Ceci est vrai avec MD06 (flèche rouge en pointillé). De nouveaux « data items produit » sont ainsi identifiés, $T_{\text{ManBill}}\{1, 4\}$ et $T_{\text{ManBill}}\{3, 4\}$ et sont utilisés pour explorer le reste des tables en se basant à nouveau sur la sous-fonction `ExplorePK` (cette nouvelle boucle n'est pas décrite dans l'algorithme). La FIGURE 3.2(a) montrent les tuples qui ont

4. les étapes exactes de `ExplorePK` sont présentées dans l'ALGORITHME 4 de l'Annexe III.

5. \mathcal{R} est une image \mathcal{T} indiquant les tuples qui sont des « tuples liés au produit » (mis à la valeur « *true* »).

6. #5 correspond au 2^{ème} passage dans la boucle *while* #5 de `RetrievalData`.

7. les étapes exactes sont présentées dans l'ALGORITHME 5 de l'Annexe III.

8. #5 correspond au 3^{ème} passage dans la boucle *while* #5 de `RetrievalData`.

9. les étapes exactes de `ExploreFK` sont présentées dans l'ALGORITHME 6 de l'Annexe III.

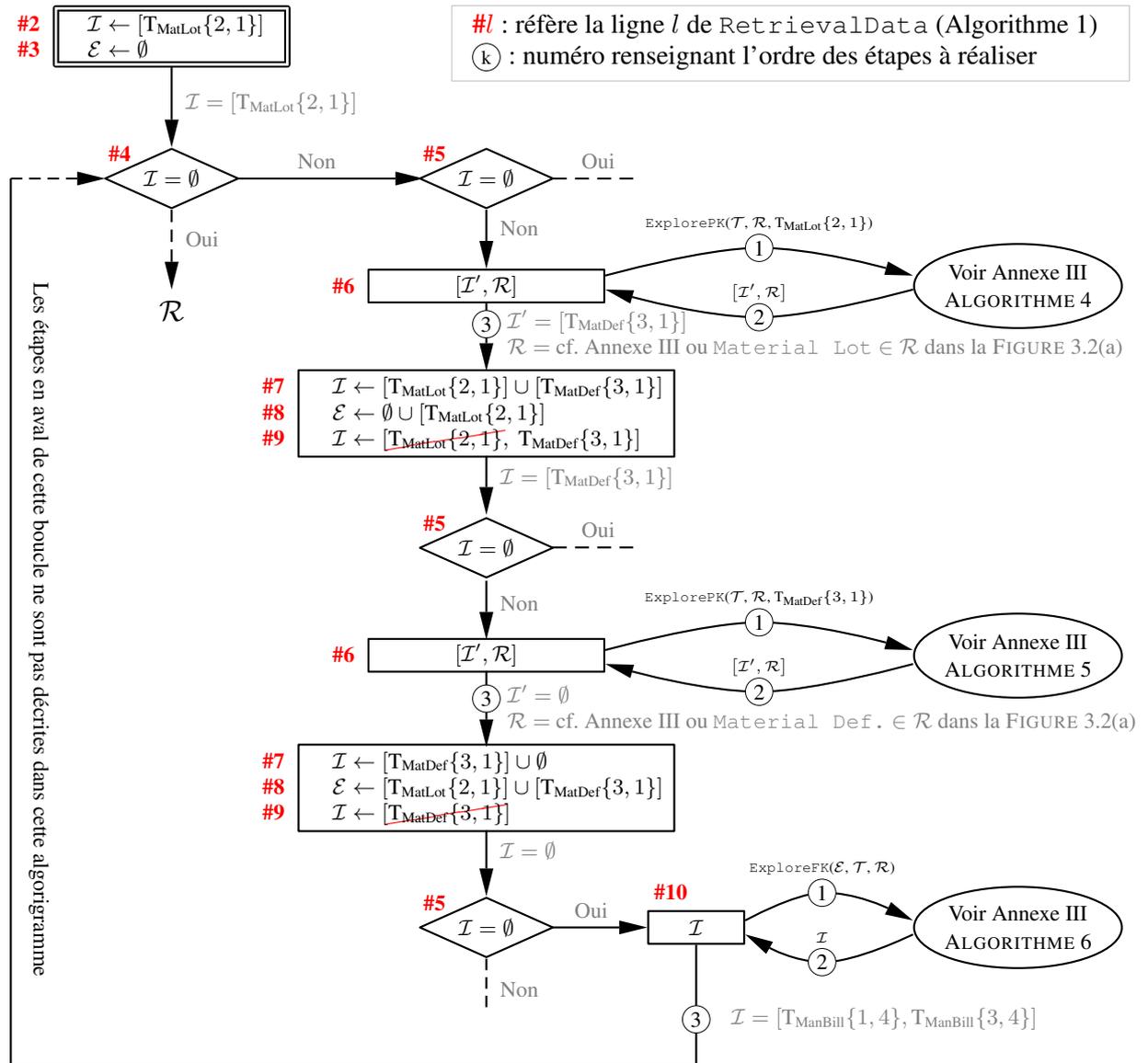
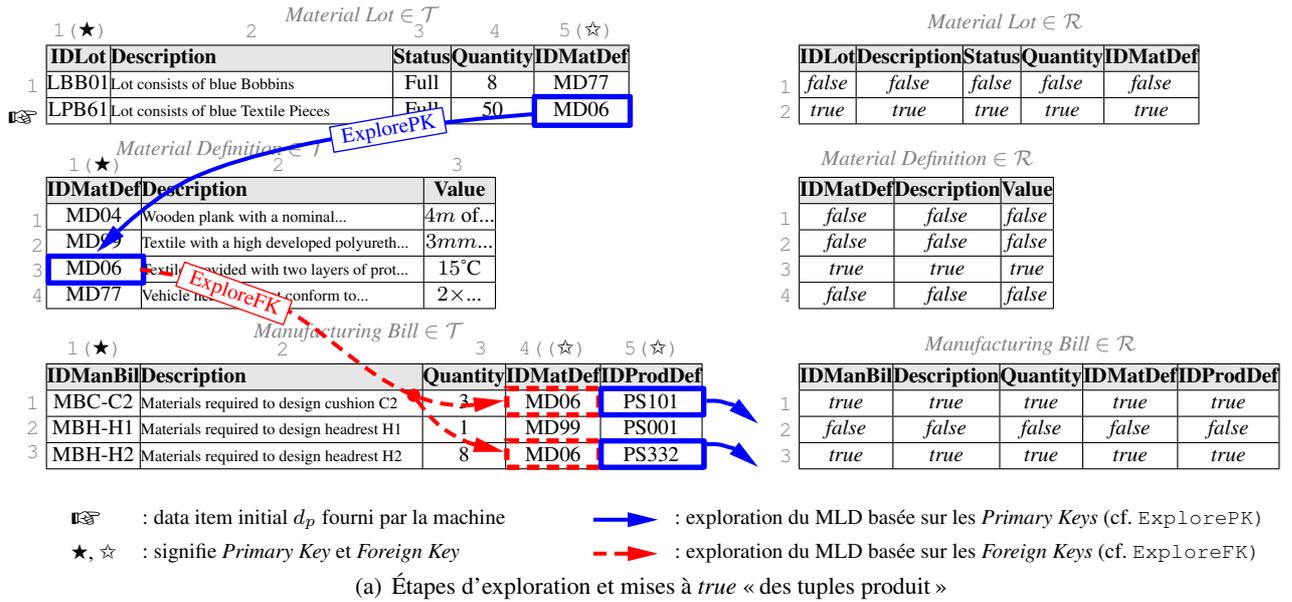


FIGURE 3.2: Identification de « tuples produit » dans les tables via la fonction RetrievalData

été identifiés comme des « tuples produit ».

Les deux prochaines sections proposent deux méthodes distinctes permettant de calculer la pertinence des « informations produit » qui ont été identifiées via les ALGORITHMES 1 à 3.

3.2 Méthode d'évaluation de la pertinence des « informations produit » par l'approche de Chan-Roddick

Comme cela a été précisé auparavant, l'objectif est désormais d'évaluer la pertinence des informations produit à être (ou ne pas être) répliquées sur le produit. Cette section détaille l'approche d'évaluation proposée par Chan et Roddick. Les auteurs proposent de calculer la pertinence de la donnée au niveau du data item (cf. terminologie dans la FIGURE 2.3). Seuls les data items appartenant à des « tuples produit » sont évalués. La FIGURE 3.3 reprend à l'identique la table `MaterialDefinition` décrite précédemment et affiche la pertinence des trois data items issus du tuple 3. Le data item localisé en colonne 1, ligne 3 de `MaterialDefinition`, noté $T_{\text{MatDef}}\{3, 1\}$, possède à cet instant du PLC une pertinence de 0.33. Plus cette pertinence est grande, plus le data item est prioritaire pour être embarqué sur le produit. En effet, compte tenu de l'application, l'ensemble des « tuples produit » ne peut pas être embarqué sur la matière communicante, la pertinence est utilisée pour sélectionner les informations à embarquer.

 Indique un « tuple produit » (lié à d_p)

Il s'agit du data item noté $T_{\text{MatDef}}\{3, 1\}$
 La pertinence de $T_{\text{MatDef}}\{3, 1\}$ est égale à 0.33

	1 (★)	2	3
	ID_Material	Description	Value
1	MD041...	Wooden plank with a nominal...	4m of...
2	MD993...	Textile with a high developed pol...	3mm...
3	MD060...	Textile which is provided with...	15°C
4	MD770...	Vehicle headrests that conform...	2 × ...

FIGURE 3.3: Table `MaterialDefinition` et pertinence des data items issus des « tuples produit »

Au cours de cette section, afin de faciliter la compréhension des modèles/algorithmes proposés et comprendre le cheminement de leur approche, un exemple applicatif est utilisé comme fil rouge et est précédé de : ⇒ Exemple :

3.2.1 Introduction

Comme nous l'avons constaté dans le Chapitre 2 et comme le soulignent (Chan and Roddick, 2003), les approches de réduction des données ignorent souvent la sémantique de l'information (elles tiennent seulement compte de paramètres liés aux performances du système : temps d'accès, nombre d'accès à l'information). En conséquence, les auteurs développent un modèle destiné à prendre en compte cette sémantique qui permet ainsi la sélection d'informations sensibles au contexte d'utilisation. Leur modèle utilise la notion de priorité. Ces priorités sont calculées au niveau du data item et correspondent à des valeurs numériques fournies ou générées sur la base d'observations et d'expériences et sont assignées

grâce à une évaluation de 8 critères. Pour chaque critère, une priorité relative ρ_x et une priorité assignée $\phi_x(l)$ sont calculées, où x représente le nom du critère et l le data item. La priorité relative indique l'importance du critère x par rapport aux autres critères. La priorité assignée mesure la valeur du data item l au regard du critère x (valeur comprise entre 0 et 1). Ces deux priorités sont combinées dans une formule unique donnée dans l'ÉQUATION 3.1, où P_l indique la pertinence de stocker le data item l sur le produit. $\text{Len}(l)$ correspond à la taille du data item l (p. ex. en octets). Nous constatons que plus les priorités ρ_x et $\phi_x(l)$ sont importantes, plus le numérateur est grand et par conséquent, plus la pertinence P_l est grande. Ces priorités sont divisées par la taille du data item l afin de pénaliser ceux ayant des tailles conséquentes. Chan et Roddick proposent de faire évoluer l'importance accordée à la taille des données grâce à une fonction logarithme népérien (cf. dénominateur de l'ÉQUATION 3.1). Le but est de fortement favoriser les data items de petites tailles.

$$P_l = \frac{\sum^x \rho_x \cdot \phi_x(l)}{\ln(\text{Len}(l) + 1)} \quad (3.1)$$

Dans leur étude, Chan et Roddick considèrent 8 critères regroupés en trois catégories :

- Critères primaires
 - *Enumeration* : permet à l'utilisateur d'énumérer les informations qu'il désire stocker sur le produit,
 - *Contextual* : permet d'intégrer la connaissance du contexte d'utilisation (peut être utilisé pour inclure des détails sur l'environnement ou l'utilisateur),
 - *Previous usage* : permet de prendre en considération l'historique d'accès aux données et ainsi favoriser les données souvent requises,
 - *Push-based* : il s'agit d'informations jugées importantes par le serveur et devant être communiquées au client (Barbará et al., 1997),
- Critères secondaires
 - *Model-based* : permet de tenir compte des distances entre données au sein du schéma relationnel. Les auteurs suggèrent de pénaliser les informations distantes des informations primaires (identifiées dans les critères primaires),
 - *Induction* : permet de définir des règles métiers qui viennent pointer des informations ayant une importance particulière relativement au contexte d'utilisation,
- Critères tertiaires
 - *Time-based inference* : permet de tenir compte de la date de l'information (ajout, mise à jour, etc.). Les auteurs soulignent que dans le contexte d'informations embarquées, il est préférable de prendre la dernière version des informations,
 - *Spatially-based inference* : permet d'intégrer des informations en fonction du contexte géographique (p. ex. en fonction de l'itinéraire du produit),

Pour la plupart de ces critères, Chan et Roddick ne présentent pas la manière de calculer les priorités assignées $\phi_x(l)$. Ils ne font qu'énoncer la fonctionnalité du critère et décrire le comportement que

serait susceptible d'avoir $\phi_x(l)$. Ceci est vrai pour les critères *Contextual*, *Previous Usage*, *Push-Based*, *Time-based inference* et *Spatially-based inference*. Leur étude porte principalement sur la définition de la fonction de sélection des informations sensibles au contexte en tenant compte d'une multitude de critères. Celle-ci a été une base de réflexion dans nos travaux pour définir des critères en fonction du contexte d'usage. En effet, de nouveaux critères peuvent venir enrichir le système d'évaluation (cf. ÉQUATION 3.1). Chan et Roddick travaillent dans le cadre des bases de données mobiles qui sont capables d'embarquer des quantités d'informations importantes comparativement à de la matière communicante. Du fait de cette spécificité, nous n'utilisons pas l'ensemble des critères qu'ils ont proposé. En effet, uniquement trois des huit critères sont déployés dans notre étude¹⁰, à savoir :

- *Enumeration* (C_e) : ce critère, utilisé dans le même but que Chan et Roddick, autorise l'utilisateur à sélectionner les informations qu'il juge importantes à stocker sur le produit. Cette énumération est réalisée par les utilisateurs en aval de l'opération d'écriture (cf. FIGURE 2.11),
- *Model-based* (C_m) : ce critère permet de tenir compte des distances entre données au sein du MLD. Nous proposons de pénaliser les données distantes de la table produit (c.-à-d. la table contenant d_p) car cette table est le centre de nos préoccupations. Plus l'information est distante de la table produit, moins elle est corrélée et par conséquent, moins elle est pertinente,
- *Contextual* (C_c) : ce critère permet de tenir compte du cycle de vie du produit et des différents types de systèmes d'information avec lesquels il peut interagir au cours de son évolution. Contrairement au critère *Enumeration* et *Model-Based*, les pondérations sur les informations sont spécifiées sur la base de recommandations d'experts ayant une connaissance globale du PLC (p. ex. sous forme de consensus d'experts et/ou sur la base de travaux de la littérature (Terzi et al., 2010)¹¹).

Il est bien entendu possible d'intégrer par la suite de nouveaux critères, que ce soit ceux suggérés par Chan et Roddick ou de nouveaux critères dédiés à de nouveaux aspects (discutés dans la conclusion générale et dans les perspectives). Les sections suivantes détaillent « quand » et « comment » les données et critères $\phi_x(l)$ et ρ_x sont calculés durant le PLC.

3.2.2 Calcul de la priorité relative ρ_x

Il s'agit dans un premier temps d'indiquer quelle est l'importance du critère x par rapport aux autres (niveau d'importance indiqué par ρ_x) à un moment donné du PLC. Nous proposons d'effectuer des *pairwise comparisons*¹², comme détaillé par la matrice D_p dans l'ÉQUATION 3.2, où M représente le nombre de critères et s_{ij} représente l'importance du critère i par rapport au critère j (renseignée par l'expert). Cela nécessite une échelle ordonnée de valeurs, notée S . Dans notre étude, nous utilisons

10. nous verrons plus loin que $\phi_x(l)$ est calculé à partir des formules proposées par Chan et Roddick pour certains des critères et sur des formules/démarches spécifiques à notre étude pour certains autres.

11. des travaux comme ceux de (Terzi et al., 2010) offrent une vision sur les phases du PLC et sur l'utilité, la sollicitation des systèmes d'informations.

12. il s'agit de comparer des éléments deux à deux (comparaison par paires). Nous conservons cependant le terme *pairwise comparison* dans la suite du document.

l'échelle définie par (Saaty, 1980) détaillée dans la TABLE 3.2 (échelle ordonnée de 1 à 9)¹³. La matrice de *pairwise comparison* D_ρ doit être remplie pour une étape du PLC qui peut inclure un ou plusieurs points d'écriture.

À partir de cette matrice, la valeur propre $\lambda_\rho(C_i)$ est calculée pour chaque critère C_i dans l'ÉQUATION 3.3 et renseigne de l'importance de ce critère. Les valeurs propres de l'ensemble des critères sont synthétisées par le vecteur propre Λ_ρ (cf. ÉQUATION 3.3).

$$D_\rho = \begin{matrix} & \begin{matrix} C_1 & C_2 & \dots & C_M \end{matrix} \\ \begin{matrix} C_1 \\ C_2 \\ \vdots \\ C_M \end{matrix} & \begin{bmatrix} s_{11} & s_{12} & \dots & s_{1M} \\ s_{21} & s_{22} & \dots & s_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ s_{M1} & s_{M2} & \dots & s_{MM} \end{bmatrix} \end{matrix} \quad s_{ij} \in S, \quad s_{ji} = \begin{cases} 1 & i = j \\ s_{ij}^{-1} & i \neq j \end{cases} \quad (3.2)$$

$$\lambda_\rho(C_i) = \frac{\sum_{j=1}^M D_\rho(i, j)}{\sum_{k=1}^M \sum_{j=1}^M D_\rho(k, j)} \quad \text{avec } i \in \{1, 2, \dots, M\} \quad (3.3)$$

$$\Lambda_\rho = [\lambda_\rho(C_1), \lambda_\rho(C_2), \dots, \lambda_\rho(C_i), \dots, \lambda_\rho(C_M)]$$

TABLE 3.2: Échelle de Saaty (Saaty, 1980)

Échelle	Définition	Description
1	Importance égale	Deux critères contribuent à l'objectif de manière égale
3	Faible importance	Un critère est légèrement favorisé par rapport à un autre
5	Forte importance	Un critère est fortement favorisé par rapport à un autre
7	Très forte importance	Un critère est très fortement favorisé par rapport à un autre
9	Importance extrême	Un critère est favorisé de façon maximale par rapport à un autre et cette affirmation est valide au plus haut point

⇒ Exemple : l'expert remplit la matrice de *pairwise comparison* présentée dans l'ÉQUATION 3.4 pour une opération d'écriture. Celle-ci indique que le critère *Enumeration* (C_e) est d'importance extrême sur le critère *Model-Based* (C_m), $s_{em} = 9$, et est légèrement plus important que le critère *Contextual* (C_c), $s_{ec} = 3$. Le détail du calcul de la valeur propre du critère C_e , notée $\lambda_\rho(C_e)$, est calculée dans l'ÉQUATION 3.5 et les valeurs propres des tous les critères sont synthétisées par Λ_ρ dans l'équation 3.6. Nous constatons que le critère *Enumeration* est le plus important avec une valeur de 0.69, puis vient le

13. cette échelle est utilisée par toutes les approches mettant en œuvre la méthode AHP.

critère *Contextual* (0.23) et enfin *Model-Based* (0.07).

$$D\rho = \begin{matrix} & C_e & C_c & C_m \\ \begin{matrix} C_e \\ C_c \\ C_m \end{matrix} & \begin{bmatrix} 1 & 3 & 9 \\ \frac{1}{3} & 1 & 3 \\ \frac{1}{9} & \frac{1}{3} & 1 \end{bmatrix} \end{matrix} \quad (3.4)$$

$$\lambda_\rho(C_e) = \frac{1 + 3 + 9}{1 + 3 + 9 + \frac{1}{3} + 1 + 3 + \frac{1}{9} + \frac{1}{3} + 1} = 0.69 \quad (3.5)$$

$$\Lambda_\rho = \begin{bmatrix} \lambda_\rho(C_e) & \lambda_\rho(C_c) & \lambda_\rho(C_m) \\ 0.69 & 0.23 & 0.07 \end{bmatrix} \quad (3.6)$$

3.2.3 Calcul de la priorité assignée $\phi_x(l)$

Cette section détaille le processus du calcul de la priorité assignée $\phi_x(l)$ (score du data item l au regard du critère x). Les critères sont détaillés dans l'ordre suivant : *Contextual*, *Enumeration* et *Model-Based*. Comme précédemment, les calculs sont appliqués sur des exemples.

Contextual

Les priorités liées au critère *Contextual* se calculent (généralement) avant que le produit ne débute son cycle de vie. Rappelons que ce critère a pour but de favoriser les informations dont l'utilisation dans les phases en aval du cycle de vie est la plus probable. En effet, les informations à inscrire sur le produit dépendent des acteurs en aval du point d'écriture considéré. Les travaux de recherche traitant des systèmes PLM ont permis de montrer que de nombreux standards d'information coexistent tout au long du PLC et ne couvrent pas les mêmes phases (Terzi et al., 2010). Ainsi, les types de systèmes d'information en aval du point d'écriture influent sur la pertinence des informations du MLD décrivant le modèle produit. Le critère *Contextual* permet de définir des groupes d'entités au sein du MLD associés aux acteurs du PLC. L'utilisateur est libre de créer des groupes d'entités propres à son application. Par exemple, on pourrait imaginer dans un contexte hospitalier, créer des groupes distincts : un 1^{er} groupe incluant les entités liées aux informations des « patients », un 2^{ème} groupe répertoriant les entités liées aux informations du « personnel hospitalier » et un 3^{ème} groupe lié aux « équipements médicaux ». Puis, en fonction du type de produit (blouse, draps, etc.) et de l'étape courante du PLC, on serait en mesure de favoriser des informations issues d'un ou de plusieurs groupes en particulier.

Une fois ces groupes d'entités définis, leur importance doit être ajustée. Cet ajustement peut être basé sur des recommandations d'experts ayant une vision globale du PLC (p. ex. un consensus d'experts) et/ou sur la base de travaux issus de la littérature (Terzi et al., 2010). L'ajustement du niveau d'importance des groupes d'entités s'effectue à l'aide de *pairwise comparisons*. Ces comparaisons sont effectuées via l'ÉQUATION 3.7, où Z représente le nombre de groupes et s_{ij} représente le niveau d'importance du

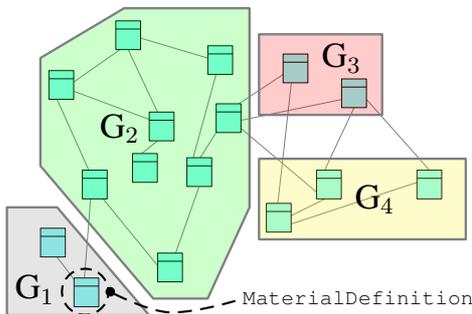
groupe d'entité i par rapport au groupe j . Cette matrice de *pairwise comparison* doit être renseignée pour chaque grande étape du PLC et peut donc impacter plusieurs opérations d'écriture. Le vecteur propre de la matrice, noté Λ_G dans l'ÉQUATION 3.8, est calculé de la même manière que précédemment (c.-à-d. sur la base de l'ÉQUATION 3.3) et donne le niveau d'importance relatif à chacun des groupes. Chaque data item appartient à une table et chaque table appartient à un et un seul groupe d'entités¹⁴. Par conséquent, pour un data item l appartenant à la table t du groupe G_i , nous obtenons comme score :

$$\phi_c(l) = \lambda_G(G_i) \quad \text{avec } l \in t \text{ et } t \in G_i$$

$$D_G = \begin{matrix} & \begin{matrix} G_1 & G_2 & \dots & G_Z \end{matrix} \\ \begin{matrix} G_1 \\ G_2 \\ \vdots \\ G_Z \end{matrix} & \begin{bmatrix} s_{11} & s_{12} & \dots & s_{1Z} \\ s_{21} & s_{22} & \dots & s_{2Z} \\ \vdots & \vdots & \ddots & \vdots \\ s_{Z1} & s_{Z2} & \dots & s_{ZZ} \end{bmatrix} \end{matrix} \quad s_{ji} \in S, \quad s_{ij} = \begin{cases} 1 & i = j \\ s_{ji}^{-1} & i \neq j \end{cases} \quad (3.7)$$

$$\Lambda_G = [\lambda_G(G_1), \lambda_G(G_2), \dots, \lambda_G(G_i), \dots, \lambda_G(G_Z)] \quad (3.8)$$

⇒ Exemple : quatre groupes d'entités sont définis dans le MLD comme illustré dans la FIGURE 3.4. L'expert soumet, pour une étape particulière du PLC, la matrice de *pairwise comparison* D_G présentée dans l'ÉQUATION 3.9. Celle-ci indique par exemple que le groupe d'entités G_1 est d'importance égale à G_2 ($s_{12} = 1$). Le vecteur propre résultant Λ_G est donné dans l'ÉQUATION 3.10 et indique que les informations provenant du groupe d'entités G_3 sont plus importantes que les informations provenant des trois autres groupes (à cet instant du PLC). Considérons la table `MaterialDefinition` comme appartenant au groupe d'entités G_1 (cf. FIGURE 3.4). De fait, les priorités assignées aux data items $T_{\text{MatDef}\{3,1\}}$, $T_{\text{MatDef}\{3,2\}}$ et $T_{\text{MatDef}\{3,3\}}$ au regard du critère *Contextual*, notées $\phi_c(T_{\text{MatDef}\{3,1\}})$, $\phi_c(T_{\text{MatDef}\{3,2\}})$ et $\phi_c(T_{\text{MatDef}\{3,3\}})$, sont toutes égales à 0.27 (importance de G_1).



$$D_G = \begin{matrix} & \begin{matrix} G_1 & G_2 & G_3 & G_4 \end{matrix} \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & 1 & \frac{1}{5} & 7 \\ 1 & 1 & \frac{1}{3} & 5 \\ 5 & 3 & 1 & 7 \\ \frac{1}{7} & \frac{1}{5} & \frac{1}{7} & 1 \end{bmatrix} \end{matrix} \quad (3.9)$$

FIGURE 3.4: Groupes d'entités

$$\Lambda_G = \begin{matrix} & \begin{matrix} \lambda_G(G_1) & \lambda_G(G_2) & \lambda_G(G_3) & \lambda_G(G_4) \end{matrix} \\ \begin{bmatrix} 0.27 & 0.21 & 0.47 & 0.05 \end{bmatrix} \end{matrix} \quad (3.10)$$

14. il sera nécessaire par la suite de mettre en œuvre une approche permettant de gérer le fait qu'une entité appartienne à plusieurs groupes (p. ex. calcul de l'importance moyenne, de l'importance maximale, etc.).

Enumeration

L'énumération des informations que l'utilisateur désire stocker sur la matière communicante est effectuée pour chaque point d'écriture (par les utilisateurs ou la compagnie en aval du point d'écriture). Dans notre étude, l'expert énumère les attributs des tables qu'il souhaite stocker sur le produit. La priorité d'un attribut v d'une table t , notée $s(v, t)$, prend la valeur 1 s'il est énuméré, 0 sinon, comme détaillé dans l'ÉQUATION 3.11. Tout data item localisé dans la colonne/attribut v de la table t a une priorité assignée au regard du critère *Enumeration*, notée $\phi_e(l)$, équivalente à $s(v, t)$.

$$s(v, t) = \begin{cases} 1 & \text{énuméré} \\ 0 & \text{non énuméré} \end{cases} \quad \text{avec } v \text{ un attribut d'une table } t \quad (3.11)$$

⇨ Exemple : l'utilisateur énumère l'attribut `ID_Material` et `value` de la table `MaterialDefinition` (cf. FIGURE 3.1), ce qui donne les priorités indiquées dans l'ÉQUATION 3.12. Par conséquent, les priorités assignées des data items $T_{\text{MatDef}\{3, 1\}}$, $T_{\text{MatDef}\{3, 2\}}$ et $T_{\text{MatDef}\{3, 3\}}$, notées $\phi_e(T_{\text{MatDef}\{3, 1\}})$, $\phi_e(T_{\text{MatDef}\{3, 2\}})$ et $\phi_e(T_{\text{MatDef}\{3, 3\}})$ sont respectivement égales à 1, 0 et 1.

$$\begin{aligned} s(\text{ID_Material}, \text{MaterialDefinition}) &= 1 \\ s(\text{Description}, \text{MaterialDefinition}) &= 0 \\ s(\text{value}, \text{MaterialDefinition}) &= 1 \end{aligned} \quad (3.12)$$

Model-Based

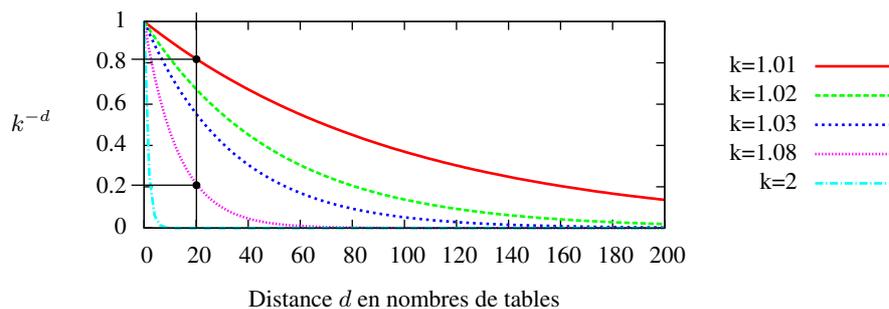
Ce critère vise à favoriser les données proches de la « table produit » (celle contenant d_p) car comme le défendent Chan et Roddick, plus la distance entre des tables est petite, plus les données sont corrélées¹⁵. Ce critère se base sur les relations impliquées dans le MLD. Il est nécessaire dans notre étude de calculer les distances entre la table produit (noté T_p) et toutes autres tables T_y . La distance correspond au plus court chemin pour atteindre T_y à partir de T_p (c.-à-d. le nombre de tables les séparant). Afin de réduire la priorité assignée au data item, lorsque sa distance augmente pour atteindre la table produit, Chan et Roddick proposent la formule donnée dans l'ÉQUATION 3.13 avec k une constante et d la distance. Le coefficient k peut être ajusté par l'expert.

$$\phi_m(l) = k^{-d} \quad \text{avec } k \in [1; +\infty] \quad d \in \mathbb{N} \quad (3.13)$$

Comme le montre la FIGURE 3.5, plus le coefficient k est petit, moins les informations distantes sont pénalisées, et réciproquement. Une même information distante de 20 tables de la table produit, avec $k = 1.02$ et $k = 1.08$, donne une priorité $\phi_m = 0.65$ et $\phi_m = 0.21$ respectivement (cf. FIGURE 3.5). Il est donc nécessaire d'étudier le MLD afin de fixer le coefficient k . En effet, si la distance maximale à travers le MLD est égale à 10 ou 200, l'ajustement du coefficient k doit être différent¹⁶.

15. cette hypothèse de travail semble logique mais peut toutefois porter à discussion dans certain contexte.

16. nous verrons par la suite que l'ajustement de ce coefficient n'est pas si trivial, cela nécessite une certaine connaissance et expérience du modèle et de l'application. Des systèmes d'apprentissage peuvent être imaginés pour un tel ajustement.

FIGURE 3.5: Ajustement du coefficient k après étude du MLD

⇒ Exemple : considérons `MaterialLot` comme étant la table produit. L'utilisateur décide de fixer k à 1.08. Par conséquent, les priorités assignées de $T_{\text{MatDef}}\{3, 1\}$, $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 3\}$ au regard du critère *Model-Based*, notées $\phi_m(T_{\text{MatDef}}\{3, 1\})$, $\phi_m(T_{\text{MatDef}}\{3, 2\})$ et $\phi_m(T_{\text{MatDef}}\{3, 3\})$ sont toutes égales à 1.08^{-1} car ces trois data items sont contenus dans la table `MaterialDefinition`, laquelle est distante d'une table de la table produit (cf. FIGURE 3.1(a)).

Calcul de la pertinence des data items

Au final, en utilisant les précédents résultats couplés à l'ÉQUATION 3.1, il est possible de calculer la pertinence des data items $T_{\text{MatDef}}\{3, 1\}$, $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 3\}$ comme illustré dans la FIGURE 3.6¹⁷. Nous constatons à la vue de ces résultats que $T_{\text{MatDef}}\{3, 3\}$ est le plus pertinent parmi les trois data items, avec une valeur de 0.51. Ainsi, $T_{\text{MatDef}}\{3, 3\}$ est privilégié par rapport à $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 1\}$ pour être stocké sur le produit. Ces calculs sont effectués pour l'ensemble des data items issu des « tuples produit ».

3.2.4 Synthèse du modèle d'évaluation proposé par Chan et Roddick

Comme Chan et Roddick le soulignent, très peu de travaux de la littérature tiennent compte de la sémantique de la donnée lors de la réduction de l'information. En ce sens, leur méthode est particulièrement intéressante puisqu'elle permet de sélectionner des informations sensibles au contexte d'usage. De plus, leur méthode permet de prendre en compte une multitude de critères qui peuvent varier en fonction de l'application. Dans nos futurs travaux, nous envisageons d'intégrer de nouveaux critères tels que l'historique des requêtes¹⁸ (favoriser des données souvent sollicitées) ou l'impact sur les performances du système de communication. Néanmoins, certaines limites de l'approche de Chan et Roddick subsistent, à savoir :

- la taille des data items n'est pas considérée comme un critère. Par conséquent, l'utilisateur ne peut pas agir sur ses préférences concernant la taille des data items à embarquer sur le produit.

17. le dénominateur du calcul de $P_{T_{\text{MatDef}}\{3,1\}}$ est égal à $\ln(11+1)$ car le contenu de $T_{\text{MatDef}}\{3, 1\}$ est de 11 caractères ASCII et 1 caractère est codé sur 1 octet.

18. ce critère fait déjà partie des critères énoncés par Chan et Roddick. Cependant, aucun renseignement sur la manière de calculer la priorité assignée n'est fournie.

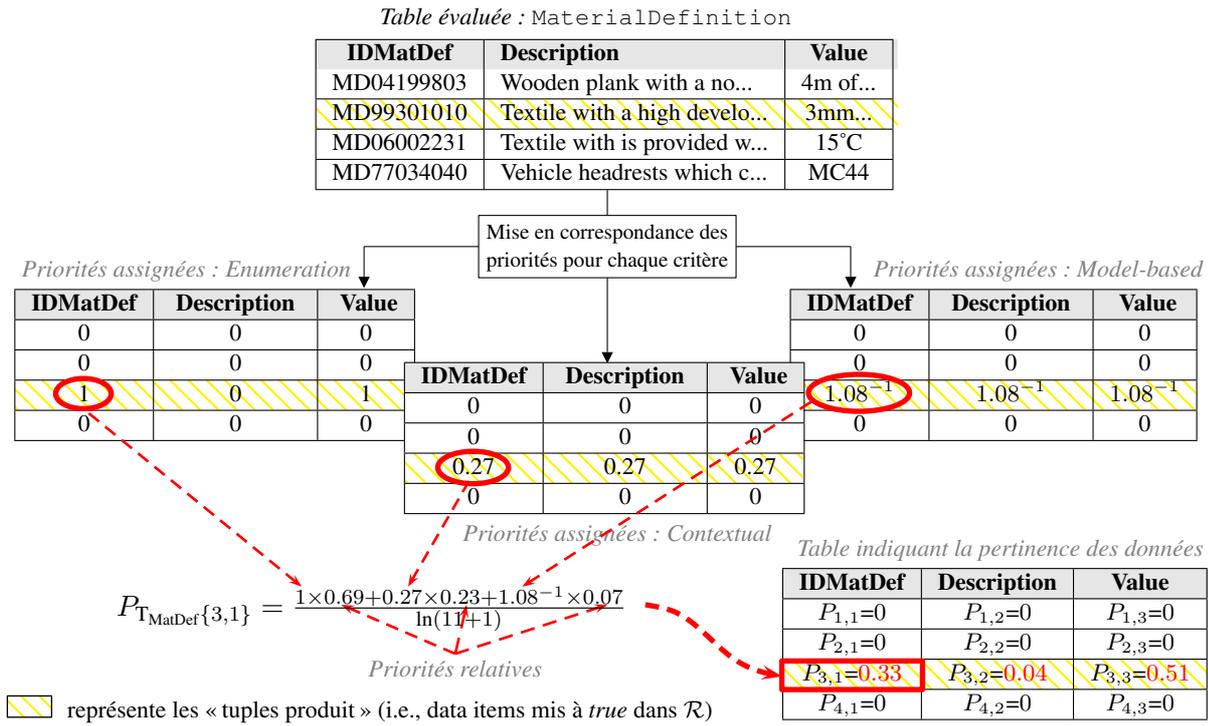


FIGURE 3.6: Calcul de la pertinence des data items $T_{MatDef}\{3, 1\}$, $T_{MatDef}\{3, 2\}$ et $T_{MatDef}\{3, 3\}$

Chan et Roddick ont certainement fait ce choix au vu de leur contexte qui emploie des bases de données mobiles, lesquelles disposent de capacités de stockages beaucoup plus importantes qu’une matière communicante. Aussi, dans notre contexte, il est intéressant de pouvoir agir sur ce critère en fonction des circonstances (type de matière, type de technologie, etc.) pour favoriser ou non le stockage de données conséquentes,

- leur approche ne permet pas de prendre en considération plusieurs avis d’experts. Or, dans notre contexte, plusieurs experts/utilisateurs peuvent se prononcer sur une même évaluation d’un objet (critère, alternative, etc.), durant une même opération d’écriture. Prenons l’exemple d’un expert désirant une information particulière (p. ex. énumération d’un attribut) et d’un second expert, qui lui, ne la désire pas : Décide-t-on d’embarquer l’information si un parmi les n experts la désire ? Décide-t-on d’embarquer l’information si tous les experts conviennent d’un même accord ? Décide-t-on de prendre une décision intermédiaire en utilisant des poids moyens ? L’approche de Chan et Roddick n’inclut pas de solution,
- leur approche ne tient pas compte de l’incertitude ou de l’imprécision sur les décisions et les choix faits par les utilisateurs/experts.

Ainsi, la proposition de résolution du problème de décision multi-critères (MCDM - *MutliCriteria Decision Making*) formulée par (Chan and Roddick, 2003) est incomplète au vu de notre contexte applicatif. Par conséquent, il est nécessaire de mettre en place une approches pour pallier ces limites. Dans la littérature, il existe d’autres méthodes classiques de MCDM telles que l’approche WSM (*Weighted-Sum*

Model), WPM (*Weighted-Product Model*), TOPSIS (*Technique Ordered Preference by Similarity to the Ideal Solution*) ou encore AHP (*Analytic Hierarchy Process*) (Triantaphyllou and Lin, 1996).

AHP est l'une des méthodes les plus répandues dans la littérature pour traiter ce type de problème (Vaidya and Kumar, 2006). Elle permet de prendre en compte des critères aussi bien qualitatif que quantitatif. La fonction d'agrégation proposée par Chan et Roddick se rapproche d'ailleurs beaucoup d'une formulation de type AHP. La mise en place d'AHP reste simple et autorise d'intégrer un critère relatif à la taille des données. La section 3.3 détaille les avantages de l'approche AHP ainsi que ses limites au vu de notre contexte d'étude, à savoir qu'elle ne permet pas de prendre en compte plusieurs points de vue sur une même évaluation (problème similaire à l'approche de Chan et Roddick). En effet, comme cela a été précisé précédemment, plusieurs experts/utilisateurs peuvent se prononcer sur une même évaluation d'un objet (critère, alternative, etc.) et toutes ces évaluations sont légitimes (c.-à-d. à prendre en compte). Les résultats issus du cas d'étude (présenté dans le Chapitre 5) montreront que lorsque l'approche de Chan et Roddick est confrontée à un groupe d'experts¹⁹ et qu'il y a des avis « isolés » (p. ex. un expert voulant stocker sur le produit une information que dix autres experts ne souhaitent pas), celle-ci ne tient quasi pas compte de cet avis.

Par conséquent, pour pallier ce problème, nous utilisons conjointement à l'approche AHP la théorie des sous-ensembles flous. La théorie des ensembles flous a été introduite par (Zadeh, 1965) afin de gérer des informations imparfaites due à une imprécision ou une incertitude. La théorie des sous-ensembles flous va permettre de représenter l'information en tenant compte de sa complexité et des divers points de vue, tandis que la méthode AHP va permettre de résoudre le problème de MCDM. Un des avantages d'utiliser la théorie des ensembles flous dans notre contexte est relatif à la diversité des critères définis dans la structure AHP. En effet, de nombreux critères peuvent être définis dans AHP (p. ex. Chan et Roddick proposent 8 critères, cf. section 3.2.1) et dans chacun d'eux les calculs sont basés aussi bien sur des observations, des expériences, des spécifications ou autres. Ainsi, en fonction du critère et de ses caractéristiques, la manipulation d'ensembles flous (c.-à-d., d'intervalles de valeurs) peut s'avérer plus appropriée que la manipulation de nombres classiques (c.-à-d., de valeurs uniques) et autorise la prise en compte d'incertitude liée aux choix/évaluations des experts. L'incertitude autorise une meilleure modélisation de l'information disponible, fiabilisant ainsi le processus de MCDM et donc la prise de décision (Bouchon-Meunier, 2007).

3.3 Méthode d'évaluation de la pertinence des « informations produit » par l'approche AHP-Floue

Cette section détaille notre proposition d'évaluation qui se base sur le principe de l'AHP-Floue. Tout comme dans l'approche de Chan-Roddick, la pertinence des informations est calculée au niveau du data item et plus cette pertinence est grande, plus le data item est pertinent à être embarqué sur le produit.

19. lors du cas d'étude, nous adaptons l'approche de Chan et Roddick pour prendre en compte les avis d'un groupe d'experts en calculant l'avis moyen.

La section 3.3.1 retrace brièvement l'histoire de AHP-Floue, puis offre un aperçu de l'approche mise en place dans notre étude (composée de 5 points²⁰ majeurs). Ces 5 points sont détaillés respectivement dans les sections 3.3.2 à 3.3.6. Une fois encore, dans le but de faciliter la compréhension des modèles, un exemple applicatif est utilisé comme fil rouge²¹ et est précédé de : ⇨ Exemple :

3.3.1 Introduction

Analytical Hierarchical Process

L'approche AHP est l'une des méthodes les plus répandues dans la littérature pour traiter les problèmes MCDM. Proposée initialement par (Saaty, 1980), elle est depuis utilisée dans des secteurs très différents tels que la politique (Saaty, 1983), l'industrie (Cheng, 1997; Labib et al., 1998), le secteur hospitalier (Lee and Kwak, 1999), le monde éducatif (Tadisina et al., 1991), etc. Cette approche possède l'avantage d'organiser les aspects critiques liés au problème de décision en une structure hiérarchique similaire à un arbre, facilitant ainsi l'ingénierie du processus de décision. De plus, cette approche met en œuvre la méthode de *pairwise comparisons*, qui est souvent préférée par les instances décisionnelles. La *pairwise comparison* permet d'obtenir les poids des facteurs (critères, sous-critères et alternatives) en les comparant entre eux et non pas en les spécifiant de manière directe²².

De nombreuses critiques ont néanmoins été formulées à l'encontre de AHP, soulignant des problèmes d'inversement de rangs (Belton and Gear, 1983; Schoner and Wedley, 1989; Choo et al., 1999), des problèmes d'échelle (Harker and Vargas, 1987), des problèmes d'agrégation (Lootsma, 1988). Elle est même jugée parfois inefficace (Kwong and Bai, 2002). D'autres études comparatives ont toutefois montré que AHP restait la méthode la plus flexible (Olson et al., 1995) car elle peut être utilisée conjointement à d'autres techniques comme la programmation linéaire, le QFD, la logique floue, TOPSIS (Vaidya and Kumar, 2006). Cela permet à l'utilisateur de tirer profit des méthodes combinées à AHP et donc, d'atteindre plus facilement le but recherché. De plus, la structure de AHP est avantageuse du fait de sa grande simplicité (Zheng et al., 2012). L'approche AHP se compose classiquement de 4 étapes (Saaty, 2008) :

- structuration du problème en une hiérarchie à plusieurs niveaux, comprenant l'objectif à atteindre, les critères, les sous-critères et les alternatives, comme illustré sur la FIGURE 3.7. La structure peut comprendre plus ou moins 4 niveaux selon que l'on définit ou non plusieurs niveaux de sous-critères,
- comparaison de chaque élément à chaque niveau de la hiérarchie sur la base de matrices de *pairwise comparison*,
- vérification de la cohérence des jugements. La cohérence parfaite n'est jamais atteinte car le preneur de décision soumet souvent une préférence subjective biaisée et incohérente (Lin, 2009),

20. le terme « point » est préféré à « étape » pour ne pas perturber le lecteur avec l'usage du terme « étape » employé dans le processus de dissémination des informations.

21. cet exemple considère les mêmes data items que dans l'approche Chan-Roddick.

22. l'évaluation directe d'un objet est un processus plus difficile à mettre en œuvre que l'évaluation par comparaison.

- calcul du poids de chaque facteur composant la hiérarchie et classement des alternatives (Niveau 4 dans FIGURE 3.7).

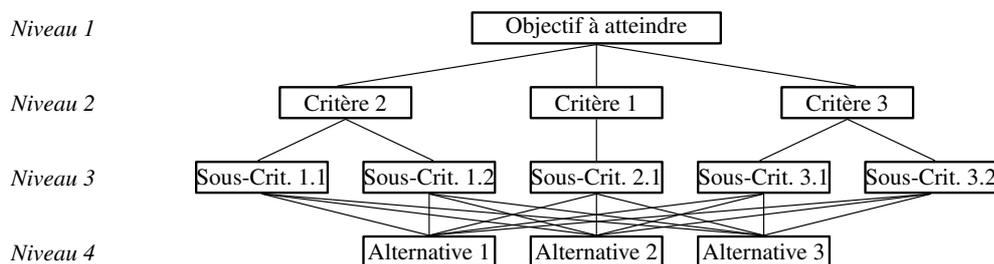


FIGURE 3.7: Illustration d'une structure AHP classique

Les données de notre étude présentent une certaine « complexité » relative, par exemple, à la multiplicité des points de vue sur le produit et l'évaluation faite par plusieurs experts. La méthode AHP classique ne permet pas de prendre en compte de telles informations. C'est pourquoi, nous proposons d'utiliser la théorie des sous-ensembles flous conjointement à la méthode AHP. La prochaine section rappelle quelques éléments fondamentaux de cette théorie utiles pour la compréhension de notre proposition.

Théorie des ensembles flous

(Zadeh, 1965) a proposé en 1965 le concept d'ensemble flou. Son objectif était de se rapprocher de la façon dont l'Homme appréhende, raisonne et communique avec son environnement. Cette façon de procéder est graduelle plutôt que tranchée et nette. Contrairement aux ensembles classiques, pour lesquels un élément appartient complètement ou n'appartient pas à l'ensemble, un sous-ensemble flou modélise une appartenance graduelle. On assigne donc à chaque élément du référentiel un degré d'appartenance au sous-ensemble flou. L'ensemble de ces degrés d'appartenance, associé à des éléments, forme la fonction d'appartenance du sous-ensemble flou. Ainsi, comme pour un ensemble classique, lorsque le degré d'appartenance $\mu_{\tilde{A}}(y)$ est égal à 1 alors y appartient complètement à \tilde{A} ; lorsque ce degré est égal à 0 alors y n'appartient pas à \tilde{A} . Enfin, le degré $\mu_{\tilde{A}}(y)$ peut être compris entre 0 et 1, indiquant que y appartient plus ou moins à \tilde{A} . Quelques définitions et propriétés de la théorie des ensembles flous, utilisées dans l'approche proposée, sont présentées ci-après :

DÉFINITION 1

(Bouchon-Meunier, 2007) : d'un point de vue formel, un sous-ensemble flou est défini par sa fonction d'appartenance :

$$\begin{aligned} \mu_{\tilde{A}} : \mathcal{Y} &\rightarrow [0, 1] \\ y &\mapsto \mu_{\tilde{A}}(y) \end{aligned} \quad (3.14)$$

Pour le sous-ensemble flou \tilde{A} , les degrés d'appartenance $\mu_{\tilde{A}}(y)$ d'un élément y prennent leurs valeurs dans l'intervalle $[0, 1]$ et \mathcal{Y} est appelé le référentiel.

DÉFINITION 2

(Bouchon-Meunier, 2007) : on appelle « noyau » d'un sous-ensemble flou $\tilde{\mathcal{A}}$, l'ensemble des éléments dont le degré d'appartenance est égal à 1 :

$$\text{noy}(\tilde{\mathcal{A}}) = \{y \in \mathcal{Y} \mid \mu_{\mathcal{A}}(y) = 1\} \quad (3.15)$$

(Bouchon-Meunier, 2007) : on appelle « support » d'un sous-ensemble flou $\tilde{\mathcal{A}}$, l'ensemble des éléments dont le degré d'appartenance est différent de 0 (Bouchon-Meunier, 2007) :

$$\text{supp}(\tilde{\mathcal{A}}) = \{y \in \mathcal{Y} \mid \mu_{\mathcal{A}}(y) > 0\} \quad (3.16)$$

DÉFINITION 3

(Bouchon-Meunier, 2007) : une quantité floue, $\tilde{\mathcal{Q}}$, est un ensemble flou normalisé sur le domaine \mathbb{R} (c.-à-d., $\text{noy}(\tilde{\mathcal{Q}}) \neq \emptyset$).

Un intervalle flou $\tilde{\mathcal{I}}$ est une quantité floue convexe.

Un nombre flou $\tilde{\mathcal{M}}$ est un intervalle flou de support borné qui possède un noyau réduit à une valeur m unique : $\text{noy}(\tilde{\mathcal{A}}) = \{m\}$.

Dans notre étude, nous utilisons des intervalles flous trapézoïdaux²³. En effet, ce sont les formes les plus répandues des intervalles flous car elles peuvent être traitées numériquement et interprétées intuitivement de manière simple (Chou et al., 2008). Un intervalle flou $\tilde{\mathcal{A}}$ trapézoïdal est illustré dans la FIGURE 3.8(a) et sa fonction d'appartenance est définie par (Kaufmann and Gupta, 1991) :

$$\mu_{\tilde{\mathcal{A}}}(y) = \begin{cases} \frac{(y-a)}{(b-a)} & a \leq y \leq b \\ 1 & b \leq y \leq c \\ \frac{(y-d)}{(c-d)} & c \leq y \leq d \\ 0 & \text{sinon} \end{cases} \quad (3.17)$$

Ainsi, $\text{noy}(\tilde{\mathcal{A}}) = [b, c]$ et $\text{supp}(\tilde{\mathcal{A}}) = [a, d]$ dans la FIGURE 3.8(a). Suivant les valeurs $[a, b, c, d]$, nous obtenons des cas particuliers de fonctions d'appartenance tels que des intervalles ordinaires (FIGURE 3.8(b)), des nombres ordinaires appelés également « singleton » (FIGURE 3.8(c)) et des nombres flous triangulaires (FIGURE 3.8(d)).

Comme pour les nombres classiques, il est possible de définir des opérations afin de manipuler les intervalles flous. Les calculs sur les intervalles flous peuvent être réalisés suivant deux méthodes. La première consiste à utiliser le principe d'extension défini par (Zadeh, 1965) et qui permet d'étendre à l'utilisation d'intervalles flous toutes les fonctions manipulant des nombres ordinaires. Lorsque plusieurs calculs s'enchaînent, notamment des opérations telles que la multiplication et la division, l'utilisation du

23. il ne s'agit pas d'un « nombre flou » à proprement dit car un trapèze ne dispose pas d'une valeur modale unique (sauf cas particuliers détaillés ci-après).

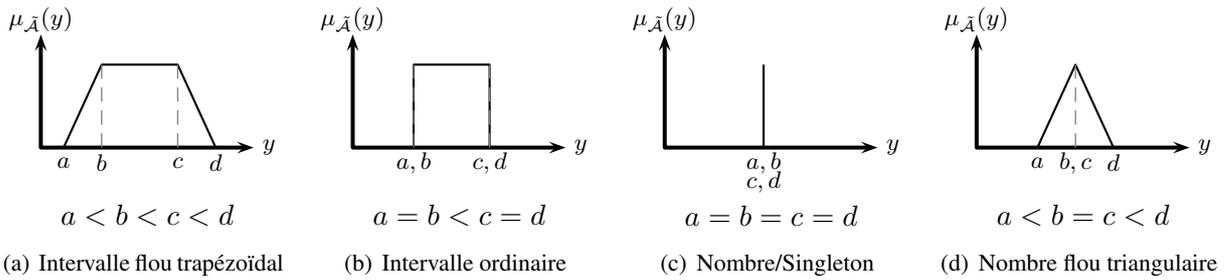


FIGURE 3.8: Intervalle flou trapézoïdal et formes particulières associées

principe d'extension conduit à ne plus pouvoir obtenir de manière simple une forme analytique des résultats, et de fait à ne plus pouvoir conserver une forme complète de la fonction d'appartenance. Aussi de nombreux auteurs utilisent une forme approchée en ne calculant que les bornes du support et du noyau dont l'avantage est de conduire à des calculs très simples.

Exemple 1 (Kaufmann and Gupta, 1991) : considérons les intervalles flous trapézoïdaux $\tilde{A} = (a, b, c, d)$ de \mathbb{R} et $\tilde{B} = (e, f, g, h)$ de \mathbb{R} . Chaque intervalle flou dispose de deux sous-ensembles non flous (intervalles de valeurs) relatifs au « support » et au « noyau » de leur fonction d'appartenance. Les deux ensembles non flous relatifs à \tilde{A} sont respectivement notés $\mathcal{A}_0 = [a, d]$ et $\mathcal{A}_1 = [b, c]$ (l'indice 0 en référence au support et l'indice 1 en référence au noyau) et ceux de \tilde{B} respectivement notés $\mathcal{B}_0 = [e, h]$ et $\mathcal{B}_1 = [f, g]$. Les opérations de base utilisées dans notre approche sont :

$$\begin{aligned} \tilde{A} + \tilde{B} &= \begin{cases} \mathcal{A}_0 + \mathcal{B}_0 = [a, d] + [e, h] = [a + e, d + h] & a \geq 0, e \geq 0 \\ \mathcal{A}_1 + \mathcal{B}_1 = [b, c] + [f, g] = [b + f, c + g] \end{cases} \\ &= [a + e, b + f, c + g, d + h] \\ \tilde{A} \times \tilde{B} &= \begin{cases} \mathcal{A}_0 \times \mathcal{B}_0 = [a, d] \times [e, h] = [ae, dh] & a \geq 0, e \geq 0 \\ \mathcal{A}_1 \times \mathcal{B}_1 = [b, c] \times [f, g] = [bf, cg] \end{cases} \\ &= [ae, bf, cg, dh] \\ \frac{\tilde{A}}{\tilde{B}} &= \begin{cases} \frac{\mathcal{A}_0}{\mathcal{B}_0} = \frac{[a, d]}{[e, h]} = \left[\frac{a}{h}, \frac{d}{e} \right] & a \geq 0, e \geq 0 \\ \frac{\mathcal{A}_1}{\mathcal{B}_1} = \frac{[b, c]}{[f, g]} = \left[\frac{b}{g}, \frac{c}{f} \right] \end{cases} \\ &= \left[\frac{a}{h}, \frac{b}{g}, \frac{c}{f}, \frac{d}{e} \right] \end{aligned}$$

Cependant, lors de multiplications et de divisions d'intervalles flous, la forme des fonctions d'appartenance des résultats n'est plus linéaire. La FIGURE 3.9(a) donne deux sous-ensembles flous \tilde{A} et \tilde{B} . Le résultat de la multiplication de ces deux sous-ensembles flous est fourni dans la FIGURE 3.9(b). La forme exacte correspond à la forme en pointillé et la forme approchée par calcul sur les limites correspond à la forme en trait plein. Nous constatons qu'une partie de l'information est perdue.

Une deuxième approche, consiste à utiliser une autre représentation des intervalles flous sous forme d'un ensemble d'intervalles classiques (appelés α -coupes) et d'appliquer les calculs sur les intervalles à cette représentation. L'utilisation de la représentation par α -coupes permet de conserver une simplicité

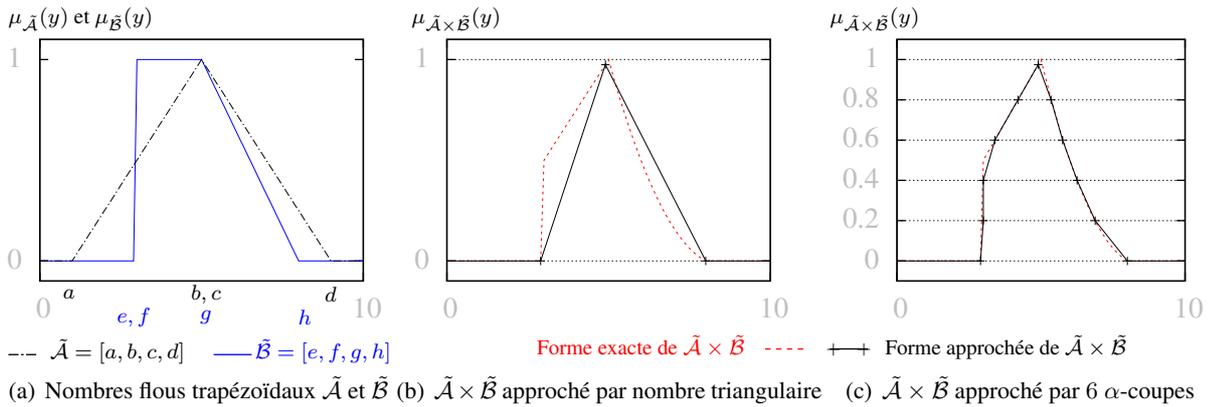


FIGURE 3.9: Multiplications de deux intervalles flous suivant deux méthodes d'approximations

de calcul puisqu'il s'agit toujours de travailler avec des intervalles et a comme avantage de conserver la forme des fonctions d'appartenance. En revanche, elle requiert d'échantillonner les intervalles flous et donc de répéter les calculs sur les intervalles autant de fois que l'échantillonnage l'exige (Fu et al., 2008).

DÉFINITION 4

On appelle α -coupe, ou sous-ensemble de niveau α (avec $\alpha \in]0, 1]$), d'un ensemble flou $\tilde{\mathcal{A}}$ de \mathbb{R} , le sous-ensemble non flou \mathcal{A}^α de \mathbb{R} tel que :

$$\mathcal{A}^\alpha = \{y \in \mathbb{R} \mid \mu_{\tilde{\mathcal{A}}}(y) \geq \alpha\} \quad \alpha \in]0, 1] \tag{3.18}$$

par définition, $\mathcal{A}^0 = \text{supp}(\tilde{\mathcal{A}})$

Un intervalle flou peut se représenter soit par sa fonction d'appartenance, soit par l'ensemble des α -coupes de niveau compris dans $]0, 1]$. Ces deux représentations sont équivalentes et l'on passe de l'une à l'autre par l'ÉQUATION 3.18 et 3.19 (Zadeh, 1971) :

$$\mu_{\tilde{\mathcal{A}}}(y) = \sup^{\alpha \in]0, 1]} (\alpha \times \mathcal{A}^\alpha(y)) \tag{3.19}$$

La FIGURE 3.9(c) illustre une fois encore la multiplication des deux intervalles flous trapézoïdaux $\tilde{\mathcal{A}}$ et $\tilde{\mathcal{B}}$ en considérant les 6 α -coupes de niveau $\{0, 0.2, 0.4, \dots, 1\}$. Nous remarquons que la forme globale de la fonction d'appartenance résultant de la multiplication est mieux conservée²⁴.

Nous avons choisi d'utiliser l'approche par α -coupe car nous pensons que pour le classement final des alternatives, la forme des intervalles flous issus des calculs peut être prise en compte. L'échantillonnage des fonctions d'appartenance utilise 11 α -coupes de niveau $\{0, 0.1, 0.2, \dots, 0.9, 1\}$.

Association de la méthode AHP et de la théorie des ensembles flous

Les plus anciens travaux sur l'association AHP-Floue sont apparus dans (van Laarhoven and Pe-drycz, 1983), lesquels comparent les facteurs décrits par des nombres flous triangulaires. (Buckley, 1985)

24. plus le nombre d' α -coupes sera important, plus la forme se rapprochera de la forme théorique.

a ensuite développé une approche permettant de déterminer les priorités à partir de comparaisons sur la base d'intervalles flous trapézoïdaux. (Chang, 1996) a introduit une nouvelle approche pour gérer la méthode AHP-Floue avec l'utilisation d'une échelle pour la *pairwise comparison* mettant en œuvre des nombres flous triangulaires, conjointement à une méthode communément appelée FEHP (*Fuzzy extended AHP*), laquelle permet de synthétiser les valeurs des matrices de *pairwise comparison*.

Comparativement à d'autres méthodes de MCDM, l'utilisation de la méthode AHP-Floue offre de nombreux avantages. Premièrement, les autres méthodes connaissent certaines difficultés à capturer les jugements imprécis et incertains des experts (Das, 2011). Deuxièmement, certains facteurs sont difficilement quantifiables à l'aide de nombres et par conséquent, mesurer ces facteurs de manière qualitative est plus simple pour le décideur (Zheng et al., 2012). Tout comme l'approche AHP classique, l'approche AHP-floue a largement été mise à contribution dans la littérature dans tous les secteurs : politique (Kuo et al., 1999), social (Yeh et al., 2000), hospitalier (Büyüközkan et al., 2011) environnemental (Anagnostopoulos and Petalas, 2011; Yang et al., 2011), industriel (Kreng and Wu, 2007; Noci and Toletti, 2000; Monitto et al., 2002), naval (Cheng, 1997; Celik et al., 2009) ou encore du transport (Vildan Özkir and Demirel, 2012). Cependant, une très grande majorité de ces travaux se limite à appliquer les approches AHP-Floue classiques (van Laarhoven and Pedrycz, 1983; Buckley, 1985; Chang, 1996) sur des cas d'études spécifiques. Dans ces approches, l'utilisation des sous-ensembles flous permet principalement de modéliser l'incertitude (Durbach and Stewart, 2012).

L'utilisation d'AHP-Floue requiert une approche rigoureuse pour obtenir les scores à partir de *pairwise comparisons*. (Wang and Chin, 2011) ont recensé deux catégories d'approche :

- celle consistant à obtenir des intervalles flous à partir d'une matrice de *pairwise comparison* floue,
- celle consistant à obtenir des valeurs exactes à partir d'une matrice de *pairwise comparison* floue.

Notre étude s'appuie sur la seconde catégorie qui inclut généralement l'approche FEHP de (Chang, 1996) et la programmation de préférence floue (Mikhailov, 2003). Comme les scores flous sont plus difficiles à calculer que les scores exacts, l'approche FEHP proposée par (Chang, 1996) est majoritairement utilisée, par exemple par (Chang, 1996; Tolga et al., 2005; Xia and Xu, 2011; Vildan Özkir and Demirel, 2012; Leung and Cao, 2000; Bozbura and Beskese, 2007; Azadeh et al., 2011; Cheng, 1997; Yeh et al., 2000; Büyüközkan and Çifçi, 2012; Cho et al., 2012; Javanbarg et al., 2012; Yazdani-Chamzini and Yakhchali, 2012). Cette approche utilise les nombres flous triangulaires comme échelle dans les matrices de *pairwise comparisons* dans le but de dériver les poids des critères et les valeurs appropriées de l'ensemble des alternatives au vu de chaque critère. Les calculs sont réalisés en utilisant uniquement les limites du noyau et du support (problème soulevé dans la FIGURE 3.9(b)). De plus, les nombres flous triangulaires peuvent s'avérer inappropriés suivant ce que veut représenter l'utilisateur/le décideur (inappropriés au vu de l'information à modéliser). C'est pourquoi certains auteurs utilisent ou développent des approches permettant d'effectuer des *pairwise comparisons* en utilisant des intervalles flous trapézoïdaux (Buckley, 1985; Weck et al., 1997; Gu and Zhu, 2006; Fu et al., 2008; Gao et al., 2012; Kayaa and Kahramanb, 2011; Aydina and Kahramanb, 2012; Zheng et al., 2012).

Nous choisissons d'utiliser AHP-Floue, d'une part, en explicitant clairement l'utilisation et la sémantique des sous-ensembles flous et, d'autre part, en montrant l'intérêt sur la base d'une comparaison avec des approches n'intégrant pas de sous-ensemble flou (c.-à-d. avec l'approche de Chan et Roddick).

Aperçu de l'approche AHP-Floue de l'étude

Cette partie offre un aperçu général de l'approche AHP-floue mise en œuvre dans notre étude. Notre approche se base, en partie, sur FEHP et se compose de 5 points majeurs donnés dans la FIGURE 3.10, à savoir :

1. construction de la structure AHP de l'étude,
2. collecte des informations nécessaires pour le calcul des scores des alternatives au vu de chaque critère (*cf.* rectangle de gauche dans la point 2 de la FIGURE 3.10) et pour le calcul de l'importance des critères (*cf.* rectangles de droite). Dans notre étude, notons qu'une multitude d'acteurs se prononcent sur les choix/ajustements liés à un même objet (critères, alternatives, etc.). Par conséquent, il est nécessaire d'agréger/fusionner l'ensemble des avis.
3. calcul de la matrice de jugement floue, notée \tilde{A} , qui se compose des sous-ensembles flous de chaque alternative au vu de chaque critère,
4. calcul de la matrice de performance floue, notée \tilde{H} , qui synthétise la matrice \tilde{A} et les sous-ensembles relatifs aux importances des critères et sous-critères.
5. classement des alternatives. Cette étape nécessite, d'une part, d'agréger les performances multicritères des alternatives et, d'autre part, de procéder à leur classement.

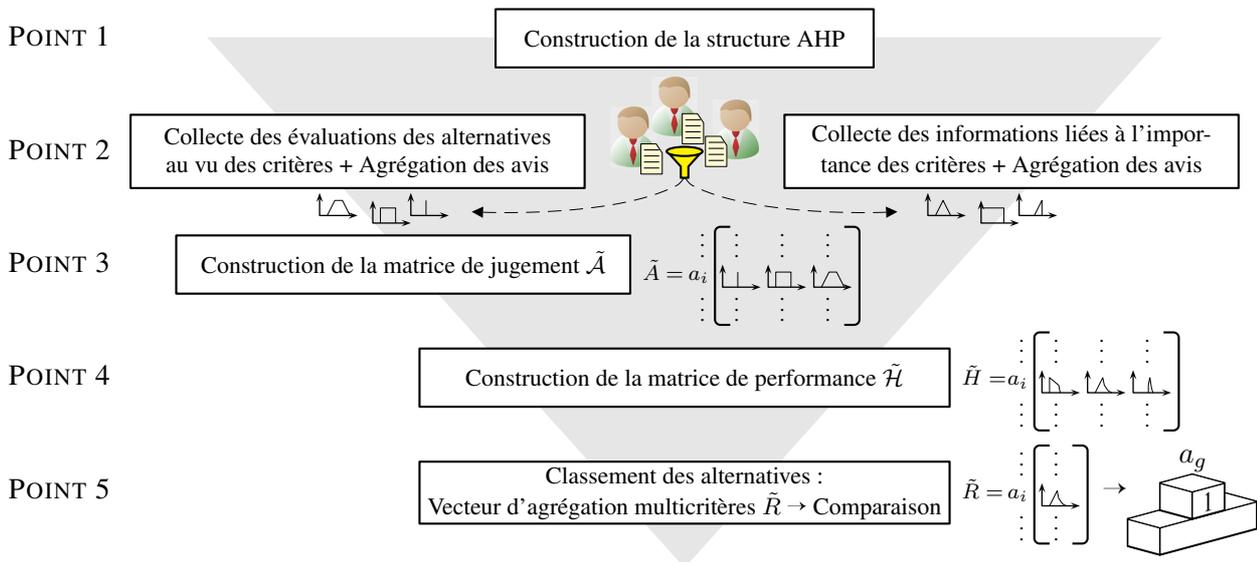


FIGURE 3.10: Approche AHP-Floue se déclinant en 5 points majeurs

Comme souligné dans le point 2, il est nécessaire d'agréger/fusionner l'ensemble des avis des formulés par les experts. Deux cas d'évaluation d'un objet par un ensemble d'experts sont identifiés (Voisin,

1999). Prenons l'exemple de l'évaluation d'un élève par des enseignants :

- *cas (a)* : on cherche par exemple à évaluer les compétences d'un élève en mathématiques par différents professeurs et examens. On obtient ainsi plusieurs « mesures » d'une même caractéristique d'un même objet. On va donc chercher à représenter la complexité de la mesure, au sens de la caractéristique à mesurer (compétence en mathématiques) et du processus (examens et corrections par un enseignant) de la caractéristique par un ensemble de valeurs possibles. Il s'agit alors de réduire l'incertitude sur l'information en utilisant plusieurs sources (d'information) potentiellement de natures différentes. Ainsi, l'agrégation de ces informations augmente la certitude que l'information exacte soit représentée (Voisin, 1999),
- *cas (b)* : on cherche par exemple à évaluer les compétences d'un élève par des enseignants de Mathématiques, de Français et d'Anglais. Il s'agit ici de trouver une représentation entre plusieurs points de vue sur un même objet. Les points de vue étant exprimés désormais sur une même échelle (satisfaction aux connaissances apprises dans une matière). L'évaluation représente la multiplicité des points de vue.

Par rapport à ces deux cas d'évaluation multiple, l'agrégation des avis des experts se fait en fonction des circonstances (cas (a) ou cas (b)) :

- *agrégation cas (a)* : nous proposons d'agréger l'ensemble des avis afin d'obtenir un consensus des évaluations nettes des experts. Cela conduit à ne négliger aucune information fournie par ces derniers. Ces évaluations sont agrégées sous forme d'un intervalle flou, représentant la connaissance complexe que nous avons de l'objet. Pour cela, l'ensemble des avis des experts forme le noyau des valeurs pleinement possibles. Cependant, compte tenu du faible nombre d'experts, nous ne pouvons pas exclure que la valeur recherchée se trouve au-delà des bornes évaluées (un nouvel expert pourrait donner une valeur au-delà des valeurs fournies). Par conséquent, nous proposons d'étendre le support des valeurs possibles au-delà du noyau, ces valeurs étant moins possibles que les valeurs du noyau. En d'autres termes, l'axe y représente l'axe commun d'évaluation partagée par les experts. Le noyau de l'agrégation des avis correspond à l'intervalle entre la valeur minimale et la valeur maximale fournies par les experts. Cet intervalle est élargi par un ensemble de valeurs moins possibles au fur et à mesure de l'éloignement du noyau. Cette décroissance peut être ajustée en fonction du nombre d'experts²⁵. La FIGURE 3.11(a) résume la construction de la fonction d'appartenance $f(y)$ sur la base de trois avis distincts, notés D_1 , D_2 et D_3 ,
- *agrégation cas (b)* : nous proposons de fusionner l'ensemble des avis afin de refléter l'aspect multipoints de vue des experts. Les experts expriment les points de vue à prendre en considération pour la décision. Leur évaluation ne comporte pas d'incertitude. Ces évaluations sont agrégées sous forme d'un intervalle ordinaire, représentant la connaissance complexe que nous avons de l'objet. L'axe y représente un axe lié à une sémantique commune aux différents points de vue. Ainsi, avec les différents points de vue, nous obtenons un encadrement des valeurs possibles. La

25. plus ce nombre est important, moins il y a de chance que la valeur se trouve en dehors du noyau.

complexité de l'information est alors représentée par un intervalle qui, contrairement au cas (a), n'est pas élargi. En effet, l'ajout d'une nouvelle évaluation par un nouvel expert correspondrait à ajouter un point de vue à l'évaluation (p. ex., ajouter la philosophie à l'évaluation du candidat). On supposera donc que l'ensemble des points de vue est représenté dans le panel d'experts interrogés. La FIGURE 3.11(b) synthétise la construction de la fonction d'appartenance $f(y)$ sur la base de 3 avis distincts, notés D_1, D_2 et D_3 .

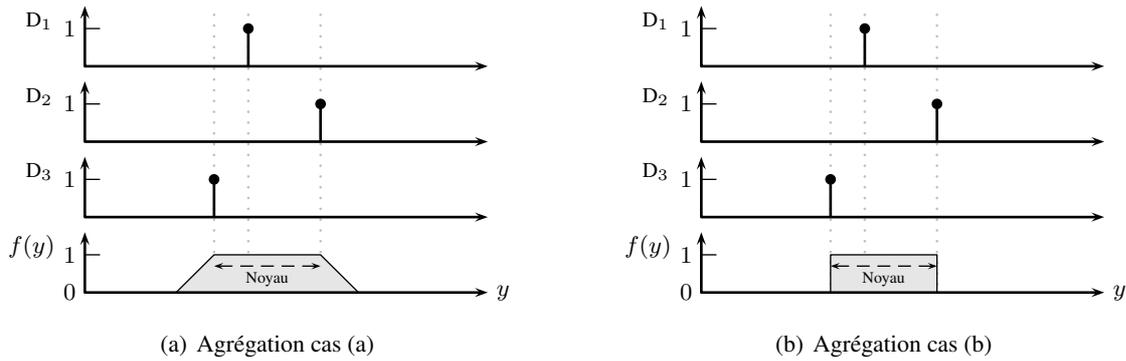


FIGURE 3.11: Agrégation des avis des experts suivant deux cas : cas (a) et cas (b)

3.3.2 Point 1 : structure AHP de l'étude

Le problème de notre étude est hiérarchisé selon la structure AHP décrite dans la FIGURE 3.12. Quatre critères sont définis au niveau 2 : *Enumeration*, *Contextual*, *Model-Based* et *Data Size* (ce dernier critère étant nouveau par rapport à l'approche Chan-Roddick²⁶). Le niveau 3 se compose de l'ensemble des alternatives qui, dans notre étude, correspond à l'ensemble des data items à évaluer. L'objectif final est bien entendu de classer l'ensemble des data items du plus pertinent au moins pertinent, comme formulé au niveau 1 de la structure AHP. Dans la suite du document, le problème est résolu de manière ascendante (du niveau 3 au niveau 1), suivant les points 2 à 5 introduits dans la FIGURE 3.10.

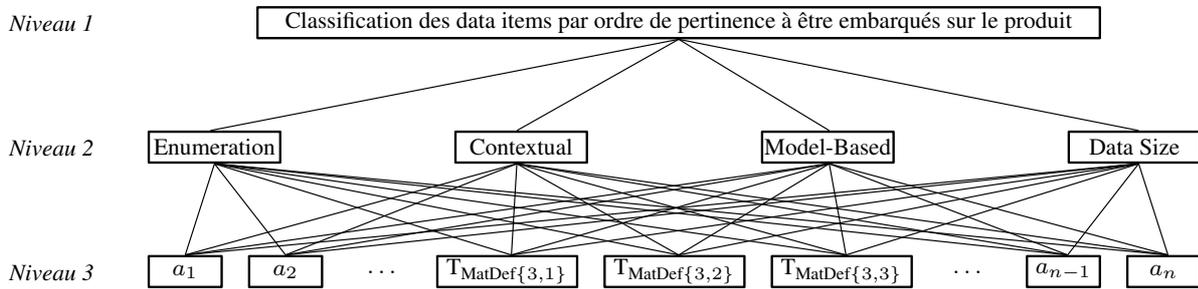


FIGURE 3.12: Structure AHP mise en œuvre dans l'étude

26. « nouveau » au sens où la préférence de la taille des données à embarquer sur le produit ne peut pas être ajustée dans l'approche de Chan et Roddick.

3.3.3 Point 2 : collecte et agrégation des informations liées aux critères et alternatives

Cette section décrit, d'une part, la manière d'attribuer à une alternative l (data item l), au regard d'un critère x , un score sous la forme d'un ensemble flou noté $\tilde{\phi}_x(l)$ et d'autre part, la manière de calculer le score reflétant l'importance de chaque critère x , décrit par un ensemble flou noté $\tilde{\rho}_x$. L'agrégation des informations/avis y est également présentée. Dans cette section, les critères sont détaillés dans l'ordre suivant :

- i. *Contextual* : collecte des informations exprimées par un groupe d'experts au sujet de l'importance des groupes d'entités dans le PLC ; puis agrégation des avis afin d'obtenir $\tilde{\phi}_c(l)$ (l étant une alternative),
- ii. *Enumeration* : collecte des informations énumérées par un groupe d'utilisateurs (experts) durant une opération d'écriture ; puis agrégation des avis afin d'obtenir $\tilde{\phi}_e(l)$,
- iii. *Model-Based* : collecte des informations exprimées par un groupe d'experts durant une opération d'écriture, au sujet du degré de négligence adopté au vu des données distantes ; puis agrégation des avis afin d'obtenir $\tilde{\phi}_m(l)$,
- iv. *Data Size* : collecte des informations exprimées par un groupe d'experts durant une opération d'écriture, au sujet du degré de négligence adopté au vu de la taille des données ; puis agrégation des avis afin d'obtenir $\tilde{\phi}_s(l)$,
- v. *Critères* : collecte des informations exprimées par un groupe d'experts, au sujet de l'importance des critères durant une opération d'écriture ; puis agrégation des avis afin d'obtenir $\tilde{\rho}_x$ ($x \in \{c, e, m, s\}$),

Contextual

L'évaluation de l'importance des groupes d'entités sur une phase particulière du PLC utilise le principe de *pairwise comparison*. Cependant, contrairement à l'approche de Chan-Roddick, plusieurs experts effectuent ces comparaisons. Les experts réalisant l'évaluation font partie du domaine des systèmes d'information relatifs à PLC/PLM. Nous considérons donc que relativement à l'information fournie, il ont un même point de vue sur le critère, ce qui correspond à l'agrégation du cas (a) : FIGURE 3.11(a). Chaque expert p remplit une matrice de *pairwise comparison* D_G^p comme présenté dans l'ÉQUATION 3.20, où Z correspond au nombre de groupes d'entités défini dans le MLD et s_{ij}^p représente l'importance du groupe i par rapport au groupe j , évaluée par l'expert p sur l'échelle de Saaty S . Avant de calculer l'importance relative de chaque groupe d'entité (c.-à-d. le vecteur propre flou $\tilde{\Lambda}_G$), il est nécessaire d'agréger les différents avis des experts (c.-à-d., les matrices $D_G^p \mid p = \{1, 2, \dots, Q\}$). La matrice \tilde{G} de l'ÉQUATION 3.21 correspond à l'agrégation des matrices D_G^p . Chaque élément \tilde{s}_{ij} est un sous-ensemble flou obtenu par agrégation de cas (a) de l'ensemble des \tilde{s}_{ij}^p . L'agrégation se base sur les hypothèses et les règles définies dans la TABLE 3.3. Enfin, l'importance relative de chaque groupe G_i , correspondant à un sous-ensemble flou noté $\tilde{\lambda}_G(G_i)$, est calculée dans l'ÉQUATION 3.22 sur la base d'une représentation à 11 α -coupes

(une α -coupe est notée $\lambda_G^\alpha(G_i)$). Ces importances relatives sont synthétisées par le vecteur flou $\tilde{\Lambda}_G$ dans l'ÉQUATION 3.23.

Pour un data item l appartenant à la table t du groupe G_i , nous obtenons comme score :

$$\tilde{\phi}_c(l) = \tilde{\lambda}_G(G_i) \quad \text{avec } l \in t \text{ et } t \in G_i$$

$$D_G^p = \begin{matrix} & \begin{matrix} G_1 & G_2 & \dots & G_Z \end{matrix} \\ \begin{matrix} G_1 \\ G_2 \\ \vdots \\ G_Z \end{matrix} & \begin{bmatrix} s_{11}^p & s_{12}^p & \dots & s_{1Z}^p \\ s_{21}^p & s_{22}^p & \dots & s_{2Z}^p \\ \vdots & \vdots & \ddots & \vdots \\ s_{Z1}^p & s_{Z2}^p & \dots & s_{ZZ}^p \end{bmatrix} \end{matrix}, \quad p = \{1, 2, \dots, Q\} \quad s_{ij}^p = \begin{cases} 1 & i = j \\ (s_{ji}^p)^{-1} & i \neq j \end{cases} \quad (3.20)$$

$$\tilde{G} = \begin{matrix} & \begin{matrix} G_1 & G_2 & \dots & G_Z \end{matrix} \\ \begin{matrix} G_1 \\ G_2 \\ \vdots \\ G_Z \end{matrix} & \begin{bmatrix} \tilde{s}_{11} & \tilde{s}_{12} & \dots & \tilde{s}_{1Z} \\ \tilde{s}_{21} & \tilde{s}_{22} & \dots & \tilde{s}_{2Z} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{s}_{Z1} & \tilde{s}_{Z2} & \dots & \tilde{s}_{ZZ} \end{bmatrix} \end{matrix} \quad (3.21)$$

$$\lambda_G^\alpha(G_i) = \frac{\sum_{j=1}^Z G^\alpha(i, j)}{\sum_{k=1}^Z \sum_{j=1}^Z G^\alpha(k, j)} \quad \forall \alpha = \{0, 0.1, 0.2, \dots, 0.9, 1\} \quad (3.22)$$

$$\tilde{\Lambda}_G = [\tilde{\lambda}_G(G_1), \tilde{\lambda}_G(G_2), \dots, \tilde{\lambda}_G(G_i), \dots, \tilde{\lambda}_G(G_Z)] \quad (3.23)$$

⇔ Exemple : quatre groupes d'entités sont définis dans le MLD. Deux experts D_1 et D_2 soumettent respectivement les matrices de *pairwise comparisons* présentées²⁷ dans la FIGURE 3.13. D_1 spécifie que G_1 est beaucoup plus important que G_4 ($s_{14}^1 = 7$) et D_2 affecte une importance sensiblement inférieure ($s_{14}^2 = 5$). L'agrégation des avis de ces deux experts est ensuite effectuée via les règles définies dans la TABLE 3.3. Le détail de l'agrégation des avis relatifs à \tilde{s}_{14} est détaillé dans la FIGURE 3.14²⁸. Enfin, l'importance relative de chaque groupe G_i est calculée à l'aide de l'ÉQUATION 3.22²⁹. Les sous-ensembles flous résultants $\tilde{\lambda}_G(G_1)$, $\tilde{\lambda}_G(G_2)$, $\tilde{\lambda}_G(G_3)$ et $\tilde{\lambda}_G(G_4)$ forment le vecteur flou $\tilde{\Lambda}_G$ présenté dans la FIGURE 3.13 (vecteur détaillé dans la FIGURE 30 de l'Annexe IV). On remarque au vu de ces sous-ensembles flous que G_3 est le groupe d'entités le plus important³⁰ les que celles des autres fonctions d'appartenance et se rapproche de, suivi de G_1 , G_2 et G_4 . Avec la même hypothèse que précédemment (c.-à-d., $\text{MaterialDefinition} \in G_1$), les sous-ensembles flous des data items $T_{\text{MatDef}}\{3, 1\}$,

27. matrices valables pour une étape du PLC qui peut inclure un ou plusieurs points d'écriture.

28. les valeurs de la matrice \tilde{G} sont détaillées dans l'ÉQUATION 10 de l'Annexe IV.

29. l'ÉQUATION 11 dans l'Annexe IV détaille le calcul pour le groupe G_1 à la coupe $\alpha = 0$, notée $\lambda_G^0(G_1)$.

30. on peut le voir en comparant à vue d'oeil les aires couvertes par chaque fonction d'appartenance. On constate que celle relative à $\tilde{\lambda}_G(G_3)$ est plus imposante et supérieure aux trois autres.

$T_{MatDef}\{3, 2\}$ et $T_{MatDef}\{3, 3\}$, notés $\tilde{\phi}_c(T_{MatDef}\{3, 1\})$, $\tilde{\phi}_c(T_{MatDef}\{3, 2\})$ et $\tilde{\phi}_c(T_{MatDef}\{3, 3\})$ sont égaux à celui du groupe G_1 comme illustré dans la FIGURE 3.13.

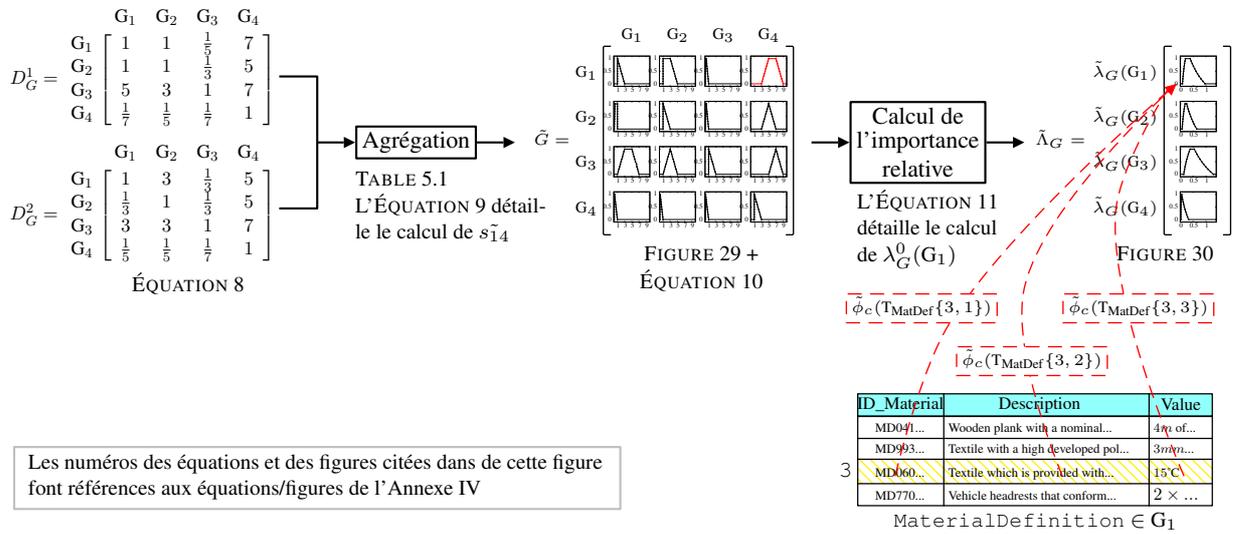


FIGURE 3.13: Évaluation de l'importance des groupes d'entités, agrégation des avis et calcul de leur importance relative

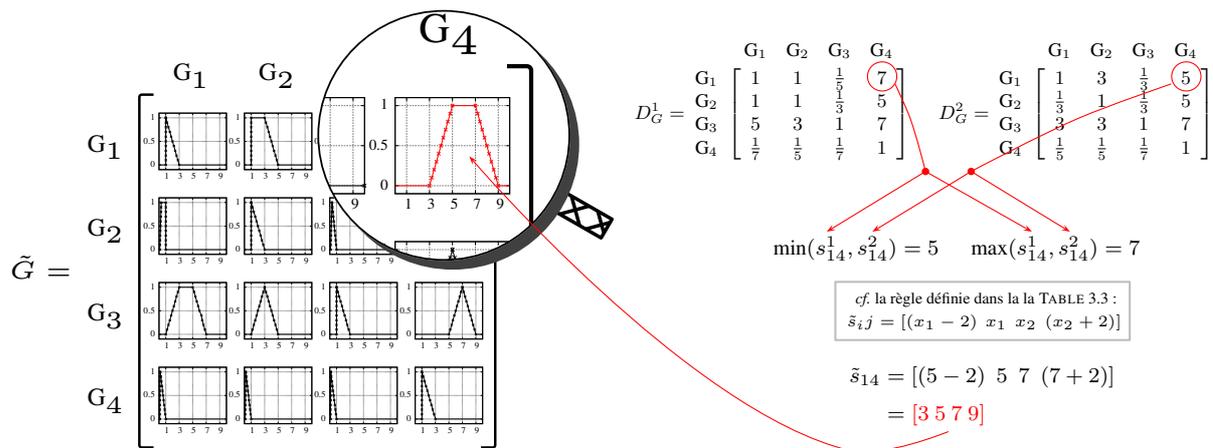
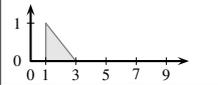
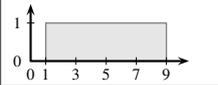
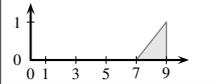
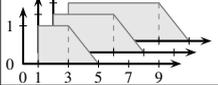
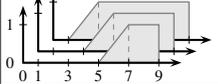
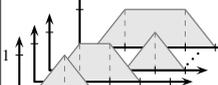
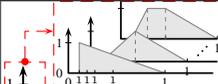


FIGURE 3.14: Construction du sous-ensemble flou \tilde{s}_{14} dans la matrice \tilde{G}

Deux hypothèses sont formulées :

- hypothèse 1 : les experts s'accordent sur le sens de l'importance ($G_i > G_j$ ou $G_i < G_j$),
- hypothèse 2 : on reste dans l'échelle de Saaty. Ainsi pour respecter l'hypothèse 1, on se trouve soit dans $\{1, 3, 5, 7, 9\}$, soit dans $\{\frac{1}{9}, \frac{1}{7}, \frac{1}{5}, \frac{1}{3}, 1\}$.

TABLE 3.3: Construction des sous-ensembles flous relatifs à la matrice de *pairwise comparison*

Condition	Fonctions d'app.	Description	Illustration
Les agrégations des évaluations $G_i > G_j$ sont construites sur les règles suivantes :			
$\min_{p \in \{1..Q\}} (s_{ij}^p) = 1$ $\max_{p \in \{1..Q\}} (s_{ij}^p) = 1$	$\tilde{s}_{ij} = [1 \ 1 \ 1 \ 3]$	Les experts sont d'accord sur l'équivalence des groupes. De plus, pour respecter l'hypothèse 1, ils se sont entendus sur $G_i > G_j$. Ainsi, l'intervalle flou est limité à 1 et « étendu » (agrégation cas (a)) jusque ³¹ 3.	
$\min_{p \in \{1..Q\}} (s_{ij}^p) = 1$ $\max_{p \in \{1..Q\}} (s_{ij}^p) = 9$	$\tilde{s}_{ij} = [1 \ 1 \ 9 \ 9]$	Les experts s'entendent sur l'importance $G_i > G_j$. Cependant, l'agrégation conduit à une incertitude totale quant à la valeur de l'importance. Pour respecter l'hypothèse 2, l'intervalle $[1; 9]$ est défini.	
$\min_{p \in \{1..Q\}} (s_{ij}^p) = 9$ $\max_{p \in \{1..Q\}} (s_{ij}^p) = 9$	$\tilde{s}_{ij} = [7 \ 9 \ 9 \ 9]$	Les experts s'entendent sur l'importance extrême de $G_i > G_j$. Pour respecter l'hypothèse 2, l'intervalle est borné à 9 et est étendu à la graduation immédiatement inférieure (c.-à-d. la valeur 7).	
$\min_{p \in \{1..Q\}} (s_{ij}^p) = 1$ $\max_{p \in \{1..Q\}} (s_{ij}^p) = x$ $x = 3, 5, 7$	$\tilde{s}_{ij} = [1 \ 1 \ x(x+2)]$	Les experts s'entendent sur l'importance $G_i > G_j$. Cependant, l'agrégation conduit à une incertitude partielle quant à la valeur de l'importance. Pour respecter l'hypothèse 2, l'intervalle $[1; x]$ est défini et est étendu à la graduation immédiatement supérieure à x .	
$\min_{p \in \{1..Q\}} (s_{ij}^p) = x$ $\max_{p \in \{1..Q\}} (s_{ij}^p) = 9$ $x = 3, 5, 7$	$\tilde{s}_{ij} = [(x - 2) \ x \ 9 \ 9]$	Les experts s'entendent sur l'importance $G_i > G_j$. Cependant, l'agrégation conduit à une incertitude partielle quant à la valeur de l'importance. Pour respecter l'hypothèse 2, l'intervalle $[x; 9]$ est défini et est étendu à la graduation immédiatement inférieure à x .	
$\min_{p \in \{1..Q\}} (s_{ij}^p) = x_1$ $\max_{p \in \{1..Q\}} (s_{ij}^p) = x_2$ $x_1, x_2 = 3, 5, 7$	$\tilde{s}_{ij} = [(x_1 - 2) \ x_1 \ x_2 \ (x_2 + 2)]$	Les experts s'entendent sur l'importance $G_i > G_j$. Cependant, l'agrégation conduit à une incertitude partielle quant à la valeur de l'importance. Pour respecter l'hypothèse 2, l'intervalle $[x_1; x_2]$ est défini et est étendu aux graduations immédiatement inférieures et supérieures à x_1 et x_2 respectivement.	
Les agrégations des évaluations \tilde{s}_{ji} sont déduites des évaluations $G_i > G_j$ par la règle suivante :			
$\min_{p \in \{1..Q\}} (s_{ij}^p) = x_1$ $\max_{p \in \{1..Q\}} (s_{ij}^p) = x_2$ $x_1, x_2 = 1, 3, 5, 7, 9$	$\tilde{s}_{ji} = [\frac{1}{9} \ \frac{1}{x_2} \ \frac{1}{x_1} \ 1]$	Dans les méthodes traditionnelles comme FEAHP, $\tilde{s}_{ji} = \frac{1}{\tilde{s}_{ij}}$. Pour simplifier notre méthode, nous décidons que les bornes de l'intervalle flou \tilde{s}_{ji} soient égales à l'inverse des valeurs extrêmes de S , à savoir $\frac{1}{9}$ et 1, quelles que soit les valeurs x_1 et x_2 . Cela réduit le nombre de possibilités qui, initialement, n'influent quasi pas sur le résultat final car les intervalles flous \tilde{s}_{ji} sont définis sur $]0, 1]$ et les intervalles \tilde{s}_{ij} sur $]1, 9]$.	

31. nous nous limitons à étendre l'intervalle à la graduation immédiatement supérieure ou inférieure de l'échelle de Saaty. En effet, cette échelle comporte dans notre étude cinq niveaux, ce qui correspond à 25% de l'échelle. Élargir à deux graduations correspondrait à 50%, ce qui serait trop important (même si le nombre d'experts est faible).

Enumeration

Rappelons ici que les utilisateurs énumèrent les attributs des tables qu'ils souhaitent stocker sur le produit. Dans notre étude, nous ajoutons une valeur de choix intermédiaire par opposition à l'approche de Chan et Roddick (c.-à-d. la valeur 0.5), afin que le choix de stocker ou non l'information sur le produit ne soit plus booléen. Le score d'un attribut v d'une table t , noté $s^p(v, t)$, avec p le numéro du décideur, prend la valeur 1 ou 0.5 s'il est énuméré (en fonction de l'intensité de la préférence) et 0 s'il n'est pas énuméré, comme formalisé dans l'ÉQUATION 3.24. Dans ce critère, les experts qui réalisent l'évaluation à un point d'écriture du produit peuvent provenir de domaines variés (p. ex. un responsable de la chaîne de production, un responsable du transport, etc.). Nous considérons donc que relativement à l'information fournie, ils n'ont pas nécessairement un même point de vue sur le critère. En effet, un expert/utilisateur peut souhaiter stocker une information particulière pour une raison qui lui est propre (p. ex. le responsable de la chaîne de production souhaite stocker sur le produit les références des machines qui ont usiné le produit, alors que le responsable du transport souhaite y mettre des informations relatives aux lieux de stockage du produit). Puisque les experts ont des points de vue différents sur ce critère, nous mettons en œuvre l'agrégation du cas (b) : FIGURE 3.11(b). La TABLE 3.4 présente le calcul d'agrégation de l'ensemble des avis (noté \mathcal{V}) relatif à ce critère.

Tout data item localisé dans la colonne/attribut v de la table t a un sous-ensemble flou au regard du critère *Enumeration*, noté $\tilde{\phi}_e(l)$, équivalent à $\tilde{s}(v, t)$.

$$s^p(v, t) = \begin{cases} 1 & \text{énuméré (attribut nécessaire)} \\ 0.5 & \text{énuméré (attribut intéressant)} \\ 0 & \text{non énuméré (attribut inutile)} \end{cases} \quad p = \{1, 2, \dots, Q\}, v \text{ un attribut de la table } t \quad (3.24)$$

TABLE 3.4: Construction des sous-ensembles flous relatifs aux attributs des tables

Fonction d'appartenance	Illustration
$\tilde{s}(v, t) = [\min(\mathcal{V}) \quad \max(\mathcal{V})] \quad \mathcal{V} = \{s^1(v, t), s^2(v, t), \dots, s^Q(v, t)\}$	

⇒ Exemple : trois utilisateurs D_1 , D_2 et D_3 s'intéressent à spécifier s'ils désirent ou non les trois attributs composant la table *MaterialDefinition*, à savoir : *ID_Material*, *Description* et *value* (cf. FIGURE 3.3). Ils énumèrent respectivement ces trois attributs comme détaillé dans la FIGURE 3.15 (p. ex. le décideur 1 juge « nécessaire » l'attribut *ID_Material* mais « inutile » *Description*). Les sous-ensembles flous résultant de l'agrégation des avis des trois experts sont calculés dans les ÉQUATIONS 13 à 15 de l'Annexe IV et sont présentés dans la FIGURE 3.15. Les sous-ensembles flous de $T_{\text{MatDef}}\{3, 1\}$, $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 3\}$ au regard du critère *Enumeration*, notés $\tilde{\phi}_e(T_{\text{MatDef}}\{3, 1\})$,

$\tilde{\phi}_e(\text{T}_{\text{MatDef}}\{3, 2\})$ et $\tilde{\phi}_e(\text{T}_{\text{MatDef}}\{3, 3\})$, sont par conséquent respectivement égaux à ceux des attributs ID_Material, Description et value comme illustré sur cette même figure.

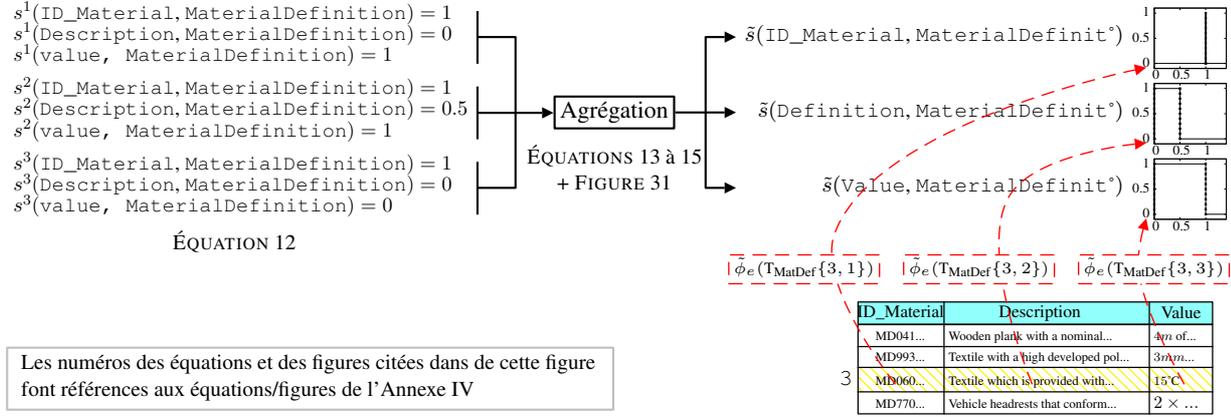


FIGURE 3.15: Énumération des attributs de MaterialDefinition et agrégation des avis

Model-Based

Il s'agit dans ce critère de régler le coefficient k afin de spécifier si des données plus ou moins « lointaines » de la table produit doivent être stockées sur le produit (cf. FIGURE 3.5). Chaque décideur p règle le coefficient k comme indiqué dans l'ÉQUATION 3.25. Dans ce critère, les experts qui réalisent l'évaluation peuvent également provenir de secteurs différents et ne pas avoir un même point de vue sur le critère. En effet, un expert peut souhaiter stocker des informations distantes car celles qui l'intéressent sont éloignées de la table produit, alors qu'un autre expert peut être intéressé uniquement par des informations proches de la table produit. Par conséquent, nous mettons en œuvre l'agrégation du cas (b)³² (cf. FIGURE 3.11(b)). La TABLE 3.4 présente le calcul d'agrégation de l'ensemble des avis (noté \mathcal{V}) relatifs à ce critère.

$$s_m^p = k \quad p = \{1, 2, \dots, Q\}, \quad k \in [1; +\infty] \quad (3.25)$$

TABLE 3.5: Construction du sous-ensemble flou relatif au coefficient k dans le critère C_m

Fonction d'appartenance	Illustration
$\tilde{s}_m = [\min(\mathcal{V}) \quad \max(\mathcal{V})]$	
$\mathcal{V} = \{s_m^1, s_m^2, \dots, s_m^Q\}$	

Le sous-ensemble flou de chaque data item l au regard du critère *Model-Based*, noté $\tilde{\phi}_m(l)$, peut alors être calculé via l'ÉQUATION 3.26, où d représente la distance entre le data item l et la table produit

³². Le type d'agrégation choisi dans ce critère peut être discuté. Il est possible d'envisager une agrégation de type cas (a) si on considère que les experts qui réalisent l'ajustement de k ont un même point de vue.

(nombre minimal de tables les séparant).

$$\tilde{\phi}_m(l) = (\tilde{s}_m)^{-d} = \left[\max(\mathcal{V})^{-d} \min(\mathcal{V})^{-d} \right] \quad d \in \mathbb{N} \quad (3.26)$$

⇒ Exemple : deux utilisateurs D_1 et D_2 fixent respectivement k à 1.03 et 1.08 comme indiqué dans la FIGURE 3.16. Ces avis sont ensuite agrégés au travers de la fonction \tilde{s}_m (calcul détaillé dans l'ÉQUATION 17 de l'Annexe IV) et, ainsi, les sous-ensembles flous de $T_{\text{MatDef}}\{3, 1\}$, $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 3\}$ au regard du critère *Model-Based*, notés $\tilde{\phi}_m(T_{\text{MatDef}}\{3, 1\})$, $\tilde{\phi}_m(T_{\text{MatDef}}\{3, 2\})$ et $\tilde{\phi}_m(T_{\text{MatDef}}\{3, 3\})$ sont calculés (calculs détaillés respectivement dans les ÉQUATIONS 18, 19 et 20 de l'Annexe IV³³). Les sous-ensembles résultants sont présentés dans la FIGURE 3.16. Nous remarquons que ces trois data items disposent du même sous-ensemble flou car ils proviennent de la même table.

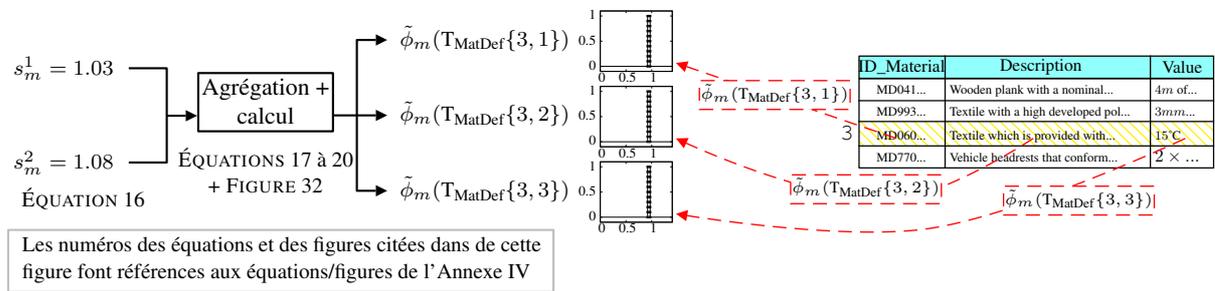


FIGURE 3.16: Réglage du coefficient k et agrégation des avis concernant le critère *Model-Based*

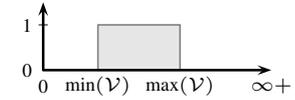
Data Size

Il s'agit, pour ce critère, de favoriser le stockage sur le produit de données en fonction de leur taille. Pour cela, nous proposons d'utiliser la même formule que pour le précédent critère, à savoir k^{-d} avec k une constante ajustable et d , non plus une distance, mais la taille du data item. Le coefficient k est réglé comme indiqué dans l'ÉQUATION 3.27. Tout comme le précédent critère, les experts qui réalisent l'évaluation peuvent avoir des points de vue différents sur ce critère. Par exemple, considérons qu'un premier expert est confronté ou intéressé par une grande quantité de tuples (p. ex. il énumère beaucoup d'attributs), il voudra peut-être stocker les informations de petites tailles afin d'avoir une vue « complète/diversifiée » de cette quantité (p. ex. préférer stocker 80 petits data items que 5 énormes). Peut-être qu'un second expert est quant à lui intéressé, à ce point d'écriture, par une petite quantité d'information (p. ex., peu d'attributs énumérés) et veut autoriser le stockage d'informations de grandes tailles. Par conséquent, nous mettons en œuvre l'agrégation du cas (b) : FIGURE 3.11(b). La TABLE 3.6 présente le calcul d'agrégation de l'ensemble des avis (noté \mathcal{V}) relatifs à ce critère.

$$s_s^p = k \quad p = \{1, 2, \dots, Q\}, \quad k \in [1; +\infty] \quad (3.27)$$

33. la puissance d dans ces équations est égal à 1 car ces trois data items proviennent de la même table, qui est distante de 1 table de la table produit.

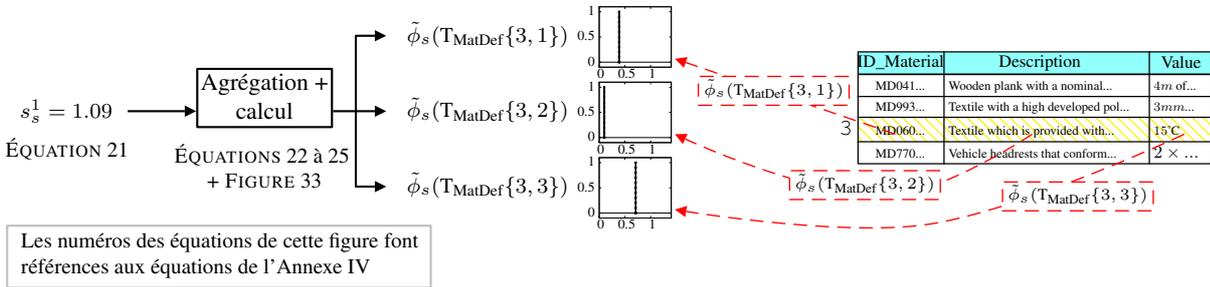
TABLE 3.6: Construction du sous-ensemble flou relatif au coefficient k dans le critère C_s

Fonction d'appartenance		Illustration
$\tilde{s}_s = [\min(\mathcal{V}) \quad \max(\mathcal{V})]$	$\mathcal{V} = \{s_s^1, s_s^2, \dots, s_s^Q\}$	

Le sous-ensemble flou de chaque data item l au regard du critère *Data Size*, noté $\tilde{\phi}_s(l)$, peut alors être calculé via l'ÉQUATION 3.28 avec d la taille en octets du data item l .

$$\tilde{\phi}_s(l) = (\tilde{s}_s)^{-d} = \left[\max(\mathcal{V})^{-d} \quad \min(\mathcal{V})^{-d} \right] \quad d \in \mathbb{N} \quad (3.28)$$

⇒ Exemple : un unique expert D_1 fixe k à 1.09 comme indiqué dans la FIGURE 3.17. Le sous-ensemble flou après agrégation est par conséquent équivalent à un singleton de valeur 1.09 (cf. ÉQUATION 22 dans l'Annexe IV). Les sous-ensembles flous de $T_{\text{MatDef}}\{3, 1\}$, $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 3\}$ au regard du critère *Data Size*, notés $\tilde{\phi}_s(T_{\text{MatDef}}\{3, 1\})$, $\tilde{\phi}_s(T_{\text{MatDef}}\{3, 2\})$ et $\tilde{\phi}_s(T_{\text{MatDef}}\{3, 3\})$ sont respectivement calculés (calculs détaillés respectivement dans dans les ÉQUATIONS 23, 24 et 25 de l'Annexe IV³⁴), dont les résultats sont illustrés dans la FIGURE 3.17.

FIGURE 3.17: Réglage du coefficient k et agrégation des avis concernant le critère *Data Size*

Critères

Désormais, il ne s'agit plus de calculer le score d'une alternative au regard d'un critère, mais de calculer l'importance de chaque critère durant une opération d'écriture du produit (cf. FIGURE 3.10). Ces importances sont spécifiées par un groupe d'experts via les matrices de *pairwise comparisons* D_p^p fournies dans l'ÉQUATION 3.29 (de manière similaire à la spécification de l'importance des groupes d'entités dans le critère *Contextual*), où s_{ij}^p représente l'importance du critère i par rapport au critère j , évaluée par l'expert p sur l'échelle S . La matrice $\tilde{\rho}$ de l'ÉQUATION 3.30 correspond à l'agrégation des matrices D_p^p et chaque élément \tilde{s}_{ij} est obtenu via les règles définies dans la TABLE 3.3. L'importance

34. Les puissances d dans les ÉQUATIONS 23, 24 et 25 sont respectivement égales à 11, 27 et 4 qui sont fonction des tailles respectives en octets de $T_{\text{MatDef}}\{3, 1\}$, $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 3\}$.

relative de chaque critère C_i , correspondant à un sous-ensemble flou noté $\tilde{\lambda}_\rho(C_i)$, est enfin calculée dans l'ÉQUATION 3.31 sur la base d'une représentation à 11 α -coupes (notées $\lambda_\rho^\alpha(C_i)$). Ces importances relatives sont synthétisées par le vecteur flou $\tilde{\Lambda}_\rho$ dans l'ÉQUATION 3.32.

$$D_\rho^p = \begin{matrix} & C_1 & C_2 & \dots & C_M \\ \begin{matrix} C_1 \\ C_2 \\ \vdots \\ C_M \end{matrix} & \begin{bmatrix} s_{11}^p & s_{12}^p & \dots & s_{1M}^p \\ s_{21}^p & s_{22}^p & \dots & s_{2M}^p \\ \vdots & \vdots & \ddots & \vdots \\ s_{M1}^p & s_{M2}^p & \dots & s_{MM}^p \end{bmatrix} \end{matrix} \quad p = \{1..Q\}, \quad s_{ij}^p = \begin{cases} 1 & i = j \\ (s_{ji}^p)^{-1} & i \neq j \end{cases} \quad (3.29)$$

$$\tilde{\rho} = \begin{matrix} & C_1 & C_2 & \dots & C_M \\ \begin{matrix} C_1 \\ C_2 \\ \vdots \\ C_M \end{matrix} & \begin{bmatrix} \tilde{s}_{11} & \tilde{s}_{12} & \dots & \tilde{s}_{1M} \\ \tilde{s}_{21} & \tilde{s}_{22} & \dots & \tilde{s}_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{s}_{M1} & \tilde{s}_{M2} & \dots & \tilde{s}_{MM} \end{bmatrix} \end{matrix} \quad (3.30)$$

$$\lambda_\rho^\alpha(C_i) = \frac{\sum_{j=1}^M \rho^\alpha(i, j)}{\sum_{k=1}^M \sum_{j=1}^M \rho^\alpha(k, j)} \quad \forall \alpha = \{0, 0.1, 0.2, \dots, 0.9, 1\} \quad (3.31)$$

$$\tilde{\Lambda}_\rho = [\tilde{\lambda}_\rho(C_e), \tilde{\lambda}_\rho(C_c), \dots, \tilde{\lambda}_\rho(C_i), \dots, \tilde{\lambda}_\rho(C_M)] \quad (3.32)$$

⇒ Exemple : Deux experts D_1 et D_2 soumettent respectivement les matrices de *pairwise comparisons* présentées dans la FIGURE 3.18. La matrice d'agrégation des différents avis $\tilde{\rho}$ est calculée comme illustré dans cette même figure (matrice détaillée dans la FIGURE 34 de l'Annexe IV). Puis, les importances relatives des quatre critères (sous-ensembles flous notés $\tilde{\lambda}_\rho(C_e)$, $\tilde{\lambda}_\rho(C_c)$, $\tilde{\lambda}_\rho(C_m)$ et $\tilde{\lambda}_\rho(C_s)$) sont calculées à l'aide de l'ÉQUATION 3.31 et sont synthétisées par le vecteur $\tilde{\Lambda}_\rho$ dans la FIGURE 3.18 (vecteur détaillé dans la FIGURE 35 de l'Annexe IV). Nous constatons au vu de ces sous-ensembles que le critère *Enumeration* est plus important que les trois autres.

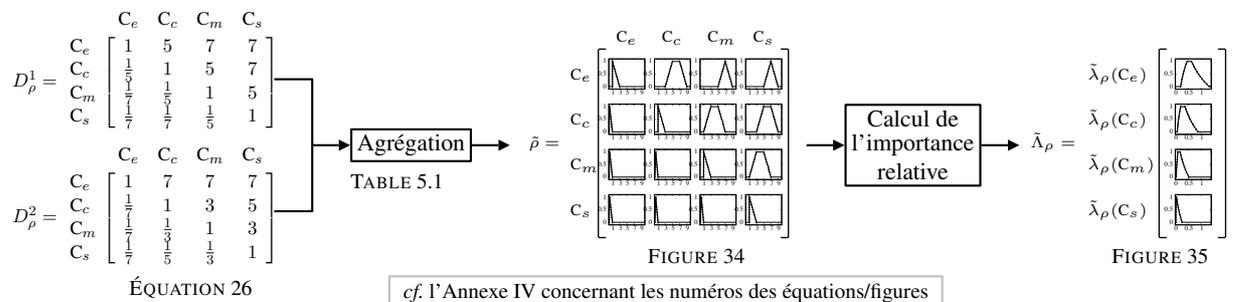


FIGURE 3.18: Évaluation de l'importance des critères, agrégation des avis et calcul de leur importance

3.3.4 Point 3 : calcul de la matrice de jugement flou \tilde{A}

Les évaluations de toutes les alternatives au regard des critères ont ainsi été calculées, comme cela est détaillé pour les data items $T_{MatDef}\{3, 1\}$, $T_{MatDef}\{3, 2\}$ et $T_{MatDef}\{3, 3\}$. Ces évaluations sont regroupées dans la matrice de jugement notée \tilde{A} , présentée dans la FIGURE 3.19³⁵.

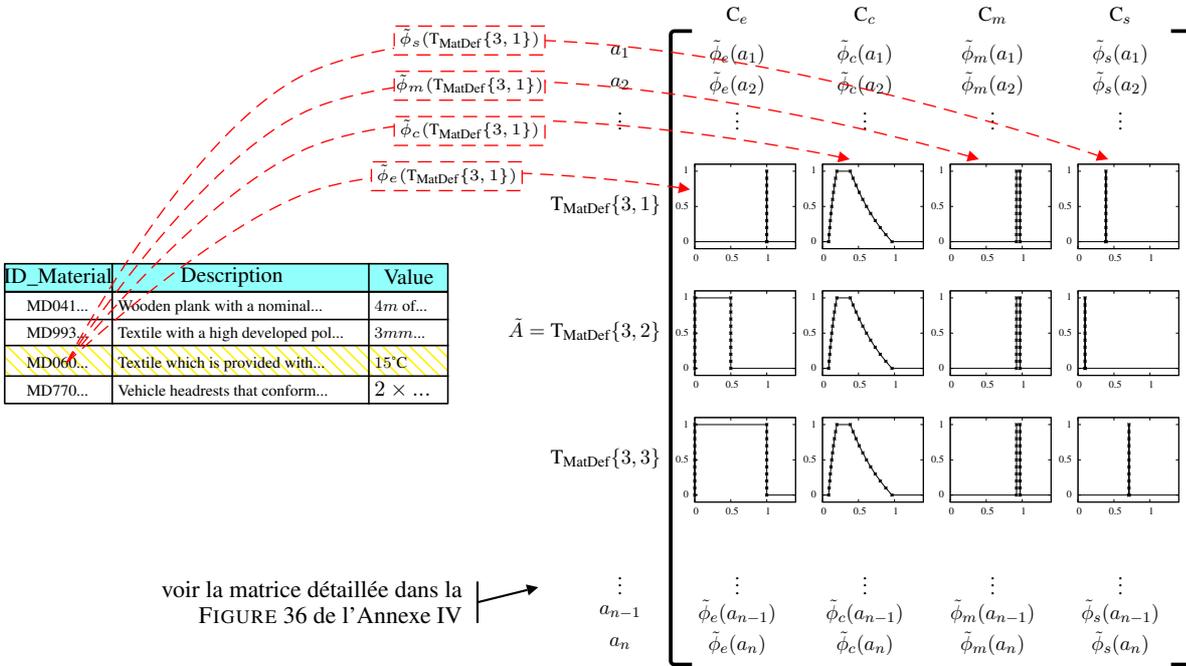


FIGURE 3.19: Matrice de jugement flou \tilde{A} : sous-ensembles flous des alternatives au vu des critères

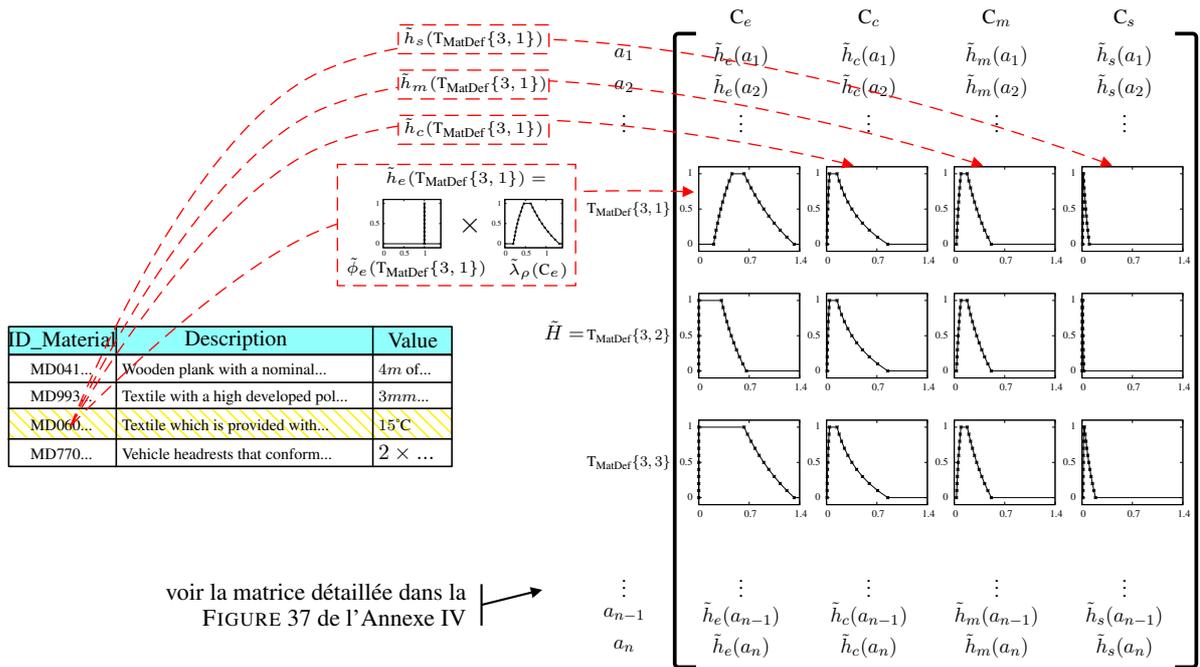
3.3.5 Point 4 : calcul de la matrice de performance floue \tilde{H}

Jusqu'à présent, la matrice de jugement flou \tilde{A} renseigne sur le score des alternatives au regard des critères, mais elle ne tient pas compte de l'importance de chacun des critères. Par conséquent, la matrice de performance floue \tilde{H} est atteinte en multipliant le score de chaque alternative l au regard de chaque critère x (c.-à-d., $\tilde{\phi}_x(l)$), par l'importance relative du critère x lui-même (c.-à-d., $\tilde{\lambda}_\rho(C_x)$). Cette multiplication s'effectue bien entendu sur la base de leurs α -coupes (cf. Annexe IV).

⇨ Exemple : la matrice de performance floue \tilde{H} peut être calculée puisque nous disposons des sous-ensembles flous $\tilde{\phi}_x(l)$ et $\tilde{\lambda}_\rho(C_x)$ pour chaque alternative l ($\forall x = \{e, c, m, s\}$). La FIGURE 3.20 illustre graphiquement la multiplication $\tilde{\phi}_e(T_{MatDef}\{3, 1\}) \times \tilde{\lambda}_\rho(C_e)$, dont le résultat noté $\tilde{h}_e(T_{MatDef}\{3, 1\})$ prend place dans la matrice \tilde{H} ³⁶.

35. Cette matrice se compose en réalité de 11 matrices de sous-ensembles non flous relatives aux 11 α -coupes, comme cela est détaillé dans la FIGURE 36 de l'Annexe IV. L'intersection d'une ligne l et d'une colonne x dans A^0 de la FIGURE 36 renseigne sur le sous-ensemble non flou de $\tilde{\phi}_x(l)$ à la coupe $\alpha = 0$.

36. Cette matrice se compose également de 11 matrices de sous-ensembles non flous relatives aux 11 α -coupes, comme cela est détaillé dans la FIGURE 37 de l'Annexe IV. Le calcul de $h_e^0(T_{MatDef}\{3, 2\})$ est détaillé dans l'ÉQUATION 28 de cette même annexe. Ce calcul consiste à multiplier les intervalles de l' α -coupe de niveau 0 de $\tilde{\phi}_e(T_{MatDef}\{3, 2\})$ (i.e., $[0, 0.5]$, cf. FIGURE 36 dans l'Annexe IV), avec les intervalles $\lambda_\rho^0(C_e)$ (i.e. $[0.2, 1.32]$, cf. FIGURE 35 dans l'Annexe IV).


 FIGURE 3.20: Matrice de performance floue \tilde{H} : prise en compte de l'importance des critères

3.3.6 Point 5 : classification des alternatives/data items

Ce point vise à classer les alternatives en fonction de leurs évaluations respectives aux différents critères. Jusqu'à alors, chaque alternative est évaluée par un sous-ensemble flou au regard de chaque critère, soit au total, quatre sous-ensembles flous par data item (cf. FIGURE 3.20). Comme l'explique (Grabisch and Perny, 1999), deux modes opératoires distincts peuvent être mis en œuvre pour l'agrégation multicritères, nécessaire à l'étape de comparaison des alternatives. Afin de détailler ces deux modes, introduisons la fonction f qui évalue la relation de préférence entre les alternatives a et b (préférence notée $\succsim(a, b)$). La fonction f réalise, d'une part, l'agrégation des performances a_j et b_j (c.-à-d., les performances au regard de chaque critère $j \forall j = \{C_1, \dots, C_M\}$) et, d'autre part, la comparaison des alternatives a et b . Les deux modes opératoires possibles sont :

- l'approche AC « Agréger puis Comparer » : elle se caractérise par une décomposition du calcul $f(a_{C_1}, \dots, a_{C_M}, b_{C_1}, \dots, b_{C_M})$ sous la forme $\Upsilon(\Omega(a_{C_1}, \dots, a_{C_M}), \Omega(b_{C_1}, \dots, b_{C_M}))$ avec Ω l'opérateur d'agrégation et Υ la fonction de comparaison, d'où l'appellation AC. Cette décomposition est illustrée dans la FIGURE 3.21 par les flèches rouges hachurées (figure adaptée de (Grabisch and Perny, 1999)).
- l'approche CA « Comparer puis Agréger » : elle se caractérise par une décomposition du calcul $f(a_{C_1}, \dots, a_{C_M}, b_{C_1}, \dots, b_{C_M})$ sous la forme $\Omega(\Upsilon_1(a_{C_1}, b_{C_1}), \dots, \Upsilon_M(a_{C_M}, b_{C_M}))$, d'où l'appellation CA. Cette décomposition correspond aux flèches pleines dans la FIGURE 3.21,

Dans notre étude, l'approche AC est mise en œuvre. L'opérateur d'agrégation Ω et de comparaison Υ font respectivement l'objet des deux prochaines sections.

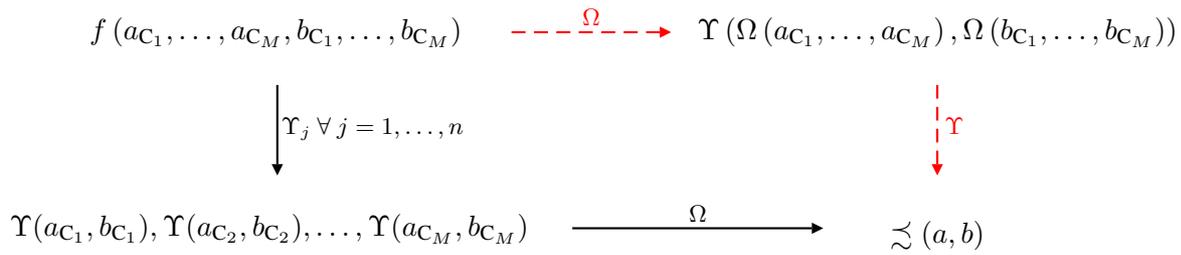


FIGURE 3.21: Deux modes opératoires pour l'agrégation multicritères : (Grabisch and Perny, 1999)

Agrégation multicritères (Ω)

La fonction d'agrégation Ω qui est mise en œuvre dans notre approche AC consiste à sommer les sous-ensembles flous relatifs à chaque alternative dans \tilde{H} (c.-à-d. sommer $\tilde{h}_x(l) \forall x = \{e, c, m, s\}$), ce qui donne lieu à une matrice floue notée \tilde{R} . Cette somme s'effectue également sur la base de leurs α -coupes (cf. Annexe IV).

⇒ Exemple : dans la FIGURE 3.22, nous illustrons la somme des quatre sous-ensembles flous relatifs à $T_{MatDef}\{3, 1\}$ ³⁷ et nous affichons dans la matrice \tilde{R} le résultat de cette somme pour les trois data items de l'étude.

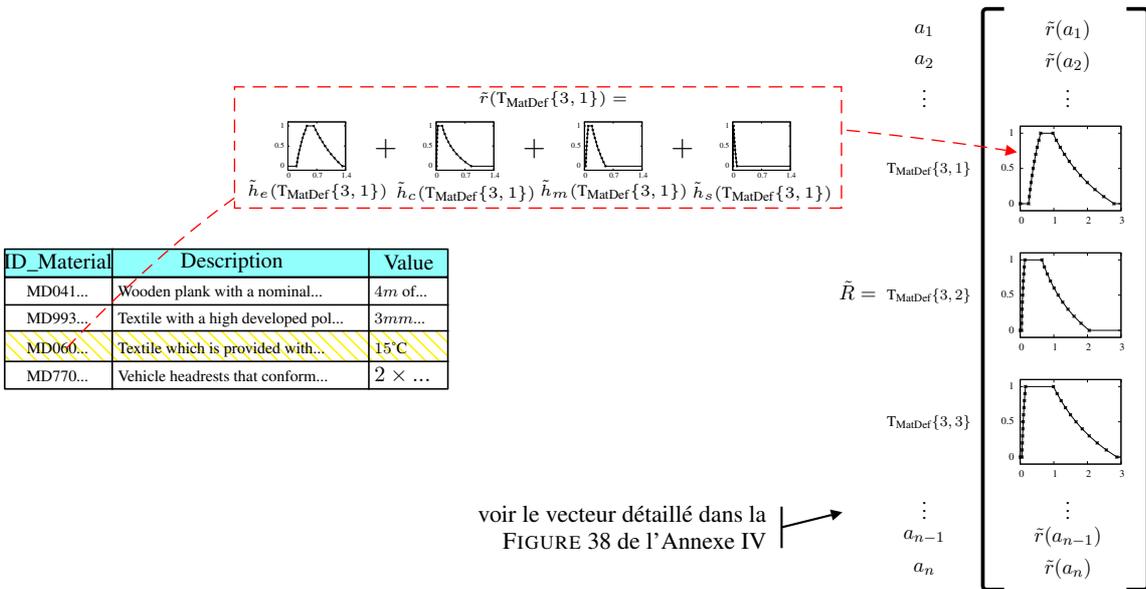


FIGURE 3.22: Matrice \tilde{R} contenant le sous-ensemble flou relatif à chaque alternative

Diverses méthodes permettant de réaliser la fonction de comparaison Υ peuvent alors être envisagées : la méthode basée sur les centres de gravité (*centroid method*) (Yager, 1980), sur les hauteurs (*height method*), sur les moyennes pondérées (*weighted average method*) (Dong and Wong, 1987), sur les moyennes des maximas (*mean-max method*), etc. La méthode mise en œuvre dans notre étude est

37. Le calcul $r^0(T_{MatDef}\{3, 1\})$ est détaillé dans l'ÉQUATION 31 de l'Annexe IV.

décrite dans la prochaine section.

Comparaison des alternatives (Υ) et classification

Plusieurs approches ont été proposées dans la littérature pour ordonner des intervalles flous en se basant sur leurs α -coupes. (Chen and Klein, 1997) proposent un indice basé sur les niveaux α en combinant une opération arithmétique floue (soustraction) avec un calcul d'aire. (Oussalah, 2002) suggère une procédure de défuzzification en calculant la moyenne des α -coupes, appelée également ALC (*Averaging Level Cuts*). Les méthodes les plus populaires d'après (Cochran and Chen, 2005) sont celles basées sur les centres de gravité (CoG - *Center of Gravity*) et sur les moyennes des maxima (MoM - *Mean of Maximum*)³⁸. Les auteurs précisent que la méthode CoG est plus précise et que la méthode MoM et particulièrement lorsqu'elles sont confrontées à des intervalles flous quelconques. Toutefois, la méthode MoM nécessite moins de calculs (c.-à-d. moins de temps) mais peut, dans certains cas, dénaturer fortement la forme originale. Dans notre étude, nous mettons en œuvre la méthode CoG afin de classer les alternatives³⁹.

Plusieurs méthodes CoG ont été proposées dans la littérature, la première étant celle de (Yager, 1980) qui utilise les coordonnées en abscisse comme indice de classement. Ensuite, de nombreuses autres méthodes CoG ont vu le jour (Murakami and Maeda, 1984; Cheng, 1998; Chu and Tsao, 2002; Chen and Chen, 2003; Chen and Chen, 2007; Wang and Lee, 2008), lesquelles utilisent d'autres indices pour classer les alternatives comme des distances, des aires, les coordonnées en abscisse et ordonnée, etc. Cependant, des études comparatives telles que (Ramli and Mohamad, 2009) ont montré qu'aucune de ces méthodes ne se détache réellement des autres car elles présentent toutes des avantages et des inconvénients pour le classement d'alternatives. Dans notre étude, nous implémentons l'approche initiale développée par (Yager, 1980), encore utilisée aujourd'hui comme « benchmark » par certains auteurs (Chen and Chen, 2007). Le calcul du CoG proposé par Yager est détaillé dans l'ÉQUATION 3.33 et retourne une valeur numérique correspondant à l'indice de classement (valeur se situant sur l'abscisse de la fonction d'appartenance comme cela est illustré sur la FIGURE 3.33).

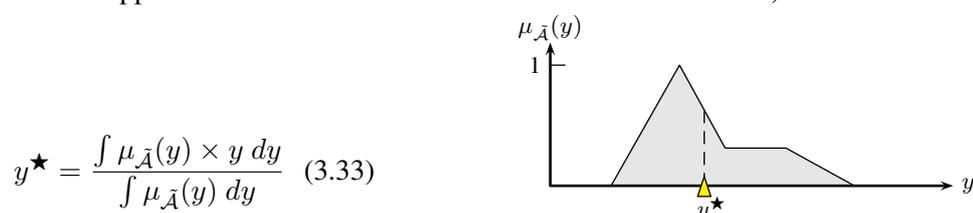


FIGURE 3.23: Méthode du CoG : (Yager, 1980)

Dans notre approche, les sous-ensembles flous sont représentés sur la base de leurs α -coupes, tel qu'illustré avec la flèche annotée $f(\mu_{\tilde{A}}(y))$ dans la FIGURE 3.24, où $\mu_{\tilde{A}}(y)$ est le sous-ensemble flou

38. la méthode MoM est principalement utilisée lorsque la fonction d'appartenance possède un noyau qui ne se réduit pas à une valeur unique.

39. nous étudierons d'autres approches d'ordonnement des alternatives dans de travaux ultérieurs à cette thèse afin de comparer les approches.

(aire grisée) et f est la fonction d'échantillonnage (exemple considérant 6 α -coupes). Par conséquent, l'ÉQUATION 3.33 se transforme en une intégrale sur les niveaux α et les intégrales sont une approximation supérieure, notée $f_+^{-1}(\mu_{\tilde{A}}(y))$ dans la FIGURE 3.24, ou inférieure, notée $f_-^{-1}(\mu_{\tilde{A}}(y))$. La valeur CoG calculée sur cette approximation pour chaque alternative est utilisée pour comparer une à une ces alternatives (plus le CoG est grand, plus le data item est pertinent à être stocké sur le produit).

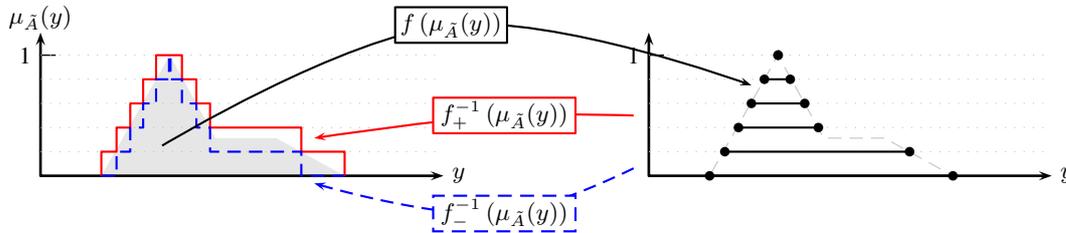


FIGURE 3.24: Calcul de COG par approximation inférieure $f_-^{-1}(\mu_{\tilde{A}}(y))$ ou supérieure $f_+^{-1}(\mu_{\tilde{A}}(y))$

⇒ Exemple : les calculs du CoG supérieur et inférieur de $T_{\text{MatDef}}\{3, 1\}$ (c.-à-d. les CoG du sous-ensemble flou noté $\tilde{r}(T_{\text{MatDef}}\{3, 1\})$) sont respectivement détaillés dans les FIGURES 39 et 40 de l'Annexe IV. Nous proposons de calculer le CoG supérieur pour l'ensemble des alternatives. Nous affichons dans la FIGURE 3.25 le CoG d'approximation supérieure de la fonction d'appartenance des trois data items de l'étude⁴⁰. Nous observons que le CoG de $T_{\text{MatDef}}\{3, 1\}$ ($y^\star = 1.089$) est plus grand que celui de $T_{\text{MatDef}}\{3, 2\}$ ($y^\star = 0.672$) et $T_{\text{MatDef}}\{3, 3\}$ ($y^\star = 0.966$). Par conséquent, $T_{\text{MatDef}}\{3, 1\}$ est le mieux classé comme illustré par le podium dans la FIGURE 3.25, et est stocké en priorité sur le produit. À titre indicatif, dans l'approche de Chan-Roddick, ce n'est pas $T_{\text{MatDef}}\{3, 1\}$ le plus prioritaire mais $T_{\text{MatDef}}\{3, 3\}$.

3.4 Conclusion de l'étape 2

Lorsque l'utilisateur/la machine désire stocker des informations sur le produit à un instant donné de son cycle de vie, il est tout d'abord nécessaire d'adapter l'environnement du produit/de l'utilisateur afin de procéder à la lecture et/ou l'écriture des informations sur ce dernier. Ceci correspond à l'étape 3 du processus de dissémination des informations, détaillé dans le Chapitre 4. Une fois l'architecture de terrain mise en place, le produit peut se présenter au point d'écriture, la machine relève dans un premier temps la référence portée par le produit afin d'identifier tous les « tuples produit » (à l'aide de l'ALGORITHME 1). Les data items composant ces tuples sont alors évalués soit via l'approche de (Chan and Roddick, 2003), soit via l'approche AHP-floue.

(Chan and Roddick, 2003) sont les premiers à s'intéresser à une approche qui tient compte de la sémantique des informations pour déterminer si ces informations sont pertinentes à être stockées sur un vecteur d'information mobile (p. ex., une base de données mobile ou un produit communicant). Le

40. la valeur du CoG, rappelons-le, se situe sur l'axe des abscisses.

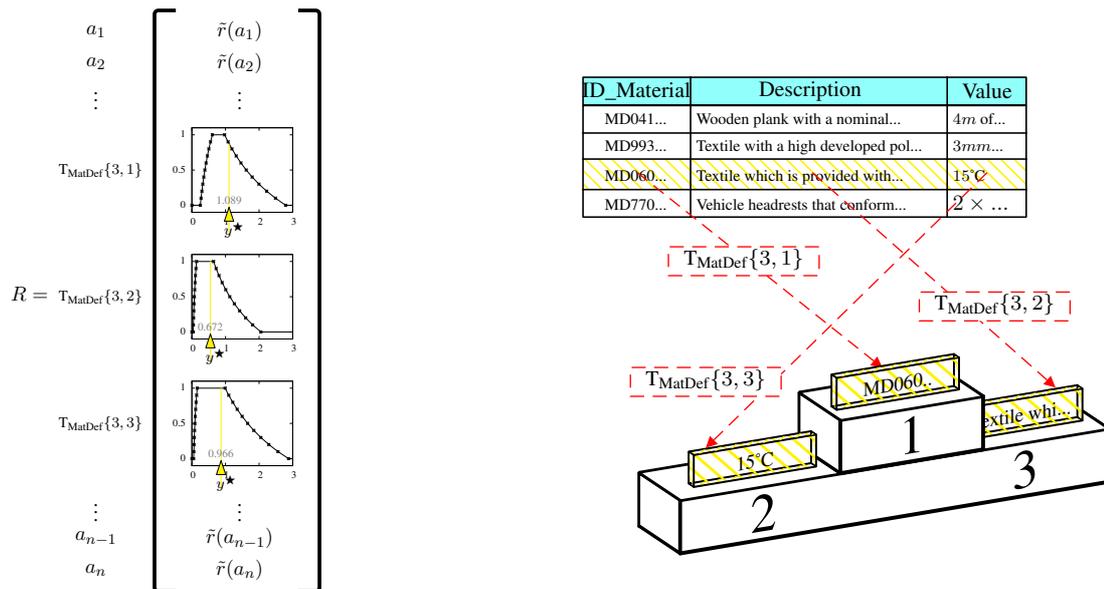


FIGURE 3.25: Calcul du CoG pour chaque alternative, puis classement des alternatives

modèle sur lequel repose leur approche permet de prendre en compte de multiples aspects/critères liés au contexte du produit (exigence des différents utilisateurs, aspect géographique et temporelle des informations, performance du système, etc.). Ce modèle est décrit en détails dans la section 3.2. Cependant, il présente certaines limites au vu des caractéristiques de notre contexte d'étude et principalement, du fait que l'utilisateur ne puisse pas gérer ses préférences concernant la taille des données à stocker sur le produit (sachant que nos produits sont fortement contraints en mémoire) et que le modèle ne permette pas de considérer plusieurs avis lors d'une même évaluation/d'un même choix. En effet, le produit est partagé par de nombreux acteurs qui, lors d'un même point d'écriture, peuvent vouloir stocker des informations variées, engendrant alors des points de vue différents concernant les choix/évaluations. Pour pallier ces limites, nous avons proposé dans la section 3.3 une approche reprenant l'idée centrale de Chan et Roddick, qui est de développer un modèle permettant de gérer le problème de décision multi-critères. Notre approche repose sur l'approche AHP-floue. AHP permet de gérer le problème MCDM et la théorie des ensembles flous permet de gérer l'agrégation des avis des experts, ainsi que l'incertitude relative à certains choix.

Les outils matériels et logiciels relatifs à l'étape 3 du processus de dissémination des informations sont présentés dans le Chapitre 4.

Chapitre 4

Étape 3 : Stockage et relecture des informations sur la matière communicante

Sommaire

4.1 Conception de la matière communicante	94
4.2 Conception de l'architecture de communication	95
4.2.1 Deux types d'architecture pour lire/écrire des données sur la matière	95
4.2.2 Dissémination des informations au sein de la matière communicante	97
4.3 Outillage de l'expérimentation	100
4.4 Conclusion de l'étape 3	102

L'étape 2 du processus de dissémination des informations a permis d'identifier et d'évaluer la pertinence des « informations produit » à être (ou ne pas être) stockées sur la matière. Il est désormais nécessaire dans l'étape 3 du processus de concevoir la matière communicante ainsi que l'architecture de communication appropriée pour lire/écrire des données sur celle-ci. Les problématiques traitées dans le cadre de cette étape sont, pour la plupart, d'ordre technologique et font principalement référence aux problématiques identifiées dans les groupes 1 et 2 de la FIGURE 27 de l'Annexe I (groupes intitulés respectivement « conception de la matière » et « architecture de communication »). On y retrouve les problèmes liés au choix de la technologie à intégrer dans la matière, au procédé de conception et d'intégration de la technologie, au protocole de communication, aux interfaces Homme-Matière, etc. Durant cette thèse, nous avons mis au point un premier prototype de matière communicante, ainsi que son infrastructure matérielle et logicielle. L'idée initiale était simplement de valider la faisabilité du concept de matière communicante. Pour ce faire, nous avons mené une collaboration étroite avec le CETELOR (Centre de TExtile LORrain) afin de concevoir un prototype de textile communicant, lequel a été présenté et publié dans (Kubler et al., 2011*b*). Les études menées pour la conception de ce textile, ainsi que les résultats, sont détaillés dans l'Annexe V et présentés brièvement dans la section 4.1. Les outils permettant de disséminer des informations au sein de la matière ont quant à eux été présentés et publiés dans (Kubler, Derigent, Thomas and Rondeau, 2012*b*) et font l'objet des sections 4.2 et 4.3.

4.1 Conception de la matière communicante

L'idée soutenue dans cette thèse pour concrétiser le concept de matière intrinsèquement et intégralement communicante est de disséminer en masse des tags RFID au sein de la matière, car il s'agit d'une technologie bon marché dont il est raisonnable d'envisager une évolution du couple coût/performance des tags RFID analogue à celle des mémoires des ordinateurs. Par exemple, le prix objectif visé par le standard EPC Classe 1 est de 0,05€/tag (ALCOM Consulting et Newton Vaureal Consulting, 2007). Cependant, l'instrumentation de la matière via des puces RFID doit être étudiée du fait des nombreuses technologies existantes, de leurs modes de fonctionnement et de leurs caractéristiques. Les environnements auxquels le produit est confronté durant son cycle de vie (eau, métal, interférence, rapidité des opérations, etc.) doivent être aussi étudiés. Lors de cette thèse, un premier prototype de textile communicant a été conçu (voir image sur la FIGURE 4.1), qui peut être répertorié dans la catégorie des *e*-textiles. Le lecteur peut trouver l'étude complète de la conception du textile communicant dans l'Annexe V ou dans (Kubler et al., 2011b). Notons simplement que deux technologies RFID distinctes ont été utilisées pour instrumenter le textile :

- le μ tag RFID mis au point par Hitachi : il s'agit, à l'heure actuelle, de la plus petite puce sur le marché avec une taille de $0.4 \times 0.4 \times 0.1mm$ (Usami et al., 2007). Cependant, cette technologie reste limitée car il est impossible de stocker des informations sur la puce et elle n'implémente pas de protocole anti-collision,
- le tag RFID Omron V720-D52P03 : ce tag est plus imposant qu'un μ tag (diamètre $\approx 15mm$) et possède une EEPROM de 64 octets ainsi qu'un protocole d'anti-collision.

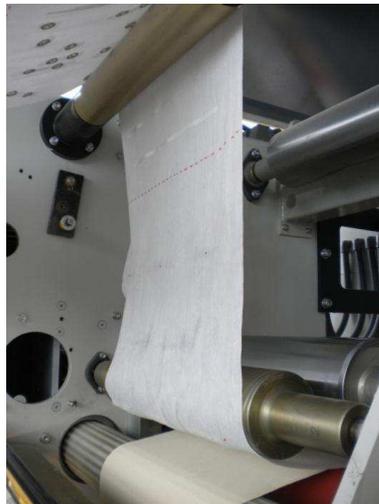


FIGURE 4.1: Processus d'intégration des tags RFID dans le textile

La section suivante présente les outils nécessaires pour communiquer avec la matière communicante, stocker et extraire de l'information sur l'ensemble de la matière, ou encore localiser l'information sur celle-ci.

4.2 Conception de l'architecture de communication

Dans notre étude, l'information devant être stockée sur la matière communicante correspond à l'ensemble des data items qui ont été jugés pertinents (via l'étape 2) à un instant précis du cycle de vie de la matière. Puisque la majorité des technologies RFID est contrainte en mémoire, il est par conséquent nécessaire de disséminer/d'éclater l'information sur plusieurs tags. Pour éclater ces data items, il est nécessaire de mettre en place une architecture de communication particulière et de développer les outils logiciels appropriés.

Deux architectures matérielles sont présentées dans la section 4.2.1, à savoir : i) une architecture permettant d'écrire les informations à la volée sur la matière, sans se soucier de leur localisation géographique dans la matière ii) une architecture permettant de localiser les données sur la matière, aussi bien durant une opération d'écriture du produit, que durant une opération de lecture. Ces deux architectures reposent sur un en-tête protocolaire particulier qui permet d'éclater les data items sur l'ensemble des tags présents dans la matière et par la suite, de les reconstruire. Cet en-tête est détaillé dans la section 4.2.2.

4.2.1 Deux types d'architecture pour lire/écrire des données sur la matière

L'architecture permettant d'écrire des data items à la volée sur la matière communicante est présentée dans la FIGURE 4.2. Elle nécessite :

- une matière communicante capable de stocker des informations (dans notre étude, ceci se traduit par une technologie RFID de type R/W - *Read/Write*),
- un lecteur RFID permettant de dialoguer avec les tags présents dans la matière,
- un accès à la base de données :
 - durant une opération d'écriture du produit : accès à la liste des data items devant être évalués et stockés sur le produit,
 - durant une opération de lecture du produit : il n'est pas forcément nécessaire d'accéder à la base de données, excepté si l'utilisateur désire mettre à jour les informations ou compléter les informations portées par le produit (*cf.* FIGURE 2.10(c)),

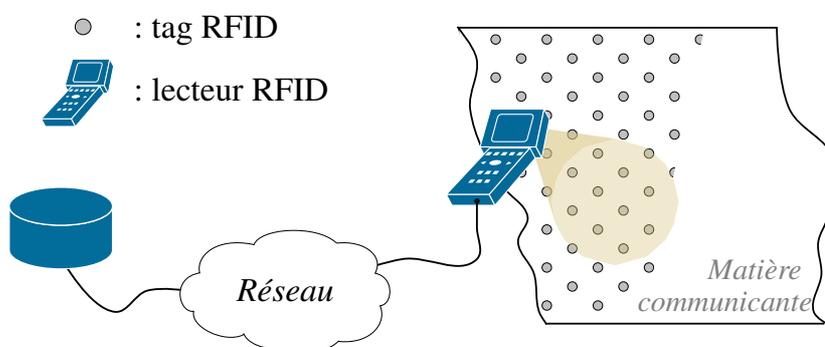


FIGURE 4.2: Architecture permettant d'écrire les données à la volée sur la matière communicante

La seconde architecture est plus complexe à mettre en œuvre puisqu'il s'agit de localiser les données sur la matière communicante. Il faut être capable de représenter virtuellement la matière communicante et de calculer pour chaque tag RFID son emplacement dans la matière. Dans notre étude, la localisation est uniquement gérée de manière 2D. L'architecture est présentée dans la FIGURE 4.3 et nécessite :

- une matière communicante capable de stocker des informations. À noter qu'avec une technologie RFID RO (*Read Only*), il est possible d'utiliser cette architecture afin de savoir où les tags se situent dans la matière et ainsi, d'associer un pointeur pour accéder à une information déportée dans une base de données,
- une rampe de lecteurs RFID, alignée au dessus de la matière communicante,
- une modélisation des zones de couverture¹ des lecteurs RFID sous forme cylindrique (*cf.* FIGURE 4.3),
- une solution permettant de faire défiler la matière sous la rampe de lecteur RFID de manière constante et continue (p. ex., un tapis roulant),
- un accès à la base de données :
 - durant une opération d'écriture du produit : accès à la liste des data items devant être évalués et stockés sur le produit,
 - durant une opération de lecture du produit : pas forcément nécessaire d'accéder à la base de données, excepté si nous désirons mettre à jour les informations ou compléter les informations portées par le produit,

Sur la base de cette architecture, il est alors possible de calculer la position de chaque tag RFID dans la matière via la méthode de la corde (Wikipedia, 2003), basée sur l'hypothèse que la zone de couverture est modélisée par un cylindre². L'Annexe VI présente les différentes étapes de calcul de cette méthode.

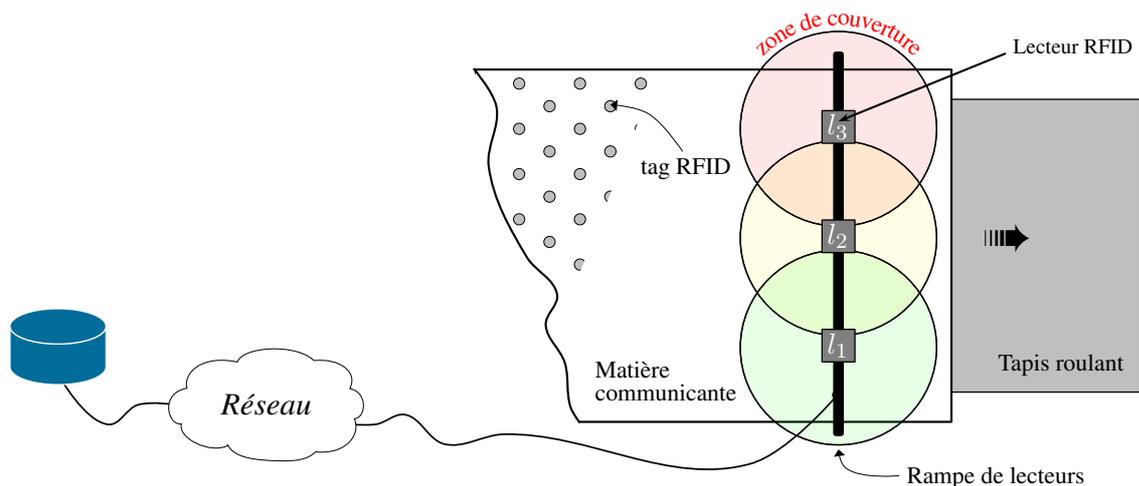


FIGURE 4.3: Architecture requise pour localiser les données au sein de la matière communicante

1. zone dans laquelle un lecteur et un tag RFID peuvent communiquer.

2. l'étude menée dans l'Annexe V décrit comment modéliser la zone de couverture sous forme cylindrique.

4.2.2 Dissémination des informations au sein de la matière communicante

Cette section détaille l'en-tête protocolaire utilisé pour disséminer les data items entre une multitude de tags RFID et permettant, à l'inverse, de les reconstruire. Cet en-tête est utilisé quelle que soit l'architecture mise en place (écriture à la volée ou localisation des informations). L'en-tête respecte le standard RFID ISO/IEC 18000-1 (ISO/IEC, 2004), lequel repose sur 3 des 7 couches du modèle OSI comme illustré sur la FIGURE 4.4. La couche 1 correspond à la partie physique de la technologie (c.-à-d. l'antenne, la partie analogique). La couche 2 gère les protocoles d'accès au réseau physique et particulièrement les mécanismes d'anti-collision. La couche 7 gère quant à elle les services d'accès aux données applicatives.

Un tag RFID peut contenir plus ou moins de données en fonction de la technologie et par conséquent, un data item peut requérir plus d'espace mémoire que celle disponible dans un unique tag. Le data item est donc découpé en trames, chaque trame étant associée à un et un seul tag RFID. Pour supporter ce découpage, nous proposons un en-tête protocolaire défini au niveau 7 du modèle OSI comme le témoigne la FIGURE 4.4 (partie grisée). Cet en-tête se compose de 7 champs détaillés ci-dessous, dont les 6 premiers sont utilisés pour reconstruire les data items et le dernier est destiné à stocker le « contenu » du data item³ :

1. *Protocol* (8 bits) : entier entre 0 et 255, utilisé pour savoir quels champs composent le paquet (définit le type d'en-tête utilisé, un nouvel en-tête pourrait être imaginé). La valeur 255 est définie dans notre application et fait référence à la structure de l'en-tête définie dans la FIGURE 4.4 (spécifique à notre application),
2. *Size* (8 bits) : entier entre 0 et 255 renseignant sur la longueur de l'en-tête en octets, en y incluant la longueur du champ « Data » (7^{ème} champ),
3. *Seq_Num* (8 bits) : entier entre 0 et 255. Il fournit le numéro de séquence de la trame courante (1 trame/tag). Le numéro de séquence est utilisé pour déterminer l'ordre dans lequel les différentes trames ont été stockées/écrites sur les tags RFID. Ce champ est nécessaire pour la reconstruction des data items,
4. *Prev_Num* (8 bits) : entier entre 0 et 255. Il fournit le numéro de séquence de la trame précédemment écrite, relative à la même opération d'écriture. La notion « même opération d'écriture » indique l'instant auquel la donnée a été stockée sur le produit (voir point suivant),
5. *ID_Phase* (64 bits) : entier entre 0 et $2^{64} - 1$. Indique la référence de l'opération d'écriture. La date d'écriture du produit est actuellement utilisée comme référence⁴. Plusieurs trames peuvent posséder la même référence *ID_Phase*, mais le couple *ID_Phase/Seq_Num* est unique. Ce champ sera utilisé pour la programmation de nouveaux services tels que : i) des solutions de redondances d'une même information (une même donnée sera différenciée par la référence de l'opération d'écriture, cf. le champ *ID_Phase*), ii) des solutions gérant l'inconsistance des informations (il sera possible

3. le lecteur peut se référer à la FIGURE 2.3 pour la terminologie employée : « contenu d'un data item ».

4. nous utilisons le *timestamp* qui représente le nombre de secondes qui s'est écoulé depuis le 1^{er} janvier 1970.

d'identifier les informations périmées), iii) des solutions de traçabilité (p. ex., maintenir l'historique d'une donnée particulière),

6. *Checksum* (32 bits) : entier entre 0 et $2^{32} - 1$. Ce champ est utilisé pour le contrôle d'erreur de données,
7. *Data* ($n - 128$ bits) : le contenu du data item se trouve dans ce champ, avec n le nombre de bits inscriptibles dans le tag RFID. Il s'agit d'une chaîne de valeur qui peut parfois être trop longue pour être stockée intégralement dans un unique tag (fonction de l'espace mémoire disponible dans celui-ci). À noter qu'un index est ajouté avant d'inscrire le contenu de chaque data item dans le but de localiser ce dernier au sein de la base de données. Cet index se compose du nom de la table, du nom de l'attribut et de la valeur de sa clé primaire. Il est codé de la manière suivante : `NomTable.NomAttribut.ValeurClePrimaire`. Par exemple, lors de l'écriture de `TMatDef{3, 3}`, l'index suivant est ajouté : `MaterialDefinition.Value.MD060111223=` (cf. la FIGURE 3.3). Ensuite, le contenu du data item est ajouté, à savoir : `15°C`.

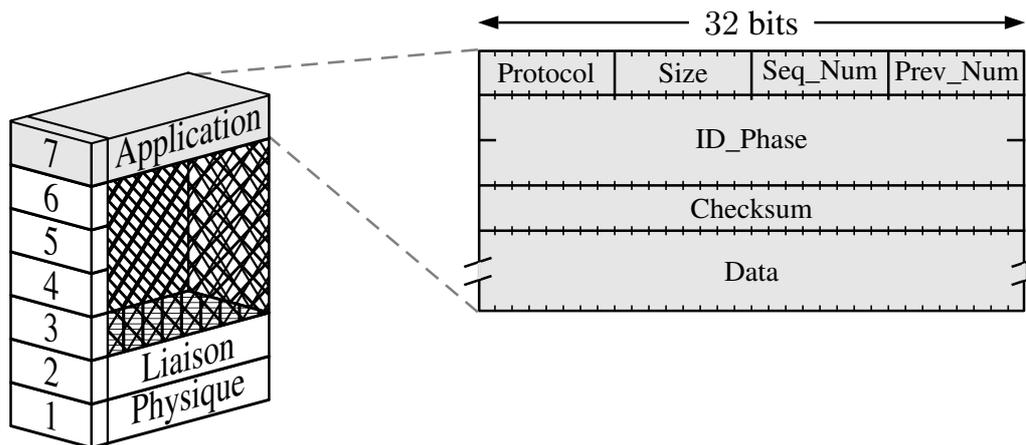


FIGURE 4.4: Couches protocolaires de la technologie RFID et structure de l'entête *splitting protocol*

Il est important de souligner qu'en fonction des data items stockés sur le produit, certaines requêtes peuvent être inexécutables. Toutefois, certaines méthodes peuvent être déployées afin de savoir a priori (c.-à-d., avant que les data items soient écrits sur la matière) si les requêtes peuvent être ou non exécutées. Par exemple, Chan et Roddick (Chan and Roddick, 2003) suggèrent d'utiliser la transformation de requêtes en *bitmaps* pour déterminer si oui ou non la requête peut être exécutée en fonction des data items destinés à être stockés sur la base de données mobile. Une telle méthode permettrait de reclasser et d'optimiser la liste des data items qui a été initialement fournie dans l'étape 2 du processus de dissémination des informations. Ainsi, nous serions en mesure de répondre à un maximum de requêtes. Néanmoins, cela sous-entend que nous ayons connaissance a priori des requêtes effectuées sur le produit, durant son cycle de vie. Ceci peut néanmoins être envisagé dans certaines situations comme par exemple, pour un processus de fabrication connu.

Un exemple est détaillé ci-dessous pour illustrer et comprendre le fonctionnement de l'en-tête protocolaire proposé.

⇒ Exemple : nous désirons dans cet exemple stocker le data item $T_{\text{MatDef}}\{3, 3\}$ sur le textile communicante. Nous supposons que les tags RFID intégrés dans la matière sont de type R/W avec une mémoire inscriptible de 30 octets. Chaque tag requis pour stocker $T_{\text{MatDef}}\{3, 3\}$ a obligatoirement 128 bits de réquisitionner pour les six premiers champs de l'en-tête. Les données à écrire se composent à la fois de l'index et du contenu du data item, soit la chaîne de caractères donnée dans l'ÉQUATION 4.1.

$$\underbrace{\text{MaterialDefinition.Value.MD060111223}}_{\text{index}} = \underbrace{15^{\circ}\text{C}}_{\text{contenu}} \quad (4.1)$$

Au total, la chaîne se compose de 41 caractères, nécessitant donc 41 octets pour être stockée (1 caractère ASCII étant codé sur 1 octet) en complément des 128 bits. Par conséquent, 3 tags RFID sont requis pour disséminer $T_{\text{MatDef}}\{3, 3\}$ comme calculé ci-dessous :

$$\left[\frac{\overbrace{41 \times 8}^{\text{index+contenu (bits)}}}{\underbrace{30 \times 8}_{\text{mémoire tag (bits)}} - \underbrace{128}_{\text{en-tête (bits)}}} \right] = 3$$

La FIGURE 4.5 illustre les trois tags requis pour disséminer $T_{\text{MatDef}}\{3, 3\}$, à savoir, les tags a , b et c . Ces trois tags disposent :

- de la même valeur dans le champ *Protocol*, $(255)_{10} = (FF)_{16}$, car nous utilisons toujours la même structure d'en-tête (celle définie et décrite précédemment), identifiée par la valeur $(255)_{10}$,
- de la même valeur dans le champ *Size*, $(30)_{10} = (1E)_{16}$, car les tags possèdent une mémoire de 30 octets inscriptible. Si nous regardons le contenu mémoire du tag a , nous constatons qu'il y a 30 octets utilisables ($32 \times 7.5 = 240 \text{ bits} = 30 \text{ octets}$),
- d'une valeur *Seq_Num* unique, renseignant le numéro de séquence du découpage. Par exemple, la première partie de la chaîne de caractères est stockée dans le tag a , $Seq_Num = (1)_{10} = (01)_{16}$, la seconde dans le tag b , $Seq_Num = (2)_{10} = (02)_{16}$, et la dernière dans le tag c , $Seq_Num = (03)_{16}$.
- de la valeur *Prev_Num* qui renseigne le numéro de séquence de la trame précédemment écrite. Par exemple, la valeur *Prev_Num* du tag b est égale à $(01)_{16}$ (c.-à-d. le *Seq_Num* du tag a) et la valeur *Prev_Num* du tag c est égale à $(02)_{16}$ (c.-à-d. le *Seq_Num* du tag b). La valeur *Seq_Num* et *Prev_Num* du tag a sont quant à elles identiques, $(01)_{16}$, car il s'agit du premier tag écrit.
- de la même valeur dans le champ *ID_Phase* car il s'agit de la même opération d'écriture. La valeur $(1344940753)_{10} = (00000000502a2ad1)_{16}$ correspond au *timestamp* qui se traduit par la date : 14/08/2012 à 13 :05 :52.
- le calcul du checksum et sa valeur ne sont pas présentés dans cet exemple,
- les valeurs du dernier champs *Data* relatives aux trois tags correspondent aux morceaux de la chaîne de caractères fournie dans l'ÉQUATION 4.1, codés en ASCII. Nous remarquons au vu du

contenu mémoire de chaque tag dans la FIGURE 4.5, que l'index consomme les espaces mémoires des deux premiers tags (tag *a* et *b*) et que le contenu de $T_{MatDef}\{3, 3\}$ est stocké dans le troisième tag⁵ (tag *c*).

Finalement, les informations peuvent être recomposées grâce à l'en-tête et à l'index et, facultativement, être réinjectées dans la base de données puisque nous disposons du nom de la table, de l'attribut et de la clé primaire de chaque data item. Bien entendu, si un tag est illisible, le ou les data items qui possèdent une partie codé dans ce tag ne pourront pas être reconstruits. Toutefois, des solutions de redondance des informations ou autres peuvent être imaginées⁶.

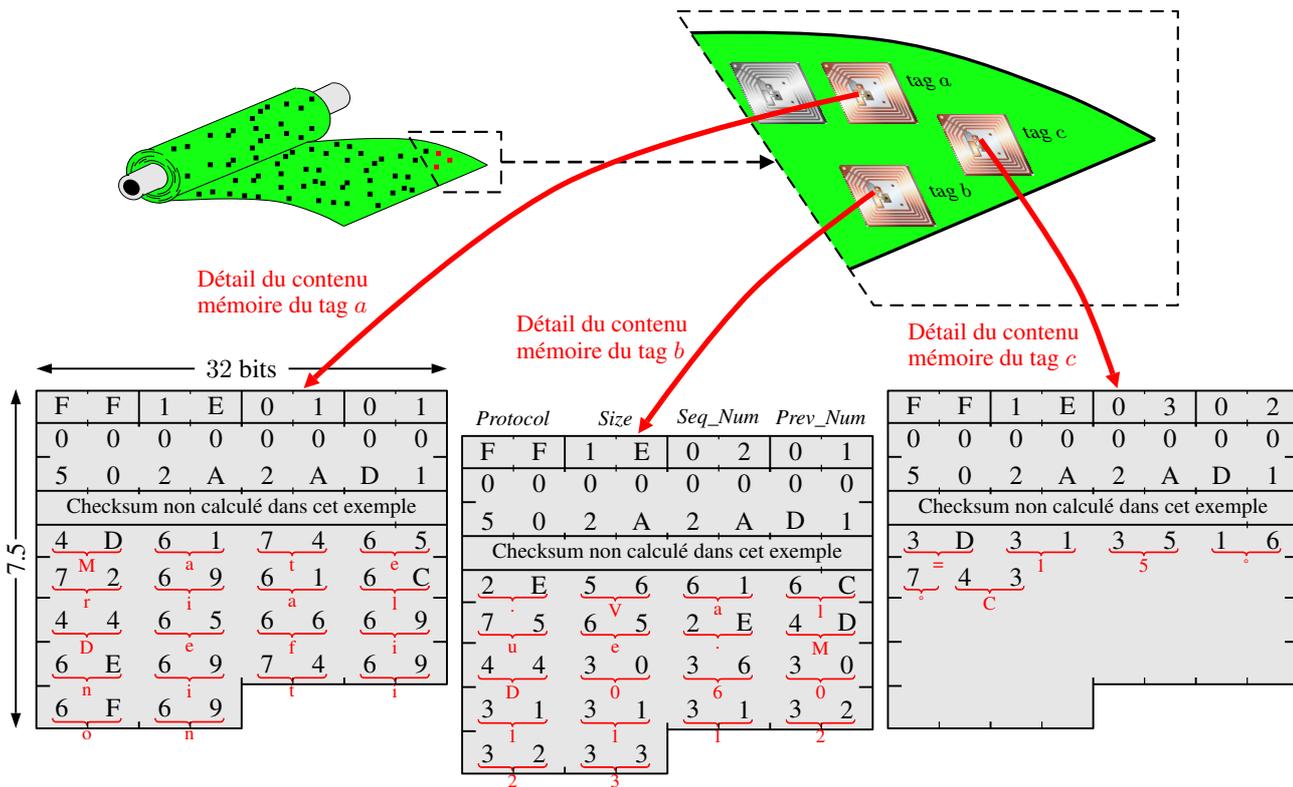


FIGURE 4.5: Illustration de la dissémination dans la matière de $T_{MatDef}\{3, 3\}$ via le *splitting protocol*

4.3 Outillage de l'expérimentation

Dans notre processus de dissémination des informations, les opérations d'écriture et de lecture coexistent (cf. FIGURE 2.10) et les données peuvent être localisées ou non (cf. FIGURES 4.2 et 4.3). Par conséquent, quatre types d'opérations/d'infrastructures sont possibles comme l'illustre la FIGURE 4.6. Les différents outils requis par ces infrastructures sont précisés ci-dessous⁷ :

5. notons que chaque caractère est codé sur 1 octet, excepté le caractère « ° » qui est un symbole particulier codé sur 12 bits, soit 1 octet et demi (cf. le contenu mémoire du tag *c* dans la FIGURE 4.5).
 6. aucune solution n'est proposée dans cette thèse (cela fait partie des perspectives de la thèse).
 7. nous spécifions également sur quels outils informatiques ils ont été développé (JAVA, etc.)

1. opération d'écriture sans localisation des données : il s'agit d'écrire les données à la volée sur la matière comme illustré avec l'opérateur de la FIGURE 4.6 (Écriture/Données non localisées). L'architecture matérielle et logicielle se compose :
 - d'un lecteur RFID unique,
 - d'une application JAVA[®] couplée avec MATLAB[®], nommée « Java/Matlab » qui assure l'extraction des informations de la base de données et qui réalise le calcul de leur pertinence (repose sur les approches Chan-Roddick et AHP-Flou⁸). Un accès à la base de données est bien évidemment requis,
 - d'une application JAVA, nommée « Java Split », qui assure l'écriture et la dissémination des data items sur la matière. Cette application repose sur l'en-tête protocolaire proposé dans la section 4.2.2,
2. opération de lecture sans localisation des données : il s'agit de lire à la volée les données portées par la matière comme illustré dans la FIGURE 4.6 (Lecture/Données non localisées). L'architecture se compose :
 - d'un lecteur RFID unique,
 - de l'application « Java Split » pour reconstituer les informations,
 - facultativement, d'un accès à la base de données si les informations extraites du produit doivent être complétées par des informations de la base de données ou, si les informations extraites du produit doivent être mises à jour dans la base de données,
3. opération d'écriture avec localisation des données : la solution consiste à positionner successivement deux rampes comme illustré sur la FIGURE 4.6 (Écriture/Données localisées). La première permet de lire la matière et de récupérer l'ensemble des événements générés par les lecteurs. La seconde permet d'écrire les données. Avec cette installation, il n'est pas nécessaire de calculer de façon immédiate les positions des tags. Le calcul peut se faire durant un temps Δt (fonction de la distance séparant les rampes et de la vitesse du tapis) comme illustré sur la FIGURE 4.6. L'architecture se compose :
 - de deux rampes de lecteurs RFID,
 - d'une application JAVA, nommée « Java Loc », qui permet de localiser les tags dans la matière (repose sur la méthode de la corde décrite dans l'Annexe VI),
 - de l'application « Java/Matlab » pour calculer la pertinence des informations issues de la base de données,
 - de l'application « Java Split » pour éclater les données.
4. opération de lecture avec localisation des données sur la matière : nous utilisons une infrastructure semblable à la précédente, sauf qu'elle ne nécessite pas le calcul de pertinence des informations

8. à noter que les spécifications peuvent être réalisées par l'opérateur en charge de l'écriture du produit, à l'instant même où il réalise l'écriture (interaction via une interface graphique), ou elles peuvent être réalisées en amont par une tierce personne comme illustré dans la FIGURE 4.6.

et qu'une seule rampe suffit, comme illustré dans la FIGURE 4.6 (Lecture/Données localisées).

L'architecture requise se compose :

- d'une rampe de lecteurs RFID,
- de l'application « Java Loc » pour localiser les informations,
- de l'application « Java Split » pour reconstituer les données,
- facultativement, d'un accès à la base de données (même raison que précédemment).

4.4 Conclusion de l'étape 3

Nous avons désormais à notre disposition tous les outils qui permettent de mettre en place physiquement le processus de dissémination des informations dans le cycle de vie d'une matière communicante. Les outils que nous proposons peuvent être utilisés avec des matières :

- conçues dans tout type de matériau (bois, textile, plastique, etc.) : l'étude de la conception de la matière communicante reste néanmoins à être réalisée,
- de formes quelconques : ceci est vrai lorsque les informations sont inscrites à la volée sur la matière. Cependant, si nous désirons localiser les informations sur la matière, notre approche requiert une matière à surface plane.

Le prochain chapitre présente un cas d'étude dans lequel nous voulons stocker sur un textile communicant des informations sensibles à son contexte d'usage et ce, à différents instants de son cycle de vie. Pour cela, nous implémentons les trois étapes du processus de dissémination des informations.

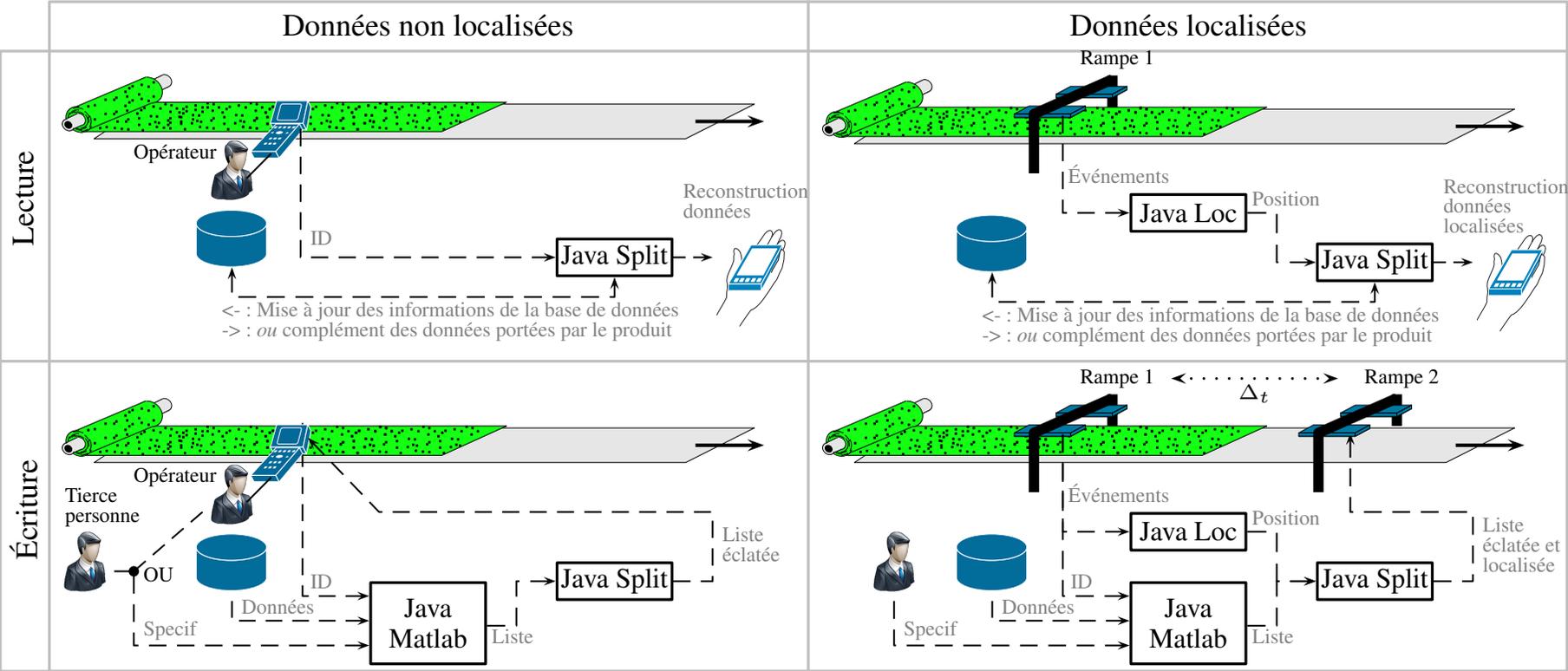


FIGURE 4.6: Les 4 types d'architectures possibles de mise en œuvre du processus de dissémination des données

Chapitre 5

Application du processus de dissémination des informations au cas d'un textile communicant

Sommaire

5.1 Cycle de vie du textile communicant	106
5.1.1 Définition des points d'écriture et de lecture dans le PLC	107
5.2 Exécution des phases d'écriture et de lecture	109
5.2.1 Principe de comparaison des approches Chan-Roddick / AHP-Floue	109
5.2.2 Point d'écriture n°1	110
5.2.3 Point de lecture n°1 : activité « Découpe »	123
5.2.4 Point d'écriture n°2	126
5.3 Synthèse et conclusion de l'étude	131
5.3.1 Conclusions des études comparatives de l'Annexe VII	131
5.3.2 Tableau de synthèse	133

Dans ce chapitre, le processus de dissémination des informations établi précédemment est mis en œuvre durant le cycle de vie d'un textile communicant. Le but est de stocker sur ce dernier les informations pertinentes et utiles aux différents acteurs du PLC (p. ex., confectionneur, brodeur, utilisateur, recycleur, etc.).

La section 5.1 donne un aperçu du cycle de vie « simplifié » du textile considéré dans notre cas d'étude. Les avantages à mettre en œuvre un textile communicant sont discutés et les différentes opérations d'écriture et de lecture du produit y sont présentées.

La section 5.2 détaille les opérations d'écriture et de lecture définies sur le cycle de vie de ce textile. Concernant les actions d'écriture, les ajustements réalisés par les experts et les utilisateurs concernés par les différents critères des approches Chan-Roddick et AHP-Floue sont décrits. Nous comparons également les résultats exprimés selon la pertinence des data items, qui découlent de ces deux approches. Nous montrons qu'il est alors possible sur la base des architectures matérielles et logicielles de disséminer l'ensemble ou une partie des data items sur le textile. Pour les phases de lecture, nous présentons les techniques utilisées pour extraire les données portées par le textile communicant et pour reconstruire les

« tuples produits » (via un logiciel dédié à l'application). Nous présentons également le principe mis en œuvre pour localiser les tags RFID dans le textile communicant.

Enfin, les avantages et inconvénients des deux approches d'évaluation (Chan-Roddick et AHP-Floue) sont analysés dans la section 5.3.

5.1 Cycle de vie du textile communicant

Dans ce cas d'application, nous nous intéressons à une fabrique de textile qui produit des bobines utilisées pour la conception d'assises de sièges pour l'automobile, de vêtements, de draps, etc. Le textile passe par une série d'étapes de transformation tout au long de son processus de production. Plusieurs membres de la chaîne logistique ont noté des points à améliorer concernant la gestion du processus de production comme :

- des problèmes de traçabilité : plusieurs cas de contrefaçon de vêtements, de draps se sont produits ces dernières années. De plus, plusieurs cas de tags RFID défaillants ou de fausses lectures ont été relevés,
- des problèmes de transformation : les informations liées au textile sont parfois perdues/inaccessibles lorsque ce dernier est transformé (p. ex. lors d'une opération de découpe). En effet, la référence produit ne se duplique pas forcément sur tous les morceaux après la découpe du textile, ou alors, cela demande une intervention manuelle,
- des problèmes d'interopérabilité : plusieurs acteurs de la chaîne logistique pensent qu'il serait plus judicieux de lier les informations au textile lui-même afin d'améliorer l'interopérabilité entre les systèmes d'information des différents acteurs,
- des problèmes d'agrégation des informations : durant sa vie, le textile est découpé et assemblé avec d'autres produits/matériaux (opération de couture, de collage). Si nous utilisons des textiles communicants, ils peuvent alors agir comme des conteneurs intelligents (*cf.* définition et modèle de classification dans la FIGURE 1.6).

En conséquence, les membres de la chaîne logistique pensent qu'il serait bénéfique de reconsidérer le système de traçabilité, de gestion et d'interopérabilité lié au textile. Ils proposent pour cela de remplacer les textiles classiques par des textiles communicants. Ainsi, les informations liées au produit pourront leur être attachées, quelle que soit la nature du produit (p. ex. pièces de textiles, vêtements, sofa, assises de chaises, etc.). Bien évidemment, le processus de dissémination des informations est mis en place afin de sélectionner les informations sensibles aux contextes d'utilisation pour les stocker sur le textile. Il est donc nécessaire d'identifier les points et les instants d'écriture et de lecture dans le but de mettre en place les infrastructures matérielles et logicielles adéquates (présentées dans la FIGURE 4.6).

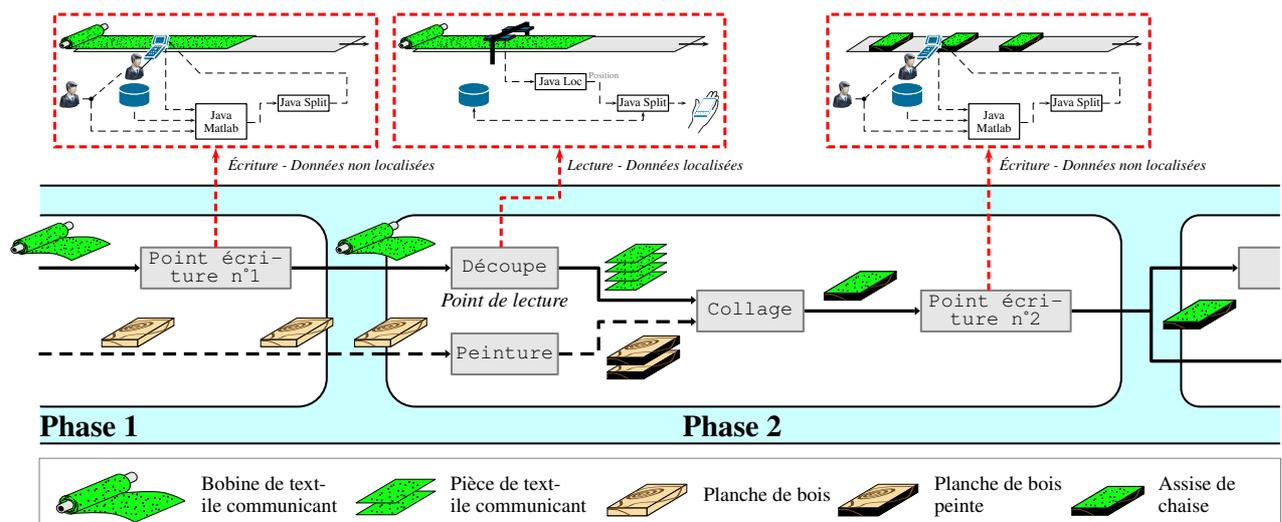


FIGURE 5.1: Tronçon simplifié du cycle de vie du textile et intégration des opérations d'écriture/lecture du processus de dissémination des données

5.1.1 Définition des points d'écriture et de lecture dans le PLC

Dans notre étude, nous simplifions la partie initiale du cycle de vie des bobines de textile communicant. La fabrication du textile communicant ne fait pas partie de notre étude. C'est sa transformation qui nous intéresse en analysant les phases d'écriture et de lecture du processus de dissémination. La FIGURE 5.1 offre un aperçu du tronçon simplifié du cycle de vie du textile. Nous considérons dans notre application que les bobines de textile communicant ont déjà été conçues dans la phase 1 du cycle de vie. Nous utilisons le textile présenté dans la section 4.1 (voir FIGURE 4.1). Ces bobines sont découpées dans la phase 2 (cf. FIGURE 5.1), avant que les pièces obtenues soient assemblées par collage sur des planches de bois, lesquelles ont été peintes préalablement. L'opération de collage donne naissance à des assises de siège, qui sont destinées à des activités/opérations dans la phase 3 du PLC. Nous supposons que deux points d'écriture ont été définis dans ce tronçon de cycle de vie, à savoir à la fin de la phase 1 et à la fin de la phase 2. Les opérateurs de ces deux points d'écriture désirent tous deux écrire à la volée les informations sur la matière, ce qui fait donc appel à l'architecture « Ecriture/Données non localisées » de la FIGURE 4.6. Il est à noter que l'opérateur concerné par le point d'écriture n°1, en plus de stocker des informations pertinentes issues de la base de données, écrit des informations liées aux défauts qu'il détecte sur cette bobine (écriture à la volée dans la zone du défaut concerné : trou, tâche, etc.). L'opérateur de l'activité « Découpe » désire extraire les informations qui ont été stockées sur les bobines communicantes par le précédent acteur et désire localiser ces données sur le textile, afin de pouvoir localiser les défauts relevés par l'opérateur du point d'écriture n°1. Ceci fait appel à l'architecture « Lecture/Données localisées » comme illustré sur la FIGURE 4.6. Le textile communicant est utilisé de manière classique dans l'opération « Collage » puisque le manutentionnaire ne désire pas accéder aux informations portées par le produit, ni en stocker. Autre remarque, les planches de bois ne sont pas faites de matière communi-

cante. Cependant, à partir de l'instant où les pièces de textile communicant sont collées sur les planches (créant des assises de chaises), ces assises sont alors capables d'embarquer des informations liées à la fois au textile, à la planche de bois et à l'assise résultante. C'est en ce sens que le textile joue le rôle de « conteneur intelligent ».

En résumé, le processus de dissémination des informations est implémenté deux fois en écriture et une fois en lecture durant ce tronçon du cycle de vie du textile. Dans notre étude, l'architecture de base de données est mise en place de manière centralisée (hébergée par un serveur MySQL). Le MLD sur lequel repose la base de données correspond à une partie du standard B2MML. Il s'agit en réalité d'un MLD se composant de 19 entités, qui sont décrites dans la TABLE 5.1¹. Puisque la base de données est mise en place, l'étape 1 du processus de dissémination n'est plus à traiter. Les sections suivantes traitent uniquement des étapes 2 et 3 des phases d'écriture et de lecture.

TABLE 5.1: Entités composant le MLD et réparties en quatre groupes d'entités distincts

Nom groupe	Nom Table	Définition
Produit	MaterialLot	Un objet <i>Material Lot</i> identifie une quantité de matière/produits (quantité réelle, courante ainsi que ses propriétés).
	MaterialDefinition	Moyen de décrire des matériaux ayant/regroupant des caractéristiques similaires, à des fins de planification et d'ordonnement.
	MaterialClass	Un objet <i>Material Class</i> est un moyen d'effectuer des définitions groupées, pouvant être utilisées lors des tâches de production ou de planification.
	ManufacturingBill	Une nomenclature identifie les classes de produits/matériaux qui sont requises lors de la production du produit.
	ActualMaterialLot MaterialSegmentSpecificat°	Un matériau produit, consommé ou des matériaux consommables sont identifiés. Contient des informations au sujet des ressources matérielles requises pour l'exécution d'un <i>process segment</i> . Un <i>process segment</i> définit souvent des classes de personnes (p. ex., usineur), d'équipements, etc.
Personnel	Person PersonClass	Une personne représente un individu particulier identifié. Moyen pour décrire un groupe de personnes ayant/regroupant des caractéristiques similaires, à des fins de planification et d'ordonnement.
	ActualPersonSegment	Identifie la personne qui est utilisée durant un <i>product segment</i> . Le <i>product segment</i> définit souvent des personnes, des équipements spécifiques (p.ex., Machine d'assemblage n°2).
	PersonSegmentSpecificat°	Contient des informations au sujet des ressources humaines requises pour l'exécution d'un <i>process segment</i> .
Production	ProductionOrder	Contient les informations liées aux ordres de production (date, heure, lieu, <i>Product Segment</i> , etc.).
	ProductSegment	Indique quelles sont les personnes, équipements, ou ressources matière requis pour exécuter le <i>Product Segment</i> , pour une certaine quantité de produit.
	ProductDefinition	Contient les références vers les règles de production, la nomenclature relative à la matière et aux ressources.
	SegmentRequirement SegmentResponse	Chaque <i>Segment requirement</i> correspond, ou référence, un <i>Product Segment</i> . Le résultat d'une production (liée à un segment de production spécifique) est défini comme un <i>Segment Response</i> . Ce segment peut être composé de zéro ou plusieurs ensembles d'informations (informations liées à la production, au personnel courant, aux équipements courants, aux matériaux consommés, etc.).
Equipement	Equipment EquipmentClass	Contient des informations liées à l'équipement (p. ex. les caractéristiques d'une machine). Moyen pour décrire un groupe d'équipements ayant/regroupant des caractéristiques similaires, à des fins de planification et d'ordonnement.
	ActualEquipmentSegment	Identifie l'équipement qui est utilisé durant un segment de production.
	EquipmentSegmentSpecificat°	Contient des informations au sujet des ressources d'équipement pour l'exécution d'un <i>process segment</i> .

1. la colonne « Nom groupe » de la TABLE 5.1 sera introduite et utilisée lorsque nous nous intéresserons aux groupes d'entités dans le critère *Contextual*.

5.2 Exécution des phases d'écriture et de lecture

Les opérations d'écriture et de lecture du produit reposent sur les étapes 2 et 3 du processus de dissémination. Dans l'étape 2, les deux méthodes présentées dans le Chapitre 3 (Chan-Roddick et AHP-Floue) sont respectivement mises en oeuvre et comparées.

La section 5.2.1 décrit le principe de comparaison utilisé dans cette étude.

La section 5.2.2 décrit respectivement l'exécution des étapes 2 et 3 du processus de dissémination des informations relatives au point d'écriture n°1.

La section 5.2.3 détaille, quant à elle, l'étape 3 du processus de dissémination mise en oeuvre pour lire de manière localisée les informations sur la bobine de textile communicant.

La section 5.2.4 décrit respectivement l'exécution des étapes 2 et 3 du processus de dissémination des informations relatives au point d'écriture n°2.

5.2.1 Principe de comparaison des approches Chan-Roddick / AHP-Floue

Dans notre étude, les deux approches proposées dans le Chapitre 3 pour évaluer la pertinence des data items sont appliquées, puis comparées. Si un seul expert effectue les ajustements dans ces deux approches, la comparaison peut être menée sans problème comme cela est détaillé dans la FIGURE 5.2(a). Cependant, si plusieurs experts réalisent ces ajustements, un problème se pose au niveau de l'approche de Chan et Roddick. En effet, elle ne permet pas de prendre en compte plusieurs expertises. Par conséquent, si nous désirons tout de même les comparer, une étape supplémentaire doit être définie avant d'appliquer l'approche de Chan-Roddick. Il s'agit d'une étape d'agrégation des avis comme illustré sur la FIGURE 5.2(b). Puisque notre étude considère des groupes d'experts dans les différents critères, cette étape doit être ajoutée dans notre scénario. Plusieurs techniques peuvent être déployées pour agréger les avis (avis moyen, minimum, maximum, etc.). Dans notre étude, nous calculons l'avis moyen car nos groupes d'experts sont de taille restreinte (4-5 experts maximum) et par conséquent, la moyenne est un bon compromis². Les calculs des avis moyens sont présentés au fur et à mesure de l'étude de cas. Cette agrégation engendre quelques modifications à l'approche de Chan et Roddick, principalement sur les échelles de valeurs qui ont été définies dans le Chapitre 3. Encore une fois, ceci est propre à notre cas d'étude où nous cherchons un moyen de comparer ces deux approches en collectant des avis de plusieurs experts. Néanmoins, nous comparons également ces deux approches dans le cas d'un avis unique (cf. FIGURE 5.2(a)) dans l'Annexe VII.

2. le but recherché n'est pas d'étudier et de trouver le mécanisme d'agrégation optimal/parfait pour la méthode de Chan et Roddick, mais uniquement de permettre la comparaison entre les deux approches.

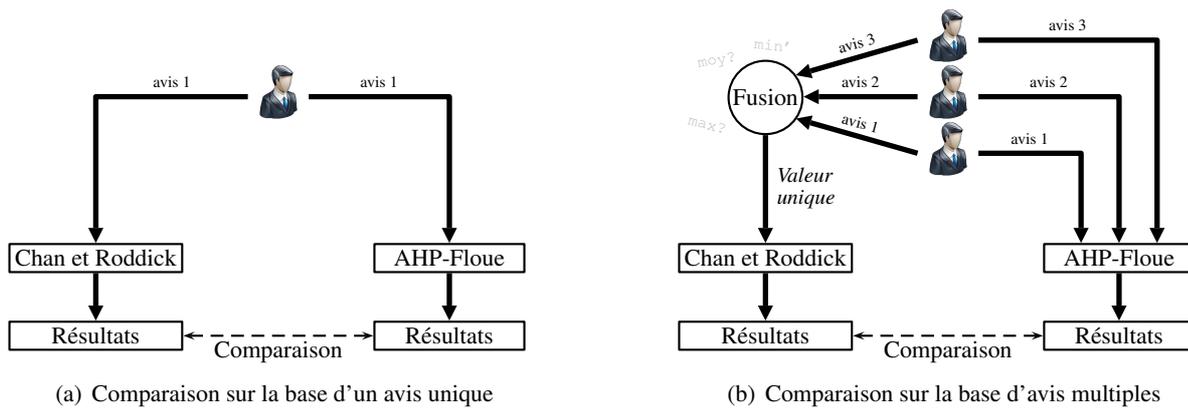


FIGURE 5.2: Principes de comparaison entre les méthodes : Chan-Roddick et AHP-Floue

5.2.2 Point d'écriture n°1

Cette section présente tout d'abord les ajustements réalisés par les experts concernés par le point d'écriture n°1 (nécessaire pour l'exécution de l'étape 2 du processus). Puis, lorsqu'une bobine de textile communicant se présente au point d'écriture, les exécutions des étapes 2 et 3 du processus de dissémination sont respectivement détaillées.

Étape 2 : ajustement et expertise des différents critères

Rappelons que les expertises menées dans le critère *Contextual* sont réalisées par un groupe d'experts, qui s'intéresse à l'ensemble du cycle de vie du textile (p. ex. un groupe d'experts provenant de plusieurs compagnies). Dans ce scénario, ce groupe se compose de trois experts. Les expertises menées dans les critères *Enumeration*, *Model-Based*, *Data Size*, ainsi qu'entre les différents critères sont effectuées par des experts/utilisateurs en aval du point d'écriture (comme expliqué dans le Chapitre 3). Cependant, ce sont des personnes provenant de la compagnie qui créent le point d'écriture et qui spécifient au niveau logiciel les expertises. Dans ce scénario, les expertises sont détaillées dans l'ordre suivant :

- i. expertises liées au critère *Contextual* : les experts définissent les groupes d'entités, puis ajustent leur importance relativement au cycle de vie du textile,
- ii. expertises liées à l'importance des critères : le(s) expert(s) définissent l'importance de chaque critère,
- iii. expertises liées au critère *Enumeration* : le(s) expert(s) indiquent les informations à stocker sur le produit,
- iv. expertises liées au critère *Model-Based* : le(s) expert(s) ajustent le coefficient k afin de favoriser ou non les informations distantes de la table produit,
- v. expertises liées au critère *Data Size* : le(s) expert(s) ajustent le coefficient k afin de favoriser ou non le stockage d'informations de grandes tailles,

i. Le groupe d'experts concerné par le critère *Contextual* a convenu d'une réunion de travail pour effectuer des ajustements. Ils définissent quatre groupes d'entités au sein du MLD, qui sont détaillés dans la TABLE 5.1. Le premier groupe G_1 , intitulé « Produit », rassemble les entités concernées par des informations liées à la composition du produit (p. ex., information sur le matériau, sur les composants/sous-composants, etc.). Le second groupe G_2 , intitulé « Personnel », regroupe les entités concernées par des informations liées aux personnes, qui sont en contact direct ou indirect avec le produit (p. ex., la personne qui a usiné le produit, qui doit en assurer la maintenance, le transport, etc.). Le troisième groupe G_3 , intitulé « Production », réunit les entités concernées par les données de production (p. ex., ordres de production, planification, ordonnancement des tâches, etc.). Le dernier groupe G_4 , intitulé « Equipement », identifie les entités concernées par des informations sur les équipements/ressources liés au produit (p. ex., les machines usinant le produit, les ressources utilisées, etc.). Les experts effectuent ensuite les comparaisons entre groupes et cela pour chaque phase du cycle de vie du textile. Dans notre cas d'étude, nous nous focalisons uniquement sur les phases 1 et 2 (cf. FIGURE 5.1). Pour la phase 1, les trois experts D_G^1 , D_G^2 et D_G^3 définissent respectivement les matrices de comparaisons fournies dans l'ÉQUATION 5.1. Dans l'approche AHP-Floue, le vecteur propre flou $\tilde{\Lambda}_G$ (reflet de l'importance de chaque groupe d'entités) est calculé via les ÉQUATIONS 3.21 à 3.23, dont le résultat est affiché dans la FIGURE 5.3(a). Nous constatons pour cette phase, que les informations issues des groupes G_1 et G_3 sont jugées plus importantes que celles de G_2 et G_4 . Concernant l'approche de Chan et Roddick, nous réalisons l'agrégation des trois avis en calculant la moyenne des matrices comme cela est détaillé dans l'ÉQUATION 5.2. Le vecteur propre Λ_G de cette matrice est donné dans la FIGURE 5.3(b) et montre une nouvelle fois que les groupes G_1 et G_3 sont équivalents et sont les plus importants (≈ 0.41).

$$D_G^1 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & 3 & 1 & 3 \\ \frac{1}{3} & 1 & \frac{1}{3} & 1 \\ 1 & 3 & 1 & 5 \\ \frac{1}{3} & 1 & \frac{1}{5} & 1 \end{bmatrix} & & & \end{matrix} \quad D_G^2 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & 5 & 1 & 3 \\ \frac{1}{5} & 1 & \frac{1}{5} & 1 \\ 1 & 5 & 1 & 7 \\ \frac{1}{3} & 1 & \frac{1}{7} & 1 \end{bmatrix} & & & \end{matrix} \quad D_G^3 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & 5 & 3 & 5 \\ \frac{1}{5} & 1 & \frac{1}{9} & 1 \\ \frac{1}{3} & 9 & 1 & 7 \\ \frac{1}{5} & 1 & \frac{1}{7} & 1 \end{bmatrix} & & & \end{matrix} \quad (5.1)$$

$$\hat{G} = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} \frac{1+1+1}{3} & \frac{3+5+5}{3} & \frac{1+1+3}{3} & \frac{3+3+5}{3} \\ \frac{\frac{1}{3}+\frac{1}{5}+\frac{1}{5}}{3} & \frac{1+1+1}{3} & \frac{\frac{1}{3}+\frac{1}{5}+\frac{1}{9}}{3} & \frac{1+1+1}{3} \\ \frac{1+1+\frac{1}{3}}{3} & \frac{3+5+9}{3} & \frac{1+1+1}{3} & \frac{5+7+7}{3} \\ \frac{\frac{1}{3}+\frac{1}{3}+\frac{1}{5}}{3} & \frac{1+1+1}{3} & \frac{\frac{1}{5}+\frac{1}{7}+\frac{1}{7}}{3} & \frac{1+1+1}{3} \end{bmatrix} & & & \end{matrix} = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & 4.33 & 1.66 & 3.66 \\ 0.24 & 1 & 0.21 & 1 \\ 0.77 & 5.67 & 1 & 6.33 \\ 0.29 & 1 & 0.16 & 1 \end{bmatrix} & & & \end{matrix} \quad (5.2)$$

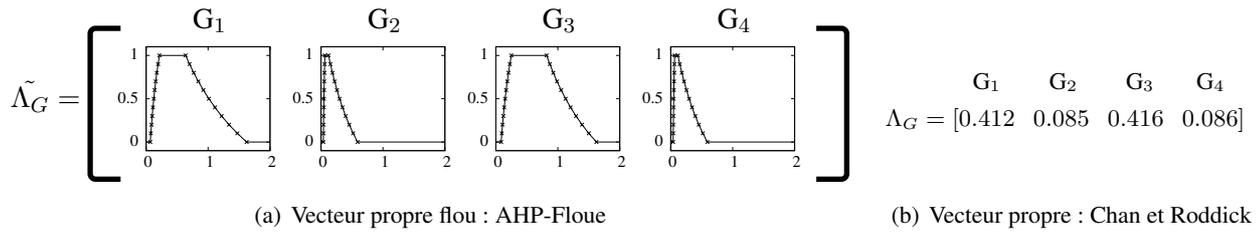


FIGURE 5.3: Vecteurs propres indiquant l'importance de chaque groupe d'entités : phase 1

Les experts réitèrent cette démarche afin de définir l'importance des quatre groupes d'entités à la phase 2 du cycle de vie du textile. Les matrices de comparaison ne sont plus présentées, mais les vecteurs propres obtenus, $\tilde{\Lambda}_G$ et Λ_G , sont respectivement donnés dans les FIGURES 5.4(a) et 5.4(b). Ces deux vecteurs indiquent que G₃ est favorisé par rapport aux trois autres groupes, dont l'ordre de priorité est :

- au regard de $\tilde{\Lambda}_G$: $\tilde{\lambda}_G(G_2) \lesssim \tilde{\lambda}_G(G_1) \lesssim \tilde{\lambda}_G(G_4) \lesssim \tilde{\lambda}_G(G_3)$,
- au regard de Λ_G : $\lambda_G(G_2) < \lambda_G(G_1) < \lambda_G(G_4) < \lambda_G(G_3)$.

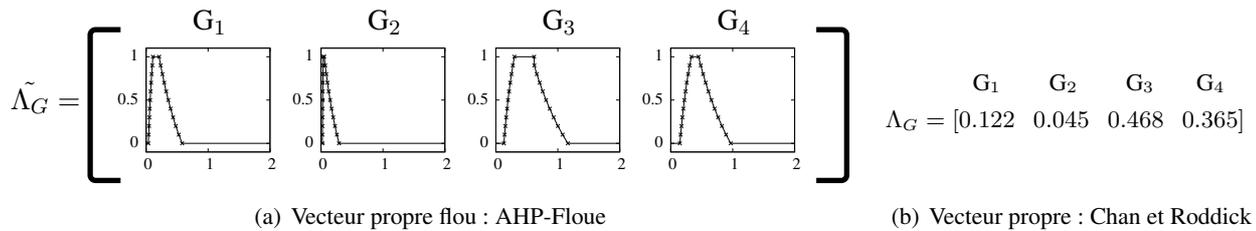


FIGURE 5.4: Vecteurs propres indiquant l'importance de chaque groupe d'entités : phase 2

ii. Deux experts issus de la compagnie mettant en œuvre le point d'écriture n°1 s'intéressent à l'importance des critères. Ils fournissent les matrices de comparaison décrites dans l'ÉQUATION 5.3. En appliquant l'approche AHP-Floue (ÉQUATIONS 3.29 à 3.31), nous obtenons le vecteur propre flou $\tilde{\Lambda}_\rho$ donné dans la FIGURE 5.5(a). Nous constatons que les experts privilégient par ordre croissant, le critère *Enumeration*, *Contextual*, *Model-Based* et *Data-Size* : $\tilde{\rho}_e \lesssim \tilde{\rho}_c \lesssim \tilde{\rho}_m \lesssim \tilde{\rho}_s$. L'avis moyen calculé pour l'approche de Chan et Roddick ne s'applique que sur les critères C_e , C_c et C_m car *Data Size* n'est pas un critère dans leur approche³. Nous remarquons une nouvelle fois que le critère C_e s'avère plus important que les autres : $\rho_m < \rho_c < \rho_e$. En privilégiant le critère *Enumeration*, les experts se donnent la liberté d'embarquer sur le textile les informations qu'ils jugent importantes, négligeant ainsi les recommandations des experts du critère *Contextual* (bien que ce critère soit tout de même le deuxième plus important).

3. notons que supprimer un critère affecterait certainement les opinions des experts et donc leur *pairwise comparison*.

$$D_{\rho}^1 = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 5 & 7 & 5 \end{bmatrix} \\ C_c & \begin{bmatrix} \frac{1}{5} & 1 & 3 & 5 \end{bmatrix} \\ C_m & \begin{bmatrix} \frac{1}{7} & \frac{1}{3} & 1 & 3 \end{bmatrix} \\ C_s & \begin{bmatrix} \frac{1}{5} & \frac{1}{5} & \frac{1}{3} & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^2 = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 3 & 3 & 5 \end{bmatrix} \\ C_c & \begin{bmatrix} \frac{1}{3} & 1 & 5 & 3 \end{bmatrix} \\ C_m & \begin{bmatrix} \frac{1}{3} & \frac{1}{5} & 1 & 1 \end{bmatrix} \\ C_s & \begin{bmatrix} \frac{1}{5} & \frac{1}{3} & 1 & 1 \end{bmatrix} \end{matrix} \quad (5.3)$$

$$\hat{\rho} = \begin{matrix} & C_e & C_c & C_m \\ C_e & \begin{bmatrix} \frac{1+1}{2} & \frac{5+3}{2} & \frac{7+3}{2} \end{bmatrix} \\ C_c & \begin{bmatrix} \frac{\frac{1}{5}+\frac{1}{3}}{2} & \frac{1+1}{2} & \frac{3+5}{2} \end{bmatrix} \\ C_m & \begin{bmatrix} \frac{\frac{1}{7}+\frac{1}{3}}{2} & \frac{\frac{1}{3}+\frac{1}{5}}{2} & \frac{1+1}{2} \end{bmatrix} \end{matrix} = \begin{matrix} & C_e & C_c & C_m \\ C_e & \begin{bmatrix} 1 & 4 & 5 \end{bmatrix} \\ C_c & \begin{bmatrix} 0.27 & 1 & 4 \end{bmatrix} \\ C_m & \begin{bmatrix} 0.24 & 0.27 & 1 \end{bmatrix} \end{matrix} \quad (5.4)$$

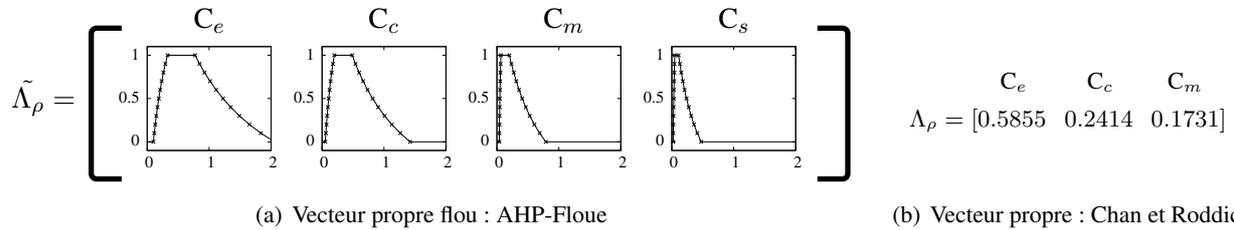
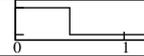
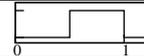
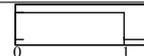
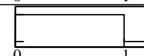
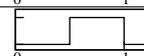
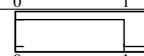
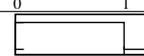
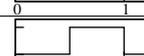


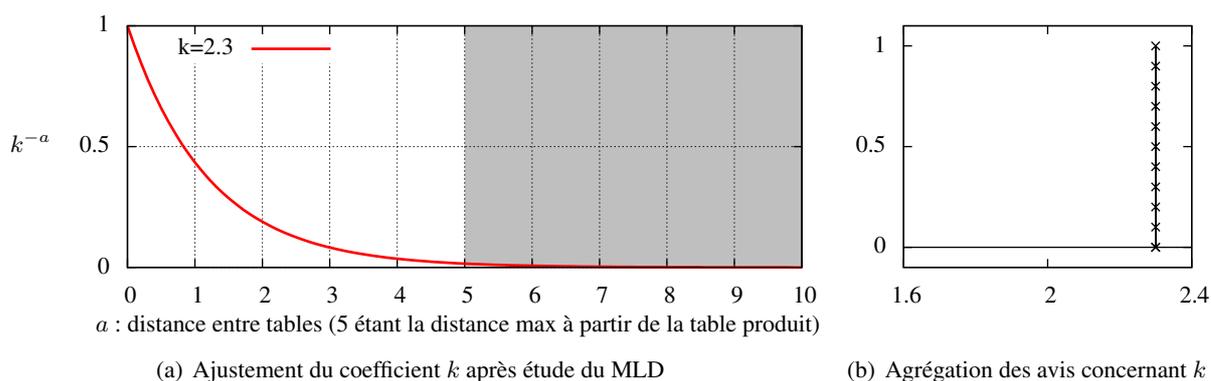
FIGURE 5.5: Vecteurs propres indiquant l'importance de chaque critère : phase 1

iii. Les deux mêmes experts procèdent ensuite à l'énumération des informations (c.-à-d. des attributs) qu'ils souhaitent stocker sur le textile. La TABLE 5.2 résume les attributs respectivement énumérés par ces deux experts (nommés D_1 et D_2). Nous constatons par exemple que D_1 et D_2 énumèrent tous deux les attributs `IDMaterialLot` et `Quantity` de la table `MaterialLot`. Ils jugent tous les deux nécessaire l'attribut `IDMaterialLot` (Choix= 1 pour D_1 et D_2), alors que concernant l'attribut `Quantity`, D_1 le juge intéressant (Choix= 0.5) et D_2 le juge nécessaire (Choix= 1). En appliquant l'approche AHP-Floue (ÉQUATION 3.24 et TABLE 3.4), nous obtenons un sous-ensemble flou propre à chaque attribut comme illustré dans la TABLE 5.2 (colonne intitulée « AHP-floue »). Concernant l'énumération pour l'approche de Chan et Roddick, nous calculons la valeur moyenne nette pour chaque attribut comme détaillé dans la TABLE 5.2 (colonne intitulée « Chan-Roddick »). Les attributs non énumérés par les experts ne sont pas présentés dans ce tableau, mais prennent bien évidemment la valeur 0 (c.-à-d., singleton de valeur 0 dans l'approche AHP-Floue et valeur nette 0 dans l'approche Chan-Roddick).

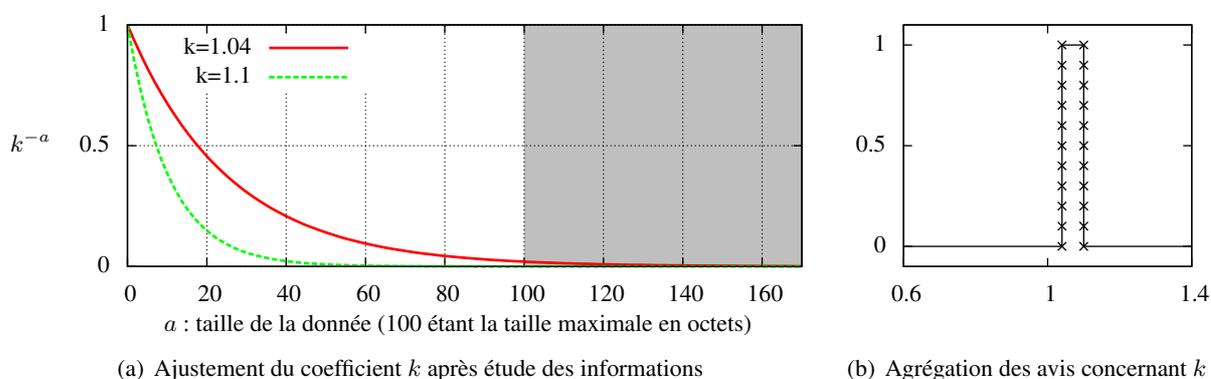
TABLE 5.2: Attributs énumérés par les experts de la compagnie implémentant le point d'écriture n°1

	Nom Table	Nom Attribut	Décideur	Choix	Score de l'attribut après fusion des avis	
					AHP-Floue	Chan-Roddick
G ₁	MaterialLot	IDMaterialLot	D1 D2	1 1		$\frac{1+1}{2} = 1$
		Status	D1 D2	0 0.5		$\frac{0+0.5}{2} = 0.25$
		Quantity	D1 D2	0.5 1		$\frac{0.5+1}{2} = 0.75$
	MaterialDefinition	IDMaterialDefinition	D1 D2	1 0		$\frac{1+0}{2} = 0.5$
		Value	D1 D2	1 0		$\frac{1+0}{2} = 0.5$
G ₃	ProductionOrder	IDProductionOrder	D1 D2	0.5 1		$\frac{0.5+1}{2} = 0.75$
		StartTime	D1 D2	0 1		$\frac{0+1}{2} = 0.5$
		EndTime	D1 D2	0 1		$\frac{0+1}{2} = 0.5$
	ProductSegment	IDProductSegment	D1 D2	1 1		$\frac{1+1}{2} = 1$
		Description	D1 D2	1 0.5		$\frac{1+0.5}{2} = 0.75$
		PublishedDate	D1 D2	0.5 0		$\frac{0.5+0}{2} = 0.25$
		Duration	D1 D2	0.5 0		$\frac{0.5+0}{2} = 0.25$
	ProductDefinition	IDProductDefinition	D1 D2	1 1		$\frac{1+1}{2} = 1$

iv. Les deux experts procèdent désormais à l'ajustement du coefficient k afin de spécifier à partir de quelle distance une donnée doit être considérée comme indésirable sur le produit. Par rapport au MLD et à la distance maximale entre l'entité produit et les autres entités (égale à 5), les experts fixent tous deux le coefficient k à 2.3 afin de négliger les données distantes de plus de 3 relations de la table produit ($2.3^{-3} = 0.08$), comme montré dans la FIGURE 5.6(a). En appliquant l'approche AHP-Floue (ÉQUATIONS 3.25 à 3.26), nous obtenons un sous-ensemble de type singleton de valeur 2.3 comme illustré dans la FIGURE 5.6(b). Le coefficient k moyen destiné à l'approche de Chan et Roddick est quant à lui égal à $\frac{2.3}{1} = 2.3$.

FIGURE 5.6: Ajustement du coefficient k et agrégation des avis concernant *Model Based* : phase 1

v. Les deux experts procèdent de la même manière à l'ajustement du coefficient k afin de spécifier à partir de quelle taille une donnée doit être considérée comme indésirable sur le produit. Vu l'espace mémoire moyen disponible sur les bobines textiles ($\approx 5\text{Mo}$), et les tailles des data items (variant de 1 octet à 140 octets), le premier expert fixe k à 1.04 afin de négliger totalement les données de taille supérieure à 100 octets comme le montre la tendance de la courbe dans la FIGURE 5.7(b). Le second expert fixe k à 1.1 afin de négliger totalement les données de taille supérieure à 60 octets. En appliquant l'approche AHP-Floue (ÉQUATIONS 3.27 et 3.28), le sous-ensemble flou décrit dans la FIGURE 5.7(b) est obtenu. Le coefficient k moyen destiné à l'approche de Chan et Roddick est quant à lui égal à $\frac{1.04+1.1}{2} = 1.07$.

FIGURE 5.7: Ajustement du coefficient k et agrégation des avis concernant *Data Size* : phase 1

Tous les ajustements ont été réalisés. La FIGURE 5.8 offre un aperçu de l'interface graphique de « Java/Matlab » qui a permis aux experts de réaliser leurs ajustements (certains ajustements précédemment décrits sont détaillés dans cette figure, cf. cadres jaunes).

Désormais, lorsqu'une bobine de textile communicant se présente au point d'écriture n°1, les étapes 2 et 3 du processus sont exécutées. Ces deux étapes font l'objet des deux prochaines sections.

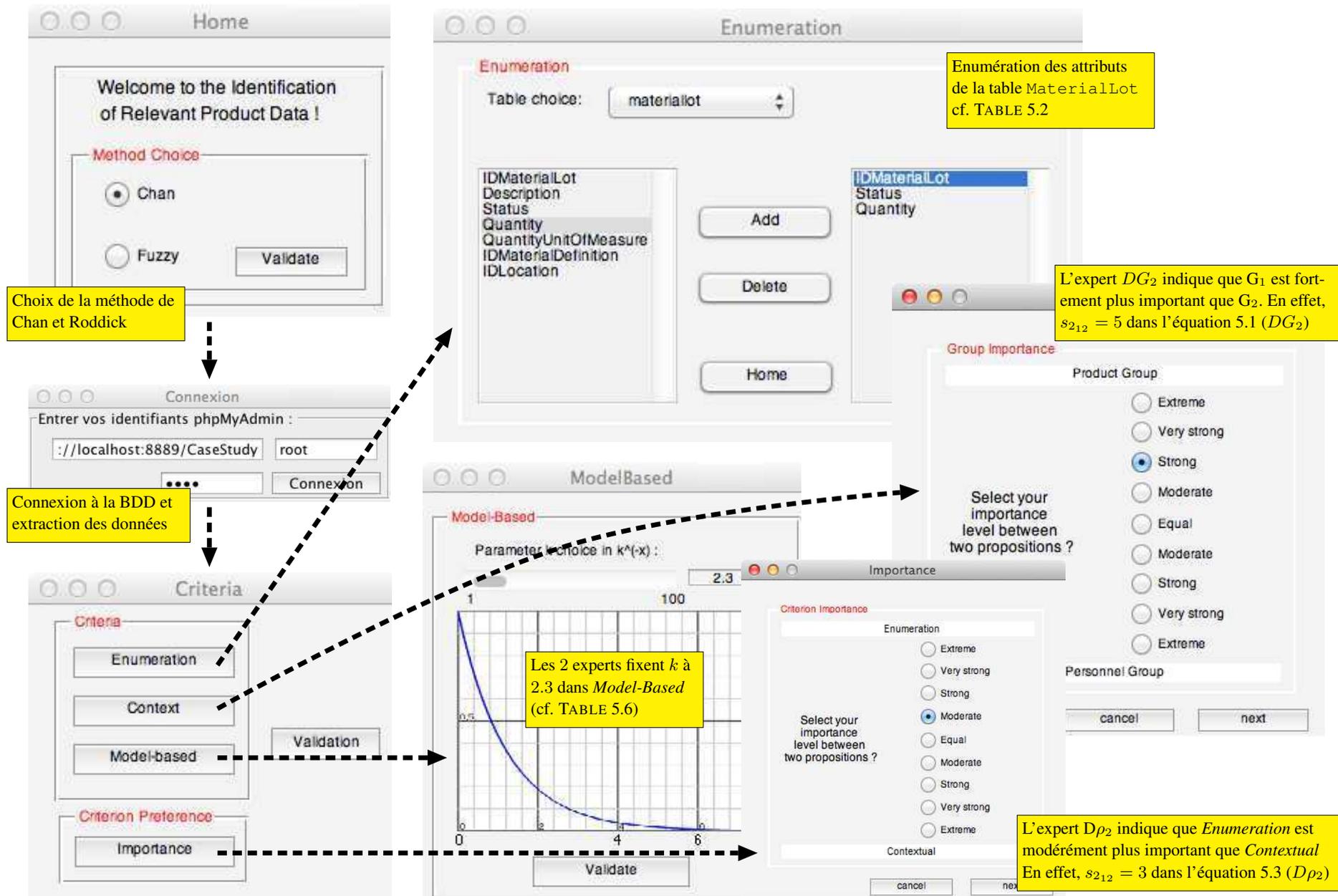


FIGURE 5.8: Application « JAVA/Matlab » assurant l'extraction des données de la BDD, l'ajustement des paramètres et le calcul de la pertinence

Étape 2 : exécution

Lorsque la bobine de textile communicant se présente au point d'écriture n°1, la machine dédiée à la tâche d'écriture relève la référence portée par celle-ci. Elle se réfère dans notre scénario à la table `MaterialLot`, du tuple dont la clé primaire contient la valeur `L00TB-B101`. L'algorithme `RetrievalData` est alors exécuté afin d'identifier tous les « tuples produits ». Dans un second temps, les pertinences des data items sont calculées à l'aide, soit de la méthode Chan-Roddick, soit de la méthode AHP-Floue.

→ Exécution de l'approche AHP-Floue

L'approche AHP-Floue génère une liste de data items notée « ListeAF », présentée dans la FIGURE 5.9, où chaque ligne correspond à un data item. Les trois premières colonnes de la liste permettent de localiser ce data item au sein de la base de données (nom de la table, nom de l'attribut, nom de la clé primaire), la quatrième et la cinquième colonne indiquent respectivement le contenu du data item et sa pertinence. Ces propos sont illustrés avec le data item localisé au rang 6 de ListeAF, lequel provient de la table `MaterialDefinition`, de l'attribut `IDMaterial` et dont la clé primaire est `MD060111223`. Nous observons que 108 data items composent ListeAF, ce qui signifie que 108 data items ont été identifiés via l'algorithme `RetrievalData`. Nous constatons, d'une part, que le data item le plus pertinent se trouve dans la table `MaterialLot` et, d'autre part, que les 18 premiers data items appartiennent aux tables `ProductionOrder`, `MaterialLot`, `ProductSegment`, `MaterialDefinition` et `ProductDefinition`. Cela s'explique du fait que toutes ces tables contiennent des attributs énumérés (cf. TABLE 5.2) et que le critère *Enumeration* est jugé prioritaire au point d'écriture n°1 (cf. FIGURE 5.5). De plus, ces cinq tables sont incluses dans les groupes G_1 et G_3 qui sont fortement recommandés par les experts issus du critère *Contextual* (cf. FIGURE 5.3). Comme l'importance du critère *Contextual* est prioritaire derrière *Enumeration*, le poids des data items provenant de ces tables est renforcé. Afin d'avoir une vision complète de ListeAF, nous proposons de modéliser les valeurs de pertinence sous forme de diagramme type « boîtes à moustaches » (FIGURE 5.10(a)) où, pour chaque table t ($t \in \mathcal{T}$), est représentée la valeur de pertinence maximale des data items présents dans la liste et appartenant à t , la pertinence minimale, le 1^{er} et 3^{ème} quartiles, ainsi que la médiane. Cette représentation permet de vérifier que le data item le plus pertinent appartient à `MaterialLot`, et de constater que les cinq tables énoncées précédemment se détachent sensiblement des autres. Il est à noter que certaines tables non énumérées se situent au niveau de ces 5 tables telles que `MaterialClass`, `SegmentRequirement`, `ManufacturingBill`, etc. Cela s'explique encore une fois par le fait que ces tables appartiennent aux groupes d'entités G_1 et G_3 .

Aux différents points d'écriture, les data items sont stockés sur le produit du plus pertinent au moins pertinent, jusqu'à ce que la matière ne dispose plus d'espace mémoire⁴. Au point d'écriture n°1, la bobine de textile communicant ne dispose que d'une soixantaine de tags RFID, ce qui dans notre cas permet

4. d'autres approches peuvent être mises en œuvre pour décider à partir de quel rang les informations ne doivent plus être stockées sur la matière (indépendamment de l'espace mémoire). À titre d'exemple, nous pourrions décider de ne pas stocker des data items ayant une pertinence inférieure à un seuil donné.

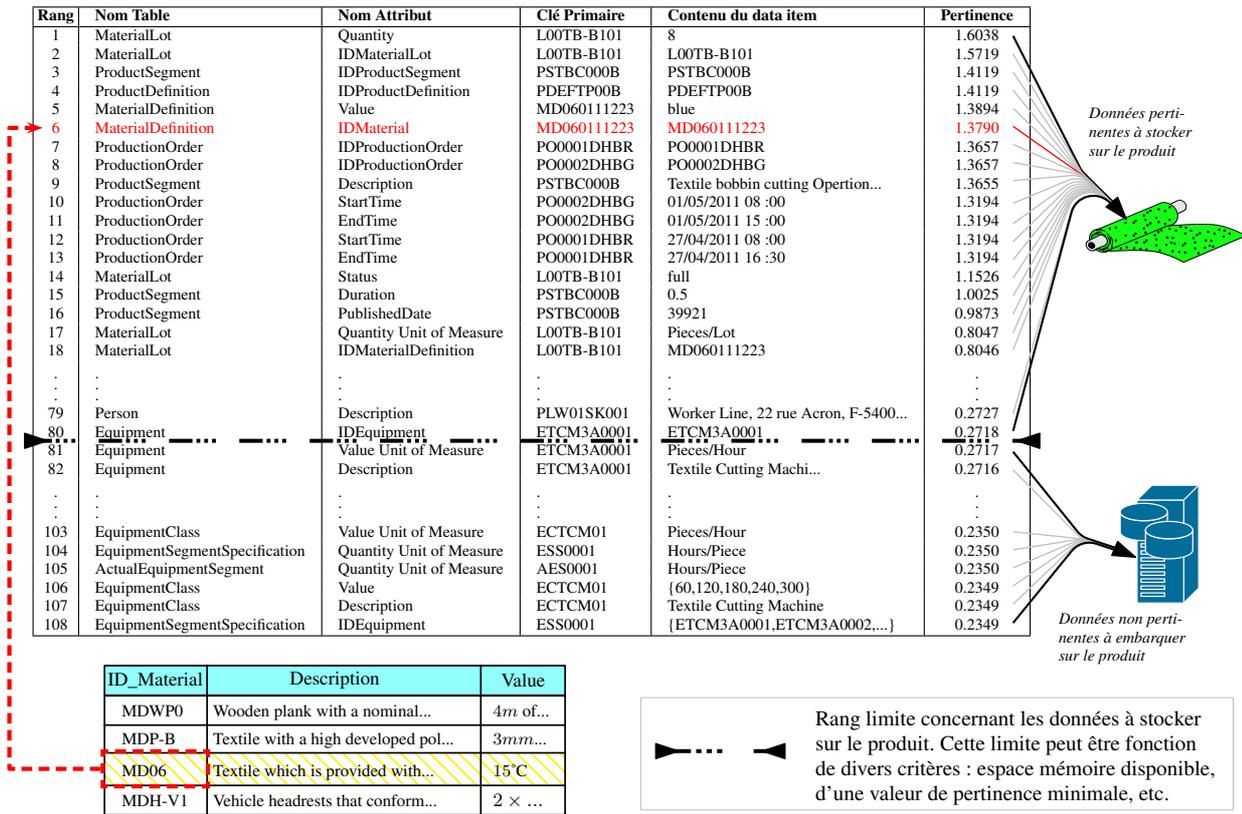


FIGURE 5.9: ListeAF ordonnée du data item le plus pertinent au moins pertinent : Point d'écriture n°1

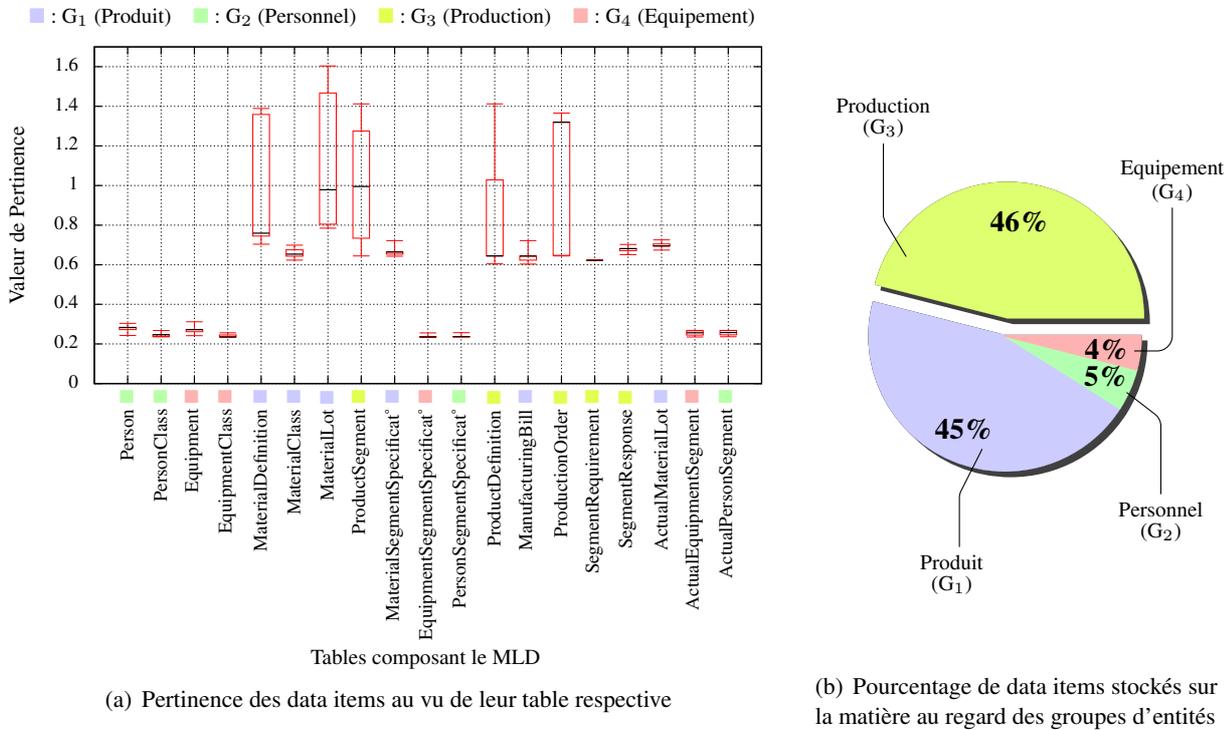


FIGURE 5.10: Représentation de ListeAF sous forme de diagramme : Point d'écriture n°1

uniquement d'embarquer les 80 premiers data items de ListeAF, comme illustré dans la FIGURE 5.9. Le diagramme circulaire présenté dans la FIGURE 5.10(b) indique les pourcentages de data items appartenant aux différents groupes d'entités parmi ceux sélectionnés. Par exemple, 46% des 80 data items (soit 36 data items) proviennent d'entités appartenant au groupe G_1 et 4% (soit 3 data items) proviennent d'entités appartenant à G_2 . Ce diagramme confirme que les data items en début de liste appartiennent principalement à des entités des groupes G_1 et G_3 .

→ Exécution de l'approche Chan-Roddick

La FIGURE 5.11 présente la liste de data items, notée « ListeCR », générée par l'approche de Chan et Roddick. Le premier data item de ListeCR est le même que celui de ListeAF, ce qui n'est pas le cas pour le data item de rang 2 dans ListeAF, qui se trouve au rang 6 dans ListeCR. Cependant, tout comme dans ListeAF, les 18 premiers data items appartiennent aux tables `ProductionOrder`, `MaterialLot`, `ProductSegment`, `MaterialDefinition` et `ProductDefinition` (tables détenant des attributs énumérés). Ces cinq tables se détachent également des autres comme le montre le diagramme « boîtes à moustaches » de la FIGURE 5.12(a). Cependant, la différence semble plus marquée que précédemment. En effet, les data items non énumérés sont de pertinence quasi nulle dans ListeCR (< 0.1), alors que ce n'est pas le cas dans ListeAF (cf. FIGURE 5.10(a)). Ceci montre que, pour ce scénario en particulier, l'approche de Chan et Roddick est plus radicale vis-à-vis des informations énumérées : les data items appartenant à des attributs énumérés sont largement favorisés.

Les résultats affichés par le diagramme circulaire de la FIGURE 5.12(b) sont toutefois identiques à ceux de la FIGURE 5.10(b). Cela est une première indication sur le fait que les deux approches retournent des listes similaires.

Rang	Nom Table	Nom Attribut	Clé Primaire	Contenu du data item	Pertinence
1	MaterialLot	Quantity	L00TB-B101	8	1.2254
2	ProductSegment	Duration	PSTBC000B	0,5	0.6927
3	MaterialLot	Status	L00TB-B101	full	0.6127
4	MaterialDefinition	Value	MD060111223	blue	0.5764
5	ProductSegment	PublishedDate	PSTBC000B	39921	0.4728
6	MaterialLot	IDMaterialLot	L00TB-B101	L00TB-B101	0.3689
7	ProductSegment	IDProductSegment	PSTBC000B	PSTBC000B	0.3464
8	ProductDefinition	PDEFTP00B	PDEFTP00B	PDEFTP00B	0.3464
9	MaterialDefinition	IDMaterialDefinition	MD060111223	MD060111223	0.3332
10	ProductionOrder	IDProductionOrder	PO0001DHBR	PO0001DHBR	0.3305
11	ProductionOrder	IDProductionOrder	PO0002DHBG	PO0002DHBG	0.3305
12	ProductionOrder	StartTime	PO0002DHBG	01/05/2011 08 :00	0.2884
13	ProductionOrder	EndTime	PO0002DHBG	01/05/2011 15 :00	0.2884
14	ProductionOrder	StartTime	PO0001DHBR	27/04/2011 08 :00	0.2810
15	ProductionOrder	EndTime	PO0001DHBR	27/04/2011 16 :30	0.2810
16	MaterialSegmentSpecification	Quantity	MSS0001	8	0.1916
17	ManufacturingBill	Quantity	PDEFTP00B1	1	0.1916
18	ProductSegment	Description	PSTBC000B	Textile bobbin cutting Opertaion...	0.1816
.
.
.
.
.
.
.
.
103	EquipmentClass	Value	ECTCM01	{60,120,180,240,300}	0.0120
104	PersonClass	Description	PCPLW01	Production Line Worker	0.0116
105	EquipmentClass	Description	ECTCM01	Textile Cutting Machine	0.0115
106	Person	Description	PLW01SK001	Worker Line, adresse : 22 rue Acron, F-...	0.0112
107	EquipmentSegmentSpecification	IDEquipment	ESS0001	{ETCM3A0001,ETCM3A0002}	0.0102
108	PersonSegmentSpecification	IDPerson	PSS0001	{PLW01SK001,PLW01JD001...}	0.0101

FIGURE 5.11: ListeCR ordonnée du data item le plus pertinent au moins pertinent : Point d'écriture n°1

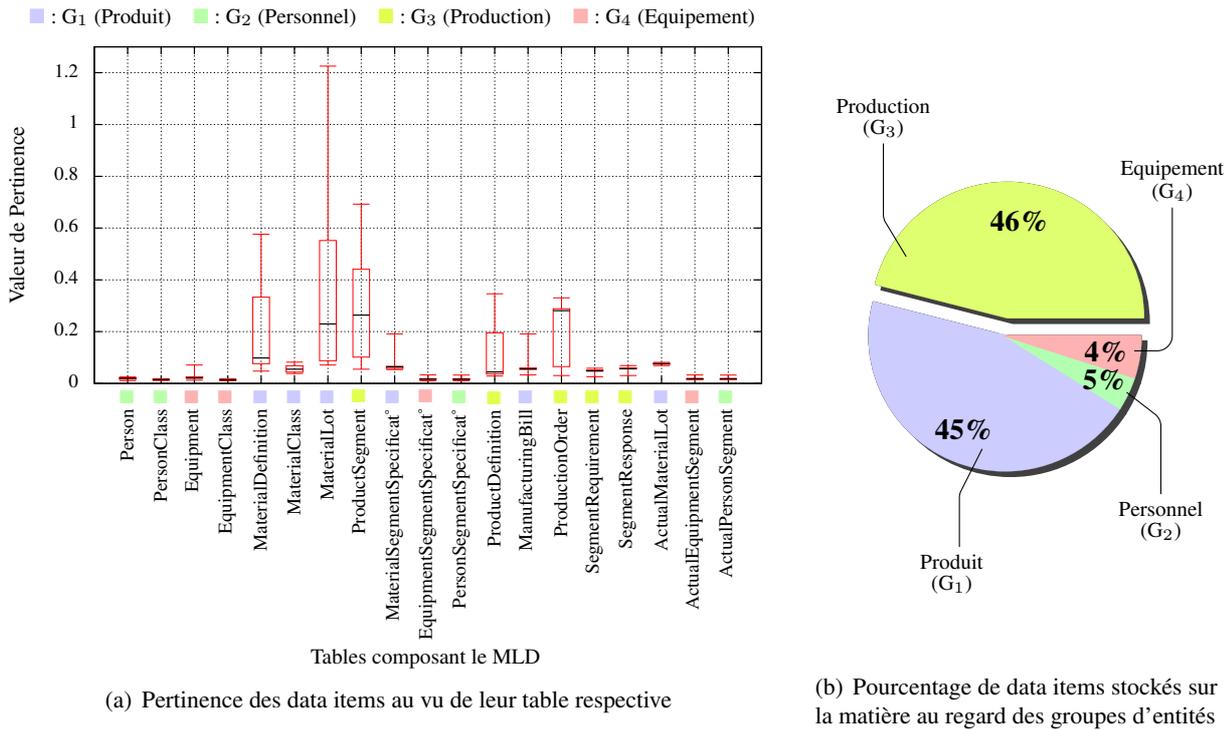


FIGURE 5.12: Représentation de ListeCR sous forme de diagramme : Point d'écriture n°1

→ Comparaison de ListeAF et ListeCR

Afin de comparer et d'évaluer la similitude entre ListeAF et ListeCR, deux représentations sont adoptées. La première est donnée dans la FIGURE 5.13(a), où est représenté en abscisse le numéro de rang d'un data item particulier dans ListeCR et en ordonnée le numéro de rang de ce même data item dans ListeAF. Par exemple, le data item étudié sur la FIGURE 5.13(a) (issu de la table `Equipment`, de l'attribut `Value` et dont le contenu est 60) est classé au rang 33/108 dans ListeCR et au rang 74/108 dans ListeAF, soit 41 rangs de différence entre les deux approches. Plus les points se situent le long de la ligne centrale (tracée en trait plein), plus le classement des deux listes est similaire. L'exemple choisi précédemment n'est pas anodin, il s'agit du data item qui possède la plus grande différence de rangs d'une approche à l'autre. Pour expliquer une telle différence, intéressons-nous dans un premier temps à l'approche AHP-Floue. Il s'agit d'un data item de très petite taille (deux caractères : 60). Cependant, le critère *Data Size* est considéré comme étant le moins important parmi les quatre critères au point d'écriture n°1. Par conséquent, qu'il soit petit ou grand, cela n'impacte pas de manière significative le résultat final (d'où le rang 74/108 dans ListeAF). À l'opposé, l'approche de Chan et Roddick favorise tous les data items ayant une petite taille et pénalise tous ceux ayant une grande taille (fonction logarithme népérien), d'où le rang 33/108 dans ListeAF. À noter que si le critère *Data Size* était le plus important, il y aurait une forte probabilité que le classement de ListeAF en soit bouleversé. Néanmoins, de manière globale, nous constatons que le classement résultant des deux approches reste similaire⁵.

5. uniquement la similitude entre approches est quantifié dans ces comparaisons. Il est difficile de quantifier ou d'établir des

La deuxième représentation, donnée dans la FIGURE 5.13(b), indique le pourcentage de data items qui seraient identiques si nous embarquions q data items de ListeAF et ListeCR (q étant inférieur au nombre total de data items dans les listes). Par exemple, si nous embarquons les 10 premiers data items de ListeAF et les 10 premiers de ListeCR, 70% des data items embarqués seraient identiques (c.-à-d., 7 sur 10). Dans notre scénario, 80 data items sont embarqués sur la bobine de textile communicant, nous pouvons donc voir sur la FIGURE 5.13(b) que quelle que soit l'approche mise en œuvre au point d'écriture n°1 (Chan-Roddick ou AHP-Floue), 95% des data items stockés sur la bobine de textile sont identiques. Ceci confirme le précédent commentaire, à savoir que le classement des deux approches est proche. Cette similitude est corrélée aux ajustements effectués au point d'écriture n°1.

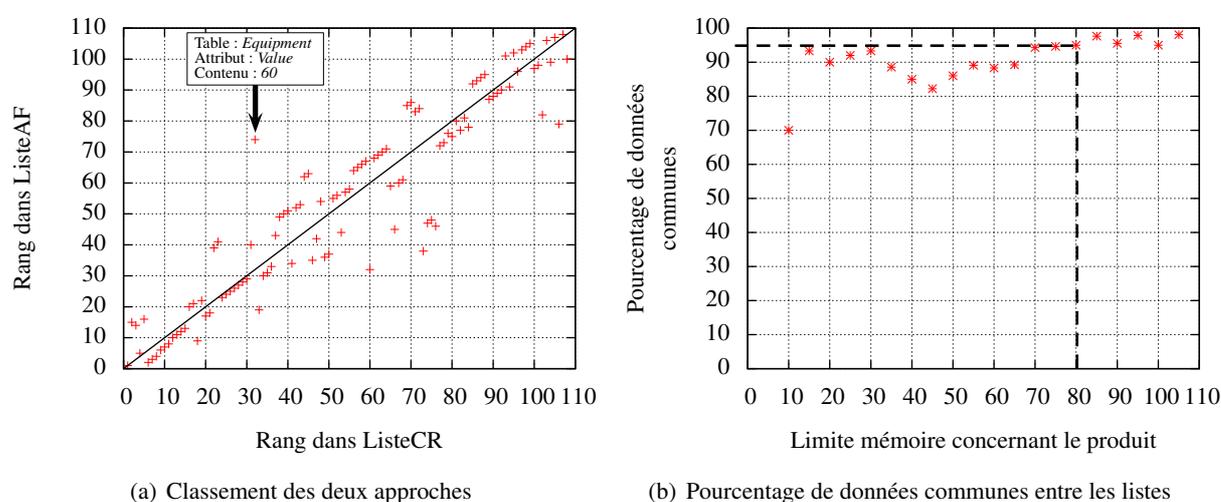


FIGURE 5.13: Comparaison des approches Chan-Roddick et AHP-Floue : Point d'écriture n°1

Étape 3 : exécution de l'opération d'écriture

L'architecture matérielle et logicielle « Écriture/Données non localisées » décrite dans la FIGURE 4.6 est utilisée pour stocker les 80 data items de ListeAF⁶ sur la bobine de textile communicant. La FIGURE 5.14(a) montre les événements générés par le lecteur RFID lors de l'opération d'écriture. Nous observons que 63 tags RFID ont été requis pour stocker les 80 data items. Cependant, 70 tags ont fait l'objet d'une écriture car 7 d'entre elles ont échoué (cf. FIGURE 5.14(a) : l'écriture du 3^{ème} tag de référence e007000002199de8 a échoué). Ceci montre que les data items sont stockés sur le textile malgré les erreurs d'écriture. La raison est que la plupart des technologies RFID implémentent, de base, des mécanismes automatiques de correction d'erreur.

indicateurs de performance d'une approche donnée. Une manière d'établir de tels indicateurs serait de définir un *benchmark* basé sur une application réelle (c.-à-d., avec des acteurs du métier), dans laquelle nous pourrions établir un schéma de distribution optimal, qui servirait de base pour l'évaluation de la performance des approches. Cet objectif fait partie des perspectives à court terme de nos travaux.

6. la démarche serait identique en considérant ListeCR.

```

Basic Application Example
File Help Log
File Parser Scheme Inspector Process Launcher Frag/alloc param IO
Write
Start writing process
Writing tag e007000002199dde
Nb tags writable: 1
Writing tag e0070000024dcbedd
Nb tags writable: 2
Nb tags writable: 3
Writing tag e007000002199de7
Nb tags writable: 4
...
Writing tag e007000002200010
Nb tags writable: 63
Total tag required: 70
Writing process ended

Writing error: tag e007000002199de8
tag is ignored (invalid response after a
write command)

Clear Info Clear Error Launch

```

(a) Événements générés par le lecteur RFID lors d'une opération d'écriture

DB	Hex	ASCII
0:	FF F0 02 03	ÿü..
1:	00 00 01 34	...4
2:	D7 20 8F 6D	* m
3:	00 00 46 E8	..Fè
4:	31 2E 61 38	Mate
5:	2E 31 34 3D	rial
6:	3F 3B 54 31	Defi
7:	2E 61 38 2E	niti
8:	31 35 3D 53	on.D
9:	61 77 79 65	efin
10:	72 20 3B 54	itio
11:	31 2E 61 38	n.MD
12:	2E 31 36 3D	060
13:	44 69 63 61	1122
14:	69 6E 73 6F	3=Te
15:	FF F0 02 03	xtil
16:	6E 20 3B 54	e wh

(b) Détail du contenu mémoire d'un tag RFID après avoir stocké le data item $T_{\text{MatDef}\{3,2\}}$

FIGURE 5.14: Événements générés lors de l'opération d'écriture et détail du contenu mémoire d'un tag

La FIGURE 5.14(b) détaille le contenu mémoire du tag de référence $e0070000024dcbedd$ qui contient une partie du data item $T_{\text{MatDef}\{3,2\}}$. Les tags RFID ont généralement une mémoire découpée en blocs de données. La mémoire des tags Omron V720-D52P03 est découpée en 16 blocs de données (identifiés par les initiales DB - *Data Block* dans la FIGURE 5.14(b)), chaque bloc ayant une taille de 4 octets. Puisque les six premiers champs de l'en-tête définies pour disséminer les informations entre tags occupe 128 bits (soit 16 octets), les 4 premiers blocs mémoire de chaque tag RFID lui sont réservés (4×4 octets). Les 12 blocs restants sont utilisés pour stocker le contenu des data items. Cependant, rappelons qu'un index est ajouté au début de chaque data item afin de pouvoir le localiser au sein de la base de données. Nous pouvons voir l'index relatif à $T_{\text{MatDef}\{3,2\}}$ dans la FIGURE 5.14(b) ($\text{MaterialDefinition.Description.MD060111223=}$), occupant les blocs 4 à 12 inclus. À partir du 14^{ème} bloc, le contenu du data item $T_{\text{MatDef}\{3,2\}}$ est ajouté, à savoir la chaîne de caractères : `Textile which is provided with two layers of protective coatings.`

Nous constatons que ce tag ne dispose pas d'une mémoire suffisante pour stocker l'ensemble de la chaîne, par conséquent, le reste est stocké dans un autre tag RFID⁷.

La bobine de textile communicant embarque désormais les informations jugées pertinentes au point d'écriture n°1. Notons que l'opérateur n'a constaté aucun défaut sur cette bobine et n'a donc ajouté aucune donnée supplémentaire⁸. Cette bobine poursuit son processus industriel, dont la prochaine étape est l'activité « Découpe ».

5.2.3 Point de lecture n°1 : activité « Découpe »

La bobine de textile se présente à l'opération de découpe. L'architecture « Lecture/Données localisées » est mise en place (cf. FIGURE 4.6). Deux points sont traités dans ce qui suit :

- la relecture des 80 data items et la reconstruction des tuples produits via l'application « Java Split »,
- la localisation des tags dans la matière via l'application « Java Loc ». La précision de localisation des tags est également analysée.

Étape 3 : exécution de l'opération relecture

L'ensemble des data items portés par la bobine est reconstruit et affiché à l'utilisateur via l'application « Java Split ». Rappelons que la reconstruction des informations disséminées dans le textile communicant s'appuie sur l'en-tête protocolaire ajouté dans chaque tag RFID et sur les index ajoutés en début de chaque data item. Les FIGURE 5.15(a) et FIGURE 5.15(b) montrent respectivement les tuples des tables `MaterialDefinition`⁹ et `Person` qui ont été extraits et reconstruits. Le nombre de tuples affichés dépend d'une part, du nombre de tuples identifiés via l'algorithme `RetrieveData` et d'autre part, de leur degré de pertinence. Nous observons que dans la table `Person`, deux tuples ont été extraits mais que deux data items sont manquants. Deux raisons peuvent être à l'origine de ce constat :

- ces deux data items ont été jugés non pertinents : ils étaient situés au-delà du 80^{ème} rang dans `ListeAF`,
- ils n'ont pas pu être reconstruits : conséquence d'une erreur de lecture, d'une zone de matière non lue, etc.

Une fois reconstituées par l'application « Java Split », les tables sont à disposition de l'utilisateur. Le textile communicant agit alors comme une base de données et peut répondre à des requêtes sur les tables reconstituées sans nécessiter d'accès à la base de données initiale.

7. il ne sera pas nécessaire d'ajouter l'index du data item `TMatDef{3, 2}` sur ce second tag.

8. « donnée supplémentaire » faisant référence aux données écrites à la volée dans les « zones matières » des défauts.

9. il s'agit du tuple 3 de `MaterialDefinition` dans l'exemple initial de la FIGURE 3.3.

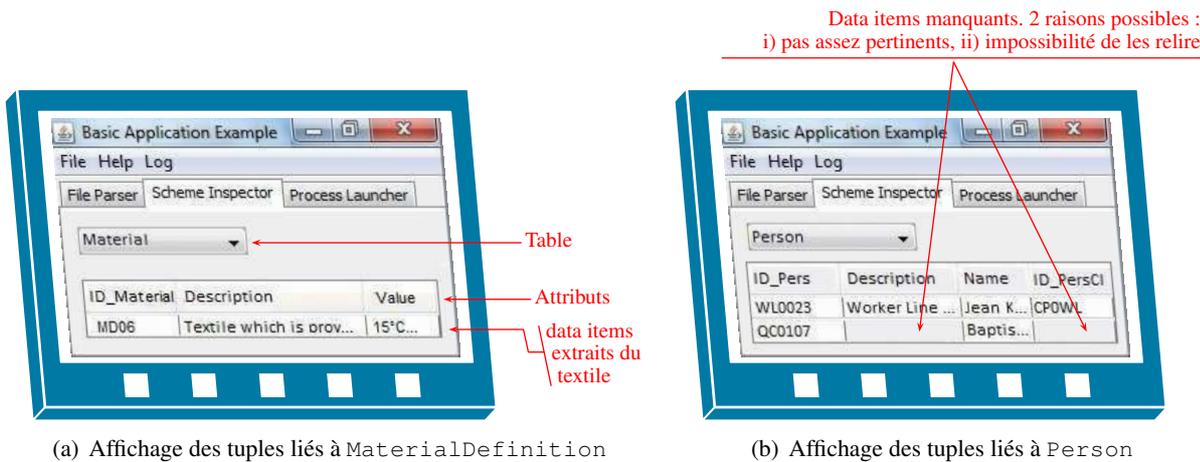


FIGURE 5.15: Application « JAVA Split » permettant de reconstituer les données extraites du produit

Étape 3 : localisation des informations sur la bobine

La seconde application « Java Loc » permet de reconstituer virtuellement la cartographie des tags RFID disséminés dans le textile communicant. Dans notre étude, sept lecteurs RFID sont alignés sur une rampe. Le textile défile à une vitesse¹⁰ de 5 m/s. Les lecteurs RFID renvoient les événements vers un « backend », qui est en charge de collecter/fusionner l'ensemble des événements. Ces événements sont utilisés par l'algorithme introduit dans l'Annexe V pour calculer la position des tags dans le textile. La FIGURE 5.16 offre un aperçu de l'interface graphique de « Java Loc ». La loupe n°1 met en avant la liste des tags RFID présents dans le textile (référéncés par leur ID, lesquels proviennent des événements générés par les lecteurs RFID). Les tags de cette liste peuvent être sélectionnés et ainsi, il est possible de voir où ils sont situés dans le textile (voir la loupe n°2 dans la FIGURE 5.16 : tag de référence¹¹ e0070000024dcbedd). La loupe n°3 montre qu'il est possible d'obtenir les coordonnées des tags dans le textile, ce qui peut être utile lorsque nous désirons recalculer un plan de découpe, par exemple, en cas de présence de défauts.

La cartographie des tags dans le textile (obtenue via l'algorithme) est présentée dans la FIGURE 5.17. Elle fournit la position réelle des tags dans le textile, ainsi que les deux positions retournées par l'algorithme¹². Ces résultats sont obtenus lorsque la rampe de lecteurs se trouve à 60mm au dessus du textile. La position d'un tag est détaillée dans la FIGURE 5.17, dont la distance euclidienne séparant la position réelle du tag et sa position calculée est égale à 7.01mm pour la plus proche solution (corde 2) et 33.44mm pour la plus éloignée (corde 1). Le zoom réalisé dans cette même figure (en bas à droite) montre la position du tag de référence e0070000024dcbedd dans le textile. Nous remarquons que ce sont les mêmes coordonnées que celles affichées dans la FIGURE 5.16 (Loupe n°3).

10. dans ce cas spécifique d'application, la vitesse a été étudiée afin de lire les tags dans des conditions acceptables.

11. il s'agit du tag dont le contenu mémoire a été détaillé précédemment : cf. FIGURE 5.14(b).

12. deux possibilités sont identifiées en y par la méthode de la corde décrite dans l'Annexe V.

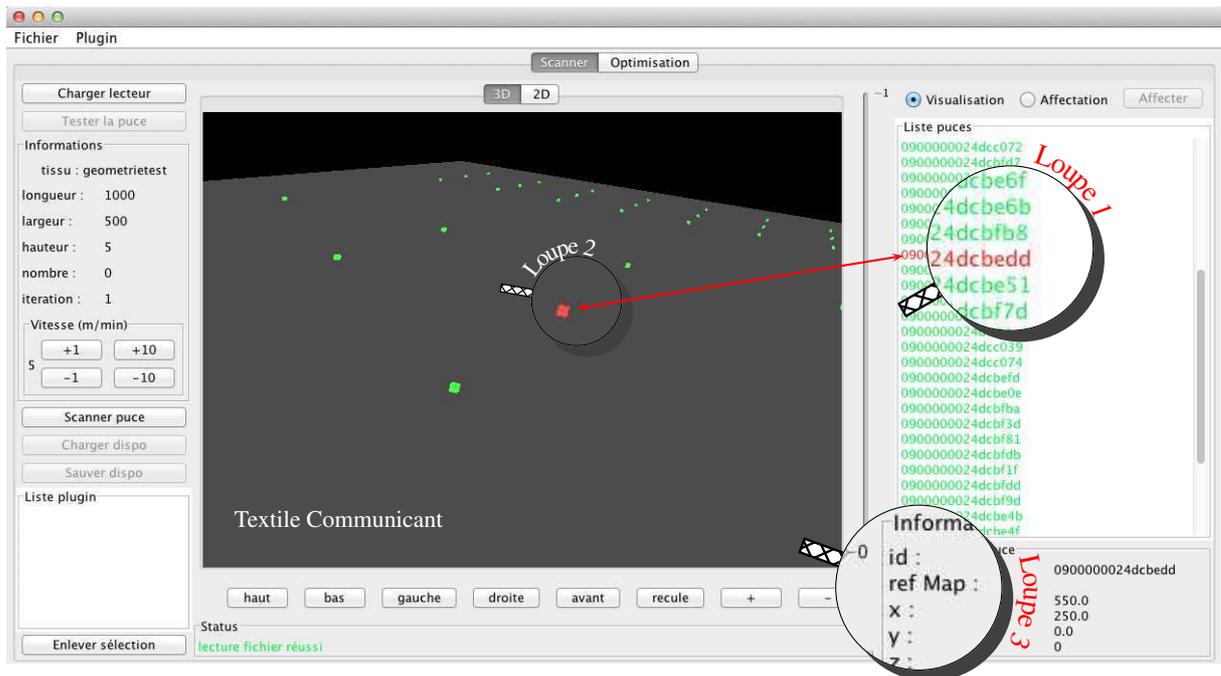


FIGURE 5.16: Application « JAVA Loc » permettant de localiser les tags RFID

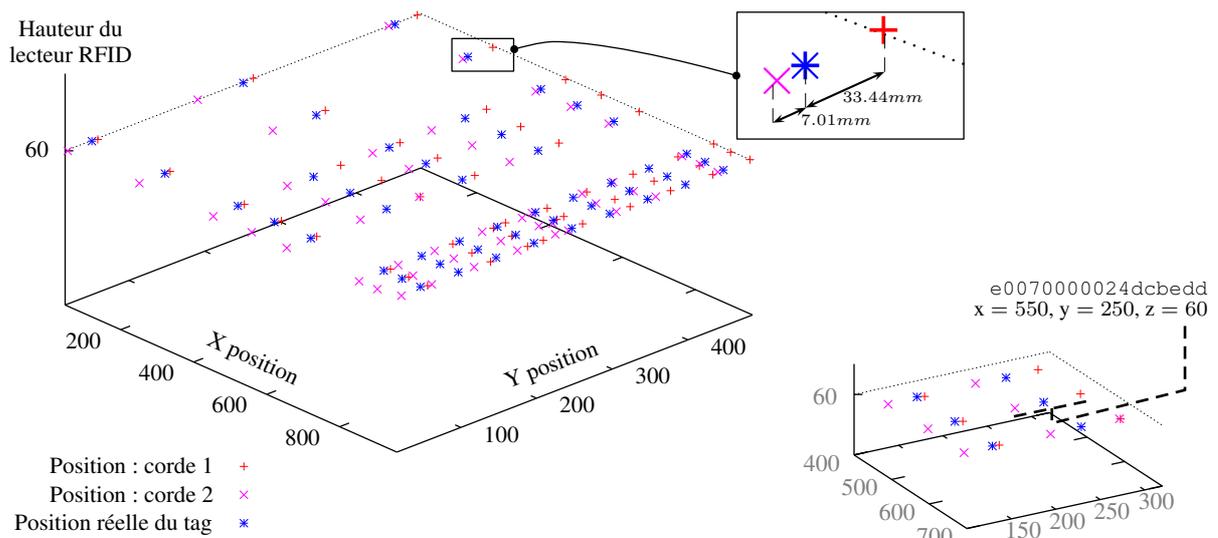


FIGURE 5.17: Cartographie virtuelle des tags présents dans la bobine de textile communicant

La distance minimale séparant la position réelle du tag par rapport à celle calculée est retenue pour chaque tag. Ces distances sont synthétisées dans la FIGURE 5.18 sous la forme de « boîtes à moustaches ». Cette étude est réalisée pour trois hauteurs de rampe distinctes, à savoir lorsque la rampe se trouve à 60, 80 et 100mm au-dessus du textile. Nous constatons que quel que soit le scénario, 50% des erreurs (valeurs comprises entre le 1^{er} et le 3^{ème} quartiles) se situent dans la même tranche de valeurs (entre 10 et 30mm). Les médianes indiquent que les erreurs de positionnement sont moindres avec une hauteur de rampe de 80mm. Cependant, la marge d'erreur pour ces trois hauteurs n'est pas très significative. De

manière générale, l'algorithme commet une erreur $\approx 20mm$ sur le calcul des positions des tags dans le textile, bien que cela importe peu pour cette phase de validation du concept.

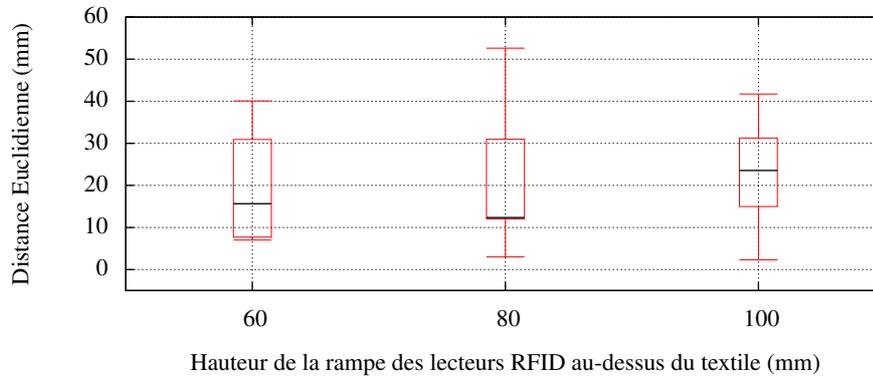


FIGURE 5.18: Précision de calcul des positions des tags dans le textile

Après extraction des informations, la bobine de textile est découpée en pièces de textile, qui sont destinées à l'opération « Collage » pour la confection des assises de chaises (cf. FIGURE 5.1). Cette activité nous intéresse peu puisqu'elle n'implémente pas le processus de dissémination des informations (ni en lecture, ni en écriture). La prochaine section détaille le point d'écriture n°2 visant à sélectionner et à embarquer les informations pertinentes sur les assises de chaise.

5.2.4 Point d'écriture n°2

Dans cette section, seuls les résultats liés à l'étape 2 du processus de dissémination sont détaillés. L'étape 3 est similaire à celle du point d'écriture n°1. Les résultats sont une nouvelle fois calculés pour les approches AHP-Floue et Chan-Roddick.

Étape 2 : ajustements et expertises des différents critères

Les expertises liées au critère *Contextual* ont déjà été effectuées et présentées dans la FIGURE 5.4 pour la phase 2 du cycle de vie du textile (phase dans laquelle se trouve le point d'écriture n°2). Concernant les critères *Model-Based* et *Data-Size*, nous considérons que les ajustements sont similaires à ceux du point d'écriture n°1. Les seules expertises qu'il nous reste à détailler sont :

- i. les expertises liées aux importances des critères,
- ii. les expertises liées au critère *Enumeration*.

i. Un expert issu de la compagnie mettant en œuvre le point d'écriture n°2 s'intéresse à l'importance des critères. Seuls les vecteurs propres $\tilde{\Lambda}_\rho$ et Λ_ρ résultants des *pairwise comparisons* sont présentés dans les FIGURE 5.19(a) et FIGURE 5.19(b) respectivement. Au regard de ces deux vecteurs, nous constatons que les critères *Contextual* et *Data Size* sont les plus importants dans l'ordre de priorité suivant :

- au regard de $\tilde{\Lambda}_\rho$: $\tilde{\lambda}_\rho(C_m) \gtrsim \tilde{\lambda}_\rho(C_e) \gtrsim \tilde{\lambda}_\rho(C_s) \gtrsim \tilde{\lambda}_\rho(C_c)$,

- au regard de $\Lambda_\rho : \lambda_\rho(C_m) < \lambda_\rho(C_e) < \lambda_\rho(C_c)$.

Du fait que le critère *Data Size* est important à ce point d'écriture dans AHP-Floue, on peut certainement s'attendre à des résultats/listes moins similaires que dans le précédent point d'écriture.

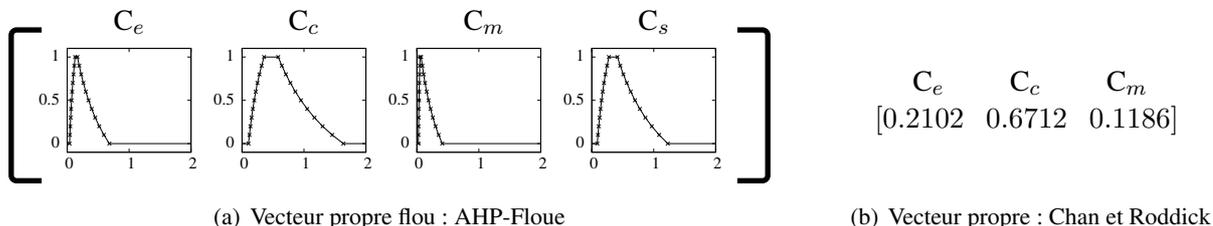


FIGURE 5.19: Vecteur propre indiquant l'importance de chaque critère : Phase 2

ii. Deux décideurs procèdent à l'énumération des attributs qu'ils souhaitent stocker sur les assises de chaise. Ces énumérations sont résumées dans la TABLE 5.3. En appliquant l'approche AHP-Floue, nous obtenons un sous-ensemble flou propre à chaque attribut comme illustré dans la colonne intitulée « AHP-floue ». Les énumérations moyennes sont fournies dans la colonne « Chan-Roddick ».

TABLE 5.3: Attributs énumérés par les experts de la compagnie implémentant le point d'écriture n°2

	Nom Table	Nom Attribut	Décideur	Choix	Score de l'attribut après fusion des avis	
					AHP-Floue	Chan-Roddick
G ₄	Equipment	IDEquipment	D1 D2	1 1		$\frac{1+1}{2} = 1$
		Value	D1 D2	1 0.5		$\frac{1+0.5}{2} = 0.75$
		Value Unit	D1 D2	1 0.5		$\frac{1+0.5}{2} = 0.75$
	EquipmentClass	IDEquipmentClass	D1 D2	0.5 0.5		$\frac{0.5+0.5}{2} = 0.5$
		Description	D1 D2	0.5 0		$\frac{0+0.5}{2} = 0.25$
G ₁	MaterialLot	IDMaterialLot	D1 D2	1 1		$\frac{1+1}{2} = 1$
		Description	D1 D2	0.5 0		$\frac{0+0.5}{2} = 0.25$

Tous les ajustements ont été réalisés. Désormais, lorsqu'une assise de chaise se présente au point d'écriture n°2, les étapes 2 et 3 du processus sont exécutées.

Étape 2 : exécution

Lorsqu'une assise de chaise se présente au point d'écriture n°2, la machine relève la référence portée par celle-ci. Dans notre étude, il s'agit de la référence renvoyant à la table MaterialLot et au tuple de clé primaire : L00TP-B101¹³. L'algorithme `RetrievalData`, ainsi que les deux approches d'évaluation, sont ensuite respectivement exécutés.

13. la référence produit diffère de celle du point d'écriture n°1, laquelle avait pour valeur L00TB-B101.

→ Exécution de l'approche AHP-Floue

La liste obtenue à partir de l'approche AHP-Floue (ListeAF) est détaillée dans la FIGURE 5.20. 208 data items composent cette liste, contre 108 pour le point d'écriture n°1. Ceci s'explique en partie par le fait que les informations liées aux assises de chaises sont à la fois relatives au textile communicant et aux planches de bois, contrairement au point d'écriture n°1 dont les informations étaient uniquement relatives à la bobine de textile. Les data items provenant de la table contenant les attributs énumérés (c.-à-d., Equipment, MaterialLot et EquipmentClass, cf. FIGURE 5.3) font encore partie des 18 premiers rangs de ListeAF. Cependant, contrairement au point d'écriture n°1, les informations énumérées ne se situent plus en début de liste (p. ex., les data items aux rangs 5, 6 ou encore 10, 11, 12 ne sont pas issus d'attributs énumérés), car le critère *Enumeration* n'est plus aussi important. Les data items classés aux rangs 5 et 6 appartiennent tous les deux à une entité issue du groupe G_4 (c.-à-d., le groupe le plus important) et sont tous les deux de très petite taille (1 caractère, soit 1 octet). Par conséquent, ils sont très avantagés car le critère *Data Size* est prioritaire sur *Enumeration* et *Model-Based*.

Rang	Nom Table	Nom Attribut	Clé Primaire	Contenu du data item	Pertinence
1	Equipment	Value	ETCM3A0001	60	0.8013
2	Equipment	Value	EBSM1E0001	80	0.8013
3	Equipment	IDEquipment	ETCM3A0001	ETCM3A0001	0.6968
4	Equipment	IDEquipment	EBSM1E0001	EBSM1E0001	0.6968
5	EquipmentSegmentSpecification	Quantity	ESS0004	1	0.6839
6	EquipmentSegmentSpecification	Quantity	ESS0005	1	0.6839
7	Equipment	Value Unit of Measure	ETCM3A0001	Pieces/Hour	0.6787
8	Equipment	Value Unit of Measure	EBSM1E0001	Pieces/Hour	0.6787
9	MaterialLot	Quantity	L00TP-B101	0	0.5959
10	ProductSegment	Duration	PSTBC000B	0.5	0.5922
11	ProductSegment	Duration	PSSHO00BR	0.2	0.5922
12	ProductSegment	Duration	PSSHO00BG	0.2	0.5922
13	MaterialLot	IDMaterialLot	L00TP-B101	L00TP-B101	0.5908
14	EquipmentClass	IDEquipmentClass	ECTCM01	ECTCM01	0.5656
15	EquipmentClass	IDEquipmentClass	ECBSM01	ECBSM01	0.5656
16	ProductionOrder	RequestState	PO0001DHBR	Done	0.5615
17	SegmentResponse	IDSegmentResponse	PSR0001	PSR0001	0.5515
18	SegmentResponse	IDSegmentResponse	PSR0003	PSR0003	0.5515
...
...
...
203	ActualPersonSegment	Quantity Unit of Measure	APS0001	Hours/Piece	0.1306
204	ActualPersonSegment	Quantity Unit of Measure	APS0003	Hours/Piece	0.1306
205	PersonClass	Description	PCPLW01	Production Line Worker	0.1303
206	PersonSegmentSpecification	IDPerson	PSS0001	{PLW01SK001,PLW01JD001,PL...}	0.1303
207	PersonSegmentSpecification	IDPerson	PSS0004	{PLW01SK001,PLW01JD001,PL...}	0.1303
208	PersonSegmentSpecification	IDPerson	PSS0005	{PLW01SK001,PLW01JD001,PL...}	0.1303

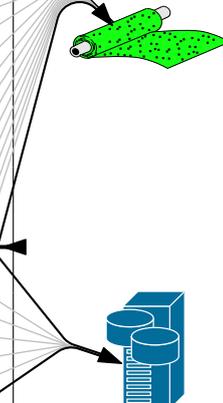


FIGURE 5.20: ListeAF ordonnée du data item le plus pertinent au moins pertinent : Point d'écriture n°2

Le diagramme de « boîtes à moustaches » de la FIGURE 5.21(a) permet d'observer que les tables énumérées ne se détachent plus autant que dans le précédent point d'écriture (voir p. ex. MaterialLot). Dans ce diagramme, les groupes d'entités forment des niveaux très marqués :

- 1^{er} niveau/le plus bas (pertinence ≈ 0.2) : composé des tables appartenant à G_2 « Personnel » (cf. code couleur sur les abscisses),
- 2^{ème} niveau (pertinence ≈ 0.3) : composé des tables appartenant à G_1 « Produit », excepté MaterialLot qui est plus pertinent car deux de ses attributs sont énumérés,
- 3^{ème} niveau (pertinence ≈ 0.45) : composé des tables appartenant à G_3 « Equipement »,

- 4^{ème} niveau (pertinence ≈ 0.55) : composé des tables appartenant à G_4 « Production »,

Ces niveaux respectent les recommandations contextuelles formulées par les experts (cf. ordre d'importances des 4 groupes d'entités dans la FIGURE 5.4) qui sont considérées comme les recommandations les plus prioritaires (critère *Contextual* prioritaire sur les autres, cf. FIGURE 5.19). Les étendues sont donc plus larges que dans le point d'écriture n°1 car dans chaque table sont présents des data items de plus ou moins grande taille, ce qui impacte fortement leur pertinence.

L'assise de chaise située au point d'écriture n°2 ne dispose pas d'une mémoire aussi conséquente que la bobine de textile (cf. point d'écriture n°1). En effet, la taille de la pièce de l'assise est plus petite et par conséquent, possède moins de tags RFID. Seuls les 35 premiers data items de ListeAF peuvent être stockés sur cette assise. Le diagramme circulaire de la FIGURE 5.21(b) montre que 74% de ces 35 data items (soit 26 data items) proviennent d'entités appartenant au groupe G_3 (groupe le plus important). Nous observons qu'aucune information relative au groupe G_2 (« Personnel ») n'est stockée sur l'assise. Enfin, seules 4% des informations stockées sur l'assise sont issues d'entités du groupe G_1 . Ces informations sont principalement liées à l'énumération des attributs de la table `MaterialLot`.

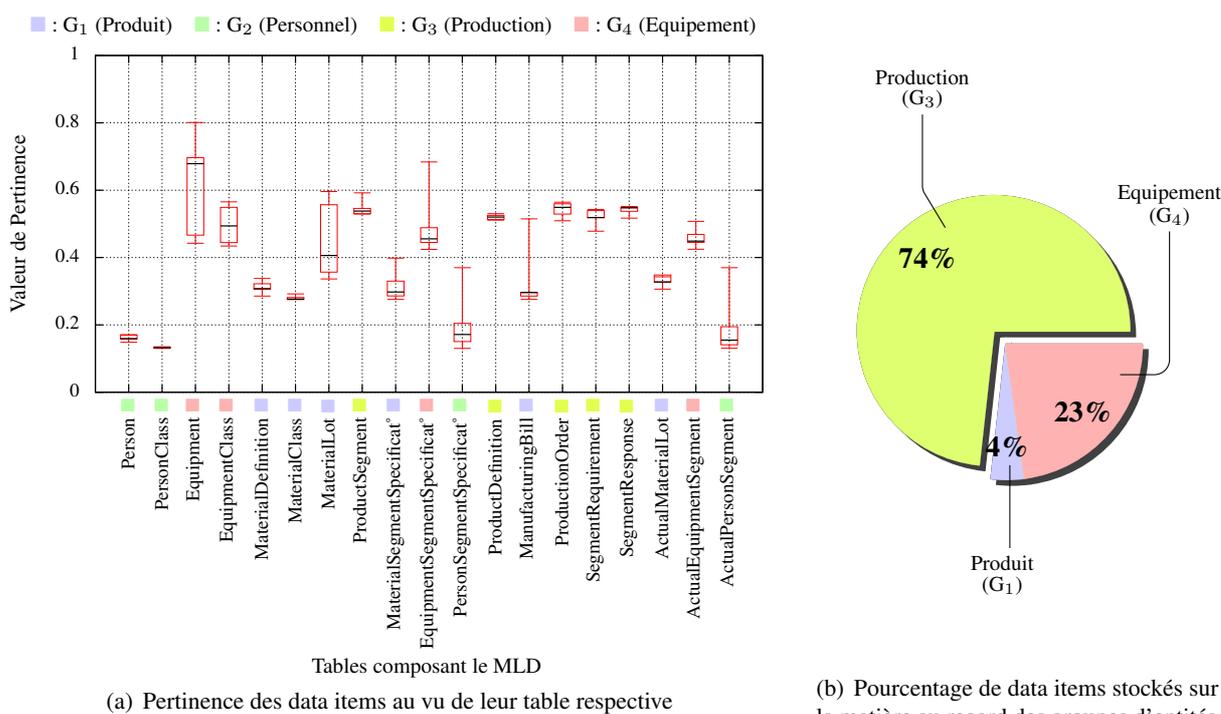


FIGURE 5.21: Représentation de ListeAF sous forme de diagramme : Point d'écriture n°2

→ Exécution de l'approche Chan-Roddick

Pour la liste obtenue par l'approche de Chan-Roddick (ListeCR), seuls les diagrammes sont présentés. Nous constatons tout d'abord dans la FIGURE 5.22(a) qu'un data item se détache dans la table `Equipment` (pertinence ≈ 0.65). Il s'agit du même data item que celui de rang 1 dans ListeAF. Le comportement général des tables est similaire à celui de l'approche AHP-Floue car nous retrouvons une

nouvelle fois les différents niveaux relatifs aux importances des groupes d'entités. Cependant, cette fois-ci, ces niveaux sont beaucoup moins prononcés que dans la FIGURE 5.21(a). Le fait que les frontières entre niveaux soient resserrées engendre une probabilité plus forte d'embarquer des informations issues de différents groupes. C'est en effet ce que fait ressortir la FIGURE 5.22(b), où le diagramme circulaire indique que des informations provenant de G_2 sont également embarquées sur l'assise (bien que cela ne représente que 5%), contrairement à l'approche AHP-Floue qui n'embarquait aucune information issue de ce groupe.

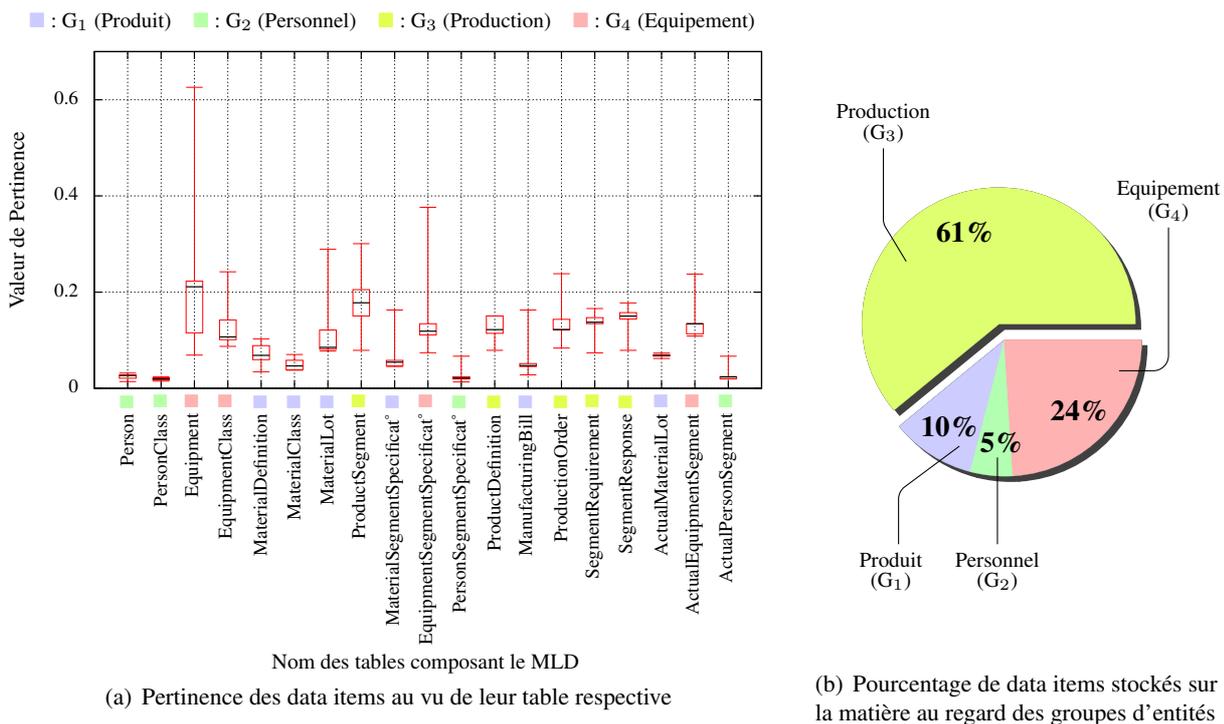


FIGURE 5.22: Représentation de ListeCR sous forme de diagramme : Point d'écriture n°2

La première comparaison fournie par la FIGURE 5.23(a) montre que le classement issu des deux approches diffère de manière plus prononcée dans le point d'écriture n°2 que dans le point d'écriture n°1. En effet, la dispersion est plus importante dans la FIGURE 5.23(a) que dans la FIGURE 5.13(a). Cette dispersion est principalement due à l'importance du critère *Data Size*, car ce dernier est le plus important dans l'approche AHP-Floue au point d'écriture n°2. Or, le poids de ce critère n'est pas modifiable dans l'approche de Chan et Roddick.

La deuxième comparaison, présentée dans la FIGURE 5.23(b), indique que quel que soit le nombre de data items embarqués sur le produit et quelle que soit l'approche utilisée, ListeAF et ListeCR ont au minimum 55% de data items communs, ce qui est inférieur au degré de similarité obtenu au point d'écriture n°1 (70% minimum). Dans notre étude, les 35 premiers data items de ListeAF sont stockés sur l'assise de chaise. Si nous décidions d'utiliser l'approche de Chan et Roddick (c.-à-d., considérer

ListeCR), seulement 60% des data items seraient identiques avec les 35 de ListeAF (soit 21 data items sur 35).

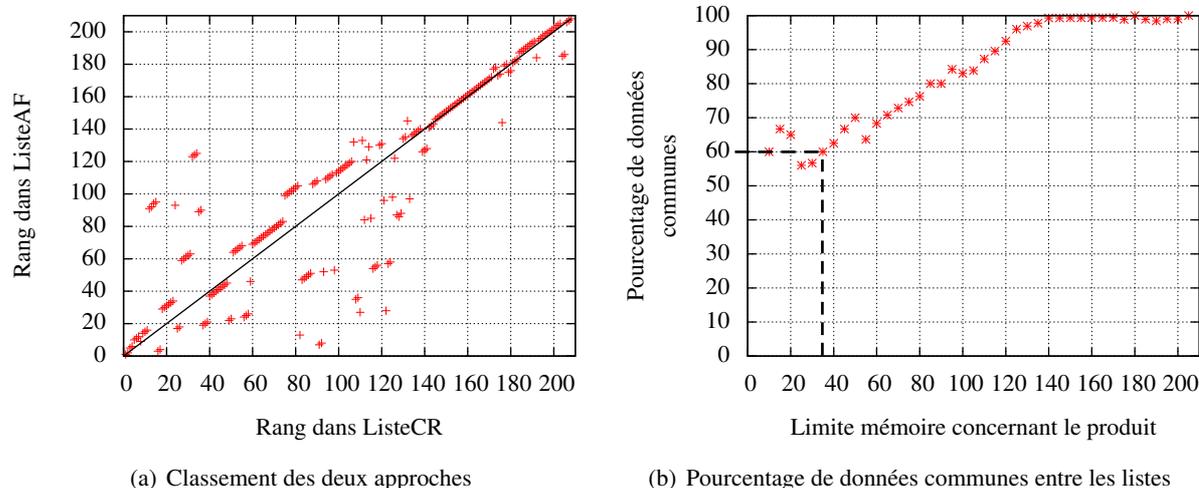


FIGURE 5.23: Comparaison des approches Chan-Roddick et AHP-Floue : Point d'écriture n°2

5.3 Synthèse et conclusion de l'étude

Ce cas d'étude a permis d'avoir un aperçu de l'intégration du processus de dissémination des informations lors du cycle de vie d'un textile communicant. Les outils logiciels et matériels (applications, protocoles, prototypes, etc.) développés pour assurer les fonctions des différentes étapes du processus ont pu être mis en œuvre. Ce cas d'étude a également été l'opportunité d'établir des premières comparaisons entre les approches « Chan-Roddick » et « AHP-Floue ». Comme cela a été souligné en début de chapitre, des comparaisons supplémentaires sont menées dans l'Annexe VII. Les conclusions de l'ensemble des comparaisons sont tout d'abord rappelées dans la section 5.3.1, puis une synthèse de l'ensemble des résultats de nos études est réalisée sous forme de tableau dans la section 5.3.2.

5.3.1 Conclusions des études comparatives de l'Annexe VII

L'Annexe VII propose trois scénarios. Le premier compare les approches lorsqu'un grand nombre d'experts (p. ex. 10 experts) réalisent une évaluation et qu'une minorité (p. ex. 1 expert sur les 10) exprime un souhait opposé aux autres. En d'autres termes, nous cherchons à identifier le comportement des approches face à des avis « isolés ». Deux autres scénarios sont ensuite étudiés sur la base d'un avis unique, évitant ainsi l'ajout de l'étape d'agrégation des avis dans l'approche de Chan-Roddick (*cf.* FIGURE 5.2).

Scénario 1 : avis « isolés »

Dans ce scénario, 10 experts énumèrent les attributs qu'ils souhaitent stocker sur la matière : 9 experts expriment le même souhait (ils ne les désirent pas) et un unique expert exprime le souhait opposé (il désire tous ces attributs). Puisque nous nous basons sur la moyenne des avis dans Chan-Roddick et qu'un seul expert sur 10 énumère ces attributs, la moyenne de chaque attribut (c.-à.-d, $\phi_e(l)$) est par conséquent égale à 0.1 ($\frac{0+0+0+0+0+0+0+0+0+1}{10} = 0.1$). Cette valeur tend vers zéro et rend ainsi l'énumération des attributs quelque peu « négligeable ». À l'inverse, l'approche AHP-Floue utilise des intervalles de valeurs $[0; 1]$ afin de tenir compte des avis « isolés ».

Les résultats obtenus reflètent clairement les propos tenus dans la section 5.2.1, à savoir que le calcul de l'avis moyen que nous avons mis en place dans l'approche de Chan et Roddick pour agréger les avis des experts, convient lorsque le nombre d'experts est faible. Cependant, cette méthode ne permet pas de tenir compte des avis « isolés » parmi un grand nombre d'opinions.

Scénario 2 et 3 : avis unique

Dans ces scénarios, nous comparons les approches de Chan-Roddick et AHP-Floue sur la base d'un avis unique comme illustré dans la FIGURE 5.2(a). Deux scénarios distincts sont proposés dont la différence concerne le critère *Enumeration*. Dans le scénario 1, une grande quantité d'attributs est énumérée (une quarantaine), contre six dans le scénario 2. Pour chaque scénario, les ajustements concernant les critères *Enumeration*, *Contextual*, *Model-Based* et *Data Size* n'évoluent pas, uniquement les importances entre critères évoluent. Le but est d'étudier le comportement des deux approches comme, par exemple, identifier si des critères particuliers tendent à rendre les listes des deux approches similaires et inversement.

Les études des deux scénarios tendent à montrer que les deux approches sont similaires lorsque l'on considère un avis unique. Toutefois, des différences surviennent lorsque l'on favorise extrêmement le critère *Data size* puisque le modèle de Chan et Roddick ne permet pas de prendre en compte cette préférence (repose sur le comportement de la fonction logarithme népérien).

5.3.2 Tableau de synthèse

Les conclusions de nos études permettent de relever les avantages et inconvénients, ou du moins les aspects comportementaux des deux approches proposées. La TABLE 5.4 synthétise ces aspects en indiquant, d'une part, si l'approche répond positivement ou négativement à chaque aspect et, d'autre part, en justifiant cette évaluation sur la base des études menées dans le Chapitre 5 et dans l'Annexe VII.

Nous constatons au vu de ce tableau que l'approche AHP-Floue est plus complète que l'approche de Chan et Roddick. Cependant, l'approche AHP-Floue est plus complexe à mettre en œuvre (cf. TABLE 5.4). En effet, l'ajout de nouveaux critères est plus complexe du fait de l'utilisation de la théorie des sous-ensembles flous, mais elle offre une plus grande flexibilité pour intégrer de nouveaux critères qui doivent prendre en compte plusieurs avis ou encore des incertitudes.

Le dernier point important à noter (non spécifié dans le tableau) est qu'au vu de toutes les figures de nos études (figures issues du Chapitre 5 et de l'Annexe VII), quels que soient les réglages effectués, le critère favorisé, le nombre d'experts considéré, il y a une limite de rang 60 qui garantit que quelle que soit l'approche utilisée (c.-à-d. si nous stockons sur le produit plus de 60 data items issus de ListeCR ou ListeAF¹⁴), nous avons au minimum 70% des informations qui sont communes. Bien évidemment, cette limite est principalement fonction du nombre de data items à évaluer¹⁵. Toutefois, si nous sommes capable de déterminer cette limite pour un scénario quelconque et si nous connaissons l'espace mémoire disponible sur la matière communicante, il est possible de garantir à l'utilisateur que quelle que soit l'approche qu'il met en œuvre, $X\%$ des informations stockées sur le produit seront identiques.

Cette étude ne permet d'identifier quelle est, parmi les deux approches, celle qui fournit la meilleure distribution des données, il s'agit uniquement d'un indicateur de similarité entre les approches. Aujourd'hui, aucun travail n'a été mené pour évaluer la performance de l'approche de Chan et Roddick ou la nôtre. Nos futurs travaux visent à mettre en place un *benchmark* basé sur une application réelle, avec des acteurs du métier/de l'application, afin d'établir la distribution optimale¹⁶, ce qui permettra d'introduire des indicateurs de performance des approches.

14. les 60 premiers de la liste.

15. nos listes évaluent une centaine de data items. Il serait intéressant de voir si une limite récurrente existe quelque soit le nombre de data items.

16. il s'agit de déterminer « à la main » les informations qui devraient être stockées sur le produit au vu des attentes des acteurs, puis de regarder si les approches Chan-Roddick et AHP-Floue fournissent ou non les mêmes résultats.

TABLE 5.4: Tableau comparatif entre l'approche de Chan et Roddick et l'approche AHP-Floue

		Chan et Roddick Commentaires		AHP-Floue Commentaires
Prise en compte de groupes d'experts	–	L'approche ne permet pas, initialement, de tenir compte de plusieurs avis pour une même évaluation. Il est nécessaire de définir sa propre méthode d'agrégation des avis, comme nous l'avons proposé dans le cas d'étude du Chapitre 5 (calcul de l'avis moyen).	++	Cette approche permet de prendre en compte plusieurs avis pour une même évaluation via un processus proposant deux types d'agrégation des avis en fonction des circonstances/du critères (<i>cf.</i> FIGURE 3.11).
Gestion d'avis « isolé »	--	Les études menées dans l'Annexe VII montrent que lorsque l'approche doit gérer un grand nombre d'avis (≈ 10) et qu'elle est confronté à des avis « isolés », elle n'arrive pas à satisfaire tous les acteurs.	++	Les études menées dans l'Annexe VII montrent que cette approche permet de tenir compte d'avis « isolé ».
Mise en place de l'approche	++	Approche très simple de mise en oeuvre puisque les calculs reposent sur une équation unique et des valeurs « nettes ».	--	Approche complexe à mettre en oeuvre (en comparaison à l'approche de Chan et Roddick) car les calculs utilisent les sous-ensembles flous et le concept d' α -coupes.
Évolutivité	++	L'ajout de nouveaux critères est relativement simple pour les mêmes raisons qu'énoncées ci-dessus.	+	L'ajout de nouveaux critères est plus complexe que l'approche de Chan et Roddick. Cependant, l'approche AHP-Floue permet de prendre en compte de manière immédiate les évaluations faites par des groupes d'experts. Ceci peut être avantageux lorsque l'on désire ajouter de nouveaux critères au modèle, dans lesquels nous devons tenir compte de plusieurs avis ou encore d'incertitudes.
Gestion de la taille des informations	–	La méthode favorise les données de petites tailles et pénalise de manière moins prononcée les données de grandes tailles (comportement de la fonction logarithme). L'utilisateur ne peut cependant pas de gérer les préférences concernant la taille des informations qu'il souhaite stocker sur le produit.	+	Il s'agit dans cette approche d'un critère comme les autres, ce qui permet à l'utilisateur de gérer ses préférences.
Prise en compte de l'incertitude lors de l'évaluation	--	Impossible.	+	Selon le type d'agrégation d'avis défini (discuté ci-avant), de l'incertitude sur les évaluations réalisées par les experts peut être ajoutée et ajustée.

Conclusion générale et perspectives

Cette thèse s'inscrit dans le cadre de la gestion du cycle de vie du produit, ou PLM (*Product Lifecycle Management*). Ces travaux de thèse proposent un processus de dissémination des informations relatives aux produits. Ce processus vise à établir le schéma de distribution optimal des informations qui peuvent être stockées sur le produit lui-même ou sur les bases de données. Un nouveau concept est introduit, changeant radicalement la manière de voir le produit et la matière. Ce concept consiste à donner la faculté au produit d'être intrinsèquement et intégralement « communicant », de manière à ce que le produit puisse stocker de l'information sur l'ensemble de la matière le composant. Prendre comme postulat que la matière est « communicante » ouvre un champ de recherche considérable, tant d'un point de vue technologique que scientifique. C'est dans ce contexte que cette thèse a été initiée.

Le Chapitre 1 retrace l'évolution des produits/objets intelligents dans notre société, ainsi que les travaux relatifs à la gestion du cycle de vie du produit. Une étude rigoureuse est également menée dans le but de décliner les principaux champs de recherche inhérents au concept de « matière communicante » à investiguer dans l'avenir.

Le Chapitre 2 propose un état de l'art de la distribution d'informations dans des environnements fixes et/ou mobiles. Le corpus de documents met en évidence que peu de travaux disséminent les informations relativement au contexte d'usage de la base de données. En d'autres termes, les approches ne tiennent pas compte de la sémantique des informations lors de la dissémination. Cette notion de sélection d'informations basée sur le contexte est le pilier central de nos travaux et justifie les deux questions majeures auxquelles cette thèse apporte des réponses, c'est-à-dire :

- quelles informations doivent être stockées sur le produit durant son cycle de vie ?
- selon quels critères la pertinence à stocker ces informations sur le produit est-elle évaluée ?

Pour répondre à ces deux questions, nous proposons un processus de dissémination des informations qui se compose de trois étapes clés :

1. la mise en place de l'architecture de bases de données (fait appel aux travaux présentés dans le Chapitre 2),
2. l'identification des informations relatives au produit dans la base de données et l'évaluation de leur pertinence à être (ou ne pas être) répliquées/embarquées sur le produit communicant. Cette évaluation est menée à l'aide d'une approche de la littérature, proposée par (Chan and Roddick, 2003), puis à l'aide d'une approche développée dans cette thèse, basée sur AHP-Floue. Ces deux

approches sont détaillées dans le Chapitre 3,

3. le stockage et la relecture des informations portées par le produit communicant. Un outillage particulier est requis pour réaliser ces fonctions, que nous développons et présentons dans le Chapitre 4.

Le Chapitre 5 présente un cas d'étude dans lequel nous identifions pour différentes étapes du cycle de vie d'un textile communicant, les informations jugées pertinentes à y être stockées. Nous présentons également un premier prototype de textile communicant sur lequel nous stockons physiquement ces informations. Ce cas d'étude permet, d'une part, de valider la faisabilité du concept de « matière communicante » et, d'autre part, de comparer les deux approches d'évaluation déployées dans l'étape 2 du processus de dissémination.

Nos études nous permettent d'établir un degré de similitude entre ces deux approches, mais il est difficile d'établir des indicateurs de performance sur lesquels nous pourrions affirmer qu'une approche est meilleure que l'autre. Pour le moment, nous offrons des outils d'aide à la décision aux utilisateurs afin qu'ils puissent valider si les informations sélectionnées par les approches sont celles attendues. Un moyen d'établir des indicateurs de performances serait de définir un *benchmark* basé sur une application réelle (c.-à-d, avec des acteurs du métier), dans laquelle nous pourrions établir un schéma de distribution optimal, qui servirait de base pour l'évaluation de la performance des approches. Cet objectif fait partie des perspectives à court terme de nos travaux.

Les premiers résultats de cette thèse montrent que notre approche se comporte de manière similaire à l'approche de (Chan and Roddick, 2003) pour un scénario « équivalent ». Une fois encore, l'équivalence n'est pas simple à évaluer car notre approche intègre un nouveau critère lié à la taille des informations et intègre la prise en compte de groupe d'experts lors des évaluations (non supporté dans l'approche de Chan et Roddick). Les points de différence majeurs notés entre ces deux approches sont que, contrairement à celle de Chan et Roddick, notre approche autorise l'utilisateur à agir sur ses préférences quant à la taille des informations à stocker sur le produit. De plus, notre approche permet de gérer des avis « isolés » d'experts (p. ex. un expert voulant stocker sur le produit une information que neuf autres ne souhaitent pas), ce qui n'est pas possible dans l'approche de Chan et Roddick bien que cet avis soit tout aussi légitime que les autres. Le principal point faible de l'approche AHP-Floue est relatif à sa mise en place et son évolutivité, lesquelles s'avèrent plus complexes que dans l'approche de Chan et Roddick.

Il est possible d'imaginer des applications réalistes dans une portée de 5 à 10 ans, sur des produits à forte valeur ajoutée où l'impératif de traçabilité est primordial (pièces aéronautiques, composants de bâtiments, ...). Le stockage de données directement sur la matière est une alternative aux approches de type *Cloud Computing* (privilégiant un hébergement des données distant) car elle permet d'assurer la disponibilité des informations en tout temps et en tout lieu. Elle n'est cependant pas à mettre en opposition au domaine du *Cloud Computing* mais bien en complémentarité. En effet, les informations vitales (peu nombreuses) seront directement stockées sur la matière tandis que les données annexes (très nombreuses) pourraient être conservées sur le système de bases de données et accessibles via le réseau

Cloud. L'aspect novateur de notre approche réside dans la manière hybride de stocker les informations relatives au produit (sur la base de données et/ou sur le produit lui-même).

Les perspectives à court terme sont bien évidemment corrélées aux travaux menés dans cette thèse, à savoir :

- fournir des outils complets aux utilisateurs : certains éléments de notre approche nécessitent d'être étudiés, voir complétés par d'autres solutions. Un premier exemple concerne le réglage des coefficients k effectué dans deux des critères de l'approche AHP-Floue (permet à l'utilisateur de négliger des données distantes de la table « produit » ou des données de grandes tailles). Cet ajustement est actuellement réalisé par l'utilisateur lui-même, mais il n'est pas si évident de définir avec certitude la valeur de ces coefficients. Il est possible d'envisager des outils d'apprentissage qui ajusteraient de tels coefficients. Une seconde cible concerne l'approche permettant d'identifier dans quelle étape du cycle de vie le produit se trouve à l'instant t . En effet, notre approche repose sur l'hypothèse que cette information est connue, ce qui en réalité n'est pas si trivial (Day, 1981). Il est donc nécessaire de fournir aux utilisateurs des moyens précis et concrets pour y répondre,
- la mise en place d'un *benchmark* sur une application réelle, avec des acteurs du métier/de l'application, afin d'introduire des indicateurs de performance des approches d'évaluation de la pertinence des informations,
- l'intégration de nouveaux critères comme l'historique des requêtes, qui peut être utile lors d'applications industrielles où l'on connaît précisément les opérations à mener et les informations requises. Nos travaux actuels cherchent à quantifier l'impact de l'utilisation d'une matière communicante sur les performances temporelles d'un processus de fabrication (Kubler et al., 2011a). Ces temps d'accès varient selon l'architecture réseau et les mécanismes de fiabilisation mis en œuvre. De tels mécanismes ont également été étudiés dans nos précédents travaux (Kubler, Robert, Georges and Rondeau, 2012),

Les perspectives à long terme sont quant à elles liées aux champs de recherches technologiques et scientifiques qui ont été déclinés dans le Chapitre 1. Un simple calcul (calcul détaillé dans l'Annexe VIII : FIGURE 59) montre que nous avons répondu à $\approx 50\%$ des problématiques de l'étude¹⁷. Les grands axes qui restent à investiguer sont répertoriés ci-dessous :

- *étude et conception de matériaux communicants* : bien qu'un premier prototype de textile communicant ait été confectionné dans cette thèse, le laboratoire n'a pas l'expérience et les compétences dans un certain nombre de domaines (antenne et physique des ondes, composition des matériaux, etc.). Il est donc nécessaire dans l'avenir de trouver de nouvelles ressources, collaborations pour répondre aux verrous technologiques liés à la réalisation de matériaux communicants,
- *cohérence, consistance et sécurité des informations* : des architectures de communications sont proposées dans cette thèse pour communiquer, stocker et reconstruire des informations sur la ma-

17. pourcentage « subjectif » car certaines problématiques n'ont pas été traitées entièrement, d'autres n'ont pas été identifiées au début du projet (non prise en compte dans le calcul), etc.

tière communicante. Néanmoins, des problèmes de confidentialité, de pérennité et d'intégrité des informations se posent encore. Par conséquent, de nouveaux domaines sont à étudier pour s'attaquer aux verrous scientifiques et technologiques qui apparaissent comme l'agrégation des informations sur la matière, le cryptage des informations, la redondance d'informations au sein de la matière et entre matières communicantes (p. ex. imaginer que des informations portées par une matière communicante se propagent, tel un virus, au sein de cette matière ou dans une autre lorsque les deux se rencontrent),

- *applications ciblées* : bien que cette thèse discute principalement des intérêts du concept de matière communicante et des applications envisagées dans le cadre des chaînes logistiques, beaucoup d'autres secteurs peuvent être ciblés (p. ex., domotique, médical, aéronautique, etc.). Toutefois, il nous reste à confronter nos résultats d'analyse avec l'expertise d'un panel d'industriels dans le but d'obtenir une validation et un élargissement des verrous technologiques et scientifiques relatifs au concept de « matière communicante ».

* * *

Annexe I : Déclinaison des problématiques scientifiques et technologiques

Le concept de « matière communicante » induit un changement important de paradigme. Nous avons souligné le fait qu'un changement de ce type nécessite une analyse s'appuyant sur l'ingénierie système pour poser correctement les bases d'un champ de recherche qui peut s'avérer large. L'ingénierie système est une discipline qui permet de maîtriser la conception et la réalisation de systèmes complexes dans l'objectif de satisfaire les besoins du client durant le cycle de vie du système. Diverses méthodes d'ingénierie existent à ce jour et ont pour objectif de soutenir les décideurs et chefs de projets dans la définition et la conception du système à faire et du système pour faire. Citons :

- l'AFIS (Agence Française d'Information Scientifique) : dans l'ensemble des outils d'Ingénierie Système proposé par cette communauté, on trouve des démarches et outils à usage du chef de projet, adéquats pour concevoir, faire évoluer et vérifier un système apportant une solution économique et performante aux besoins d'un client. La spécification des exigences y est présentée comme une étape importante du processus de conception,
- l'AFAV (Association Française de l'Analyse de la Valeur) : cet organisme a proposé des méthodes et outils intégrés, tels que l'Analyse Fonctionnelle (AF), l'Analyse de la Valeur (AV). L'AF permet d'établir un cahier des charges, cependant elle ne permet pas d'assurer la déclinaison des objectifs en problèmes organisés à traiter,
- 6 Sigmas : à l'origine, c'est une méthodologie d'assurance qualité s'appuyant sur un usage fort des statistiques et qui a trouvé une évolution en tant que démarche de conception. Puisque cette méthode inclut la spécification des exigences et que notre objectif est de spécifier des champs de recherches à partir de fonctionnalités que doit remplir le produit, elle n'est donc pas véritablement adaptée,

Aussi, nous avons opté pour une démarche nous permettant de hiérarchiser les objectifs à atteindre et de les décliner en diverses problématiques technologiques et de recherche. Cette démarche est présentée dans la sections suivantes, puis est appliquée au concept de « matière communicante ».

1.1 Méthodologie de définition des problèmes

Dans cette étude, nous développons une approche en adaptant certains des outils évoqués précédemment. Sur la base d'une analyse fonctionnelle (AFNOR, 1996) qui a permis de lister les fonctions que peut réaliser le produit, nous mettons en place une approche par matrices de hiérarchisation s'inspirant du QFD - *Quality Function Deployment* (Zaïdi, 1990). L'approche QFD permet aux entreprises de répondre au mieux aux attentes clients et conduit à déployer celle-ci tout au long du processus de conception-industrialisation par l'usage de matrice de corrélation et pondération. En finalité, l'approche QFD permet de cibler les paramètres fondamentaux, ou juste nécessaires, pour satisfaire les attentes des clients. Il s'agit de travailler sur la qualité perçue dès la conception, alors qu'il est encore possible d'explorer un large éventail de solutions. Dans notre étude, cette approche va nous permettre d'une part, d'identifier les problèmes technologiques et scientifiques liés à la matière communicante et d'autre part, les corrélations entre ces problèmes. Enfin, à l'aide d'une approche basée sur les algorithmes spectraux (Fiedler, 1975a; Fiedler, 1975b), ces différents problèmes sont regroupés par familles de problèmes à fortes corrélations, devant être traités au sein d'un même groupe de travail et suivant un ordre de priorité. La FIGURE 24 donne un aperçu de la démarche mise en œuvre, couplant l'analyse fonctionnelle (dans le but d'extraire les exigences clients) avec l'approche QFD à deux niveaux et enfin la méthode de sérialisation, qui font l'objet des sections suivantes.

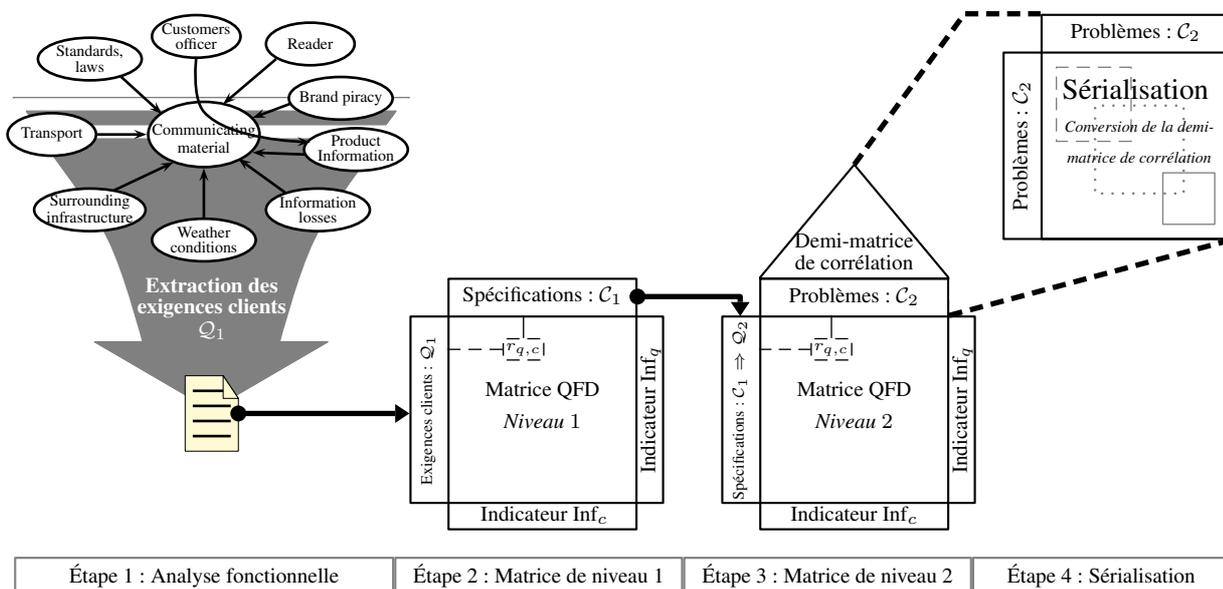


FIGURE 24: Schéma de la méthode pour la définition et la hiérarchisation des problèmes

Quality Function Deployment (QFD)

L'approche QFD prend en compte l'ensemble des besoins du marché et/ou les désirs des futurs utilisateurs dès la conception d'un produit (ou service), dans le but d'élaborer le meilleur processus de fabrication en accord avec les impératifs de qualité fixés. Cependant, cette approche nécessite dans un

premier temps l'énumération des exigences clients. Pour cela, nous avons réalisé plusieurs analyses fonctionnelles du produit « matière communicante » (considérant plusieurs scénarios d'usage) afin d'extraire l'ensemble des exigences clients (première étape illustrée sur la FIGURE 24).

Les exigences résultantes sont alors placées sur les lignes de la matrice QFD de premier niveau. Ces exigences sont également appelées les *Quoi* de la méthode et forment l'ensemble \mathcal{Q}_1 (cf. FIGURE 24 : *Matrice niveau 1*). Une cote numérique Pd_q relative au poids de l'exigence q ($q \in \mathcal{Q}_1$) est aussi spécifiée. Ensuite, il est nécessaire de dresser la liste des moyens ou des spécifications du produit. Ces « moyens » sont appelés les *Comment* de la méthode (composant l'ensemble \mathcal{C}_1) et se retrouvent dans les colonnes de la matrice. Enfin, nous élaborons une relation $r_{q,c}$ entre *Quoi-Comment*, basée sur 4 niveaux¹⁸ : 0, 1, 3, 9. La valeur 0 signifie qu'il n'y a aucune relation entre le quoi et le comment, et 9 représente une très forte relation. Deux indicateurs d'influence Inf_c et Inf_q sont ensuite calculés :

Inf_c donne l'influence relative d'une spécification c ($c \in \mathcal{C}_1$) sur les exigences :

$$Inf_c = \frac{\sum_{q \in \mathcal{Q}_1} (Pd_q \cdot r_{q,c})}{100} \quad (5)$$

Inf_q donne l'influence relative des spécifications sur une exigence q ($q \in \mathcal{Q}_1$) :

$$Inf_q = Pd_q \cdot \sum_{c \in \mathcal{C}_1} r_{q,c} \quad (6)$$

Dans un second temps, cette démarche est réitérée afin de créer la matrice de second niveau (cf. FIGURE 24 : *Matrice niveau 2*). Les spécifications (c.à-d. les *Comment*) de la matrice de premier niveau deviennent désormais les *Quoi* à résoudre de la seconde matrice, composant l'ensemble \mathcal{Q}_2 . Dans notre démarche, les *Comment* de niveau 2 correspondent aux problématiques, tant scientifiques que technologiques à résoudre (appelées ensemble \mathcal{C}_2). Le déploiement de la « Qualité » est assuré par propagation des contraintes, par le fait que les spécifications \mathcal{C}_1 deviennent les *Quoi* de la seconde matrice (c.-à-d. \mathcal{Q}_2). Ceci assure que les problèmes \mathcal{C}_2 mis en évidence, une fois résolus, permettront bien de répondre aux exigences des clients \mathcal{Q}_1 . En complément de l'étude menée précédemment, nous ajoutons à ce stade la demi-matrice de corrélation, se situant au dessus de la matrice. Il s'agit de rechercher l'existence de corrélations entre différents problèmes. Elles peuvent être positives ou négatives et d'intensités variables. En effet, certains processus peuvent se nuire, d'autres se renforcer. Dans notre étude, elles sont exprimées selon 5 niveaux : ++, +, 0, -, --. Les indicateurs calculés dans la matrice de premier niveau Inf_c , Inf_q , $r_{q,c}$, sont calculés de manière similaire (basée sur les ensembles \mathcal{Q}_2 , \mathcal{C}_2).

Notons à ce stade que nous n'utilisons pas les outils QFD de manière classique. Généralement, le déploiement des exigences se fait sur 4 à 5 niveaux (c.à-d. à travers 4 à 5 matrices). De plus, la demi-matrice de corrélation, appelée également « toit de la maison de la qualité », est réalisée pour chaque matrice/niveau. Dans notre étude, la matrice de niveau 2 met en évidence les problèmes technologiques et scientifiques sous-jacents relatifs aux exigences initiales des clients, de ce fait, il est important pour

18. valeurs généralement choisies dans les approches QFD (Zaïdi, 1990).

nous d'en faire apparaître les interactions positives ou négatives afin d'établir une hiérarchisation des problèmes à résoudre.

Organisation de résolution des problèmes

La difficulté est d'organiser les problèmes à résoudre (\mathcal{C}_2) en considérant leur interdépendance. Il s'agit de regrouper ces problèmes selon leurs corrélations négatives ou positives. Pour réaliser cette quatrième étape (comme illustré sur la FIGURE 24), nous proposons de définir ces regroupements en nous basant sur les algorithmes spectraux et plus particulièrement sur l'algorithme de sérialisation développé par (Atkins et al., 1998). Cet algorithme repose sur le principe suivant :

DÉFINITION 5

Étant donné un ensemble d'éléments n à ordonner, et une fonction de corrélation $f_{(i,j)}$ qui reflète un niveau d'attraction entre deux éléments i et j pour qu'ils se retrouvent l'un à côté de l'autre dans la séquence, l'objectif est de trouver toutes les séquences possibles entre les éléments en toute cohérence avec les fonctions de corrélation. C'est-à-dire si π représente la permutation d'éléments et $\pi_{(i)} < \pi_{(j)} < \pi_{(k)}$ alors $f_{(i,j)} \geq f_{(i,k)}$ et $f_{(j,k)} \geq f_{(i,k)}$.

La demi-matrice de corrélation est donc équivalente au graphe des relations entre les différents problèmes. Nous avons vu que ces relations peuvent être positives ou négatives, il est donc nécessaire de transformer la demi-matrice en une matrice permettant d'implémenter l'algorithme d'Atkins. Concrètement, nous remplaçons les symboles $++$, $+$, 0 , $-$, $--$ par les valeurs 20, 10, 1, 50, 100 respectivement. Seule la diagonale de la matrice conserve la valeur 0. La valeur minimale 1 mentionne qu'il n'y a pas d'interaction entre les problèmes mais permet que le graphe soit connexe (condition indispensable pour appliquer l'algorithme spectral). Le choix arbitraire des valeurs 20, 10, 50, 100 met en évidence les corrélations en privilégiant fortement les corrélations négatives, le but étant de mettre en avant les points « durs » à traiter. Une fois la séquence de problèmes déterminée via l'algorithme d'Atkins, l'objectif est de former les différents groupes de problèmes à traiter simultanément, ces groupes constituant l'ensemble \mathcal{G} . Pour cela, la stratégie choisie consiste d'une part à s'assurer que toutes les valeurs, « 50, 100 », soient comprises dans au moins un groupe. Et d'autre part, nous choisissons des groupes de taille raisonnable afin de faciliter l'étude. Pour chaque groupe résultant, un taux de contribution tc_g au projet final est calculé par l'ÉQUATION 7, où $g \in \mathcal{G}$ et \mathcal{C}_{2g} est un sous-ensemble de \mathcal{C}_2 composé uniquement des problèmes appartenant au groupe g . Par exemple, si nous traitons l'ensemble des problèmes appartenant à un groupe g dont le taux de contribution est de 20, cela signifie que l'on répond à 20% de la problématique générale. Dans notre cas, cela signifierait que l'on répond à 20% des problématiques \mathcal{C}_2 liées à la conception d'une « matière communicante » par rapport aux exigences attendues.

$$tc_g = 100 \cdot \frac{\sum_{c \in \mathcal{C}_{2g}} \text{Inf}_c}{\sum_{c \in \mathcal{C}_2} \text{Inf}_c} \quad (7)$$

.2 Déclinaison des problématiques liées au concept de « matière communicante »

Désormais, nous appliquons la démarche décrite ci-avant dans le but de hiérarchiser les objectifs à atteindre concernant le concept de matière communicante, et de les décliner en diverses problématiques scientifiques et technologiques. Nous avons établi un cahier des charges fonctionnel à partir d'une analyse fonctionnelle « générique » obtenue en synthèse de diverses études relatives à des cas particuliers (citons le scénario d'un douanier contrôlant la cargaison de matière communicante d'un camion). Les exigences clients ont ainsi pu être formalisées en termes de fonctions représentatives d'objectifs de *Qualité*, *Sécurité* et *Prix* attendus :

- **Qualité :**
 - doit pouvoir être lue/écrite : la matière doit disposer d'une architecture matérielle et logicielle lui permettant d'entrer en communication avec son environnement,
 - contenir les informations désirées et utiles : la matière doit disposer d'une capacité mémoire suffisante pour stocker les informations utiles aux utilisateurs du produit, durant l'ensemble de son cycle de vie,
 - résister aux traitements : la matière doit résister aux divers environnements et traitements,
 - résister aux pertes : la matière doit pouvoir offrir des mécanismes de tolérance aux fautes et particulièrement aux pertes d'informations (p. ex. si une partie de la matière est supprimée, coupée) suivant une certaine tolérance,
 - respecter les normes environnementales : la matière communicante doit répondre aux mêmes critères environnementaux qu'une matière classique,
 - respecter les normes d'interopérabilité : le ou les système(s) d'information et de communication relatifs à la matière doivent mettre en œuvre des technologies présentes sur le marché, voir standardisées afin que la matière puisse être utilisée par un maximum d'acteurs (p. ex. la technologie RFID ou Ethernet).
 - conserver une esthétique similaire à l'originale : la matière communicante doit (lorsque exigé) conserver le même aspect esthétique que la matière classique.
- **Sécurité :**
 - éviter le piratage des données : La sécurité des données doit être assurée en fonction des exigences du client (droit d'accès, etc.).
- **Prix :**
 - Prix d'entretien à moindre coût : un compromis entre différents critères tels que la robustesse de la technologie à implémenter, le choix des mécanismes de tolérance à implémenter (solution de redondance/copies/backup de l'information, etc.), le nombre de révisions que la matière requiert sur l'ensemble de son cycle de vie, etc. doit être trouvé,
 - Prix d'achat compétitif : le choix de la technologie à implémenter doit s'orienter vers les tech-

nologies bon marché telles que la RFID, capteurs, etc.

Ces exigences forment désormais les *Quoi* de la matrice QFD de niveau 1, laquelle est présentée dans la FIGURE 25. Dans ce qui suit, nous présentons les exigences, spécifications et problèmes les plus importants. Le but n'est pas d'être exhaustif mais de faciliter la compréhension par l'exemple. Il est à ce stade, nécessaire de spécifier *Comment* répondre aux *Quoi* de la matrice. Si l'on prend l'exemple de l'exigence « Doit pouvoir être lue/écrite », on constate qu'une des spécifications (notée en colonne de la matrice QFD 1) concerne les « protocoles d'encodage des données » et de « communication ». Prenons encore l'exemple de « comment résister aux traitements subis par la matière communicante ? », une solution est de choisir une technologie sans-fil robuste aux interférences. Ensuite, on spécifie la cote numérique Pd_q relative au poids de chaque exigence q . On dit par exemple que les exigences « Doit pouvoir être lue/écrite » et « Doit pouvoir contenir des informations » sont les plus importantes ($Pd = 6$). Enfin, les valeurs des relations $r_{q,c}$ entre *Quoi-Comment* sont définies. On spécifie par exemple que l'exigence « Doit pouvoir être lue/écrite » (= q) est fortement liée ou plus exactement, sera en partie résolue lorsque le « protocole de communication » (= c) sera défini ($r_{q,c} = 9$). Enfin, les indicateurs Inf_c et Inf_q peuvent être calculés (cf. dernière ligne et colonne grisées de la matrice). D'après l'indicateur Inf_c , le choix du type de tag, le volume d'informations ou encore la résistance aux pertes sont les spécifications ayant le plus d'impact sur les exigences clients. De la même façon, l'indicateur Inf_q montre que les exigences « doit pouvoir être lue/écrite », « doit pouvoir contenir les informations » et « doit résister aux différents traitements subis par la matière tout au long du cycle de vie » sont hiérarchiquement, les exigences les plus importantes.

		Spécifications																								
		Type de lecteur	Délais de lecture/écriture courts	Protocole de communication	Protocole d'encodage des données	Adaptabilité des infos aux besoins	Type de tag	Volume d'informations	Exhaustivité de l'information	Typage des informations	Adaptabilité des gammes d'infos	Gammes d'informations	Résister à l'ambiance extérieure	Résistance aux interférences	Cohérence des données	Résistance aux erreurs bits	Résistance aux pertes	Norme ISO 14001	Accessibilité physique à la matière	Types de normes	Invisibilité des tags	Cryptage des données	Identification de l'acteur	Prix unitaire du tag	Prix d'entretien	
Exigences clients		Qualité																								
		Pd																					Se	Px		
Qualité	Doit pouvoir être lue/écrite	6	9	3	9	9	9	9	3	3	1	3			9	3	3	3		9	3	1	3	3		570
	Contenir les informations	6		3		9	3	9	9	3	3	3			1	1	3	1	9				3	1		426
	Résister aux traitements	5		1				3						9	9	9		9	9	3		3	1			280
	Résister aux pertes	4		1	1			3	9	3	3	1			1	3		9	9			1		1		180
	Respecter les normes environnementales	5														3	3			9						75
	Respecter les normes d'interopérabilité	5	3	1		9	1	1				1	1	1				3			9	9		3	1	215
	L'esthétique doit être la même avec μ tags	2	3		1		3	9	9	3	3				1	3				1		9	1			92
Se	Eviter le piratage des données	3				3			1														9	9		66
Px	Prix d'achat compétitif	1						9	9													1			9	28
	Prix d'entretien à moindre coût	1						3																	9	12
			0.8	0.5	0.6	1.6	0.8	1.7	1.4	0.9	0.5	0.5	0.2	0.7	1.4	1	0.6	1.5	0.9	1.2	0.7	0.4	0.9	0.6	0.1	0.1

FIGURE 25: Matrice de hiérarchisation : niveau 1

La matrice QFD de second niveau ainsi que la demi-matrice de corrélation sont ensuite réalisées (FIGURE 26), dont les *Comment* de la matrice de premier niveau deviennent les *Quoi* de cette nouvelle

matrice. Notons que les spécifications \mathcal{C}_1 liées aux exigences « prix » (\mathcal{Q}_1) ne sont pas prises en compte dans la matrice de niveau 2, car l'objectif est avant tout de faire émerger les problématiques scientifiques et technologiques, sans se soucier de l'aspect financier. La matrice de niveau 2 montre cette fois-ci au regard des valeurs Inf_c (cf. FIGURE 26) que la « mémoire de la matière », la « redondance de l'information » ou encore les « zones d'accessibilité à la matière » s'affichent comme les problématiques majeures à résoudre dans le but de satisfaire au mieux et en priorité les exigences clients. Dans la demi-matrice de corrélation, on indique si une spécification (c.-à-d. un *comment*) influe sur une autre spécification. Typiquement, on doit fournir à la matière une capacité mémoire suffisante pour l'application, mais à l'opposé on souhaite faire de la redondance d'informations sur la matière (pour répondre à l'exigence de la tolérance aux fautes). Par conséquent, ces deux spécifications sont clairement en contradiction et influent l'une sur l'autre de manière très négative. Pour cette raison, on spécifie une corrélation doublement négative (—) entre ces deux spécifications (la case correspondante est grisée/jaune dans la FIGURE 26).

La dernière étape (étape 4 sur la FIGURE 24) concerne la sérialisation des problèmes via l'algorithme de (Atkins et al., 1998). La matrice donnée en FIGURE 27 présente les différents groupes obtenus après sérialisation. Quatre groupes de problèmes sont définis :

- **Conception de la matière** : ce groupe est essentiellement composé de problèmes liés à la réalisation de la matière communicante dans son état brut. Cela concerne le choix de la technologie à mettre en oeuvre (p. ex. RFID), l'intégration/la fixation des composants électroniques (p. ex. les systèmes tags) dans la matière. Ce groupe se compose principalement de problèmes de choix de solutions vis-à-vis de contraintes technologiques,
- **Architecture de communication** : ce groupe traite des problématiques liées à la conception de l'architecture de communication, laquelle permet à la matière d'entrer en contact et d'être contactée avec/par son environnement. Il s'agit de définir le protocole de communication, tant au niveau de la couche physique (couche constituée par les tags et les lecteurs) qu'au niveau de la couche application. On traite ici des problèmes tels que l'accessibilité à la matière, l'ordre des tags à lire et à écrire, la manière de décomposer une information entre plusieurs systèmes tags, etc.,
- **Exploitation des données** : ce groupe est le « cœur » du projet, de par sa taille (cf. FIGURE 27), son positionnement (en interaction avec les trois autres groupes) et les problématiques le composant (leur nature typée « scientifique » et leur poids Inf_q d'importance). Il s'agit ici de développer un système d'informations embarqué dans la matière, capable de fournir aux utilisateurs des services spécifiques. Ce système d'informations, au sens où nous l'entendons, nécessite d'une part la définition des spécifications conceptuelles des modèles de données supports aux informations échangées tout au long de son cycle de vie, d'autre part requiert une prise en main des mécanismes de répartition des données entre « matière communicante » et ses environnements et enfin, la spécification des protocoles et services applicatifs répondant aux besoins des utilisateurs. Les réponses aux spécifications seront suivies d'une étude de validation du système d'informations en termes notamment de niveau de confiance offert (cohérence, disponibilité des données distribuées et pertinence des données stockées sur la matière) et d'une validation des protocoles et services concernant la fiabilité des communications et la facilité d'accès à la matière,
- **Sécurité & Exhaustivité** : durant son cycle de vie, la matière va être amenée à évoluer en raison de transformations physiques (découpe, assemblage, ...), informationnelles (consultation, ajout/suppression d'informations) et de contraintes d'accessibilité (lecture/écriture partielles de la matière, ...). Ceci engendre une multitude de problèmes tels que le choix des informations à conserver, la modification des données portées par la matière (à un instant donné de son cycle de vie), la réduction du volume de données par des méthodes d'agrégation et de désagrégation ou encore des problèmes de sécurité de données (cryptage, identification de l'acteur).

Le taux de contribution pour chaque groupe a également été calculé et spécifié sur la FIGURE 27 (voir légende : tc_g). Nous constatons que le groupe *Exploitation des données* a le taux de contribution le plus élevé 58% et s'impose donc comme le cœur du projet. Chaque groupe résultant de la sérialisa-

tion peut être associé à une vue particulière du système. En effet, nous évoluons d'une vue « système physique » (groupes 1-2) vers une vue « système informationnel » (groupes 3-4) comme illustré sur la FIGURE 27. La section 1.4 présente, positionne et décrit les problématiques/spécifications qui vont être traitées de manière plus ou moins approfondie dans cette thèse.

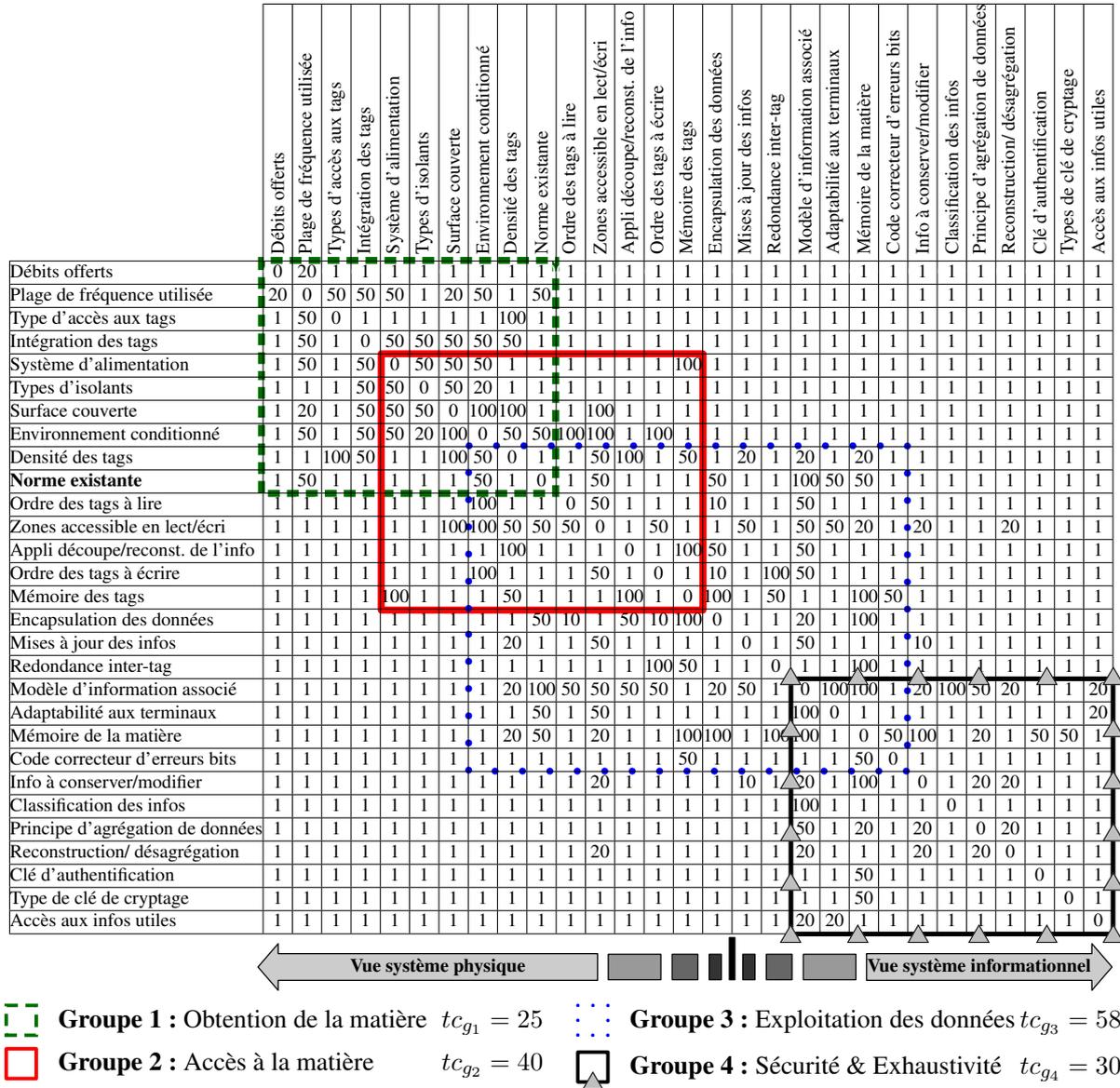


FIGURE 27: Matrice de sérialisation Fiedler

Annexe II : Analyse détaillée des approches de bases de données distribuées

Cette annexe détaille l'ensemble des solutions classées dans les TABLES 2.1, 2.2 et 2.3 respectivement. Trois sections sont ainsi définies :

- détail des solutions relatives à la fragmentation des données (solutions issues de la TABLE 5),
- détail des solutions relatives à l'allocation des fragments de données résultant de la phase de fragmentation (solutions issues de la TABLE 2.1),
- détail des solutions relatives aux bases de données mobiles (solutions issues de la TABLE 5).

Ces trois tables, présentées dans le Chapitre 2 sont reprises dans cette annexe afin de faciliter au lecteur la recherche des solutions dans les différentes tables. L'intitulé de chaque colonne correspond à un aspect précis de la taxonomie. Cependant, deux colonnes sont ajoutées dans les TABLES 2.1 et 2.2, renseignant sur le type de base de données concerné, à savoir : DDBS (*Distributed Database System*) ou DOODBS (*Distributed Object-Oriented Database Systems*). Suivant le type de bases de données, les mécanismes de fragmentation et d'allocation diffèrent. La différence majeure réside dans les entrées des mécanismes, à savoir que les approches prennent uniquement en compte les affinités entre attributs (des classes du modèle) dans les DDBS, alors que les mécanismes développés dans le cadre des DOODBS prennent en compte les affinités entre attributs, entre méthodes et entre méthodes-attributs. Cette différence concerne principalement l'étape de fragmentation.

.3 Étape de fragmentation

Fragmentation de données dans les DDBS

Dans le cadre des DDBS, (Navathe et al., 1984) ont été parmi les premiers à s'intéresser à la fragmentation verticale. Ils utilisent une *Attribute Affinity Matrix* (AAM) formée à partir de la *Attribute Usage Matrix* (AUM), ainsi que du *Bond Energy Algorithm* (BEA) afin de produire des fragments verticaux. AUM est une matrice dont les colonnes correspondent à un ensemble d'attributs et les lignes à un ensemble de transactions. La valeur de la matrice relate la fréquence d'accès des transactions sur ces attributs. AAM est une matrice dans laquelle, pour chaque paire d'attributs, la somme totale des fréquences des transactions qui accèdent à cette paire est stockée. L'algorithme BEA est utilisé pour grouper les attributs en fonction de leurs affinités (matrice AAM). Une affinité entre attributs exprime

TABLE 5: Classification : Méthodes de fragmentation

	Fragmentation	Overlapping	DDBS	DOODBS	Dynamique	Statique	Nombre de citations	Nombre de citations/an
(Navathe et al., 1984)	V	✓	✓			✓	308	308/28 = 11.0
(Apers, 1988)	M		✓		✓	✓	247	247/24 = 10.3
(Navathe and Ra, 1989)	V	✓	✓			✓	172	172/23 = 7.48
(Navathe et al., 1995)	M	✓	✓			✓	83	83/17 = 4.88
(Son and Kim, 2004)	V		✓			✓	29	29/8 = 3.62
(Tamhankar and Ram, 1998)	M		✓			✓	47	47/14 = 3.35
(Baião and Mattoso, 1998; Baião et al., 2002)	M			✓		✓	38	38/12 = 3.17
(Ezeife and Barker, 1998)	V			✓		✓	43	43/14 = 3.07
(Bellatreche et al., 1997)	H			✓		✓	41	41/15 = 2.73
(Chakravarthy et al., 1994)	V		✓			✓	47	47/28 = 2.61
(Hauglid et al., 2010)	H		✓		✓		5	5/2 = 2.50
(Abuelyaman, 2008)	V		✓			✓	9	9/4 = 2.25
(Sleit et al., 2007)	M		✓	✓	✓		8	8/5 = 1.60
(Khan and Hoque, 2010)	H		✓			✓	3	3/2 = 1.50
(Fung et al., 2002)	V			✓		✓	12	12/10 = 1.20
(Malinowski and Chakravarthy, 1997)	V			✓		✓	12	12/15 = 0.80
(Darabant et al., 2004)	H			✓		✓	6	6/8 = 0.75
(Ezeife and Zheng, 1999a)	H			✓	✓		1	1/13 = 0.08
(Pinto and Torres, 2002)	H	✓	✓		✓		0	0/10 = 0.00
(Adrian Runceanu, 2007)	M			✓		✓	0	0/5 = 0.00

le degré d'utilisation conjointe. Ils introduisent également le *Binary Vertical Partitioning* (BVP), bien connu à ce jour. Cet algorithme partitionne récursivement la matrice d'affinité en deux groupes distincts, sans *overlapping*, dans le but de minimiser le nombre de transactions effectuées sur les attributs de ces groupes. (Navathe and Ra, 1989) améliorent leurs précédents travaux en proposant un algorithme basé sur une technique graphique. Plus tard, ils introduisent un nouveau type de fragmentation hybride, le *grid cell*. Chaque cellule de la grille appartient à exactement un fragment horizontal et un fragment vertical, les cellules de la grille donnent lieu à un cloisonnement impartial d'une relation. Dans la même lignée, (Chakravarthy et al., 1994) proposent un *Partition Evaluator* (PE) afin d'évaluer un quelconque algorithme de fragmentation verticale mettant en œuvre la matrice AAM. Ce PE est composé de deux éléments, coût d'accès aux attributs locaux non pertinents et coût d'accès aux attributs distants pertinents. Plus la valeur de PE est élevée, plus les performances du système seront dégradées, et par conséquent, moins bon sera le schéma de fragmentation. Aujourd'hui encore, de nombreux papiers s'appuient sur ces

précédents travaux comme base de comparaison ou dans le but de les revisiter. Une technique de fragmentation pour la moins particulière est développée par (Abuelyaman, 2008) and (Khan and Hoque, 2010), puisqu'ils ne tiennent pas compte des fréquences exactes d'exécution des requêtes en entrée de leur algorithme, ce qu'aucune autre méthode ne fait à notre connaissance. Ils expliquent tous deux que ce type d'approche s'applique lors de l'étape initiale du cycle de vie de la base de données, lorsque l'on a peu, voire aucune connaissance précise sur l'application (fréquence des requêtes, etc.). (Abuelyaman, 2008) montre qu'il obtient des résultats similaires à ceux fournis par l'algorithme BVP. Néanmoins, les limites de l'algorithme BVP sont énoncées par (Son and Kim, 2004), parmi elles : il ne fournit pas toujours une solution déterministe concernant le problème de fragmentation verticale *n-way* (contrainte sur le nombre de fragments à retourner), de plus il ne permet pas de supporter à la fois le partitionnement *n-way* ainsi que le partitionnement *best-fit* (minimisation d'un coût). En conséquence, les auteurs proposent une méthode appelée *Adaptable Vertical Partitionning* (AVP) supportant ces deux types de fragmentation. Ils montrent après comparaison, que leur méthode est moins limitée (concernant le *n-way*) que la méthode BVP. Le PE est revisité par (Runceanu, 2007), lequel met en œuvre une heuristique concernant le problème de fragmentation. L'auteur spécifie que la fonction objectif employée peut facilement être étendue pour inclure d'autres variables (types de requêtes, coût de traitement distant, etc.).

Fragmentation de données dans les DOODBS

Le PE est également appliqué dans le cadre des DOODBS par (Malinowski and Chakravarthy, 1997). Le but est d'afficher les bénéfices qui peuvent être réalisés en implémentant des mécanismes de fragmentation issus des DDBS. Les premiers travaux traitant de la fragmentation verticale de classes dans les DOODBS sont ceux de (Karlalalem et al., 1994; Karlalalem and Li, 1995; Ezeife and Barker, 1998). Ces auteurs ont respectivement cherché à énumérer, évaluer les enjeux, puis à développer différents schémas de fragmentation. (Ezeife and Barker, 1998) montrent que les coûts de communication réseaux dominant sur les coûts de traitement des requêtes. (Fung et al., 2002) proposent un schéma de fragmentation verticale basé sur un modèle de coût analytique. Au final, ils montrent qu'il est possible d'établir un schéma de fragmentation verticale optimale en considérant une charge de travail donnée (relative aux requêtes). Certains auteurs expliquent que si la phase de fragmentation est correctement réalisée, il n'en sera que plus facile pour la phase d'allocation (Baião et al., 2002; Baião and Mattoso, 1998). Ces derniers démontrent les bénéfices d'une fragmentation mixte et proposent une démarche de conception basée sur une succession d'étapes : une phase d'analyse, une fragmentation verticale et horizontale. Dans la première phase, un ensemble de classes nécessaires (ou non) à la fragmentation horizontale, verticale est identifié. Dans la seconde et troisième phases, les fragmentations verticales et horizontales sont effectuées sur les classes précédemment identifiées à l'aide d'algorithmes, qui sont basés et adaptés de (Navathe et al., 1995) (algorithmes basés sur les affinités entre attributs). De même, l'ordre des classes à fragmenter peut avoir un impact non négligeable sur le résultat de la fragmentation, comme le défendent (Darabant et al., 2004). En conséquence, ils développent une approche permettant d'établir un ordre entre classes (approche reposant sur un ensemble de requêtes applicatives) avant de s'atteler à

la phase de fragmentation. (Bellatreche et al., 1997) étudient la fragmentation de classe horizontale en proposant une approche permettant de réduire le nombre d'accès aux objets non pertinents/importants. Les méthodes de (Tamhankar and Ram, 1998; Adrian Runceanu, 2007) seront introduites dans la section dédiée à l'allocation des fragments puisqu'elles gèrent à la fois la refragmentation (c.-à-d. la fragmentation dynamique) ainsi que la ré-allocation des fragments (c.-à-d. l'allocation dynamique des fragments de données). Pour les mêmes raisons, trois des cinq approches dynamiques relevées dans la TABLE 5 (Apers, 1988; Hauglid et al., 2010; Sleit et al., 2007) seront abordées dans la section .4. (Pinto and Torres, 2002) s'intéressent spécifiquement à la refragmentation où ils mettent en place une réplication partielle. À noter que leur algorithme doit être implémenté sur chaque ordinateur disposant d'une base de données esclave. (Ezeife and Zheng, 1999a) ont proposé une architecture incluant un équipement de supervision. Cet équipement garde une trace des changements dans le système et périodiquement recalcule les valeurs de performances du système. Cet équipement de supervision est en réalité celui présenté dans (Ezeife and Zheng, 1999b), appelé *Object Horizontal Partition Evaluator* (OHPE). OHPE est basé sur le *Partition Evaluator* (PE), qui rappelons-nous, se compose de deux éléments : coût d'accès aux attributs locaux non pertinents et coût d'accès aux attributs distants pertinents. Cependant, ces deux évaluateurs mesurent uniquement le nombre d'instances d'un fragment sans tenir compte de la taille des objets et des caractéristiques du réseau.

Principaux critères au regard de la littérature

Au regard du corpus de documents classé dans la TABLE 5, la majorité des approches repose sur des fonctions objectives qui considèrent des critères similaires ; les principaux étant :

- l'ensemble des attributs de classes (fragmentation verticale),
- les affinités entre attributs,
- l'ensemble des tuples (fragmentation horizontale),
- l'ensemble des méthodes de classes (contexte DOODBS),
- l'ensemble des requêtes de données (lecture/écriture) ou des transactions ¹⁹,
- la fréquence de chaque requête ou transaction.

Ainsi, en fonction des exigences et contraintes formulées par le concepteur de la base de données, une ou plusieurs de ses méthodes pourront être choisies dans la TABLE 5. Par exemple, s'il n'a pas suffisamment de connaissance sur les fréquences des requêtes, il pourra implémenter les méthodes développées par (Abuelyaman, 2008) et (Khan and Hoque, 2010).

.4 Allocation de données

Allocation dynamique des données

19. l'utilisateur interagit avec la base de données via des opérations de lecture/écriture, lesquelles sont appelées des "transactions".

TABLE 6: Classification : Méthodes d'allocation

	Fragmentation	Allocation	Réplication	DDBS	DOODBS	Dynamique	Statique	Nombre de citations	Nombre de citations/an
(Apers, 1988)	M	M	✓	✓		✓	✓	247	247/24 = 10.3
(Loukopoulos and Ahmad, 2004)		✓	P _c	✓		✓	✓	59	59/8 = 7.37
(Huang and Chen, 2001)		M	U _e	✓		✓		70	70/11 = 6.36
(Ahmad et al., 2002)		✓		✓			✓	42	42/10 = 4.20
(Tamhankar and Ram, 1998)	M	M	P _c	✓			✓	47	47/14 = 3.36
(Ulus and Uysal, 2003)		M		✓		✓		25	25/9 = 2.78
(Brunstrom et al., 1995)		H		✓		✓		43	43/17 = 2.53
(Hauglid et al., 2010)	H	H	P _c	✓		✓		5	5/2 = 2.50
(Singh and Kahlon, 2009)		✓		✓		✓		7	7/3 = 2.33
(Chin, 2002)		✓		✓		✓		18	18/10 = 1.80
(Hababeh et al., 2005)		✓	✓	✓			✓	12	12/7 = 1.71
(Sleit et al., 2007)	M	M	✓	✓	✓	✓		8	8/5 = 1.60
(Blankinship et al., 1991)		✓		✓			✓	19	19/21 = 1.19
(Chaturvedi et al., 1994)		M	✓	✓		✓		16	16/18 = 0.89
(Uysal and Ulus, 2007)		M		✓		✓		4	4/5 = 0.80
(Petrescu, 2010)		✓	P _c -A _c	✓		✓		1	1/2 = 0.50
(Grebla et al., 2004)		H		✓		✓		2	2/8 = 0.25
(Tasharofi and Basseda, 2006)		✓		✓		✓		1	1/6 = 0.17
(Hamideh Sabouri, 2008)		✓	U _e	✓			✓	0	0/4 = 0.00

Débutons par les méthodes dynamiques ne proposant aucune réplication de fragments. Une des techniques la plus revisitée à ce jour est l'*Optimal algorithm* proposée par (Brunstrom et al., 1995). Quatre modules sont implémentés dans chaque nœud/site (*Exodus server*, *Generator process*, *Stats process* and *Move process*) afin de stocker, traiter et d'assurer le transfert des fragments entre les sites. L'approche repose sur deux algorithmes, le premier appelé *Simple Counter Algorithm* classe les sites en fonction des accès effectués sur chaque fragment de données, puis réalloue chacun d'eux sur le site qui y accède le plus. Le second algorithme *Load Sensitive Counter Algorithm* surveille à la fois la charge dans le système et les fréquences d'accès aux fragments. Le fragment est réalloué si cela n'engendre pas de surcharge sur le système (c.-à-d., si la charge est inférieure à un seuil donné/réglable). À noter que pour chaque fragment, un compteur relatif aux accès effectués par chaque site du système est mis en place. En d'autres termes, si l'on considère n sites composant une architecture et k fragments alloués au site i (i appartenant à n), le nombre de compteurs implémentés sur ce site i équivaut à $n \times k$. Certains auteurs ont souligné les limites de cet algorithme, notamment (Ulus and Uysal, 2003) qui expliquent que ce dernier est trop

lourd en termes de mémoire (nombre de compteurs trop important), qu'il y a une limitation des compteurs (jamais remis à zéro), ainsi qu'un problème de *timing* (les fragments peuvent être amenés à bouger trop souvent). En conséquence, ces auteurs proposent un nouvel algorithme *Threshold Algorithm* (basé sur l'*Optimal algorithm*) qui consiste à implémenter un unique compteur par fragment. Le compteur est incrémenté après chaque accès distant et remis à zéro après chaque accès local. Leur algorithme décrète la ré-allocation du fragment si le compteur dépasse un seuil donné. Cependant, il est impossible de savoir quel site a accédé majoritairement au fragment, on connaît uniquement le site ayant effectué l'ultime accès (site qui se voit alors attribuer le fragment en question). Ces mêmes auteurs (Uysal and Ulus, 2007) analysent par la suite leur algorithme à l'aide d'une chaîne de Markov à état fini afin d'analyser le comportement d'un fragment en réponse à un changement de probabilité d'accès ou, à un changement de la valeur du seuil. Il montre que le fragment tend à rester sur le nœud lorsque la probabilité d'accès ou la valeur du seuil augmentent. Le problème de *timing* lié à l'*Optimal algorithm* est également pointé par (Tasharofi and Basseda, 2006), lesquels proposent à leur tour un algorithme dénommé *Near Neighborhood Allocation* (NNA). La principale différence réside sur la manière dont évolue le fragment dans le système. Dans l'*Optimal Algorithm*, le fragment est envoyé directement au site qui y accède le plus. Dans l'approche NNA, l'algorithme calcule dans un premier temps le chemin entre la source (site sur lequel est localisé le fragment) et le site qui le requiert le plus en s'appuyant sur l'algorithme de *Dijkstra* (*Shortest Path First*), puis le fragment est envoyé de proche en proche (sites issus du chemin). Cette technique réduit le transfert de données sur le réseau et de fait, réduit les temps de transmission du fragment. Les simulations indiquent que pour des tailles de fragments significatifs (supérieures à 8000 octets) et des taux de requêtes conséquents, l'algorithme NNA fournit de meilleurs résultats que l'*Optimal Algorithm* et que pour des petits fragments et des faibles taux de requêtes, l'*Optimal Algorithm* fonctionne mieux. Une variation de l'*Optimal algorithm* et du *Threshold Algorithm* est proposée par (Singh and Kahlon, 2009) après avoir exposé les limites de chacun d'eux, à savoir que le fragment évolue entre sites de manière aléatoire, qu'aucun historique n'est disponible sur les accès passés et que le compteur est remis à zéro après chaque accès local. L'algorithme proposé *Threshold and Time Constraint Algorithm* (TTCA) maintient une matrice $n \times k$ où n est le nombre total de sites et k le nombre total de fragments. Leur approche solutionne il est vrai, ces problèmes, mais requiert une mémoire plus conséquente sur les différents sites que les précédentes techniques. Un concept pour le moins original est introduit par (Grebala et al., 2004), un agent mobile intelligent est mis en œuvre, lequel navigue entre sites et relève les données utiles aux décisions de ré-allocation de fragments. Le lecteur peut se reporter au rapport technique rédigé par (Basseda and Tasharofi, 2005) pour une description plus détaillée de la plupart des algorithmes présentés dans cette section.

Intéressons-nous désormais aux méthodes d'allocation dynamiques considérant la réplification de fragments. (Petrescu, 2010) propose un algorithme dans le but d'évaluer si les fragments doivent être réalloués en fonction du plan d'accès des utilisateurs. Il repose sur des fonctions/conditions qui permettent de décréter si un fragment peut être stocké dans un nœud spécifique en fonction des permissions

d'accès (en lecture/écriture) ainsi que de facteurs tels que la mémoire disponible, la charge courante du réseau, etc. Les statistiques par nœud sont calculées sur une période de temps donnée (ajustable). Une approche tirée d'une adaptation de *machine learning* est réalisée dans (Chaturvedi et al., 1994). Leur méthode repose sur le concept d'invariance de temps. Elle détecte automatiquement les habitudes d'utilisation des données à partir de l'historique des requêtes. Cela permet d'identifier les parties de la base de données qui restent inchangées (appelées *time invariant fragments*). Ces *time invariant fragments* sont ensuite répliqués sur l'ensemble des nœuds du système. Notons que cette stratégie de réplication est jugée réductrice par (Barney and Low, 2008) et peut engendrer une surcharge sur le réseau.

Allocation statique des données

Dans cette section, on se focalise tout d'abord sur les approches statiques gérant la réplication de fragments. Certains auteurs réalisent des groupements de sites ou fragments au sein du système d'information, tels que (Hababeh et al., 2005) qui cherchent à regrouper les fragments en fonction des coûts de transmission entre sites. Leur approche se décline en deux étapes, la première réalise un regroupement des sites (bases de données), la seconde alloue les fragments à chacun des clusters obtenus. Dans la lignée de ces travaux, (Hababeh, 2010) propose une nouvelle approche de groupement laquelle s'intéresse uniquement à réduire les coûts (temps) de communication entre sites. Les simulations s'appuient sur l'outil *OPNET Modeler simulator*^{® 20} afin de mettre en évidence la réduction de charge réalisée sur le réseau de communication. À travers la littérature, on relève bon nombre de travaux faisant appel à des heuristiques du fait de l'explosion combinatoire de l'espace des solutions, prouvé NP-Complet dans (Apers, 1988; Loukopoulos and Ahmad, 2004; Bhar and Barker, 1995; Garey and Johnson, 1979; Kangasharju et al., 2002). Aussi, les algorithmes génétiques sont souvent utilisés (Sarhan, 2009; Ahmad et al., 2002; Loukopoulos and Ahmad, 2004; Corcoran and Hale, 1994; Cheng et al., 2002; Rahmani et al., 2009; Alavizadeh and Sabouri, 2006). Un algorithme génétique assez particulier est proposé par (Hamideh Sabouri, 2008), puisqu'il implémente une matrice d'exigences dans le but d'influencer le résultat d'allocation des fragments (valeurs comprises entre 0 et 100). Bien que l'idée soit intéressante²¹, aucune information sur la manière de remplir cette matrice n'est fournie (choix concepteur ou en collaboration avec les clients ?). L'algorithme génétique proposé par (Loukopoulos and Ahmad, 2004), appelé *Genetic Replication Algorithm* (GRA), est unique dans le sens qu'une moitié des individus composant la population initiale est générée à travers un algorithme dénommé *Simple Replication Algorithm* (SRA). Ce dernier effectue une réplication du fragment basé sur un bénéfice $B(k, i)$, où k correspond à l'objet/fragment à analyser (doit-on le répliquer ou non ?) et i le site sur lequel est évalué ce bénéfice (point de vue local). Ce bénéfice fournit une solution de haute qualité lorsque les fréquences de lecture sont largement supérieures au nombre total de mises à jour. Par la suite, les auteurs adaptent le GRA (prévu initialement pour une allocation statique) afin de le rendre fonctionnel de manière dynamique, en prenant

20. OPNET Modeler permet le développement et l'analyse de communications réseaux, d'équipements, de protocoles et d'applications.

21. C'est un des seuls auteurs jusqu'à présent qui essaie d'influer les résultats au vu du contexte d'application/d'usage de la base de données.

en entrée la distribution des fragments courants. Il fournit alors une nouvelle distribution en utilisant les caractéristiques du réseau et les changements survenus.

Focalisons-nous désormais sur les méthodes d'allocation statiques n'offrant pas la possibilité de répliquer les fragments de données. Les travaux menés par (Ahmad et al., 2002) se basent sur une approche originale mettant en œuvre un graphe de dépendance entre fragments. Ce graphe modélise les dépendances entre les fragments et les quantités de données transférées, occasionnées par l'exécution d'une requête. La fonction coût est alors implémentée dans quatre heuristiques : *Genetic Algorithm* (GA), *Simulated Evolution Algorithm* (SE), *Mean field Annealing Algorithm* (MFA), *Random Search* (RS). Les quatre heuristiques sont comparées sur plusieurs critères (temps d'exécution moyen, dégradation des performances, etc.). La conclusion est telle que le GA donne principalement les meilleurs résultats.

Cinq des approches relevées dans la TABLE 6 gèrent à la fois la refragmentation, ainsi que la ré-allocation des fragments. Trois d'entre elles sont gérées de manière dynamique (Apers, 1988; Hauglid et al., 2010; Sleit et al., 2007). (Apers, 1988) propose un algorithme d'allocation utilisant le plan de traitement des requêtes. L'approche cherche à déterminer la fragmentation optimale pour que le coût total des transmissions induites par les requêtes, ainsi que par les mises à jour soient minimisés. Récemment, une approche décentralisée appelée *Dynamic Fragmentation and Replica Management in Distributed Database System* (DYFRAM) a été proposée par (Hauglid et al., 2010). DYFRAM met en œuvre trois algorithmes : un pour créer de nouvelles répliques, un pour les supprimer et un dernier pour éclater ou regrouper des fragments. L'algorithme de création et de suppression des répliques est implanté sur chacun des sites, tandis que l'algorithme pour l'éclatement de fragments est uniquement mis en œuvre sur le maître afin d'éviter tout conflit. Leur approche ne permet pas de gérer l'*overlapping* d'attributs, elle considère le coût de communication des accès locaux comme nul et repose sur l'hypothèse que tous les sites ont les mêmes capacités de traitement et de communication. (Sleit et al., 2007) proposent l'algorithme *Enhanced - Adaptive Distributed Request Window* (E-ADRW) qui est une évolution de l'algorithme ADRW proposé par (Wujian and Veeravalli, 2003). Ce dernier n'est aucunement dédié au domaine des bases de données relationnelles, mais aux systèmes réseaux distribués. Le principe de l'algorithme E-ADRW est le suivant : tout d'abord, la condition de réplication est examinée, si celle-ci est vérifiée, la condition de fragmentation est à son tour testée, et si celle-ci est également vérifiée, alors la fragmentation est privilégiée à la réplication. Les auteurs comparent l'algorithme E-ADRW aux algorithmes *Threshold algorithm* et *Optimal algorithm*, ils démontrent que leur algorithme nécessite moins de mémoire, génère moins de charge sur le réseau et que la condition de fragmentation est plus stricte.

Principaux critères au regard de la littérature

Comme dans le cas des méthodes de fragmentation, beaucoup de critères sont communs aux fonctions objectifs et aux algorithmes implémentés. Les principaux critères sont :

- l'ensemble de fragments/objets dans le système,

- la taille de chaque fragment/objet,
- l'ensemble de sites composant le système de bases de données (clients et serveurs),
- la capacité de stockage de chaque site,
- l'ensemble des requêtes (lecture/écriture) ou des transactions,
- la fréquence des requêtes ou des transactions,
- le coût associé aux requêtes de lecture/écriture,
- le coût de communication entre sites,
- le coût de stockage de chaque fragment de données sur chaque site.

Une fois encore, en fonction des exigences et contraintes formulées par le concepteur, la méthode la plus appropriée pourra être choisie dans la TABLE 6. Par exemple, si les bases de données ont des capacités de stockage limitées, le concepteur pourra se tourner vers des méthodes traitant ce problème ; s'il désire mettre en œuvre une réplication de données sur le principe de Primary copy, il pourra sélectionner des fonctions objectifs formalisant un tel mode (p. ex. la fonction objectif proposée par (Loukopoulos et Ahmad, 2004)).

.5 Bases de données mobiles

Cette section détaille uniquement les travaux qui n'ont pas été discutés dans le Chapitre .2, à savoir les travaux de (Hara and Madria, 2006; Demers et al., 2002; Pabmanabhan and Gruenwald, 2006; Záborský, 2008) et (Phatak and Badrinath, 1999).

Dans le cadre des MANETs, (Hara and Madria, 2006) proposent plusieurs algorithmes de réplication de données basés sur le principe de Pc-As, à travers lesquels les données restent disponibles (haute probabilité de disponibilité) et les répliques restent cohérentes (dans le cas de mise à jour). De plus, leur approche met en œuvre un protocole garantissant aux nœuds un accès à la réplique la plus proche (en termes de distance). Une technique de réplication dénommée DREAM (*Data REplication technique for real-time Ad-hoc Mobile databases*) est présentée dans (Pabmanabhan and Gruenwald, 2006) se composant de trois phases : la première cherche à identifier les éléments de données, ainsi que les serveurs sur lesquels ils doivent être affectés. Les seconde et troisième phases ont pour objectifs respectifs de déterminer la manière d'accéder aux répliques et d'identifier le mode de synchronisation des répliques. Dans leurs travaux, la localisation des nœuds est évaluée via GPS et la ré-allocation des fragments se fait durant une période de temps spécifique. (Záborský, 2008) inclut dans son algorithme le SNR (*Signal Noise Rate*) afin de décider si une ré-allocation doit être effectuée. L'ensemble des avantages et inconvénients des stratégies de réplication synchrone/asynchrone est également présenté dans (Moiz and Rajamani, 2009). Les auteurs présentent plusieurs techniques de réplication de données ainsi que leur applicabilité en environnement mobile. Ils fournissent un cadre pour la sélection de la méthode de réplication la plus appropriée au vu des exigences du prestataire de services. Dans le même ordre d'idées, une analyse comparative est menée par (Sharma et al., 2010) concernant quatre techniques de *Cluster-*

based Data Replication. Cette étude cherche à évaluer leurs performances respectives en fonction de différents critères : classification de l'architecture client/serveur, disponibilité et cohérence de la donnée, hypothèses émises concernant les requêtes de lecture/écriture, etc. Ils précisent néanmoins qu'il est impossible d'affirmer qu'une approche est meilleure qu'une autre car cela dépend fortement des exigences et caractéristiques de l'application. (Phatak and Badrinath, 1999) sont les seuls auteurs dans notre classification qui s'intéressent à la fragmentation du schéma relationnel en environnement mobile (fragmentation horizontale). Ils émettent toutefois l'hypothèse que suffisamment d'informations sont disponibles au sujet de la localisation des accès. Dans le cas contraire, il est nécessaire d'en générer.

TABLE 7: Classification : Méthodes destinées aux environnements mobiles

	Fragmentation	Allocation/ Réplication	Dynamique	Statique	Capacité de stockage	Pertinence de l'info.	Type de noeud	Nombre de citations	Nombre de citations/an
(Demers et al., 2002)		P _c -A _s	✓		✓		M	377	377/10 = 37.7
(Huang et al., 1994)		✓	✓				M-F	201	201/18 = 11.2
(Hara and Madria, 2006)		P _c -A _s	✓		✓		M	104	104/6 = 17.3
(Bobineau et al., 2000)			✓	✓	✓	✓	M-F	105	105/12 = 8.75
(Phatak and Badrinath, 1999)	H			✓			M-F	21	21/13 = 1.62
(Lubinski and Heuer, 2000)		✓	✓		✓	✓	M-F	18	18/12 = 1.50
(Madria et al., 1998)				✓	✓		M-F	17	17/1998 = 1.21
(Chan and Roddick, 2003)		✓	✓	✓	✓	✓	M-F	14	14/9 = 1.56
(Lubinski, 2000)		✓	✓	✓	✓		M-F	8	8/12 = 0.67
(Abawajy et al., 2006)		A _s -S					M-F	7	7/6 = 1.17
(Pabmanabhan and Gruenwald, 2006)		P _c -A _s	✓		✓		M	7	7/6 = 1.17
(Zábovský, 2008)		✓	✓				M	3	4/4 = 1.00
(Ashraf et al., 2010)		✓	✓				M-F	1	1/2 = 0.50
(Moiz and Rajamani, 2009)		P _c -A _s		✓			M-F	1	1/3 = 0.33

Annexe III : Détail des Algorithmes

Cet Annexe détaille les algorithmes `RetrievalData`, `ExplorePK` et `ExploreFK` qui sont appliqués sur l'exemple donné dans la section 3.1.3.

L'algorithme présenté dans la section 3.1.3, qui décrit les étapes de `RetrievalData`, est rappelé dans la FIGURE 28. Dans cet algorithme, la sous-fonction `ExplorePK` est sollicitée deux fois (cf. FIGURE 28) :

- `ExplorePK(\mathcal{T} , \mathcal{R} , $T_{\text{MatLot}}\{2, 1\})$,`
- `ExplorePK(\mathcal{T} , \mathcal{R} , $T_{\text{MatDef}}\{3, 1\})$,`

et la sous-fonction `ExploreFK` est sollicitée une fois :

- `ExploreFK(\mathcal{E} , \mathcal{T} , \mathcal{R})`.

Ces trois sollicitations sont respectivement détaillés dans les ALGORITHMES 4, 5 et 6.

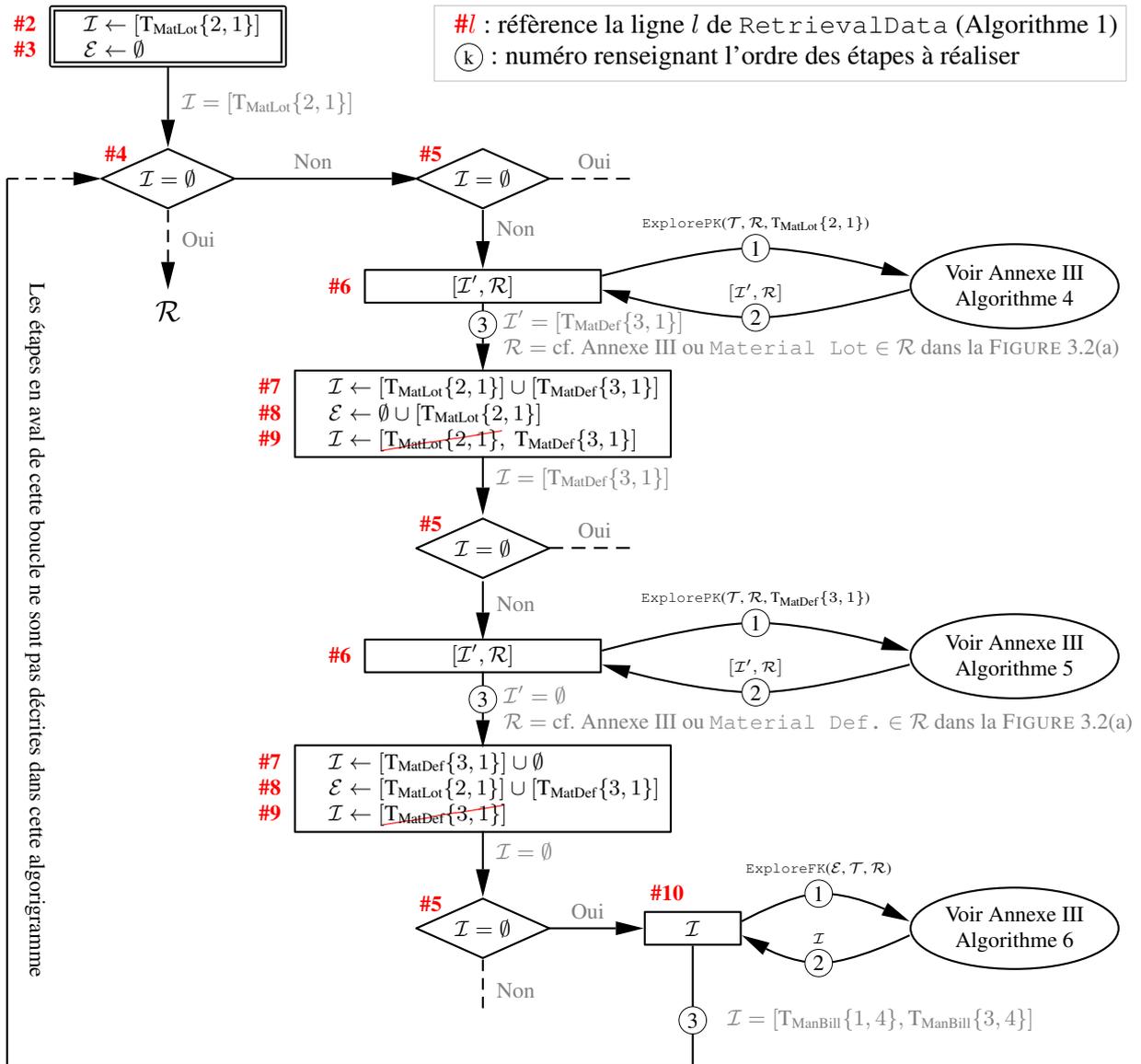
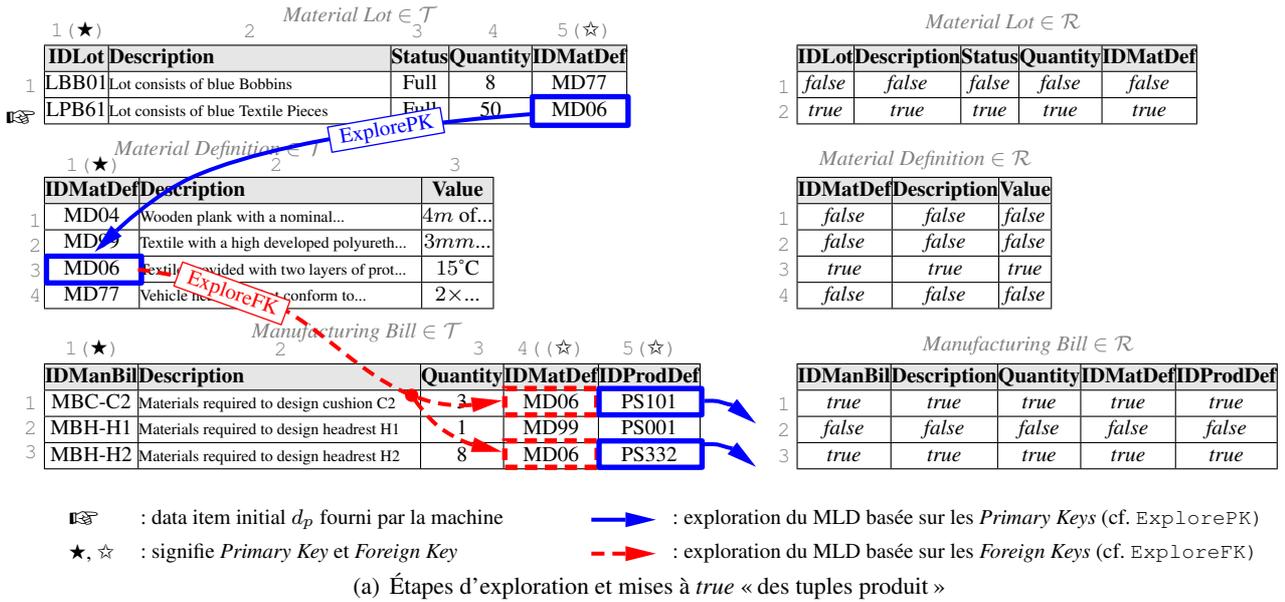


FIGURE 28: Identification de « tuples produit » dans les tables via la fonction RetrievalData)

Algorithme 4: ExplorePK(\mathcal{T} , \mathcal{R} , $T_{\text{MatLot}}\{2, 1\}$)

```

output :  $\mathcal{I}'$ ,  $\mathcal{R}$ 
// l'évolution de  $\mathcal{I}'$  et  $\mathcal{R}$  est mise en avant en rouge

1 begin
2    $\mathcal{I}' \leftarrow \emptyset$ ;
3   forall the  $j_1 = 1 : 5$  do // 5 attributs dans MaterialLot
4      $\mathcal{R}_{\text{MatLot}}\{2, 1\} \leftarrow true$ ; //  $j_1 = 1$ 
5      $\mathcal{K}_{\text{MatLot}}\{1\} = 1$ ; // l'attribut 1 de MaterialLot n'est pas une FK ( $\neq 2$ )
6      $\mathcal{R}_{\text{MatLot}}\{2, 2\} \leftarrow true$ ; //  $j_1 = 2$ 
7      $\mathcal{K}_{\text{MatLot}}\{2\} = 0$ ; // l'attribut 2 de MaterialLot n'est pas une FK ( $\neq 2$ )
8      $\mathcal{R}_{\text{MatLot}}\{2, 3\} \leftarrow true$ ; //  $j_1 = 3$ 
9      $\mathcal{K}_{\text{MatLot}}\{3\} = 0$ ; // l'attribut 3 de MaterialLot n'est pas une FK ( $\neq 2$ )
10     $\mathcal{R}_{\text{MatLot}}\{2, 4\} \leftarrow true$ ; //  $j_1 = 4$ 
11     $\mathcal{K}_{\text{MatLot}}\{4\} = 0$ ; // l'attribut 4 de MaterialLot n'est pas une FK ( $\neq 2$ )
12     $\mathcal{R}_{\text{MatLot}}\{2, 5\} \leftarrow true$ ; //  $j_1 = 5$ 
13     $\mathcal{K}_{\text{MatLot}}\{5\} = 2$ ; // l'attribut 5 de MaterialLot EST une FK
14    forall the  $t' \in [\dots, \text{MatLot}, \text{MatDef}, \text{ManBill}, \dots]$ ,  $j_2 \in$ 
       $\{1, n_{t'}\} | \text{value}(T_{\text{MatLot}}\{5\}) = \text{value}(T_{t'}\{j_2\}) \ \& \ \mathcal{K}_{t'}\{j_2\} = 1$  do // Dans notre
      exemple,  $\mathcal{T} = [\dots, \text{MatLot}, \text{MatDef}, \text{ManBill}, \dots]$ , car uniquement les tables MaterialLot,
      MaterialDefinition et ManufacturingBill nous intéressent
      //  $t' = \text{MaterialLot}$  avec  $n_{\text{MatLot}} = 5$  (i.e. 5 attributs,  $j_2 = 1:5$ )
15       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{MatLot}}\{1\})$ ; //  $\text{ID\_MatDef} \neq \text{ID\_Lot}$ 
16       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{MatLot}}\{2\})$ ; //  $\text{ID\_MatDef} \neq \text{Description}$ 
17       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{MatLot}}\{3\})$ ; //  $\text{ID\_MatDef} \neq \text{Status}$ 
18       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{MatLot}}\{4\})$ ; //  $\text{ID\_MatDef} \neq \text{Quantity}$ 
19       $\text{value}(T_{\text{MatLot}}\{5\}) = \text{value}(T_{\text{MatLot}}\{5\}) \ \& \ \mathcal{K}_{\text{MatLot}}\{5\} = 2$ ;
      //  $\text{ID\_MatDef} = \text{ID\_MatDef}$ . Cependant, l'attribut 5 de MaterialLot n'est pas une
      PK. En effet,  $\mathcal{K}_{\text{MatLot}}\{5\} = 2$  car il s'agit d'une FK
      //  $t' = \text{MaterialDefinition}$  avec  $n_{\text{MatDef}} = 3$  (i.e. 3 attributs,  $j_2 = 1:3$ )
20       $\text{value}(T_{\text{MatLot}}\{5\}) = \text{value}(T_{\text{MatDef}}\{1\}) \ \& \ \mathcal{K}_{\text{MatDef}}\{1\} = 1$ ;
      //  $\text{ID\_MatDef} = \text{ID\_MatDef}$  ET l'attribut 1 de MaterialDefinition est une PK
21      forall the  $i_2 = 1 : 4 | \text{cont}(T_{\text{MatLot}}\{2, 5\}) = \text{cont}(T_{\text{MatDef}}\{i_2, 1\})$  do // Il y a 4
      tuples dans MaterialDefinition (i.e.  $m_{\text{MatDef}} = 4$ ), cf. FIGURE 3.2(a)
22       $\text{cont}(T_{\text{MatLot}}\{2, 5\}) \neq \text{cont}(T_{\text{MatDef}}\{1, 1\})$ ; //  $\text{MD06} \neq \text{MD04}$ 
23       $\text{cont}(T_{\text{MatLot}}\{2, 5\}) \neq \text{cont}(T_{\text{MatDef}}\{2, 1\})$ ; //  $\text{MD06} \neq \text{MD99}$ 
24       $\text{cont}(T_{\text{MatLot}}\{2, 5\}) = \text{cont}(T_{\text{MatDef}}\{3, 1\})$ ; //  $\text{MD06} = \text{MD06}$ 
25       $\mathcal{I}' \leftarrow \emptyset \cup T_{\text{MatDef}}\{3, 1\}$ ; //  $T_{\text{MatDef}}\{3, 1\}$  est un data item à explorer
26       $\text{cont}(T_{\text{MatLot}}\{2, 5\}) \neq \text{cont}(T_{\text{MatDef}}\{4, 1\})$ ; //  $\text{MD06} \neq \text{MD77}$ 
27       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{MatDef}}\{2\})$ ; //  $\text{ID\_MatDef} \neq \text{Descript}^*$ 
28       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{MatDef}}\{3\})$ ; //  $\text{ID\_MatDef} \neq \text{Value}$ 
      //  $t' = \text{ManufacturingBill}$  avec  $n_{\text{ManBill}} = 5$  (i.e. 5 attributs,  $j_2 = 1:5$ )
29       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{ManBill}}\{1\})$ ; //  $\text{ID\_MatDef} \neq \text{ID\_ManBill}$ 
30       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{ManBill}}\{2\})$ ; //  $\text{ID\_MatDef} \neq \text{Descript}^*$ 
31       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{ManBill}}\{3\})$ ; //  $\text{ID\_MatDef} \neq \text{Quantity}$ 
32       $\text{value}(T_{\text{MatLot}}\{5\}) = \text{value}(T_{\text{ManBill}}\{4\}) \ \& \ \mathcal{K}_{\text{ManBill}}\{4\} = 2$ ;
      //  $\text{ID\_MatDef} = \text{ID\_MatDef}$ . Cependant, l'attribut 4 de ManufacturingBill n'est
      pas une PK. En effet,  $\mathcal{K}_{\text{ManBill}}\{4\} = 2$  car il s'agit d'une FK
33       $\text{value}(T_{\text{MatLot}}\{5\}) \neq \text{value}(T_{\text{ManBill}}\{5\})$ ; //  $\text{ID\_MatDef} \neq \text{ID\_ProdDef}$ 

```


Algorithme 6: ExploreFK($\mathcal{E}, \overline{\mathcal{T}}, \mathcal{R}$)

```

output :  $\mathcal{I}$ 
// l'évolution de  $\mathcal{I}$  est mise en avant en rouge

1 begin
2    $\mathcal{I} \leftarrow \emptyset$ ; // On initialise la liste des data items restant à parcourir à vide
3    $\mathcal{N} \leftarrow \emptyset$ ; // On initialise la liste des tables non explorées qui sont directement en
   relation avec des tables déjà explorées
4   forall the  $T_t \in [\text{MatLot}, \text{MatDef}]$ ,  $T_{t'} \in [\dots, \text{MatLot}, \text{MatDef}, \text{ManBill}, \dots]$  |  $D_{T_t, T_{t'}} =$ 
   1 &  $T_{t'} \notin [\text{MatLot}, \text{MatDef}]$  do //  $\mathcal{E} = [\text{T}_{\text{MatLot}}\{2,1\}, \text{T}_{\text{MatDef}}\{3,1\}]$  donc les tables déjà
   visitées sont MaterialLot et MaterialDefinition. C'est pourquoi  $T_t \in [\text{MatLot}, \text{MatDef}]$  et
   que  $T_{t'}$  ne doit pas appartenir à ces deux tables ( $T_{t'} \notin [\text{MatLot}, \text{MatDef}]$ ). Nous
   recherchons de nouvelles relations.
   //  $T_t = \text{MaterialLot}$  (i.e. MatLot)
5    $D_{\text{T}_{\text{MatLot}}, \text{T}_{\text{MatLot}}} = 0$ ; // distance entre MaterialLot et MaterialLot  $\neq 1$ 
6    $D_{\text{T}_{\text{MatLot}}, \text{T}_{\text{MatDef}}} = 1$  &  $\text{T}_{\text{MatDef}} \in \mathcal{E}$ ; // distance MaterialLot - MaterialDefinit° = 1.
   Cependant, MaterialDefinition a déjà été visitée (i.e.  $\in \mathcal{E}$ ), donc condition non
   validée
7    $D_{\text{T}_{\text{MatLot}}, \text{T}_{\text{ManBill}}} = 2$ ; // distance entre MaterialLot et ManufacturingBill  $\neq 1$ )
   //  $T_t = \text{MaterialDefinition}$  (i.e. MatDef)
8    $D_{\text{T}_{\text{MatDef}}, \text{T}_{\text{MatLot}}} = 1$  &  $\text{T}_{\text{MatLot}} \in \mathcal{E}$ ; // distance entre MaterialDefinit° et
   MaterialLot = 1. Cependant, MaterialLot a déjà été visitée (i.e.  $\in \mathcal{E}$ )
9    $D_{\text{T}_{\text{MatDef}}, \text{T}_{\text{MatDef}}} = 0$ ; // distance entre MaterialDef. et MaterialDef.  $\neq 1$ 
10   $D_{\text{T}_{\text{MatDef}}, \text{T}_{\text{ManBill}}} = 1$  &  $\text{T}_{\text{ManBill}} \notin \mathcal{E}$ ; // distance entre MaterialDefinit° et
   ManufacturingBill = 1 ET cette table n'a pas encore été explorée (i.e.  $\notin \mathcal{E}$ )
11   $\mathcal{N} \leftarrow \mathcal{N} \cup \text{T}_{\text{ManBill}}$ ; // ManufacturingBill est une table non explorée et
   directement en relation avec une table déjà explorée. On l'ajoute donc à  $\mathcal{N}$ 
12  forall the  $T_t \in [\text{ManBill}]$ ,  $j_1 \in \{1, n_t\}$  |  $\mathcal{K}_t\{j_1\} = 2$  do // Nous recherchons les FK de  $T_t$ 
   // 5 attributs dans ManufacturingBill (i.e.  $n_{\text{ManBill}} = 5$  et  $j_1 = 1:5$ )
13   $\mathcal{K}_{\text{ManBill}}\{1\} = 1$ ; // Le 1er attribut de ManufacturingBill est la PK
14   $\mathcal{K}_{\text{ManBill}}\{2\} = 0$ ; // Le 2ème attribut n'est ni la PK, ni une FK
15   $\mathcal{K}_{\text{ManBill}}\{3\} = 0$ ; // Le 3ème attribut n'est ni la PK, ni une FK
16   $\mathcal{K}_{\text{ManBill}}\{4\} = 2$ ; // Le 4ème attribut EST une FK
17  forall the  $T_{t'} \in [\text{MatLot}, \text{MatDef}]$ ,  $j_2 \in$ 
    $\{1, n_{t'}\}$  |  $\text{value}(\text{T}_{\text{ManBill}}\{4\}) = \text{value}(\text{T}_{t'}\{j_2\})$ ,  $\mathcal{K}_{t'}\{j_2\} = 1$  do // Nous regardons si
   la FK (colonne 4) de ManufacturingBill est une PK d'une table déjà explorée (i.e.
   une table  $T_{t'} \in \mathcal{E} = [\text{MatLot}, \text{MatDef}]$ )
   //  $T_{t'} = \text{MaterialLot}$  (i.e. MatLot) et  $n_{\text{MatLot}} = 5$  (i.e.  $j_2 = 1:5$ )
18   $\text{value}(\text{T}_{\text{ManBill}}\{4\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{1\})$ ; // IDMatDef  $\neq$  IDLot
19   $\text{value}(\text{T}_{\text{ManBill}}\{4\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{2\})$ ; // IDMatDef  $\neq$  Description
20   $\text{value}(\text{T}_{\text{ManBill}}\{4\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{3\})$ ; // IDMatDef  $\neq$  Status
21   $\text{value}(\text{T}_{\text{ManBill}}\{4\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{4\})$ ; // IDMatDef  $\neq$  Quantity
22   $\text{value}(\text{T}_{\text{ManBill}}\{4\}) = \text{value}(\text{T}_{\text{MatLot}}\{5\})$  &  $\mathcal{K}_{\text{MatLot}}\{5\} = 2$ ;
   // IDMatDef = IDMatDef. Cependant, l'attribut 5 de MaterialLot n'est pas une
   PK (i.e.  $\mathcal{K}_{\text{MatLot}}\{5\} \neq 1$ ), ce qui ne valide pas la condition.
   //  $T_{t'} = \text{MaterialDefinition}$  (i.e. MatDef) et  $n_{\text{MatDef}} = 3$  (i.e.  $j_2 = 1:3$ )
23   $\text{value}(\text{T}_{\text{ManBill}}\{4\}) = \text{value}(\text{T}_{\text{MatDef}}\{1\})$  &  $\mathcal{K}_{\text{MatDef}}\{1\} = 1$ ;
   // IDMatDef = IDMatDef ET l'attribut 1 de MaterialDefinit° est une PK
24  forall the  $i_1 = 1 : 4$  |  $\mathcal{R}_{\text{MatDef}}\{i_1, 1\} = \text{true}$  do // Il y a 4 tuples dans
   MaterialDefinition donc  $i_1 = 1:4$ 
25  |  $\mathcal{R}_{\text{MatDef}}\{1, 1\} = \text{false}$ ; // cf. case 1,1 de MatDef  $\in \mathcal{R}$  en FIGURE 3.2(a)
26  |  $\mathcal{R}_{\text{MatDef}}\{2, 1\} = \text{false}$ ; // cf. case 2,1 de MatDef  $\in \mathcal{R}$  en FIGURE 3.2(a)

```

```

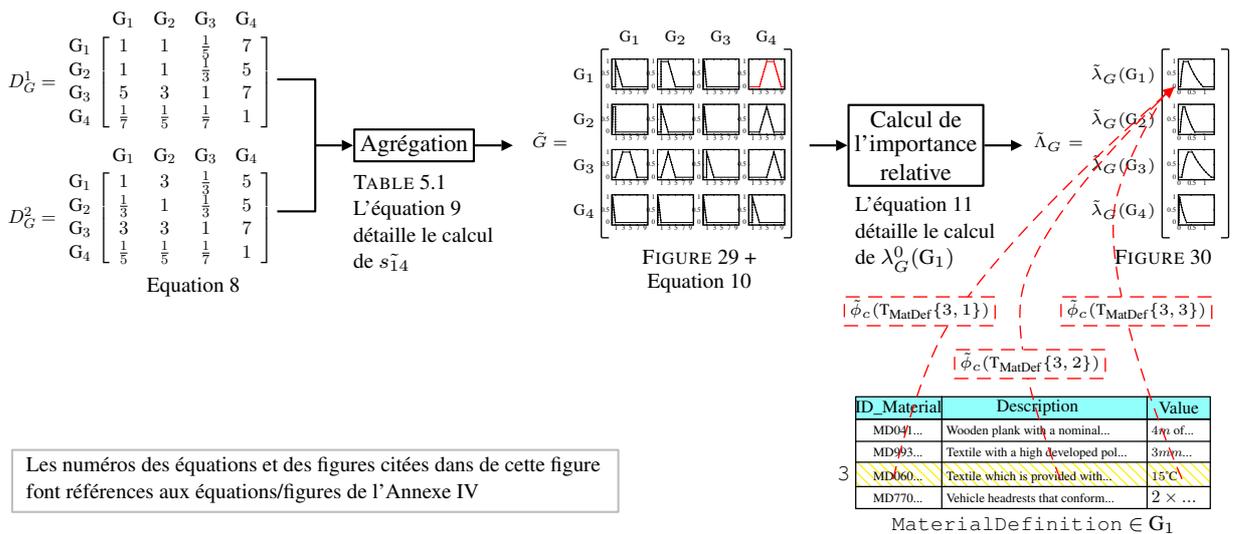
27    $\mathcal{R}_{\text{MatDef}}\{3,1\} = \text{true};$  // cf. case 3,1 de  $\text{MatDef} \in \mathcal{R}$  en FIGURE 3.2(a)
28   forall the  $i_2 = 1 : 3$  |  $\text{cont}(\text{T}_{\text{ManBill}}\{i_2,4\}) = \text{cont}(\text{T}_{\text{MatDef}}\{3,1\})$  do // On
    regarde si cette PK (i.e.  $\text{T}_{\text{MatDef}}\{1,3\}$ ) est une FK dans ManufacturingBill,
    laquelle dispose de 3 tuples (i.e.  $i_2 = 1:3$ )
29   |  $\text{cont}(\text{T}_{\text{ManBill}}\{1,4\}) = \text{cont}(\text{T}_{\text{MatDef}}\{3,1\});$  // MD06 = MD06
30   |  $\mathcal{I} \leftarrow \emptyset \cup \text{T}_{\text{ManBill}}\{1,4\};$  // Data item devant être exploré
31   |  $\text{cont}(\text{T}_{\text{ManBill}}\{2,4\}) \neq \text{cont}(\text{T}_{\text{MatDef}}\{3,1\});$  // MDY-1  $\neq$  MD06
32   |  $\text{cont}(\text{T}_{\text{ManBill}}\{3,4\}) = \text{cont}(\text{T}_{\text{MatDef}}\{3,1\});$  // MD06 = MD06
33   |  $\mathcal{I} \leftarrow \text{T}_{\text{ManBill}}\{1,4\} \cup \text{T}_{\text{ManBill}}\{3,4\};$  // Data item devant être exploré
34    $\mathcal{R}_{\text{MatDef}}\{4,1\} = \text{false}$ 
35    $\text{value}(\text{T}_{\text{ManBill}}\{4\}) \neq \text{value}(\text{T}_{\text{MatDef}}\{2\});$  // IDMatDef  $\neq$  Description
36    $\text{value}(\text{T}_{\text{ManBill}}\{4\}) \neq \text{value}(\text{T}_{\text{MatDef}}\{3\});$  // IDMatDef  $\neq$  Status
37    $\mathcal{K}_{\text{ManBill}}\{5\} = 2;$  // Le 5ème attribut est une FK
38   forall the  $\text{T}_{t'} \in [\text{MatLot}, \text{MatDef}]$ ,  $j_2 \in$ 
     $\{1, n_{t'}\}$  |  $\text{value}(\text{T}_{\text{ManBill}}\{1,5\}) = \text{value}(\text{T}_{t'}\{1, j_2\})$ ,  $\mathcal{K}_{t'}\{j_2\} = 1$  do // Nous regardons
    si la FK (colonne 5) de ManufacturingBill est une PK d'une table déjà explorée
    (i.e. une table  $\in \mathcal{E} = [\text{MatLot}, \text{MatDef}]$ )
    //  $\text{T}_{t'} = \text{MaterialLot}$  (i.e. MatLot) et  $n_{\text{MatLot}} = 5$  (i.e.  $j_2 = 1:5$ )
39   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{1\});$  // IDProdDef  $\neq$  IDLot
40   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{2\});$  // IDProdDef  $\neq$  Description
41   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{3\});$  // IDProdDef  $\neq$  Status
42   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{4\});$  // IDProdDef  $\neq$  Quantity
43   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatLot}}\{5\});$  // IDProdDef  $\neq$  IDMatDef
    //  $\text{T}_{t'} = \text{MaterialDefinition}$  (i.e. MatDef) et  $n_{\text{MatDef}} = 3$  (i.e.  $j_2 = 1:3$ )
44   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatDef}}\{1\});$  // IDProdDef  $\neq$  IDMatDef
45   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatDef}}\{2\});$  // IDProdDef  $\neq$  Description
46   |  $\text{value}(\text{T}_{\text{ManBill}}\{5\}) \neq \text{value}(\text{T}_{\text{MatDef}}\{3\});$  // IDProdDef  $\neq$  Value

```

Annexe IV : Détail des calculs liés à l'approche AHP-Floue

Cette annexe détaille les calculs qui permettent d'établir les résultats présentés dans les figures miniaturisées de la section 3.3 (résultats relatifs à AHP-Floue). Pour chaque critère, nous rappelons la figure miniaturisée.

Contextual



$$D_G^1 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ G_1 & \begin{bmatrix} 1 & 1 & \frac{1}{5} & 7 \end{bmatrix} \\ G_2 & \begin{bmatrix} 1 & 1 & \frac{1}{3} & 5 \end{bmatrix} \\ G_3 & \begin{bmatrix} 5 & 3 & 1 & 7 \end{bmatrix} \\ G_4 & \begin{bmatrix} \frac{1}{7} & \frac{1}{5} & \frac{1}{7} & 1 \end{bmatrix} \end{matrix} \quad D_G^2 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ G_1 & \begin{bmatrix} 1 & 3 & \frac{1}{3} & 5 \end{bmatrix} \\ G_2 & \begin{bmatrix} \frac{1}{3} & 1 & \frac{1}{3} & 5 \end{bmatrix} \\ G_3 & \begin{bmatrix} 3 & 3 & 1 & 7 \end{bmatrix} \\ G_4 & \begin{bmatrix} \frac{1}{5} & \frac{1}{5} & \frac{1}{7} & 1 \end{bmatrix} \end{matrix} \quad (8)$$

$$\min(s_{14}^1, s_{14}^2) = 5 \quad \max(s_{14}^1, s_{14}^2) = 7 \quad (9)$$

$$\tilde{s}_{14} = [(5 - 2) \ 5 \ 7 \ (7 + 2)]$$

$$= [3 \ 5 \ 7 \ 9]$$

$$\tilde{G} = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ G_1 & \begin{bmatrix} [1 \ 1 \ 1 \ 3] & [1 \ 1 \ 3 \ 5] & [\frac{1}{9} \ \frac{1}{5} \ \frac{1}{3} \ 1] & [3 \ 5 \ 7 \ 9] \end{bmatrix} \\ G_2 & \begin{bmatrix} [\frac{1}{9} \ \frac{1}{3} \ 1 \ 1] & [1 \ 1 \ 1 \ 3] & [\frac{1}{9} \ \frac{1}{3} \ \frac{1}{3} \ 1] & [3 \ 5 \ 5 \ 3] \end{bmatrix} \\ G_3 & \begin{bmatrix} [1 \ 3 \ 5 \ 7] & [1 \ 3 \ 3 \ 5] & [1 \ 1 \ 1 \ 3] & [5 \ 7 \ 7 \ 9] \end{bmatrix} \\ G_4 & \begin{bmatrix} [\frac{1}{9} \ \frac{1}{7} \ \frac{1}{5} \ 1] & [\frac{1}{9} \ \frac{1}{5} \ \frac{1}{5} \ 1] & [\frac{1}{9} \ \frac{1}{7} \ \frac{1}{7} \ 1] & [1 \ 1 \ 1 \ 3] \end{bmatrix} \end{matrix} \quad (10) \quad \tilde{G} =$$

$$\lambda_{G_1}^0 = \frac{G^0(1,1) + G^0(1,2) + G^0(1,3) + G^0(1,4)}{G^0(1,1) + G^0(1,2) + \dots + G^0(4,3) + G^0(4,4)} \quad (11)$$

$$= \frac{[1 \ 3] + [1 \ 5] + [\frac{1}{9} \ 1] + [3 \ 9]}{[1 \ 3] + [1 \ 5] + [\frac{1}{9} \ 1] + \dots + [\frac{1}{9} \ 1] + [1 \ 3]}$$

$$= \frac{[\frac{46}{9} \ 18]}{[\frac{170}{9} \ 56]} = \begin{bmatrix} \frac{46}{9} & 18 \\ 56 & \frac{170}{9} \end{bmatrix} = \begin{bmatrix} 46 & 162 \\ 504 & 170 \end{bmatrix} = [0.09 \ 0.95]$$

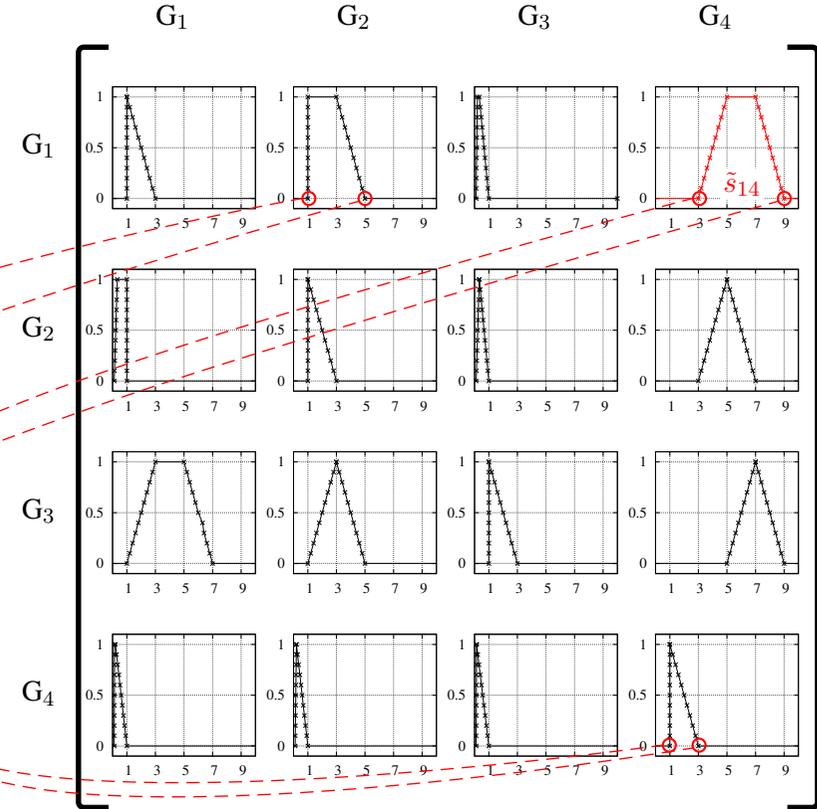


FIGURE 29: Matrice d'agrégation des avis d'experts : critère Contextual

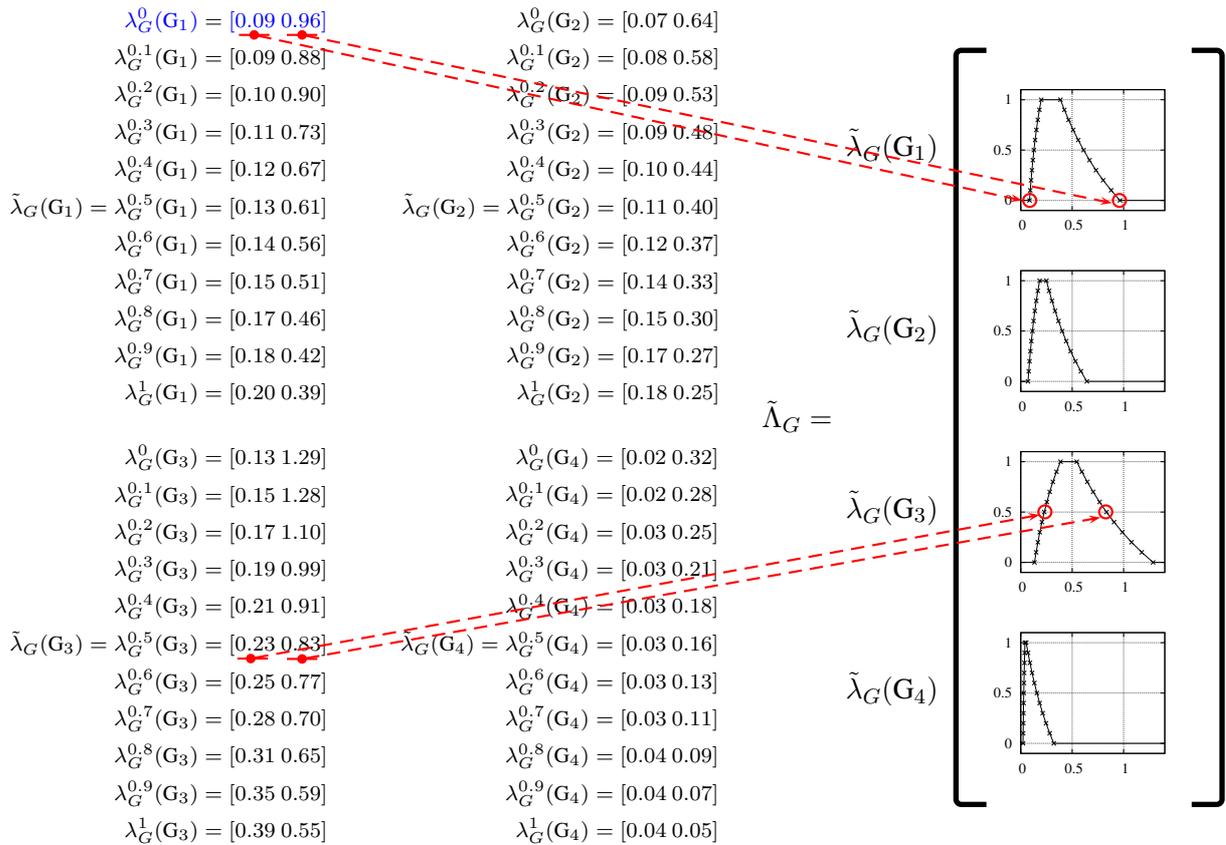
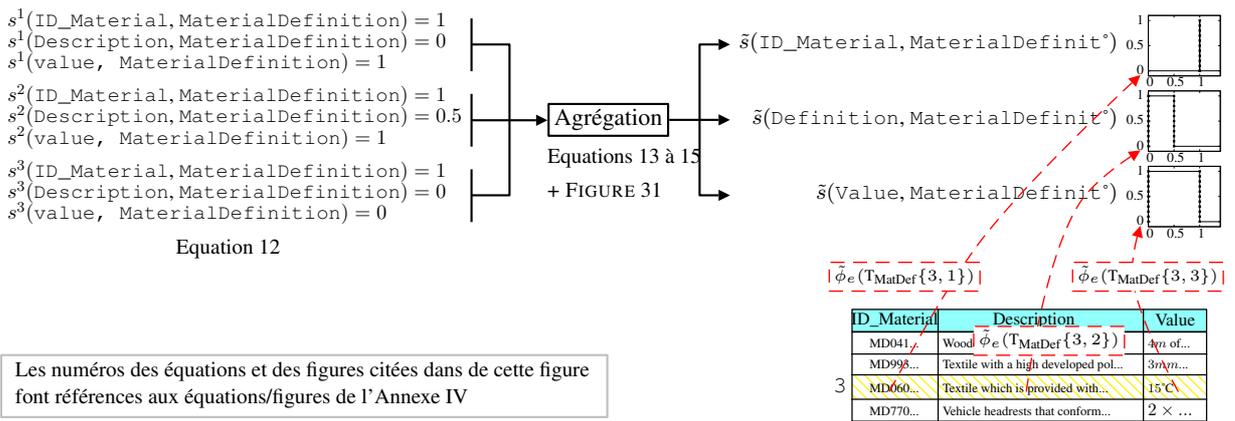


FIGURE 30: Importance relative de chaque groupe d'entité : $\tilde{\Lambda}_G$

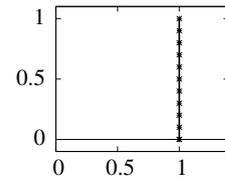
Enumeration



$$\begin{aligned}
 &s^1(\text{ID_Material}, \text{MaterialDefinit}^\circ) = 1 & s^2(\text{ID_Mat} \dots) = 1 & s^3(\text{ID_Mat} \dots) = 1 \\
 &s^1(\text{Description}, \text{MaterialDefinit}^\circ) = 0 & s^2(\text{Descri} \dots) = 0.5 & s^3(\text{Descri} \dots) = 0 \\
 &s^1(\text{MaterialDefinit}^\circ) = 1 & s^2(\text{value}, \dots) = 1 & s^3(\text{value}, \dots) = 0
 \end{aligned} \quad (12)$$

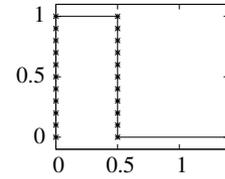
$$\tilde{s}(\text{ID_Material}, \text{MaterialDef} \dots) = [\min(1, 1, 1) \quad \max(1, 1, 1)] \quad (13)$$

$$= [1 \quad 1]$$



$$\tilde{s}(\text{Description}, \text{MaterialDef} \dots) = [\min(0, 0.5, 0) \quad \max(0, 0.5, 0)] \quad (14)$$

$$= [0 \quad 0.5]$$



$$\tilde{s}(\text{value}, \text{MaterialDef} \dots) = [\min(1, 1, 0) \quad \max(1, 1, 0)] \quad (15)$$

$$= [0 \quad 1]$$

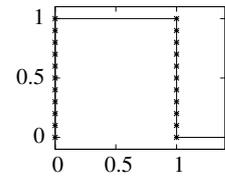
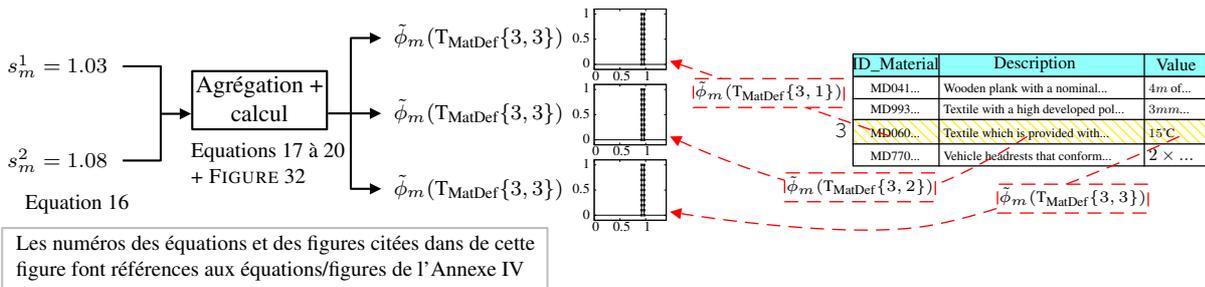


FIGURE 31: Sous-ensembles flous $\tilde{s}(v, t)$ de l'attribut v de la table t

Model-Based



$$s_m^1 = 1.03 \quad s_m^2 = 1.08 \quad (16)$$

$$\tilde{s}_m = [\min(1.03, 1.08) \quad \max(1.03, 1.08)] = [1.03 \quad 1.08] \quad (17)$$

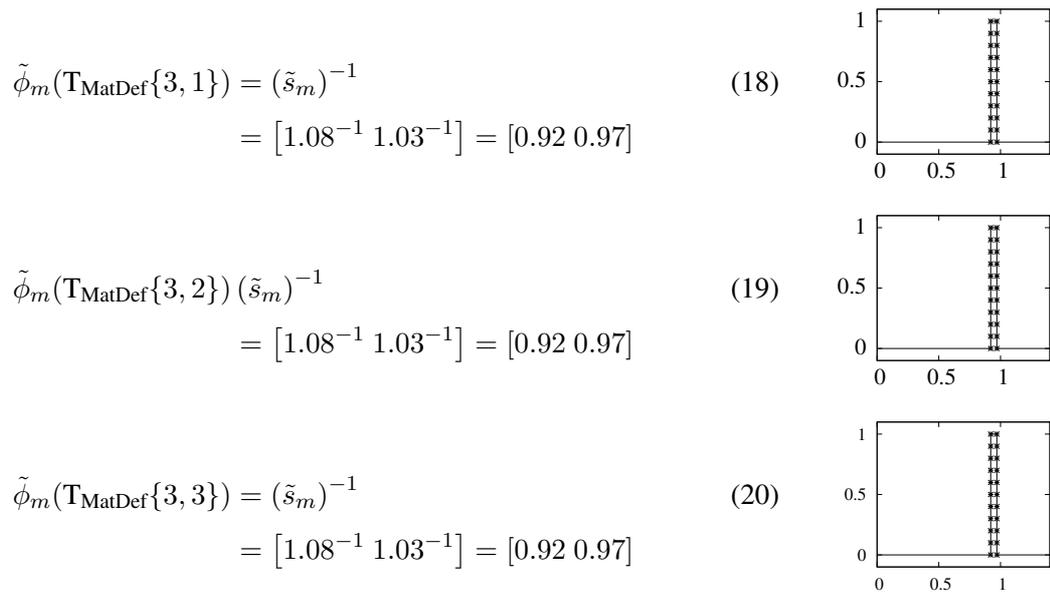
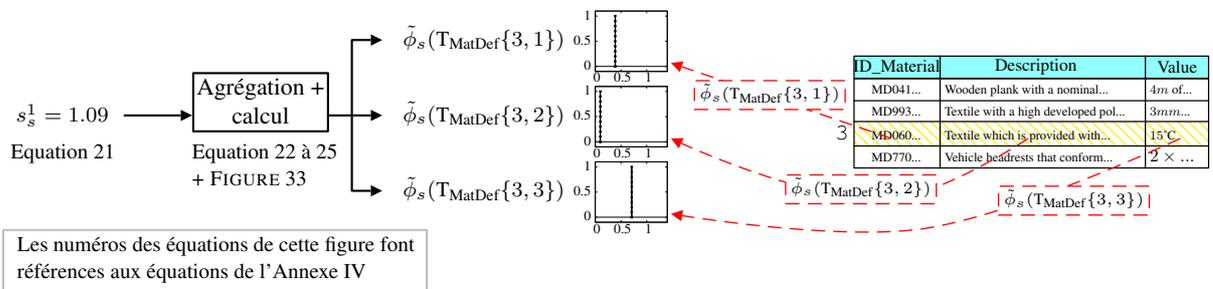


FIGURE 32: Sous-ensembles flous $\tilde{\phi}_m(l)$: score du data item l au regard du critère C_m

Data Size

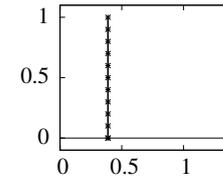


$$s_s^1 = 1.09 \quad (21)$$

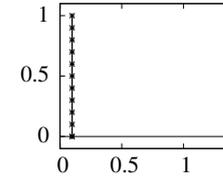
$$\tilde{s}_s = [\min(1.09) \ \max(1.09)] = [1.09 \ 1.09] \quad (22)$$

Les puissances d dans les ÉQUATIONS 23, 24 et 25 sont respectivement égales à 11, 27 et 4 qui sont fonction des tailles respectives en octets de $\mathbf{T}_{\text{MatDef}}\{3, 1\}$, $\mathbf{T}_{\text{MatDef}}\{3, 2\}$ et $\mathbf{T}_{\text{MatDef}}\{3, 3\}$.

$$\begin{aligned} \tilde{\phi}_s(\mathbf{T}_{\text{MatDef}}\{3, 1\}) &= (\tilde{s}_s)^{-11} \\ &= [1.09^{-11} \ 1.09^{-11}] = [0.39 \ 0.39] \end{aligned} \quad (23)$$



$$\begin{aligned} \tilde{\phi}_s(\mathbf{T}_{\text{MatDef}}\{3, 2\}) &= (\tilde{s}_s)^{-27} \\ &= [1.09^{-27} \ 1.09^{-27}] = [0.10 \ 0.10] \end{aligned} \quad (24)$$



$$\begin{aligned} \tilde{\phi}_s(\mathbf{T}_{\text{MatDef}}\{3, 3\}) &= (\tilde{s}_s)^{-4} \\ &= [1.09^{-4} \ 1.09^{-4}] = [0.71 \ 0.71] \end{aligned} \quad (25)$$

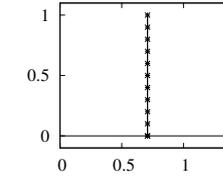
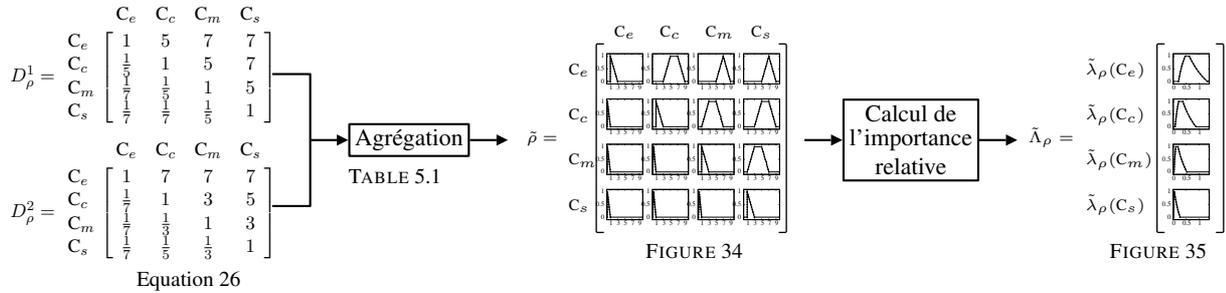


FIGURE 33: Sous-ensembles flous $\tilde{\phi}_s(l)$: score du data item l au regard du critère C_s

Importance des Critères



$$D_\rho^1 = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 5 & 7 & 7 \\ \frac{1}{5} & 1 & 5 & 7 \\ \frac{1}{7} & \frac{1}{5} & 1 & 5 \\ \frac{1}{7} & \frac{1}{7} & \frac{1}{5} & 1 \end{bmatrix} \\ C_c & \\ C_m & \\ C_s & \end{matrix} \quad D_\rho^2 = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 7 & 7 & 7 \\ \frac{1}{7} & 1 & 3 & 5 \\ \frac{1}{7} & \frac{1}{3} & 1 & 3 \\ \frac{1}{7} & \frac{1}{5} & \frac{1}{3} & 1 \end{bmatrix} \\ C_c & \\ C_m & \\ C_s & \end{matrix} \quad (26)$$

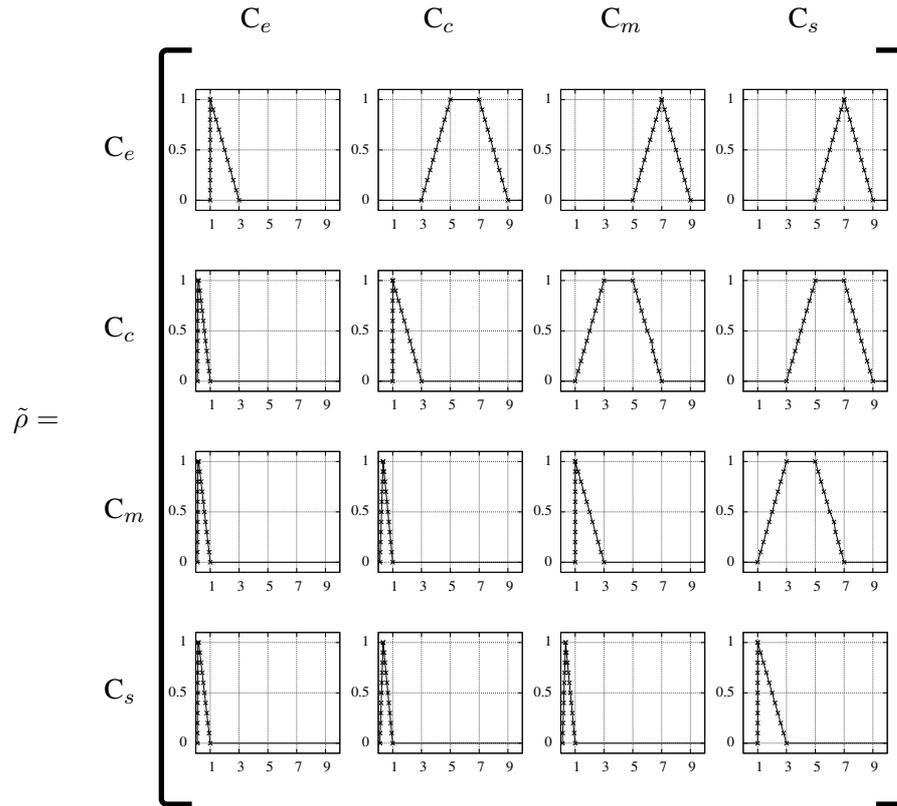
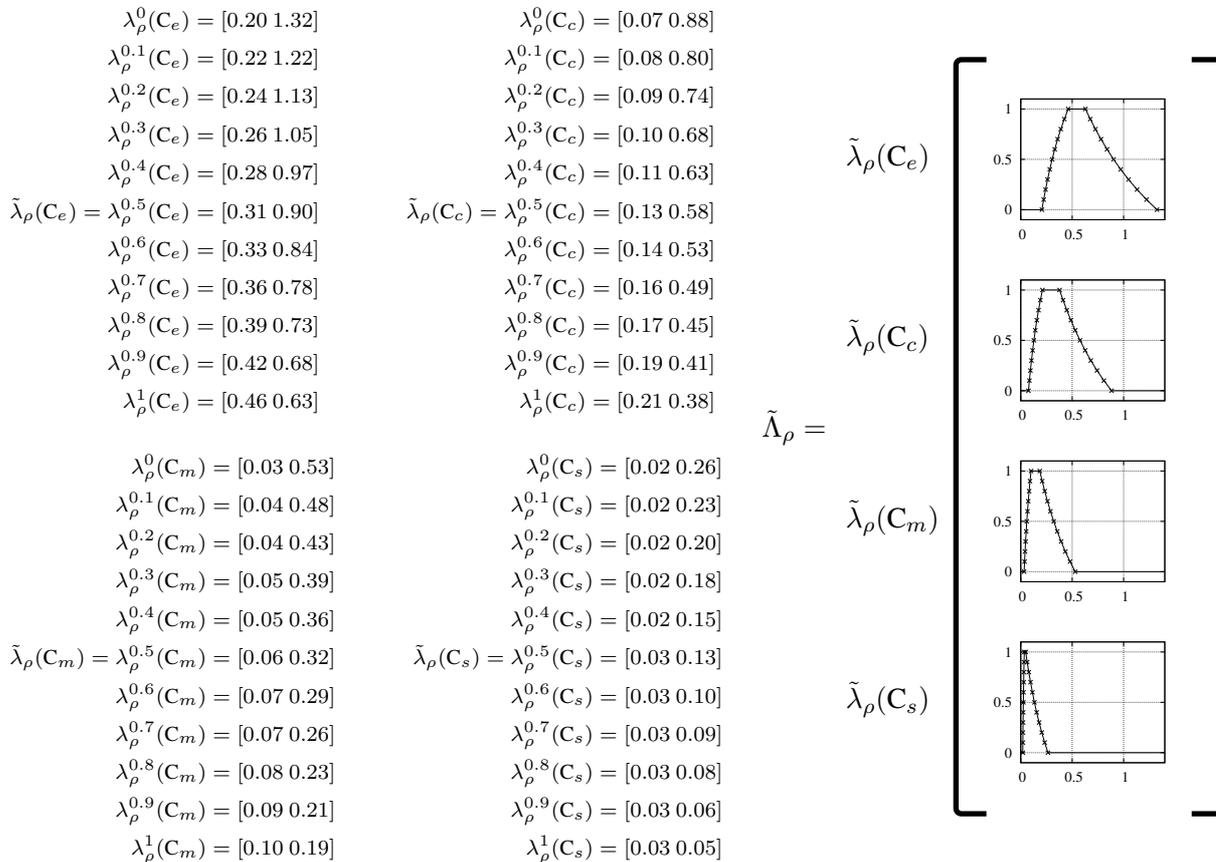


FIGURE 34: Matrice d'agrégation des avis d'experts : critères

FIGURE 35: Importance relative de chaque critère : $\tilde{\Lambda}_\rho$

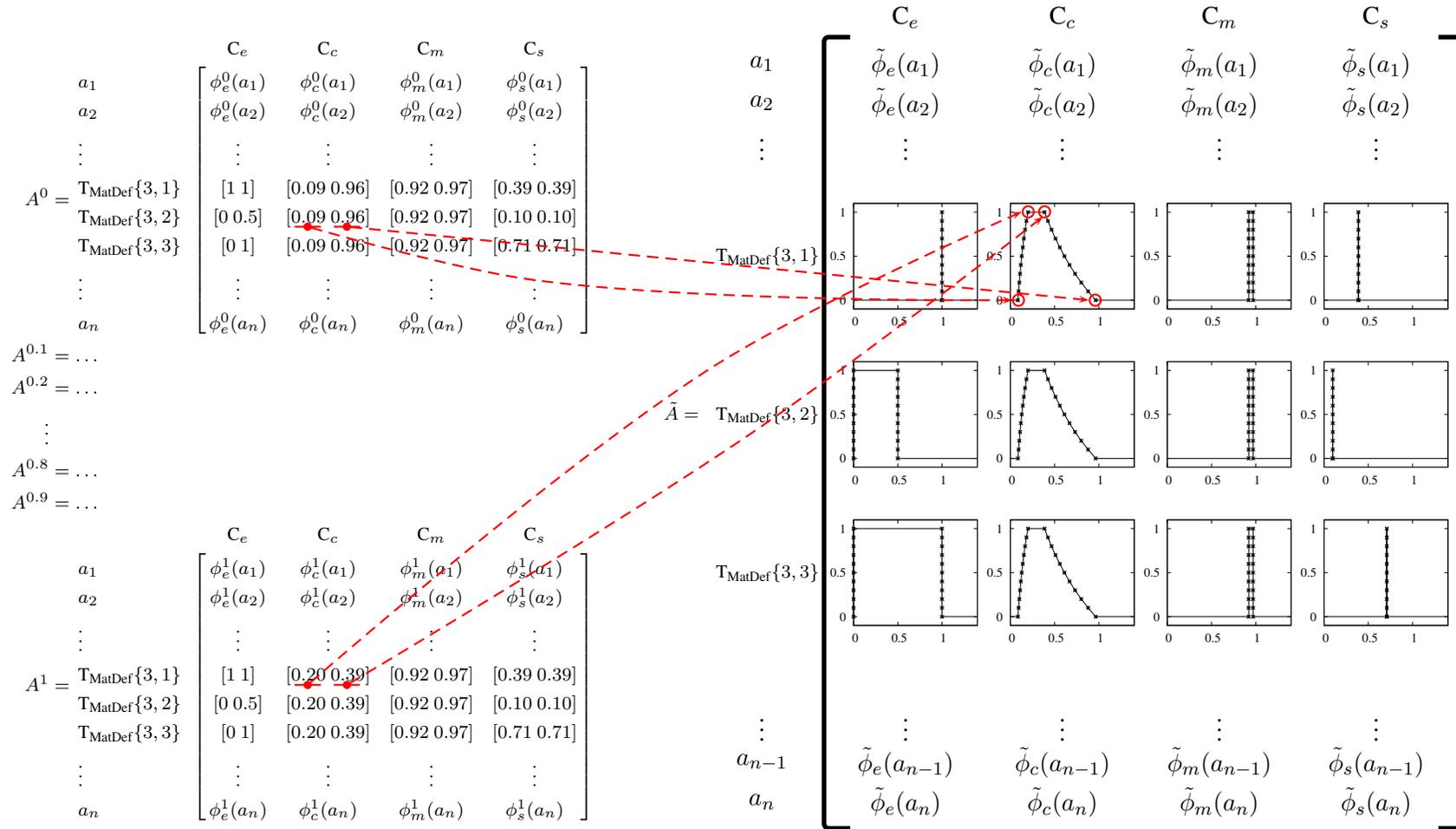


FIGURE 36: Matrice de jugement flou \tilde{A} : sous-ensembles flous des alternatives au vu des critères

La matrice de performance floue \tilde{H} est atteinte en multipliant le score de chaque alternative l au regard de chaque critère x (c.-à-d., $\tilde{\phi}_x(l)$), par l'importance relative du critère x lui-même (c.-à-d., $\tilde{\lambda}_\rho(\mathbf{C}_x)$). Cette multiplication s'effectue bien entendu sur la base de leurs α -coupes comme détaillé dans l'ÉQUATION 27. Le calcul de $h_e^0(\mathbf{T}_{\text{MatDef}}\{3, 2\})$ est détaillé dans l'ÉQUATION 28. Ce calcul consiste à multiplier les intervalles de l' α -coupe de niveau 0 de $\tilde{\phi}_e(\mathbf{T}_{\text{MatDef}}\{3, 2\})$ (i.e., $[0 \ 0.5]$, cf. FIGURE 36), avec les intervalles $\lambda_\rho^0(\mathbf{C}_e)$ (i.e. $[0.2 \ 1.32]$, cf. FIGURE 35).

$$\begin{aligned}
 H^\alpha &= \begin{array}{c} a_1 \\ a_2 \\ \vdots \\ a_n \end{array} \begin{array}{cccc} \mathbf{C}_e & \mathbf{C}_c & \mathbf{C}_m & \mathbf{C}_s \\ \left[\begin{array}{cccc} \phi_e^\alpha(a_1) \times \lambda_\rho^\alpha(\mathbf{C}_e) & \phi_c^\alpha(a_1) \times \lambda_\rho^\alpha(\mathbf{C}_c) & \phi_m^\alpha(a_1) \times \lambda_\rho^\alpha(\mathbf{C}_m) & \phi_s^\alpha(a_1) \times \lambda_\rho^\alpha(\mathbf{C}_s) \\ \phi_e^\alpha(a_2) \times \lambda_\rho^\alpha(\mathbf{C}_e) & \phi_c^\alpha(a_2) \times \lambda_\rho^\alpha(\mathbf{C}_c) & \phi_m^\alpha(a_2) \times \lambda_\rho^\alpha(\mathbf{C}_m) & \phi_s^\alpha(a_2) \times \lambda_\rho^\alpha(\mathbf{C}_s) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_e^\alpha(a_n) \times \lambda_\rho^\alpha(\mathbf{C}_e) & \phi_c^\alpha(a_n) \times \lambda_\rho^\alpha(\mathbf{C}_c) & \phi_m^\alpha(a_n) \times \lambda_\rho^\alpha(\mathbf{C}_m) & \phi_s^\alpha(a_n) \times \lambda_\rho^\alpha(\mathbf{C}_s) \end{array} \right] \end{array} & (27) \\
 &= \begin{array}{c} a_1 \\ a_2 \\ \vdots \\ a_n \end{array} \begin{array}{cccc} \mathbf{C}_e & \mathbf{C}_c & \mathbf{C}_m & \mathbf{C}_s \\ \left[\begin{array}{cccc} h_e^\alpha(a_1) & h_c^\alpha(a_1) & h_m^\alpha(a_1) & h_s^\alpha(a_1) \\ h_e^\alpha(a_2) & h_c^\alpha(a_2) & h_m^\alpha(a_2) & h_s^\alpha(a_2) \\ \vdots & \vdots & \vdots & \vdots \\ h_e^\alpha(a_n) & h_c^\alpha(a_n) & h_m^\alpha(a_n) & h_s^\alpha(a_n) \end{array} \right] \end{array} \quad \forall \alpha = \{0.1, 0.2, \dots, 0.9, 1\}
 \end{aligned}$$

$$\begin{aligned}
 h_e^0(\text{TMatDef}\{3, 2\}) &= \phi_e^0(\text{TMatDef}\{3, 2\}) \times \lambda_\rho^0(e) \\
 &= [0 \ 0.5] \times [0.2 \ 1.32] = [0 \ 0.66]
 \end{aligned}
 \tag{28}$$

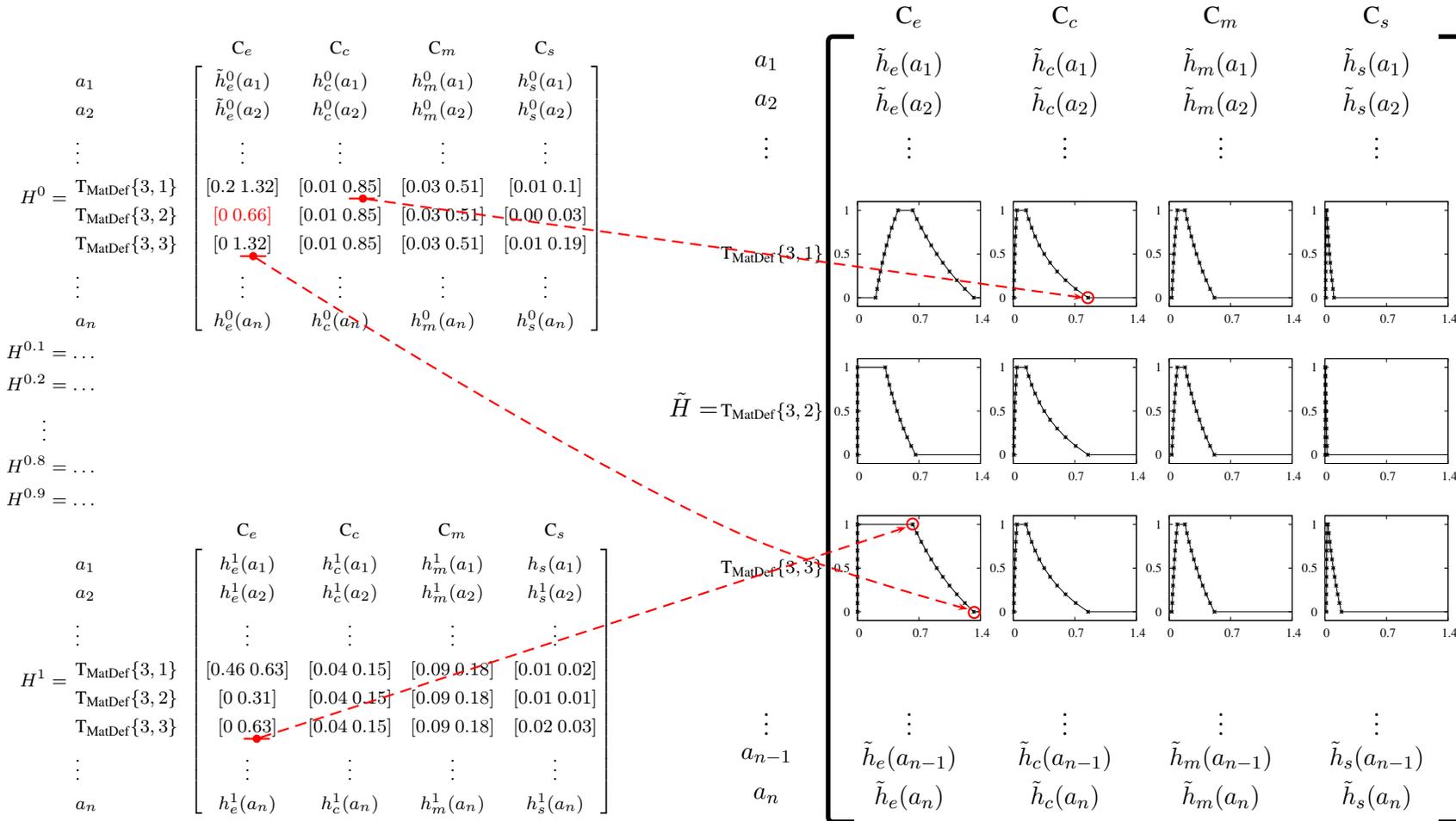


FIGURE 37: Matrice de performance floue \tilde{H} : prise en compte de l'importance des critères

La fonction d'agrégation Ω qui est mise en œuvre dans notre approche AC consiste à sommer les sous-ensembles flous relatifs à chaque alternative dans \tilde{H} (c.-à-d. sommer $\tilde{h}_x(l) \forall x = \{e, c, m, s\}$), ce qui donne lieu à une matrice floue notée \tilde{R} . Cette somme s'effectue également sur la base de leurs α -coupes comme détaillé dans l'ÉQUATION 29 et 30.

$$r^\alpha(l) = \sum_{i=1}^M h_i^\alpha(l) \quad \forall \alpha = \{0, 0.1, \dots, 0.9, 1\} \quad (29) \quad R^\alpha = \begin{bmatrix} a_1 & r^\alpha(a_1) \\ a_2 & r^\alpha(a_2) \\ \vdots & \vdots \\ a_n & r^\alpha(a_n) \end{bmatrix} \quad (30)$$

L'ÉQUATION 31 présente le calcul de la somme des quatre sous-ensembles flous relatifs à $T_{\text{MatDef}\{3, 1\}}$ à la coupe $\alpha = 0$. Nous affichons dans la matrice \tilde{R} , donnée dans la FIGURE 3.22, les résultats relatifs aux trois data items de l'étude.

$$r^0(T_{\text{MatDef}\{3, 1\}}) = [0.2 \ 1.32] + [0.01 \ 0.85] + [0.03 \ 0.51] + [0.01 \ 0.01] \quad (31) \\ = [0.25 \ 2.69]$$

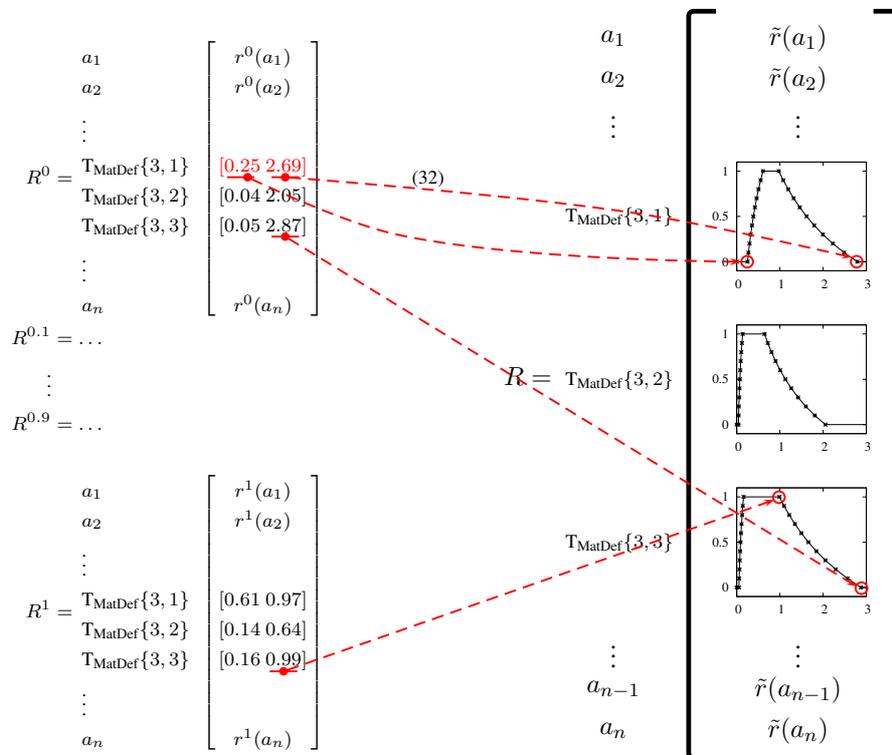


FIGURE 38: Matrice \tilde{R} contenant le sous-ensemble flou relatif à chaque alternative

Les calculs du CoG supérieur et inférieur de $T_{\text{MatDef}}\{3, 1\}$ (c.-à-d. les CoG du sous-ensemble flou noté $\tilde{r}(T_{\text{MatDef}}\{3, 1\})$) sont respectivement détaillés dans les FIGURES 39 et 40, avec r^α la valeur minimale de l' α -coupe et \bar{r}^α sa valeur maximale. $\Delta_Y r^\alpha$ correspond à la différence de niveau entre la coupe α et la coupe $\alpha + 1$. Nous observons que le CoG supérieur y^\star de $\tilde{r}(T_{\text{MatDef}}\{3, 1\})$ est égal à 1.089 et le CoG inférieur est égal à 1.016. La valeur du CoG, rappelons-le, se situe sur l'axe des abscisses comme illustré sur les FIGURES 39 et 40.

COG supérieur

$$\begin{aligned} r^\alpha & \quad \bar{r}^\alpha \\ r^0(T_{\text{MatDef}}\{3, 1\}) &= [0.25 \ 2.79] \\ r^{0.1}(T_{\text{MatDef}}\{3, 1\}) &= [0.27 \ 2.49] \\ r^{0.2}(T_{\text{MatDef}}\{3, 1\}) &= [0.30 \ 2.22] \\ r^{0.3}(T_{\text{MatDef}}\{3, 1\}) &= [0.32 \ 1.99] \\ r^{0.4}(T_{\text{MatDef}}\{3, 1\}) &= [0.35 \ 1.79] \\ r^{0.5}(T_{\text{MatDef}}\{3, 1\}) &= [0.39 \ 1.62] \\ r^{0.6}(T_{\text{MatDef}}\{3, 1\}) &= [0.42 \ 1.46] \\ r^{0.7}(T_{\text{MatDef}}\{3, 1\}) &= [0.46 \ 1.32] \\ r^{0.8}(T_{\text{MatDef}}\{3, 1\}) &= [0.50 \ 1.19] \\ r^{0.9}(T_{\text{MatDef}}\{3, 1\}) &= [0.55 \ 1.08] \\ r^1(T_{\text{MatDef}}\{3, 1\}) &= [0.61 \ 0.97] \end{aligned}$$

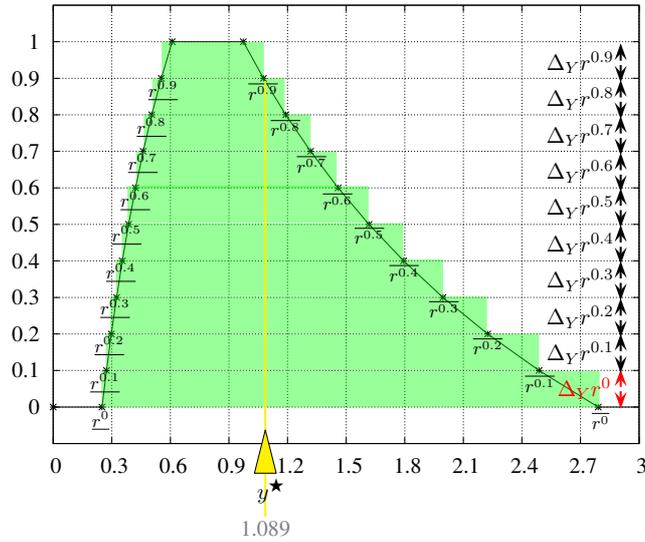
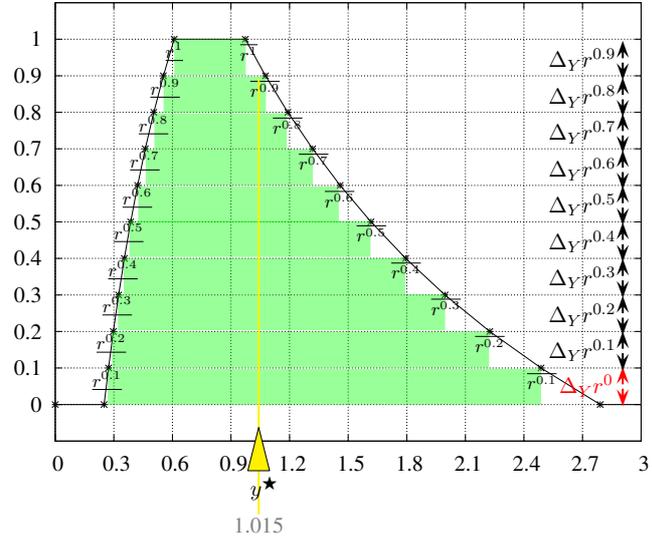


FIGURE 39: Calcul du CoG supérieur du data item $T_{\text{MatDef}}\{3, 1\}$

$$\begin{aligned} y^\star &= \frac{\sum_{\alpha=0}^{0.9} \left[\frac{r^\alpha + \bar{r}^\alpha}{2} \times \Delta_Y r^\alpha \right]}{\sum_{\alpha=0}^{0.9} \Delta_Y r^\alpha} \quad (33) \\ &= \frac{\left[\frac{r^0 + \bar{r}^0}{2} \times \Delta_Y r^0 \right] + \left[\frac{r^{0.1} + \bar{r}^{0.1}}{2} \times \Delta_Y r^{0.1} \right] + \dots + \left[\frac{r^{0.8} + \bar{r}^{0.8}}{2} \times \Delta_Y r^{0.8} \right] + \left[\frac{r^{0.9} + \bar{r}^{0.9}}{2} \times \Delta_Y r^{0.9} \right]}{\Delta_Y r^0 + \Delta_Y r^{0.1} + \dots + \Delta_Y r^{0.8} + \Delta_Y r^{0.9}} \\ &= \frac{\left[\frac{0.25+2.79}{2} \times 0.1 \right] + \left[\frac{0.27+2.49}{2} \times 0.1 \right] + \dots + \left[\frac{0.50+1.19}{2} \times 0.1 \right] + \left[\frac{0.55+1.08}{2} \times 0.1 \right]}{0.1 + 0.1 + \dots + 0.1 + 0.1} \\ &= \frac{0.152 + 0.138 + \dots + 0.084 + 0.081}{1} = 1.089 \end{aligned}$$

COG inférieur

$$\begin{aligned} r^\alpha & \overline{r^\alpha} \\ r^0(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.25 \ 2.79] \\ r^{0.1}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.27 \ 2.49] \\ r^{0.2}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.30 \ 2.22] \\ r^{0.3}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.32 \ 1.99] \\ r^{0.4}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.35 \ 1.79] \\ r^{0.5}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.39 \ 1.62] \\ r^{0.6}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.42 \ 1.46] \\ r^{0.7}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.46 \ 1.32] \\ r^{0.8}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.50 \ 1.19] \\ r^{0.9}(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.55 \ 1.08] \\ r^1(\text{T}_{\text{MatDef}}\{3, 1\}) &= [0.61 \ 0.97] \end{aligned}$$

FIGURE 40: Calcul du CoG inférieur du data item $\text{T}_{\text{MatDef}}\{3, 1\}$

$$\begin{aligned} y^\star &= \frac{\sum_{\alpha=0.1}^1 \left[\frac{r^\alpha + \overline{r^\alpha}}{2} \times \Delta_Y r^{\alpha-0.1} \right]}{\sum_{\alpha=0}^{0.9} \Delta_Y r^\alpha} \quad (34) \\ &= \frac{\left[\frac{r^{0.1} + \overline{r^{0.1}}}{2} \times \Delta_Y r^0 \right] + \left[\frac{r^{0.2} + \overline{r^{0.2}}}{2} \times \Delta_Y r^{0.1} \right] + \dots + \left[\frac{r^{0.9} + \overline{r^{0.9}}}{2} \times \Delta_Y r^{0.8} \right] + \left[\frac{r^1 + \overline{r^1}}{2} \times \Delta_Y r^{0.9} \right]}{\Delta_Y r^0 + \Delta_Y r^{0.1} + \dots + \Delta_Y r^{0.8} + \Delta_Y r^{0.9}} \\ &= \frac{\left[\frac{0.27 + 2.49}{2} \times 0.1 \right] + \left[\frac{0.30 + 2.22}{2} \times 0.1 \right] + \dots + \left[\frac{0.55 + 1.08}{2} \times 0.1 \right] + \left[\frac{0.61 + 0.97}{2} \times 0.1 \right]}{0.1 + 0.1 + \dots + 0.1 + 0.1} \\ &= \frac{0.138 + 0.126 + \dots + 0.081 + 0.079}{1} = 1.015 \end{aligned}$$

Annexe V : Prototypage d'un textile communicant

Cette annexe résume les travaux présentés à la conférence IESM 2011 (Kubler et al., 2011b), traitant de la conception d'un prototype de textile communicant. L'approche menée pour disséminer en masse des tags RFID dans le textile, ainsi que pour le choix du procédé de fabrication à mettre en œuvre, sont présentés dans ce qui suit.

Les *e*-textiles sont des matériaux intelligents dans lesquels sont greffés des capteurs ou tout autre type d'éléments électroniques. Les deux principaux secteurs qui se développent autour de cette technologie sont l'habillement et le médical (Potdar et al., 2009; Catrysse et al., 2004). Les *e*-textiles sont généralement classés en deux catégories relatives aux types de fabrication développés. Le premier consiste en la création d'un réseau filaire de cuivre ou autres métaux conducteurs sur lequel on vient greffer des composants. Cette technologie permet l'utilisation de composants de grande puissance mais aura comme désavantage de fabriquer des *e*-textiles dont la rigidité est inconfortable pour l'utilisateur. De plus, la rupture du fil électrique provoque la destruction intégrale du prototype. Le second type utilise des composants flexibles tels que la fibre optique, nanofibre de carbone, matériaux piézoélectrique, ... ce qui permet de conserver le confort, la légèreté et la souplesse du textile. Le prototype conçu dans notre étude est basé sur la seconde catégorie.

Dans un premier temps, il faut pour chaque technologie étudier et modéliser la zone de couverture du lecteur RFID (c.-à-d. la zone dans laquelle un lecteur et un tag RFID peuvent communiquer). Ceci est nécessaire afin de disséminer les tags RFID dans la matière tout en respectant les contraintes protocolaires. Par exemple, il n'est pas possible de disséminer plus d'un tag par zone de couverture avec la technologie Hitachi et pas plus de 8 tags avec la technologie Omron. La démarche permettant de modéliser cette zone repose sur une structure spécifique, détaillée dans la FIGURE 41). Elle est composée d'un socle sur lequel repose l'émetteur, positionné contre deux butées, réglables en hauteur. Sur ces dernières repose un plateau quadrillé où est positionnée la puce (cf. FIGURE 41(a)). Dans un premier temps, nous avons fait varier la hauteur du plateau et avons pour chacune d'elles, parcouru l'ensemble des positions possibles de la puce sur le plateau selon les axes x et y . Ainsi, il est possible de noter où le signal est perdu. L'orientation de l'antenne a également son importance sur la zone de couverture. La FIGURE 41(b) présente la plate-forme utilisée pour tester la rotation sur les 3 axes. La zone de couverture $3D$ peut finalement

être déterminée comme illustrée par la forme verte centrale dans la FIGURE 42(a) (forme réelle relative à la technologie RFID Hitachi). Au total, en considérant 8 angles et 10 paliers sur les plate-formes, 160 sphères de lecture sont obtenues. Afin de simplifier l'étude, nous modélisons cette sphère comme un cylindre parfait (forme cylindrique rouge dans la FIGURE 42(a)), qui est une simplification de la forme réelle de manière à s'assurer que quelle que soit la position du lecteur au-dessus du textile (hauteur, angle), ce dernier est capable de lire les tags RFID dans des conditions acceptables.

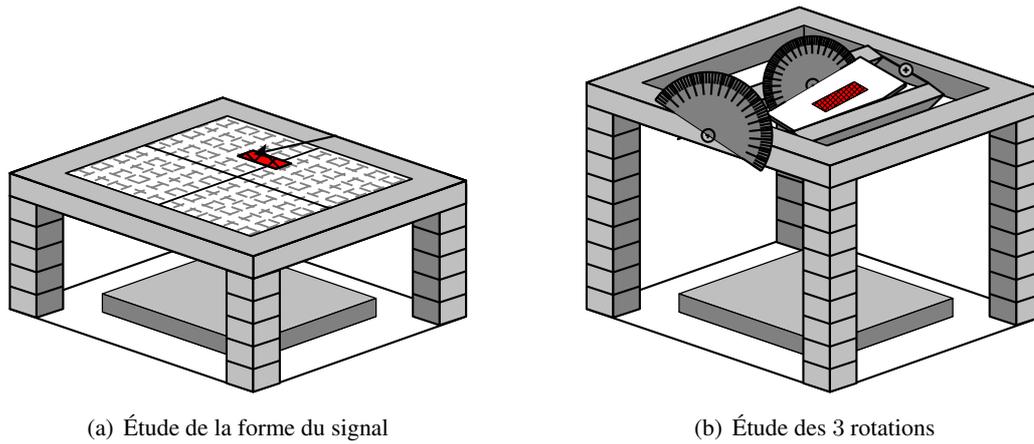


FIGURE 41: Montage permettant d'étudier le signal lecteur-tag

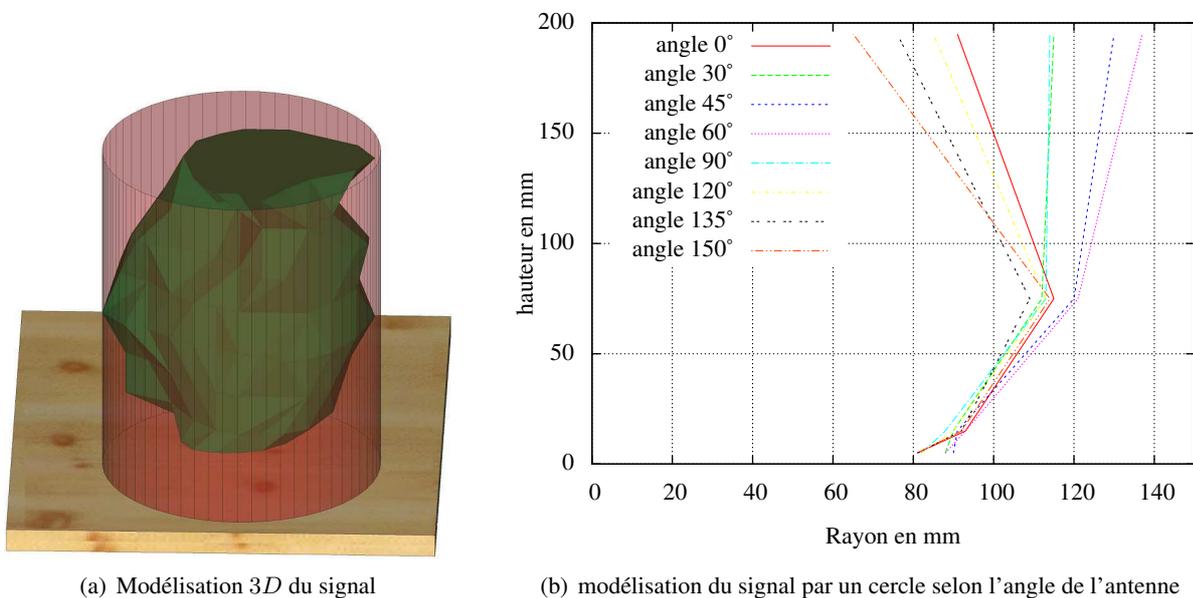


FIGURE 42: Étude de la zone de couverture lecteur-tag RFID

Une synthèse de l'ensemble des résultats (sphères obtenues) est donné dans la FIGURE 42(b). Nous constatons, d'une part, que le signal varie selon l'axe de l'antenne et, d'autre part, que plus l'émetteur est loin du tag, plus la dispersion du signal est grande. À contrario, il existe un diamètre commun du signal

à l'origine.

À partir de cette étude, il est possible de déterminer pour une technologie spécifique la dissémination optimale des tags dans la matière. À titre d'exemple, il est nécessaire de s'assurer que la zone de couverture du lecteur ne couvre jamais plus de 8 tags pour la technologie RFID Omron V720-D52P03 (contraintes protocolaires). La FIGURE 43(b) illustre une dissémination possible des tags dans la matière qui respecte cette exigence²². En effet, si nous dénombrons les tags présents dans chaque zone (3 zones représentées dans la FIGURE 43(b)), nous remarquons qu'il y a au maximum 8 tags présents. Dans notre prototype, plus de 1500 tags/m² ont pu être disséminés avec la technologie Hitachi et plus de 50 tags/m² via la technologie Omron.

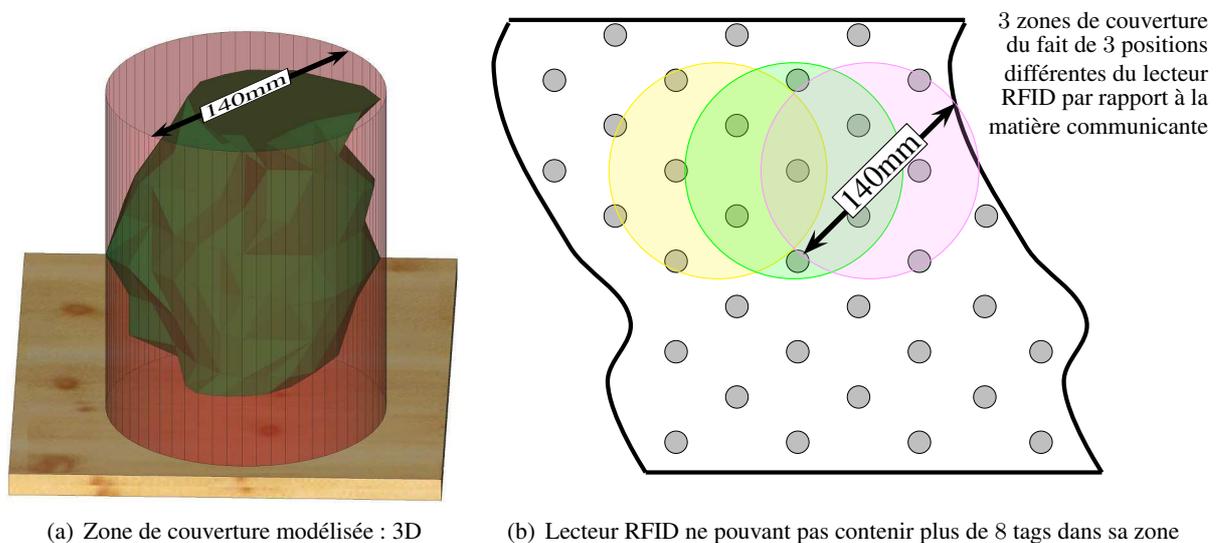
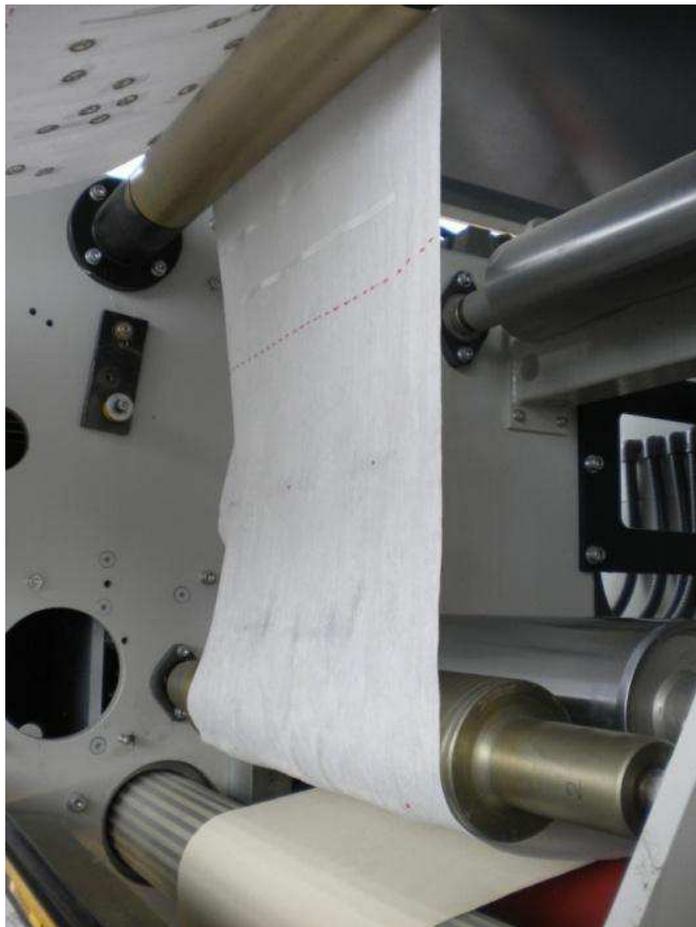
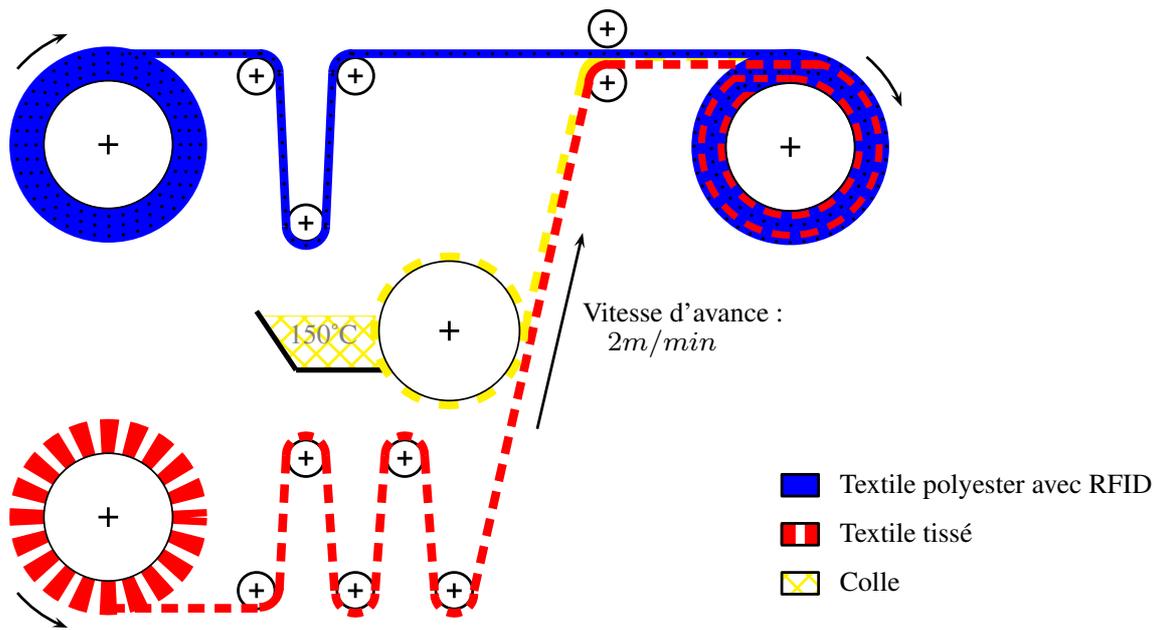


FIGURE 43: Modélisation de la zone de couverture de la technologie RFID

Dans un second temps, il est nécessaire d'étudier les différents procédés industriels permettant d'intégrer les puces RFID dans le textile suivant la dissémination souhaitée (fonction des caractéristiques des puces, des caractéristiques des procédés de fabrication, etc.). Après avoir étudié une multitude de procédés industriels avec le CETELOR : chauffage infrarouge, aiguilletage, hydroliage, etc., nous avons opté pour un procédé de contre-collage. La FIGURE 44 montre la machine lors de la conception du textile communicant. À travers ce procédé, les puces RFID ne subissent pas la température, ne subissent pas de chocs destructeurs, ni de pression de serrage. Des tests ont ensuite été menés, à la fois sur le fonctionnement du textile communicant (fonctionnement des tags RFID) et sur son usage²³. Ces tests ont montré que 100% des tags RFID fonctionnaient après conception du textile communicant et après les tests d'usage : lavage, abrasion, flexion-traction.

22. nous choisissons une sphère de 140mm car cela correspond à la dispersion maximale de la zone de couverture sur la FIGURE 42(b).

23. Le textile communicant est comparé à un textile classique sur la base de tests certifiés ISO.

FIGURE 44: Processus d'intégration des μ tags dans le tissu

Annexe VI : Localisation des tags RFID basée sur la méthode de la corde

Cette annexe détaille l'algorithme de calcul de la position des tags RFID dans la matière communicante. Cet algorithme se base d'une part sur les événements générés par la rampe de lecteurs RFID et d'autre part, sur le principe du calcul de la corde (Wikipedia, 2003) relative à la zone de lecture (modélisée comme un cylindre).

Soit \mathcal{P} l'ensemble de puces RFID présentes dans la matière communicante. Soit \mathcal{L} l'ensemble des lecteurs RFID qui composent la rampe illustrée dans la FIGURE 45 (avec $\mathcal{L} = \{l_1, l_2, l_3\}$) et soit r le rayon de la zone de couverture de la technologie RFID (modélisée dans l'Annexe IV). Chaque tag et chaque lecteur possèdent une référence notée respectivement ID_p et ID_l avec $p \in \mathcal{P}$ and $l \in \mathcal{L}$. Un lecteur RFID génère un événement à chaque fois qu'il détecte une puce dans sa zone de couverture (à chaque cycle d'acquisition). En d'autres termes, si une puce est présente durant n cycles d'acquisition dans la zone de couverture, n événements seront générés. Un événement se compose des informations suivantes : $\{ID_l, ID_p, t_{l,p}\}$, avec $t_{l,p}$ la date d'acquisition de l'événement. Basée sur l'ensemble des événements générés par les lecteurs, la position du tag p peut être calculée avec la différence de la date du premier événement et la date du dernier événement générés par un lecteur l (c.-à-d., la date d'entrée du tag p dans la zone de couverture du lecteur l et sa date de sortie), notées respectivement : $First(t_{l,p})$ et $Last(t_{l,p})$. La position du tag sur l'axe x (cf. FIGURE 45) est calculée via l'ÉQUATION 35. La date de détection de la matière, notée t_m est requise comme point de référence (détection réalisée via un capteur de présence comme illustré sur la FIGURE 45), ainsi que la vitesse du tapis roulant (notée v). Concernant la position du tag sur l'axe y , l'algorithme retourne deux possibilités du fait que la méthode de la corde ne permette pas de savoir si la position du tag p se situe sur la corde de droite ou de gauche de la zone de couverture. Ces deux cordes, notées $yd_{p,l}$ (corde de droite) et $yg_{p,l}$ (corde de gauche) sont calculées via l'ÉQUATION 36. La FIGURE 45 illustre l'exemple où les lecteurs l_1 et l_2 détectent respectivement le tag p_2 et calculent sa position en x et y . La position en x est approximativement équivalente pour chaque lecteur et 2 positions en y sont calculées pour chacun des lecteurs. Une solution permettant de déterminer si le tag se situe sur la corde de droite ou de gauche du lecteur consiste à analyser quels sont les lecteurs qui ont détecté le tag. Les cordes se situant entre ces lecteurs seront celles concernées par la position du tag sur l'axe y (sous condition que les lecteurs soient suffisamment espacés afin que leurs

zones de couverture ne se chevauchent pas de plus de la moitié). Dans l'exemple de la FIGURE 45, les lecteurs l_1 et l_2 ont détecté le tag p_2 , de ce fait, les cordes entre ces lecteurs (corde de droite pour l_1 et de gauche pour l_2 , notées respectivement yg_{p_2,l_1} et yg_{p_2,l_2} dans la FIGURE 45) seront celles concernées par la position du tag p_2 .

$$x_{p,l} = \left(\text{First}(t_{l,p}) - t_m + \frac{\text{Last}(t_{l,p}) - \text{First}(t_{l,p})}{2} \right) \times v \quad (35)$$

$$y_{d_{p,l}} = y_l + \sqrt{r^2 - \left[\left(\frac{\text{Last}(t_{l,p}) - \text{First}(t_{l,p})}{2} \right) \times v \right]^2} \quad (36)$$

$$y_{g_{p,l}} = y_l - \sqrt{r^2 - \left[\left(\frac{\text{Last}(t_{l,p}) - \text{First}(t_{l,p})}{2} \right) \times v \right]^2}$$

- $x_{p,l}$ coordonnée en x du tag p , calculée par rapport aux événements générés par le lecteur l ,
 $y_{d_{p,l}}$ coordonnée en y du tag p (corde de droite), calculée par rapport aux événements générés par le lecteur l ,
 $y_{g_{p,l}}$ coordonnée en y du tag p (corde de gauche), calculée par rapport aux événements générés par le lecteur l ,
 $\text{First}(t_{l,p})$ date de la première détection du tag p par le lecteur l ,
 $\text{Last}(t_{l,p})$ date de la dernière détection du tag p par le lecteur l ,
 v vitesse de défilement de la matière sous la rampe de lecteurs (c.-à-d., vitesse du tapis roulant),
 y_l coordonnée en y du lecteur l (position sur la rampe),
 t_m date de détection de la matière communicante (via un capteur de présence),
 r rayon de la zone de couverture de la technologie mise en œuvre.

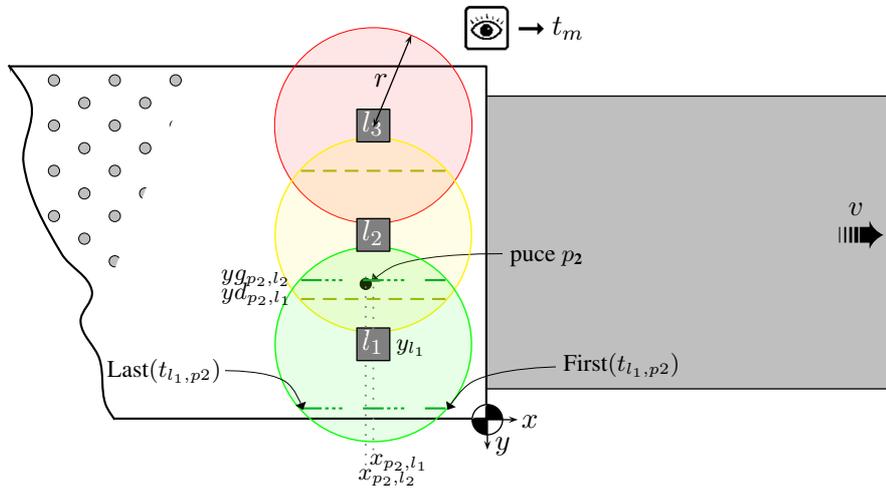


FIGURE 45: Architecture permettant de localiser des tags RFID dans la matière communicante

L'erreur commise par l'algorithme est fonction de la précision :

- de la vitesse de défilement de la matière (c.-à-d. la précision de la vitesse du tapis roulant),
- de la date de détection de la matière communicante,
- du cycle d'acquisition du lecteur RFID (plus le cycle est rapide, moins l'erreur sera grande),
- de la modélisation de la zone de couverture (modélisée comme un cylindre parfait).

Annexe VII : Comparaisons supplémentaires des approches de Chan-Roddick et AHP-Floue

Des études complémentaires au cas d'étude du Chapitre 5 sont présentées dans cette annexe.

Tout d'abord, dans la section .6, nous comparons les deux approches (Chan-Roddick et AHP-Floue) lorsqu'un grand nombre d'experts (p. ex. 10 experts) réalisent une évaluation et qu'une infime majorité d'experts (p. ex. 1 expert sur 10) exprime un souhait opposé aux autres. En effet, dans le cas d'étude du Chapitre 5, nous considérons au maximum 3 experts. L'agrégation des avis proposée pour l'approche de Chan et Roddick correspond à la moyenne des avis, qui est appropriée pour 3 experts. Mais quand est-il lorsque le nombre d'experts devient conséquent ?

Ensuite, dans la section .7, nous comparons les approches sur la base d'un avis unique. Ceci permet de considérer l'approche initiale de Chan et Roddick (c.-à-d., sans ajout de l'étape d'agrégation comme illustré dans la FIGURE 5.2(a)). Les deux représentations adoptées dans le chapitre 5 pour comparer ListeAF et ListeCR sont reprises dans ces scénarios.

.6 Comparaison des approches avec un grand nombre d'avis/experts

Scénario 1

Nous montrons dans cette section la limite de la méthode d'agrégation proposée pour Chan-Roddick (calcul de la moyenne des avis) par rapport à celle proposée dans AHP-Floue (utilisation d'intervalles classiques), lorsque 10 experts énumèrent des attributs²⁴. Les ajustements effectués dans ce scénario sont les suivants :

- *Enumeration* : **10 experts** énumèrent les attributs tels que présentés dans la TABLE 8. Pour chaque attribut, les 9 premiers experts expriment le même souhait (ne désirent pas l'attribut) et le dernier expert *D10* exprime le souhait opposé (désire l'attribut). Cette étude a pour but de montrer si les approches satisfont ou non l'ensemble des experts/utilisateurs,
- *Contextual* : un seul expert définit l'importance des groupes d'entités (cf. ÉQUATION 37). À noter

24. Les critères *Enumeration* et *Contextual* sont plus sujets à être évalués par plusieurs experts que les deux autres critères. Par exemple, de nombreux utilisateurs peuvent vouloir stocker des informations variées en fonction de leur tâche/cœur métier.

que les groupes d'entités définis dans le chapitre 5 sont conservés,

- *Model-Based* : un seul expert fixe le coefficient k relatif à ce critère (cf. ÉQUATION 38),
- *Data Size* : un seul expert fixe le coefficient k relatif ce critère (cf. ÉQUATION 39).

TABLE 8: Attributs énumérés dans le cas 2

	Nom Table	Nom Attribut	Décideur	Choix	Score	
					AHP-Floue	Chan
G ₁	MaterialLot	Description	D1	0		$\frac{1}{10} = 0.1$
			D2	0		
			D3	0		
			D4	0		
			D5	0		
			D6	0		
			D7	0		
			D8	0		
			D9	0		
			D10	1		
	MaterialClass	IDMaterialClass	D1	0		$\frac{1}{10} = 0.1$
			D2	0		
⋮			⋮			
D9			0			
MaterialClass	Description	D1	0		$\frac{1}{10} = 0.1$	
		D2	0			
		⋮	⋮			
		D9	0			
MaterialClass	Value	D1	1		$\frac{1}{10} = 0.1$	
		D2	0			
		⋮	⋮			
		D9	0			
MaterialClass	Value Unit	D1	1		$\frac{1}{10} = 0.1$	
		D2	0			
		⋮	⋮			
		D9	0			
			D10	1		

$$D_G^1 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & 1 & 5 & \frac{1}{5} \\ 1 & 1 & 3 & 3 \\ \frac{1}{5} & \frac{1}{3} & 1 & \frac{1}{3} \\ 5 & \frac{1}{3} & \frac{1}{3} & 1 \end{bmatrix} & & & \end{matrix} \quad (37)$$

$$k_m = 1.5 \quad (38) \quad k_s = 2.3 \quad (39)$$

Nous considérons dans un premier temps que le critère *Enumeration* est d'extrême importance (9 sur l'échelle S de Saaty) par rapport aux autres critères (cf. ÉQUATION 40). Puisque nous nous basons sur la moyenne des avis dans Chan-Roddick et qu'un seul expert sur 10 énumère les attributs de la TABLE 8 (le preneur de décision D_{10}), la moyenne est par conséquent égale à 0.1 (cf. TABLE 8). Cette valeur s'approche de zéro et rend l'énumération des attributs quelque peu « négligeable ». À l'inverse, l'approche AHP-Floue travaille avec un intervalle de valeurs $[0; 1]$ afin de tenir compte des avis « isolés » comme celui de D_{10} . Les résultats affichés dans la FIGURE 46(a) reflètent ces propos puisque les data items appartenant aux attributs énumérés par D_{10} se situent dans les 10 premiers rangs de ListeAF et entre les rangs 10 et 30 dans ListeCR. La FIGURE 46(a) montre donc que si on embarque moins de 35 data items sur le produit, nous avons moins de 80% de data items communs entre ListeCR et ListeAF. Nous pouvons néanmoins noter que les data items issus d'attributs énumérés sont globalement en début de liste dans ListeCR car le critère *Enumeration* est d'importance extrême par rapport aux autres.

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & 1 & 9 & 9 & 9 \\ C_c & \frac{1}{9} & 1 & 1 & 1 \\ C_m & \frac{1}{9} & 1 & 1 & 1 \\ C_s & \frac{1}{9} & 1 & 1 & 1 \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & 1 & 9 & 9 \\ C_2 & \frac{1}{9} & 1 & 1 \\ C_3 & \frac{1}{9} & 1 & 1 \end{matrix} \quad (40)$$

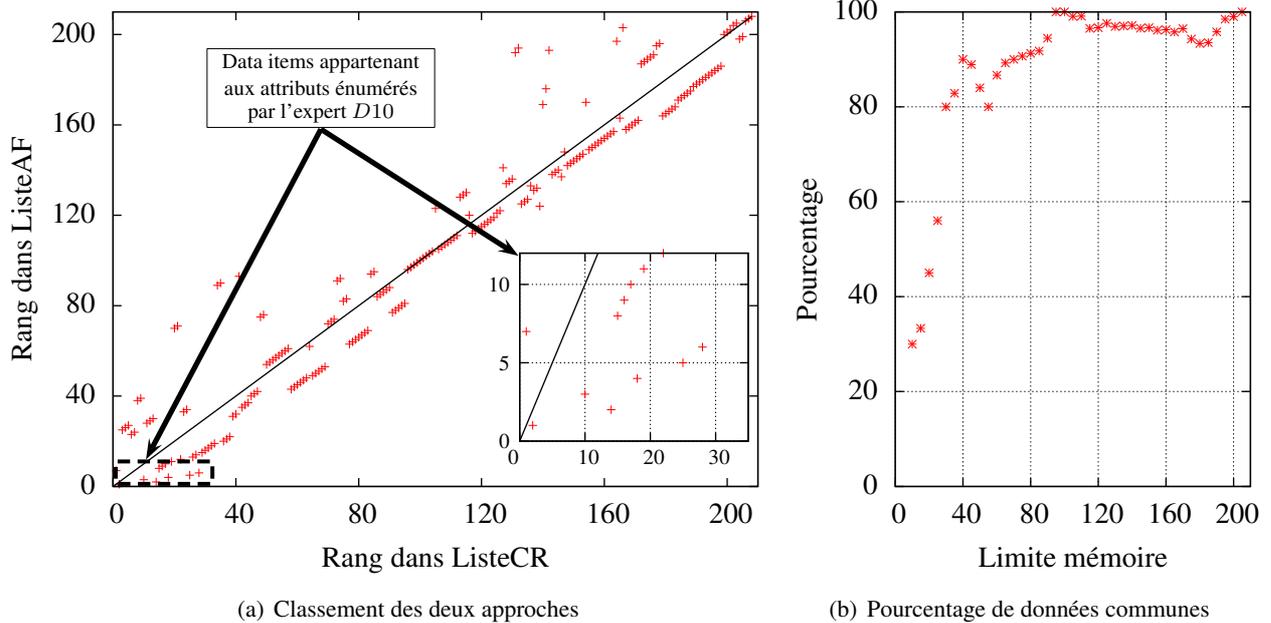


FIGURE 46: Résultats lorsque le critère *Enumeration* est extrêmement favorisé par rapport aux autres

Lorsque le critère *Enumeration* n'est plus d'importance extrême par rapport aux autres (p. ex. dans l'ÉQUATION 41, il s'agit du critère *Contextual*), les data items provenant d'attributs énumérés ne se situent plus entre les rangs 10 et 30 dans ListeCR, mais entre les rangs 20 et 120 comme montré dans la FIGURE 47(a). À l'inverse, dans ListeAF, ces mêmes data items se situent toujours dans les 15 premiers rangs, ce qui montre que l'avis isolé de D_{10} est toujours pris en compte malgré le peu d'importance

attribué au critère *Enumeration*. Le pourcentage de data items communs entre ListeCR et ListeAF en début de liste est désormais plus faible que précédemment puisque les écarts ont augmentés (p. ex. < 60% de data items communs si l'on embarque moins de 40 data items).

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & \frac{1}{9} & 1 & 1 \end{bmatrix} \\ C_c & \begin{bmatrix} 9 & 1 & 9 & 9 \end{bmatrix} \\ C_m & \begin{bmatrix} 1 & \frac{1}{9} & 1 & 1 \end{bmatrix} \\ C_s & \begin{bmatrix} 1 & \frac{1}{9} & 1 & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & \frac{1}{9} & 1 \end{bmatrix} \\ C_2 & \begin{bmatrix} 9 & 1 & 9 \end{bmatrix} \\ C_3 & \begin{bmatrix} 1 & \frac{1}{9} & 1 \end{bmatrix} \end{matrix} \quad (41)$$

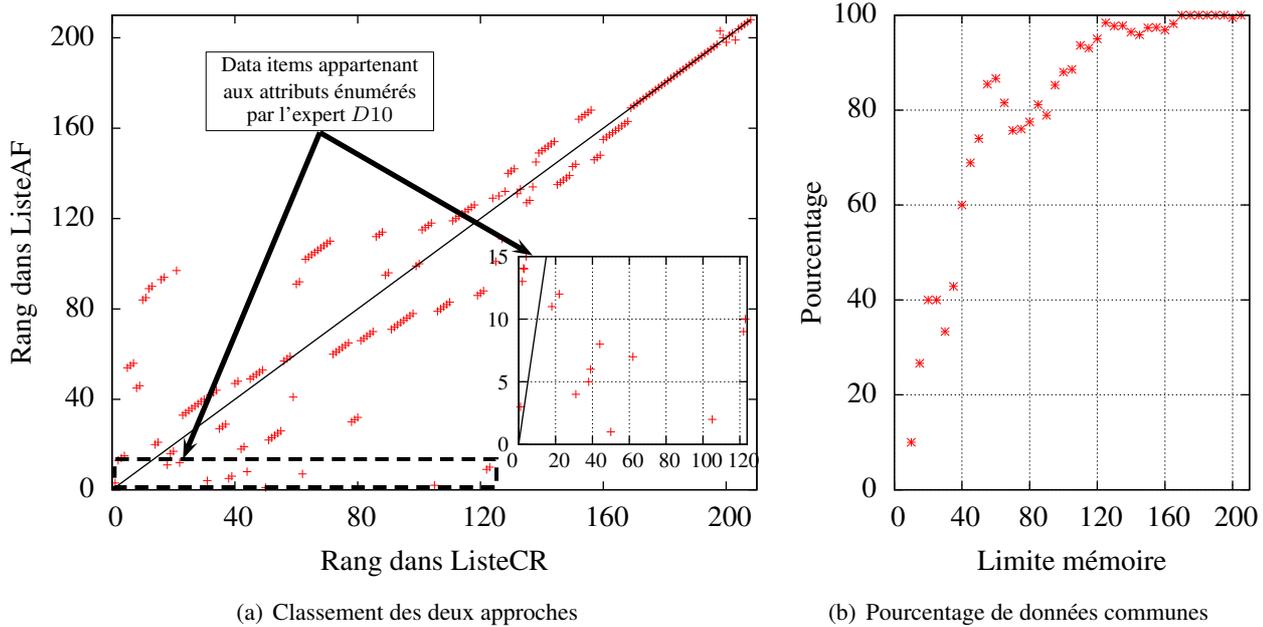


FIGURE 47: Résultats lorsque le critère *Contextual* est extrêmement favorisé par rapport aux autres

Nous pouvons conclure que lorsque nous sommes confrontés à plusieurs experts et à des avis « isolés », l'approche d'agrégation proposée pour Chan-Roddick ne permet pas de tenir compte²⁵ de ces avis, contrairement à l'approche AHP-Floue.

7 Comparaison des approches avec un unique avis/expert

Cette section détaille les comparaisons des approches de Chan et Roddick et AHP-Floue, sur la base d'un avis unique comme illustré dans la FIGURE 48. Deux scénarios distincts sont proposés, ils diffèrent sur le critère *Enumeration*. Dans le scénario 1, une grande quantité d'attributs est énumérée (une quarantaine), contre six dans le scénario 2. Pour chaque scénario, les ajustements concernant les critères *Enumeration*, *Contextual*, *Model-Based* et *Data Size* n'évoluent pas, uniquement les importances entre critères évoluent afin d'étudier le comportement des approches, à savoir :

- quels critères les approches favorisent-elles ?
- quel critère tend à rendre les listes des deux approches similaires et inversement ?

25. propos modéré en fonction de l'importance attribué au critère.

- y a-t-il une limite de rang récurrente qui garantit que quelle que soit l'approche/la liste sélectionnée (ListeCR ou ListeAF) et quelle que soit l'importance du critère, on a $X\%$ de data items similaires ?

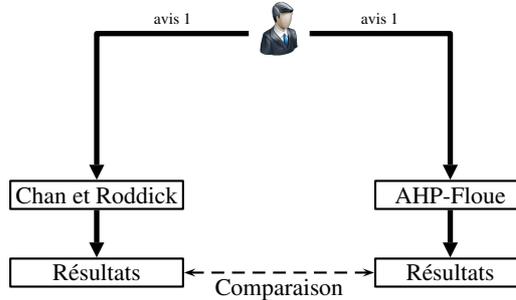


FIGURE 48: Comparaison sur la base d'un avis unique

Scénario 2

Les ajustements effectués dans ce scénario sont les suivants :

- *Enumeration* : les attributs énumérés par l'expert sont résumés dans la TABLE 9.
- *Contextual* : la matrice fournie par l'expert concernant la définition de l'importance des groupes d'entités est donnée dans l'ÉQUATION 42,
- *Model-Based* : le coefficient k relatif à ce critère est donné dans l'ÉQUATION 43,
- *Data Size* : le coefficient k relatif à ce critère est donné dans l'ÉQUATION 44.

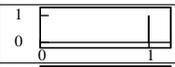
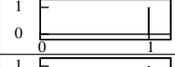
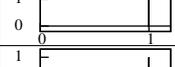
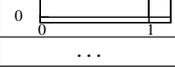
$$D_G^1 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & \frac{1}{3} & 3 & 3 \\ 3 & 1 & 5 & 5 \\ \frac{1}{3} & \frac{1}{5} & 1 & 3 \\ \frac{1}{3} & \frac{1}{5} & \frac{1}{3} & 1 \end{bmatrix} & & & \end{matrix} \quad (42)$$

$$k_m = 1.5 \quad (43)$$

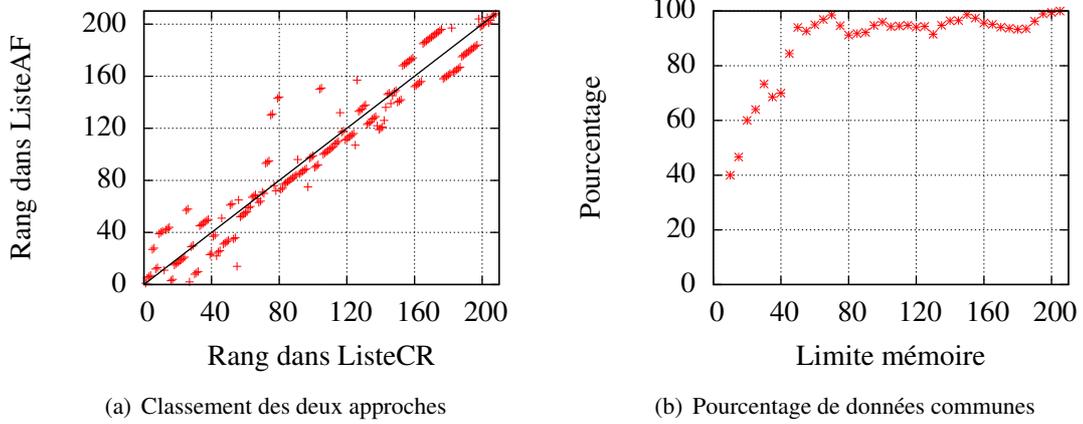
$$k_s = 1.02 \quad (44)$$

Les ÉQUATIONS 45 à 49 présentent les réglages concernant l'importance des critères. Nous favorisons respectivement de manière extrême le critère *Enumeration*, *Contextual*, *Model-Based*, *Data Size* et enfin, nous considérons que chaque critère est d'importance égale. Les résultats issus de ces réglages sont respectivement affichés dans les FIGURE 49 à 53 via les deux représentations adoptées.

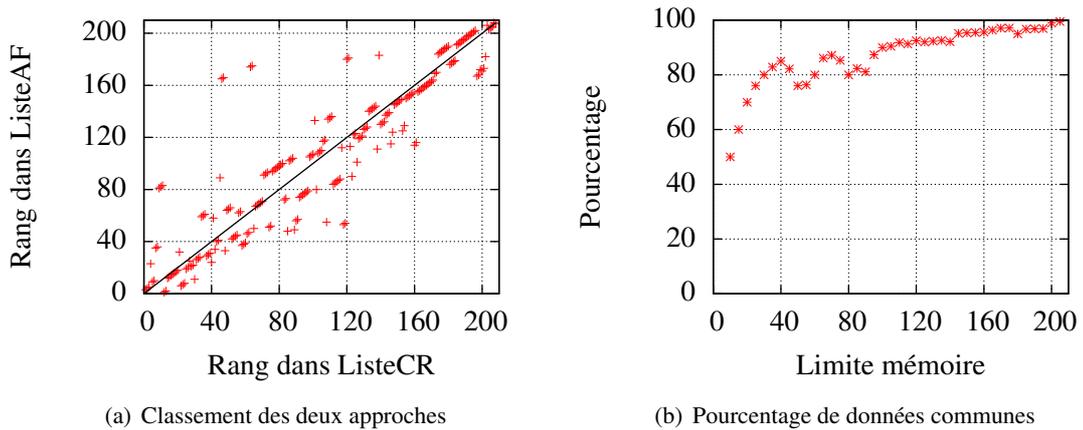
TABLE 9: Attributs énumérés dans le cas 1

	Nom Table	Nom Attribut	Décideur	Choix	Score	
					AHP-Floue	Chan
G ₁	MaterialLot	IDMaterialLot	<i>D1</i>	1		1
		Quantity	<i>D1</i>	1		1
	MaterialDefinition	IDMaterialDefinition	<i>D1</i>	1		1
		Value	<i>D1</i>	1		1
	ManufacturingBill	IDManufacturingBill	<i>D1</i>	1	...	1
		Description	<i>D1</i>	1	...	1
		IDProductDefinition	<i>D1</i>	1	...	1
		IDMaterialDefinition	<i>D1</i>	1	...	1
		Quantity	<i>D1</i>	1	...	1
		Quantity Unit	<i>D1</i>	1	...	1
G ₂	PersonSegmentSpecificat°	IDPersonSegmentSpecificat°	<i>D1</i>	1	...	1
		Description	<i>D1</i>	1	...	1
		Quantity	<i>D1</i>	1	...	1
	Person	IDPerson	<i>D1</i>	1	...	1
		Description	<i>D1</i>	1	...	1
		IDPersonClass	<i>D1</i>	1	...	1
	ActualPersonSegment	IDActualPersonSegment	<i>D1</i>	1	...	1
		IDPerson	<i>D1</i>	1	...	1
		IDSegmentResponse	<i>D1</i>	1	...	1
		Quantity	<i>D1</i>	1	...	1
G ₃	ProductionOrder	IDProductionOrder	<i>D1</i>	1	...	1
		StartTime	<i>D1</i>	1	...	1
		EndTime	<i>D1</i>	1	...	1
	ProductSegment	IDProductSegment	<i>D1</i>	1	...	1
		Duration	<i>D1</i>	1	...	1
		Duration Unit	<i>D1</i>	1	...	1
	SegmentResponse	Description	<i>D1</i>	1	...	1
		IDProductionOrder	<i>D1</i>	1	...	1
	ProductDefinition	IDProductDefinition	<i>D1</i>	1	...	1
		Description	<i>D1</i>	1	...	1
Published Date		<i>D1</i>	1	...	1	
G ₄	EquipmentClass	IEquipmentClass	<i>D1</i>	1	...	1
		Description	<i>D1</i>	1	...	1
		Value	<i>D1</i>	1	...	1
		Value Unit	<i>D1</i>	1	...	1

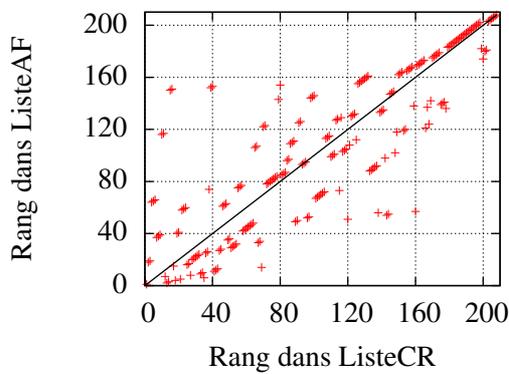
$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 9 & 9 & 9 \\ \frac{1}{9} & 1 & 1 & 1 \\ \frac{1}{9} & 1 & 1 & 1 \\ \frac{1}{9} & 1 & 1 & 1 \end{bmatrix} \\ C_c & \\ C_m & \\ C_s & \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 9 & 9 \\ \frac{1}{9} & 1 & 1 \\ \frac{1}{9} & 1 & 1 \end{bmatrix} \\ C_2 & \\ C_3 & \end{matrix} \quad (45)$$

FIGURE 49: Résultats lorsque le critère *Enumeration* est extrêmement favorisé par rapport aux autres

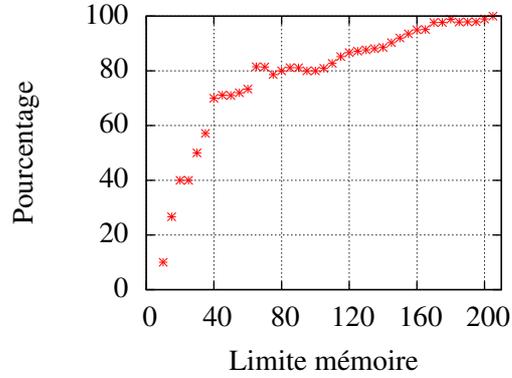
$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & \frac{1}{9} & 1 & 1 \\ 9 & 1 & 9 & 9 \\ 1 & \frac{1}{9} & 1 & 1 \\ 1 & \frac{1}{9} & 1 & 1 \end{bmatrix} \\ C_c & \\ C_m & \\ C_s & \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & \frac{1}{9} & 1 \\ 9 & 9 & 9 \\ 1 & \frac{1}{9} & 1 \end{bmatrix} \\ C_2 & \\ C_3 & \end{matrix} \quad (46)$$

FIGURE 50: Résultats lorsque le critère *Contextual* est extrêmement favorisé par rapport aux autres

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 1 & \frac{1}{9} & 1 \end{bmatrix} \\ C_c & \begin{bmatrix} 1 & 1 & \frac{1}{9} & 1 \end{bmatrix} \\ C_m & \begin{bmatrix} 9 & 9 & 1 & 9 \end{bmatrix} \\ C_s & \begin{bmatrix} 1 & 1 & \frac{1}{9} & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_2 & \begin{bmatrix} 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_3 & \begin{bmatrix} 9 & 9 & 1 \end{bmatrix} \end{matrix} \quad (47)$$



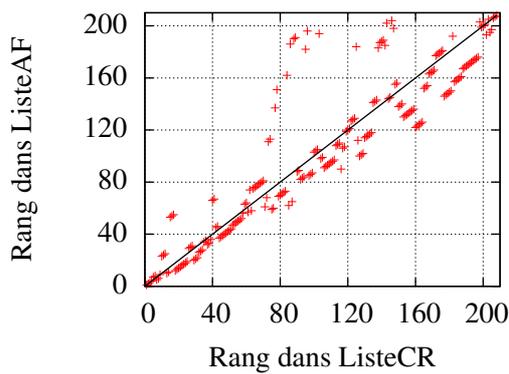
(a) Classement des deux approches



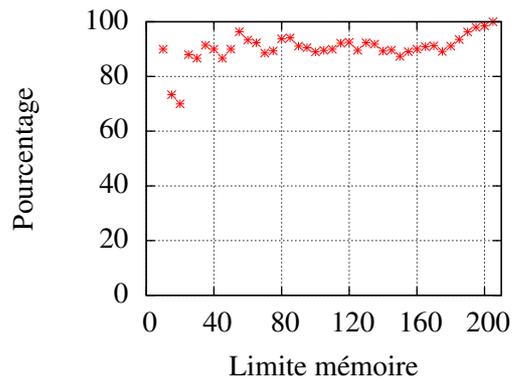
(b) Pourcentage de données communes

FIGURE 51: Résultats lorsque le critère *Model-Based* est extrêmement favorisé par rapport aux autres

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_c & \begin{bmatrix} 1 & 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_m & \begin{bmatrix} 1 & 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_s & \begin{bmatrix} 9 & 9 & 9 & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_2 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_3 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \end{matrix} \quad (48)$$



(a) Classement des deux approches



(b) Pourcentage de données communes

FIGURE 52: Résultats lorsque le critère *Data Size* est extrêmement favorisé par rapport aux autres

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \\ C_c & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \\ C_m & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \\ C_s & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_2 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_3 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \end{matrix} \quad (49)$$

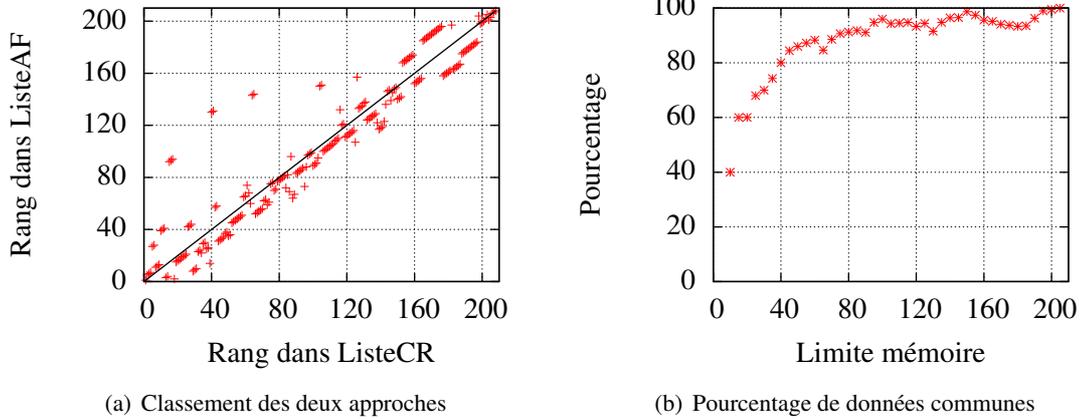


FIGURE 53: Résultats lorsque les critères sont d'importance égale

Nous remarquons que les deux approches fournissent un classement similaires lorsque les critères *Enumeration* et *Contextual* sont d'importances extrêmes (cf. FIGURE 49 et 50) et lorsque les critères sont d'importances égales (cf. FIGURE 53). En effet, les points sur les FIGURE 49(a), 50(a) et 53(a) sont majoritairement concentrés autour de la ligne centrale. Ceci est moins vrai dans la FIGURE 50(a) (c.-à-d. lorsque le critère *Data Size* est extrêmement favorisé) et plus particulièrement pour les data items de fin de liste, bien que nous ayons quasiment tout le temps plus de 80% de data items identiques entre les deux approches (cf. FIGURE 50(b)). Les points éloignés de la ligne centrale dans la FIGURE 50(a) correspondent à des data items de grandes tailles qui sont :

- fortement pénalisés dans l'approche AHP-Floue du fait que l'utilisateur émet une importance extrême sur ce critère,
- moyennement pénalisés dans l'approche de Chan et Roddick car leur modèle ne tient pas compte de cette préférence et repose sur le comportement de la fonction logarithme népérien.

C'est pourquoi les points sont éloignés de la ligne centrale vers le haut et non vers le bas. Nous remarquons que lorsque les critères *Model-Based* est favorisé, la dispersion des points dans la FIGURE 51(a) est importante, en dessous et au dessus de la ligne centrale, sur toute la plage du classement (du rang 0 à 208). Il n'est cependant pas facile d'expliquer cette dispersion car contrairement au critère *Data Size*, la préférence de l'importance du critère *Model-Based* est prise dans les deux approches.

Finalement, il est important de remarquer au vu des FIGURES 49(b) à 53(b), que si nous stockons plus de 60 data items sur le produit²⁶, que ce soit via l'approche de Chan-Roddick ou AHP-Floue, nous avons au minimum 70% des data items qui sont communs. En d'autres termes, quels que soient les réglages relatifs à l'importance des critères, on est certain d'avoir 70% des data items communs.

26. les 60 premiers data items de la liste.

Scénario 3

Les ajustements effectués dans ce scénario sont les suivants :

- *Enumeration* : les attributs énumérés par l'expert sont résumés dans la TABLE 10.
- *Contextual* : la matrice fournie par l'expert concernant la définition de l'importance des groupes d'entités est donnée dans l'ÉQUATION 50,
- *Model-Based* : le coefficient k relatif à ce critère est donné dans l'ÉQUATION 51,
- *Data Size* : le coefficient k relatif ce critère est donné dans l'ÉQUATION 52

TABLE 10: Attributs énumérés dans le cas 2

	Nom Table	Nom Attribut	Décideur	Choix	Score	
					AHP-Floue	Chan
G ₁	MaterialLot	Description	D1	1	$\frac{1}{0}$	1
		Status	D1	1	$\frac{1}{0}$	1
	MaterialClass	IDMaterialClass	D1	1	$\frac{1}{0}$	1
		Description	D1	1	$\frac{1}{0}$	1
		Value	D1	1	$\frac{1}{0}$	1
		Value Unit	D1	1	$\frac{1}{0}$	1

$$D_G^1 = \begin{matrix} & G_1 & G_2 & G_3 & G_4 \\ \begin{matrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{matrix} & \begin{bmatrix} 1 & 1 & 5 & \frac{1}{5} \\ 1 & 1 & 3 & 3 \\ \frac{1}{5} & \frac{1}{3} & 1 & \frac{1}{3} \\ 5 & \frac{1}{3} & \frac{1}{3} & 1 \end{bmatrix} & & & \end{matrix} \quad (50)$$

$$k_m = 1.5 \quad (51)$$

$$k_s = 2.3 \quad (52)$$

Les ÉQUATIONS 53 à 57 présentent les réglages concernant l'importance des critères. Tout comme précédemment, nous favorisons respectivement de manière extrême le critère *Enumeration*, *Contextual*, *Model-Based*, *Data Size* et enfin, nous considérons que chaque critère est d'importance égale. Les résultats sont respectivement affichés dans les FIGURES 49 à 53.

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 9 & 9 & 9 \\ \frac{1}{9} & 1 & 1 & 1 \\ \frac{1}{9} & 1 & 1 & 1 \\ \frac{1}{9} & 1 & 1 & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 9 & 9 \\ \frac{1}{9} & 1 & 1 \\ \frac{1}{9} & 1 & 1 \end{bmatrix} \end{matrix} \quad (53)$$

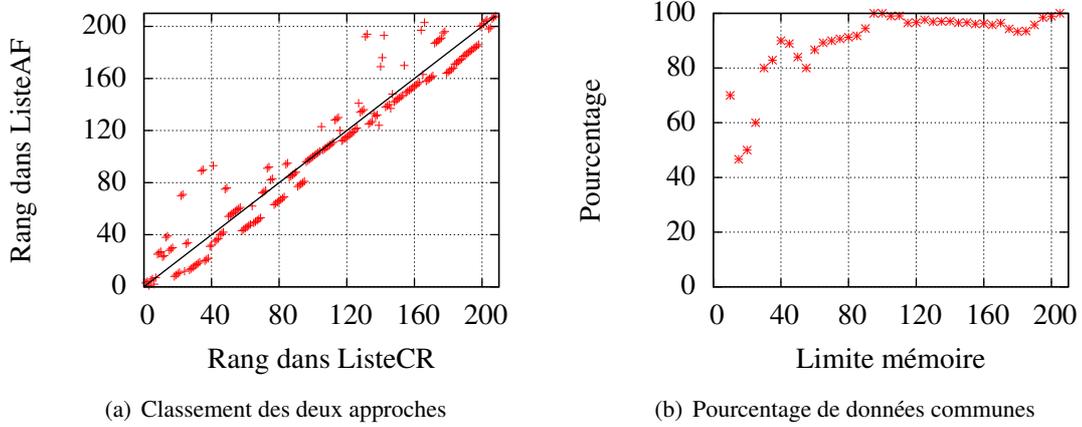


FIGURE 54: Résultats lorsque le critère *Enumeration* est extrêmement favorisé par rapport aux autres

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & \frac{1}{9} & 1 & 1 \\ 9 & 1 & 9 & 9 \\ 1 & \frac{1}{9} & 1 & 1 \\ 1 & \frac{1}{9} & 1 & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & \frac{1}{9} & 1 \\ 9 & 9 & 9 \\ 1 & \frac{1}{9} & 1 \end{bmatrix} \end{matrix} \quad (54)$$

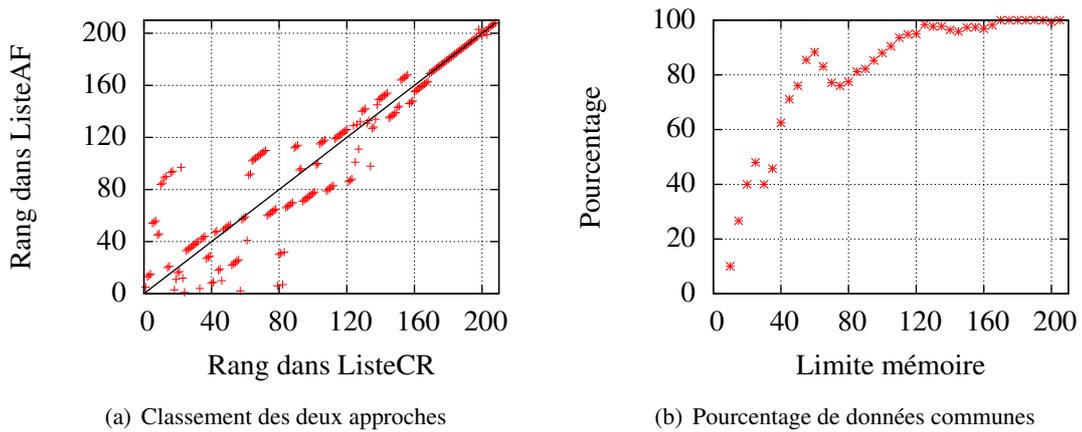
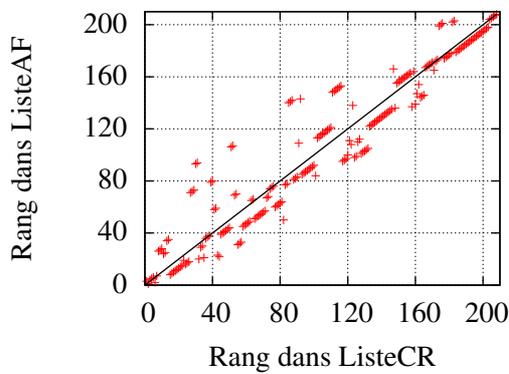
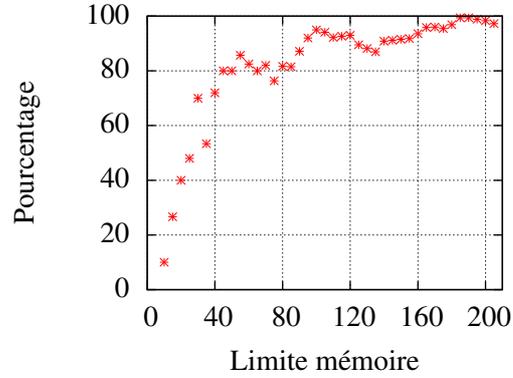


FIGURE 55: Résultats lorsque le critère *Contextual* est extrêmement favorisé par rapport aux autres

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 1 & \frac{1}{9} & 1 \end{bmatrix} \\ C_c & \begin{bmatrix} 1 & 1 & \frac{1}{9} & 1 \end{bmatrix} \\ C_m & \begin{bmatrix} 9 & 9 & 1 & 9 \end{bmatrix} \\ C_s & \begin{bmatrix} 1 & 1 & \frac{1}{9} & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_2 & \begin{bmatrix} 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_3 & \begin{bmatrix} 9 & 9 & 1 \end{bmatrix} \end{matrix} \quad (55)$$



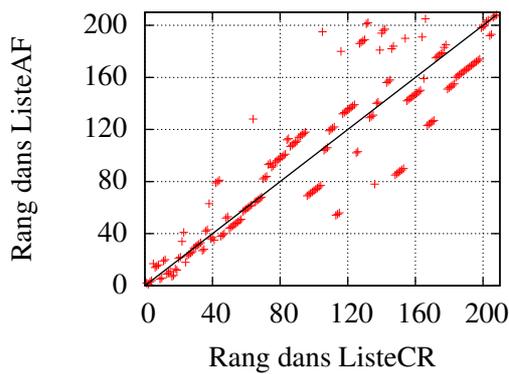
(a) Classement des deux approches



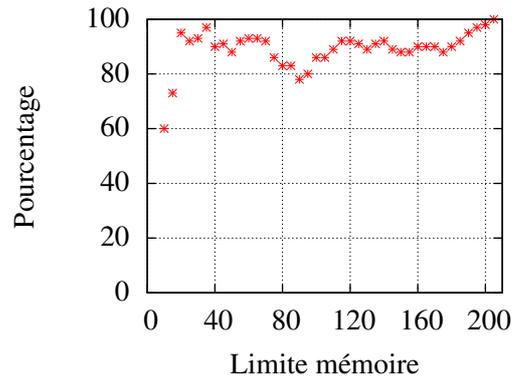
(b) Pourcentage de données communes

FIGURE 56: Résultats lorsque le critère *Model-Based* est extrêmement favorisé par rapport aux autres

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_c & \begin{bmatrix} 1 & 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_m & \begin{bmatrix} 1 & 1 & 1 & \frac{1}{9} \end{bmatrix} \\ C_s & \begin{bmatrix} 9 & 9 & 9 & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_2 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_3 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \end{matrix} \quad (56)$$



(a) Classement des deux approches



(b) Pourcentage de données communes

FIGURE 57: Résultats lorsque le critère *Data Size* est extrêmement favorisé par rapport aux autres

$$D_{\rho}^1(\text{AF}) = \begin{matrix} & C_e & C_c & C_m & C_s \\ C_e & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \\ C_c & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \\ C_m & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \\ C_s & \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \end{matrix} \quad D_{\rho}^1(\text{CR}) = \begin{matrix} & C_1 & C_2 & C_3 \\ C_1 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_2 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ C_3 & \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \end{matrix} \quad (57)$$

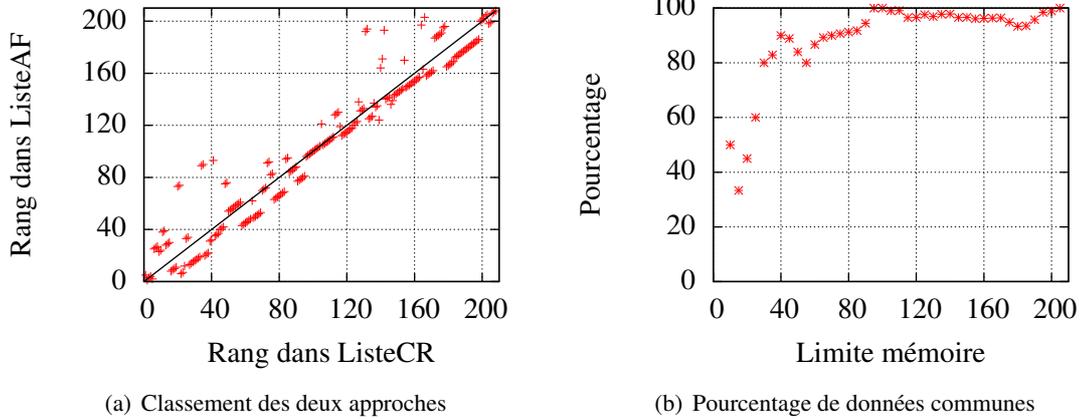


FIGURE 58: Résultats lorsque les critères sont d'importance égale

Cette fois-ci, nous remarquons que les dispersions relatives au classement de ListeCR et ListeAF dans les FIGURES 54(a) à 58(a) sont similaires à celles du scénario n°1, mais moins prononcées (particulièrement pour le critère *Model-Based*, cf. FIGURE 51(a) et FIGURE 56(a)). Cela montre que le fait d'énumérer une grande quantité d'attributs n'influe pas sur les résultats, excepté pour les data items qui appartiennent aux attributs énumérés.

Une nouvelle fois, nous remarquons sur les FIGURES 54(b) à 58(b), que si nous stockons plus de 60 data items sur le produit, que ce soit via l'approche de Chan-Roddick ou AHP-Floue, nous avons au minimum 70% des data items qui sont communs.

Annexe VIII : Taux de contribution au projet

Le calcul du taux de contribution au projet est présenté dans cette annexe. La FIGURE 59 reprend à l'identique la FIGURE 26 de l'Annexe I, sur laquelle nous faisons apparaître en gras/rouge les problèmes auxquels nous avons répondu dans cette thèse. En quelques mots :

- type d'accès aux tags : pris en compte dans la conception du prototype de textile communicant afin de déterminer la dissémination maximale des tags RFID dans le textile (voir section 4.1),
- mémoire des tags : prise en compte lorsque les data items doivent être explosés/disséminés entre plusieurs tags RFID (*cf.* en-tête protocolaire proposé dans la section 4.2),
- environnement conditionné : plusieurs infrastructures matérielles et logicielles sont proposées pour lire/écrire une matière communicante (voir section 4.2 et FIGURE 4.6),
- intégration des tags : étudiée lors de la conception du textile communicant,
- application découpe/reconstruction des informations : possible grâce à l'en-tête protocolaire,
- encapsulation des données : liée à l'en-tête protocolaire,
- densité des tags : étudiée lors de la conception du textile communicant,
- mémoire de la matière : étudiée lors de la conception du textile communicant,
- accès aux informations utiles : géré par l'étape 2 du processus de dissémination (voir Chapitre 3),
- informations à conserver/modifier : géré par l'étape 2 du processus puisqu'à chaque fois que l'utilisateur définit un point d'écriture du produit, les informations devant être stockées sur le produit sont évaluées (voir ré-évaluées si elles ont été réinjectées dans les bases de données). Il sera toutefois nécessaire dans de futurs travaux de gérer la conservation des informations déjà présentes sur le produit (non traité dans cette thèse),
- classification des informations : gérée par l'étape 2 du processus de dissémination,
- mises à jour des informations : le Chapitre 2 réalise un état de l'art complet de la distribution des informations dans des environnements fixes et mobiles. Les protocoles de mise à jour existants sont également décrits,
- code correcteur d'erreur bit : géré par la technologie RFID et par l'en-tête protocolaire,
- ordre des tags à lire : notre approche permet d'écrire et relire les tags RFID dans un ordre quelconque,

- ordre des tags à écrire : notre approche permet d'écrire/relire les tags RFID dans un ordre quelconque.

Il est finalement possible de calculer le taux contribution au projet à l'aide de la matrice QFD de second niveau donnée dans la FIGURE 59. Notons C_2 l'ensemble des problèmes (c.-à-d. les colonnes) et $C_{2\text{rép}}$ l'ensemble des problèmes auxquels nous avons répondu dans cette thèse (c.-à-d. les colonnes mises en avant en gras/rouge). Le taux de contribution final est alors calculé dans l'ÉQUATION 58 et indique que nous avons répondu à 49% de l'ensemble des problèmes identifié dans notre étude²⁷. Bien évidemment, ce pourcentage est « subjectif » car certaines problématiques n'ont pas été traitées entièrement, d'autres n'ont pas été identifiées au début du projet (non prises en compte dans le calcul), etc.

$$t_c = \frac{\sum_{c \in C_{2\text{rép}}} \text{Inf}_c}{\sum_{c \in C_2} \text{Inf}_c} \tag{58}$$

$$= \frac{0.3 + 0.4 + 0.4 + 0.3 + 0.2 + 0.4 + 0.6 + 0.8 + 0.1 + 0.4 + 0.1 + 0.4 + 0.2 + 0.2}{0.3 + 0.4 + 0.4 + 0.3 + 0.2 + 0.4 + 0.4 + 0.4 + 0.4 + 0.3 + \dots + 0.6 + 0.2 + 0.2 + 0.3 + 0.2}$$

$$= 0.49$$

		Problèmes																														
		Surface couverte	Débits offerts	Plage de fréquence	Type d'accès aux tags	Système d'alimentation	Mémoire des tags	Types d'isolants	Environnement conditionné	Norme existante	Intégration des tags	Appli découpe/recons de l'info	Encapsulation des données	Densité des tags	Mémoire de la matière	Adaptabilité aux terminaux	Accès aux infos utiles	Info à conserver/modifier	Reconstruction/ désagrégation	Modèle d'information associé	Classification des infos	Principe d'agrégation de données	Mises à jour des informations	Code correcteur d'erreur bit	Redondance inter-tag	Zones accessible en lect/écri	Ordre des tags à lire	Ordre des tags à écrire	Type de clé de cryptage	Clé d'authentification		
Spécifications $C_1 \Rightarrow Q_2$		Pd	Domaine technologique										Domaine scientifique																			
Domaine technologique	Type de lecteur	0.8	9	9	9	9	3			9	9				9	3	3	1								9	9	1	1			63
	Délais de lecture/écriture cours	0.5		9		9			9	3	3	3	1		9	3	1	9	1	1	1				1	1	9	9	9			46
	Protocole de communication	0.6		9		9			1	3	9			3																		20
	Type de tag	1.7	1	3		9	9	9		9		9	3	9				3					9	9	9				3	3	180	
	Résister à l'ambiance extérieure	0.7							9			3														9						15
	Résistance aux interférences	1.4	9	9						9	3			1										9	3							59
	Types de norme	0.7		9						9		1	1			3					3											19
Domaine scientifique	Invisibilité des tags	0.4	1	1		1		3	1	1	9															1						7
	Protocole d'encodage des données	1.6					3				9	9		3	9		1		1				1	1		1	1	3	3			70
	Volume d'information	1.4	9	3		9	9					3	3	9			9	3	9			3		1	9	9			9	3	137	
	Adaptabilité des infos aux besoins	0.8							3						9	9	3		3		3					9						33
	Exhaustivité de l'information	0.9	9	3			3					1	3	9	3		9	9	3	9	3		1	1	9			1	1	70		
	Typage des informations	0.5					3		3		3	9	1	9	3		3	3	9	3		3	3	1	1	1			1	1	26	
	Gammes d'informations	0.2														9	1	3	9	3												6
	Adaptabilité des gammes d'infos	0.5														9	1	3	9	3												12
	Cohérence des données	1				1								9	3		3		3		1		9	9	1	1	9		3			47
	Résistance aux erreurs bits	0.6	9	9					9															9								22
	Résistance aux pertes	1.6	1	1	3			9	1		3		9	9						1						9	3					76
	Accessibilité physique à la matière	1.2	1					1	9		3		9	3				3		3						9		9	9			65
	Cryptage des données	0.9								1			3	9																9		20
Identification de l'acteur	0.6												3		3														9		7	
			0.3	0.4	0.4	0.3	0.2	0.4	0.4	0.4	0.4	0.3	0.2	0.4	0.6	0.8	0.3	0.1	0.4	0.2	0.3	0.1	0.2	0.2	0.4	0.6	0.6	0.2	0.2	0.3	0.2	

FIGURE 59: Calcul de la contribution au projet avec la matrice QFD de niveau 2

27. Inf_c est calculé dans l'Annexe I. Il s'agit de la valeur de la dernière ligne d'une colonne c dans la FIGURE 59.

Glossaire

AHP : Analytic Hierarchy Process
ALC : Averaging Level Cuts
AMCC : Agent-based Manufacturing Control and Coordination
APS : Advanced Planning System
ASDK : Aglets Software Development Kit
B2M : Business to Manufacturing
B2MML : Business To Manufacturing Markup Language
CAD : Computer-Aided Design
CAE : Computer-Aided Engineering
CAM : Computer-Aided Manufacturing
CAPP : Computer-Aided Process Planning
CMMS : Computerized Maintenance Management System
CoG : Center of Gravity
CRM : Customer Relationship Management
CTE : Common Table Expressions
DDBS : Distributed DataBase System
DNS : Domain Name System
DOODBS : Distributed Object Oriented DataBase System
EDI : Electronic Data Interchange
EPC : Electronic Product Code
ERP : Enterprise Resource Planning
ESDS : Extensible Supply chain Discovery Service
FIPA : Foundation for Intelligent Physical Agents
GRAI : Global Returnable Asset Identifier
GSM : Global System for Mobile communications
GTIN : Global Trade Item Number
HMS : Holonic Manufacturing System
IFAC : International Federation of Automatic Control
IHM : Interface Homme-Machine

IMM : Interface Machine-Machine
IMS : Intelligent Manufacturing Systems
IoT : Internet of Things
JADE : Java Agent DEvelopment framework
JDBC : Java DataBase Connectivity
LCA : Life Cycle Assessment
LEAP : Lightweight Extensible Agent Platform
LMDDBS : Large Mobile Distributed DataBase System
MANET : Mobile Ad Hoc NETwork
MAS : Multi-Agents System
MASCOT : Multi-Agent Supply Chain cOordination Tool
MCDM : Multi-Criteria Decision Making
MLD : Modèle Logique de Données
MES : Manufacturing Execution System
MoM : Mean of Maximum
ONS : Object Naming Service
OSI : Open Systems Interconnection
P2P : Peer-to-Peer
PDM : Product Data Management
PDS : Product Driven Systems
PLC : Product Life Cycle
PLM : Product Lifecycle Management
PML : Physical Markup Language
PROMISE : PROduct lifecycle Management Information tracking using Smart Embedded systems
QFD : Quality Function Deployment
RAM : Random Access Memory
RFID : Radio Frequency IDentification
SSCC : Serial Shipping Container Code
SCM : Supply Chain Management
SQL : Structured Query Language
TDM : Technical Data Management
TOPSIS : Technique Ordered Preference by Similarity to the Ideal Solution
URL : Uniform Resource Locator
WPM : Weighted-Product Model
WSM : Weighted-Sum Model
WWAI : World Wide Article Information
XML : eXtensible Markup Language

Table des figures

1.1	Évolution et environnement des objets intelligents - communicants	3
1.2	Phases du PLC : (Rink and Swan, 1979)	6
1.3	Vue générale du système PLM et de ses objectifs	7
1.4	Bouclage des flux informationnel et physique au sein du PLC	9
1.5	Boîte de conserve de spaghetti : (Wong et al., 2002)	12
1.6	Modèle de classification des produits intelligents	14
1.7	Types de plates-formes basées « agent »	16
1.8	EPCglobal : une requête ONS typique avec l'utilisation d'un serveur PML	17
1.9	ID@URI : une requête classique sur la plate-forme <i>Dialog</i>	18
1.10	WWAI : Vue globale de l'architecture WWAI	19
1.11	Bénéfices des produits communicants à travers la chaîne logistique : (Tajima, 2007)	21
2.1	Système d'information à deux niveaux lié au « système matière communicante »	29
2.2	Taxonomie générale de distribution de l'information	31
2.3	Terminologie relative à une table relationnelle	32
2.4	Stratégies de fragmentation et principe d' <i>overlapping</i>	32
2.5	Terminologie relative à une table relationnelle	34
2.6	Techniques de répllication des données	35
2.7	Types d'architectures mobiles	38
2.8	Triangle illustrant l'impossibilité de remplir les trois critères simultanément	41
2.9	Vue temporelle du processus de dissémination des informations relatives au produit	44
2.10	Processus de dissémination dans le cadre du « système matière communicante »	45
2.11	Illustration de plusieurs points d'écriture/de lecture du produit dans le PLC	48
3.1	Vue d'une partie du MLD, de la table <code>MaterialDefinition</code> et des « tuples produit »	51
3.2	Identification de « tuples produit » dans les tables	57
3.3	Table <code>MaterialDefinition</code> et pertinence des data items issus des « tuples produit »	58
3.4	Groupes d'entités	63
3.5	Ajustement du coefficient k après étude du MLD	65

3.6	Calcul de la pertinence des data items $T_{\text{MatDef}}\{3, 1\}$, $T_{\text{MatDef}}\{3, 2\}$ et $T_{\text{MatDef}}\{3, 3\}$. . .	65
3.7	Illustration d'une structure AHP classique	69
3.8	Intervalle flou trapézoïdal et formes particulières associées	71
3.9	Multiplications de deux intervalles flous suivant deux méthodes d'approximations	72
3.10	Approche AHP-Floue se déclinant en 5 points majeurs	74
3.11	Agrégation des avis des experts suivant deux cas : cas (a) et cas (b)	76
3.12	Structure AHP mise en œuvre dans l'étude	76
3.13	Évaluation de l'importance des groupes d'entités et agrégation des avis	79
3.14	Construction du sous-ensemble flou \tilde{s}_{14} dans la matrice \tilde{G}	79
3.15	Énumération des attributs de <i>MaterialDefinition</i> et agrégation des avis	82
3.16	Réglage du coefficient k et agrégation des avis concernant le critère <i>Model-Based</i>	83
3.17	Réglage du coefficient k et agrégation des avis concernant le critère <i>Data Size</i>	84
3.18	Évaluation de l'importance des critères, agrégation des avis et calcul de leur importance	85
3.19	Matrice de jugement flou \tilde{A}	86
3.20	Matrice de performance floue \tilde{H}	87
3.21	Deux modes opératoires pour l'agrégation multi-critères	88
3.22	Matrice \tilde{R} contenant le sous-ensemble flou relatif à chaque alternative	88
3.23	Méthode du CoG : (Yager, 1980)	89
3.24	Calcul de CoG par approximation inférieure ou supérieure	90
3.25	Calcul du CoG pour chaque alternative, puis classement des alternatives	90
4.1	Processus d'intégration des tags RFID dans le textile	94
4.2	Architecture permettant d'écrire les données à la volée sur la matière communicante . . .	95
4.3	Architecture requise pour localiser les données au sein de la matière communicante . . .	96
4.4	Couches protocolaires de la technologie RFID et structure de l'entête <i>splitting protocol</i> .	98
4.5	Illustration de la dissémination dans la matière de $T_{\text{MatDef}}\{3, 3\}$ via le <i>splitting protocol</i> .	100
4.6	Types d'architectures possibles relatives au processus de dissémination des données . . .	103
5.1	Cycle de vie du textile et intégration des opérations d'écriture/lecture du processus . . .	107
5.2	Principes de comparaison entre les méthodes : Chan-Roddick et AHP-Floue	110
5.3	Vecteurs propres indiquant l'importance de chaque groupe d'entités : phase 1	112
5.4	Vecteurs propres indiquant l'importance de chaque groupe d'entités : phase 2	112
5.5	Vecteurs propres indiquant l'importance de chaque critère : phase 1	113
5.6	Ajustement du coefficient k et agrégation des avis concernant <i>Model Based</i> : phase 1 . .	115
5.7	Ajustement du coefficient k et agrégation des avis concernant <i>Data Size</i> : phase 1	115
5.8	Application « JAVA/Matlab » : extraction des données, ajustements et calcul de pertinence	116
5.9	ListeAF ordonnée du data item le plus pertinent au moins pertinent : Point d'écriture n°1	118
5.10	Représentation de ListeAF sous forme de diagramme : Point d'écriture n°1	118

5.11	ListeCR ordonnée du data item le plus pertinent au moins pertinent : Point d'écriture n°1	119
5.12	Représentation de ListeCR sous forme de diagramme : Point d'écriture n°1	120
5.13	Comparaison des approches Chan-Roddick et AHP-Floue : Point d'écriture n°1	121
5.14	Événements générés lors de l'opération d'écriture et contenu mémoire d'un tag	122
5.15	Application « JAVA Split » permettant de reconstituer les données extraites du produit	124
5.16	Application « JAVA Loc » permettant de localiser les tags RFID	125
5.17	Cartographie virtuelle des tags présents dans la bobine de textile communicant	125
5.18	Précision de calcul des positions des tags dans le textile	126
5.19	Vecteur propre indiquant l'importance de chaque critère : Phase 2	127
5.20	ListeAF ordonnée du data item le plus pertinent au moins pertinent : Point d'écriture n°2	128
5.21	Représentation de ListeAF sous forme de diagramme : Point d'écriture n°2	129
5.22	Représentation de ListeCR sous forme de diagramme : Point d'écriture n°2	130
5.23	Comparaison des approches Chan-Roddick et AHP-Floue : Point d'écriture n°2	131
24	Schéma de la méthode pour la définition et la hiérarchisation des problèmes	140
25	Matrice de hiérarchisation : niveau 1	144
26	Matrice de hiérarchisation : niveau 2	146
27	Matrice de sérialisation Fiedler	148
28	Identification de « tuples produit » dans les tables via la fonction RetrievalData)	160
29	Matrice d'agrégation des avis d'experts : critère <i>Contextual</i>	166
30	Importance relative de chaque groupe d'entité : $\tilde{\Lambda}_G$	167
31	Sous-ensembles flous \tilde{s}_v	168
32	Sous-ensembles flous $\tilde{\phi}_m(l)$	169
33	Sous-ensembles flous $\tilde{\phi}_s(l)$	170
34	Matrice d'agrégation des avis d'experts : critères	171
35	Importance relative de chaque critère : $\tilde{\Lambda}_\rho$	171
36	Matrice de jugement flou \tilde{A}	172
37	Matrice de performance floue \tilde{H}	174
38	Matrice \tilde{R} contenant le sous-ensemble flou relatif à chaque alternative	175
39	Calcul du CoG supérieur du data item $T_{\text{MatDef}}\{3, 1\}$	176
40	Calcul du CoG inférieur du data item $T_{\text{MatDef}}\{3, 1\}$	177
41	Montage permettant d'étudier le signal lecteur-tag	180
42	Étude de la zone de couverture lecteur-tag RFID	180
43	Modélisation de la zone de couverture de la technologie RFID	181
44	Processus d'intégration des μ tags dans le tissu	182

45	Architecture permettant de localiser des tags RFID dans la matière communicante	184
46	Résultats lorsque le critère <i>Enumeration</i> est extrêmement favorisé par rapport aux autres	187
47	Résultats lorsque le critère <i>Contextual</i> est extrêmement favorisé par rapport aux autres .	188
48	Comparaison sur la base d'un avis unique	189
49	Résultats lorsque le critère <i>Enumeration</i> est extrêmement favorisé par rapport aux autres	191
50	Résultats lorsque le critère <i>Contextual</i> est extrêmement favorisé par rapport aux autres .	191
51	Résultats lorsque le critère <i>Model-Based</i> est extrêmement favorisé par rapport aux autres	192
52	Résultats lorsque le critère <i>Data Size</i> est extrêmement favorisé par rapport aux autres . .	192
53	Résultats lorsque les critères sont d'importance égale	193
54	Résultats lorsque le critère <i>Enumeration</i> est extrêmement favorisé par rapport aux autres	195
55	Résultats lorsque le critère <i>Contextual</i> est extrêmement favorisé par rapport aux autres .	195
56	Résultats lorsque le critère <i>Model-Based</i> est extrêmement favorisé par rapport aux autres	196
57	Résultats lorsque le critère <i>Data Size</i> est extrêmement favorisé par rapport aux autres . .	196
58	Résultats lorsque les critères sont d'importance égale	197
59	Calcul de la contribution au projet avec la matrice QFD de niveau 2	200

Liste des publications

Reuves internationales avec comité de lecture

Kubler, S., Robert, J., Rondeau, É. and Georges, J.P. Dual path communications over multiple spanning trees for networked control systems. *Engineering Applications of Artificial Intelligence. The International Journal of Intelligent Real-Time Automation*. 2012 (online : <http://dx.doi.org/10.1016/j.engappai.2012.05.001>).

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Distributed database systems through fixed and mobile equipments - A survey. *Engineering Applications of Artificial Intelligence. The International Journal of Intelligent Real-Time Automation*. 2012. (accepté avec modification)

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Information Dissemination framework for context-aware products. *Computers & Industrial Engineering*. 2012. (soumis)

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Information dissemination process by using communicating materials *Journal of Intelligent Manufacturing*. 2012. (sélectionné pour publication dans une "special issue")

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Method for embedding context-sensitive information on communicating textiles by using Fuzzy AHP theory. *Journal of Intelligent & Fuzzy Systems*. 2012. (soumis)

Conférences internationales avec comité de lecture

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Storage of information on manufactured products using "communicating materials". *European Conference on Artificial Intelligence (ECAI)*. 2012.

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Information dissemination process for context-aware products. *14th IFAC Symposium on Information Control Problems in Manufacturing (INCOM)*. 2012.

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Context-sensitive information selection for mobile communicating products using AHP-Fuzzy theory. *The IEEE International Conference on Internet of Things (iThings)*. 2012. (soumis)

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Prototyping of a communicating textile. In : *International Conference on Industrial Engineering and Systems Management (IESM)*, 1333–1342. 2011.

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Problem definition methodology for the "Communicating Material" paradigm. In : *10th IFAC Workshop on Intelligent Manufacturing Systems (IMS)*. 2010.

Kubler, S., Rondeau, É. and Georges, J.P. Dependability of switched network architectures for networked control systems. In : *IEEE International Conference on Mechatronics (ICM)*. 2010.

Conférences nationales avec comité de lecture

Kubler, S., Rondeau, É. and Georges, J.P. Continuité de service sur Ethernet Industriel. In : *6th Conférence Internationale Francophone d'Automatique (CIFA)*, 2010.

Chapitre d'ouvrage

Kubler, S., Derigent, W., Thomas, A and Rondeau, É. Key Factors for Information Dissemination on Communicating Products and Fixed Databases. In : *Service Orientation in Holonic and Multi-Agent Manufacturing Control (SOHOMA)*, pp. 89–102, 2011.

Groupes de travail nationaux

Kubler, S. *Problématiques de recherche inhérentes au changement de paradigme Produit/Matière communicant(e)*, groupe de travail Intelligent Manufacturing & Services Systems *IMS2*, Strasbourg, 2010.

Kubler, S. *Continuité de service d'Ethernet Industriel*, groupe de travail Réseaux Grand Est, *RGE*, Nancy, 2010.

Bibliographie

- Abawajy, J., Deris, M. and Omar, M. (2006), 'A novel data replication and management protocol for mobile computing systems', *Mobile information systems* **2**(1), 3–19.
- Abuelyaman, E. (2008), 'An optimized scheme for vertical partitioning of a distributed database', *IJCSNS* **8**(1), 310.
- Adrian Runceanu, M. P., ed. (2007), *About Data Fragmentation and Allocation in Distributed Object Oriented Databases, International Conference Ivan Rilsky University, Sofia*.
- AFNOR (1996), 'Analyse de la valeur, analyse fonctionnelle, NF X50-151'.
- Ahmad, I., Karlapalem, K., Kwok, Y. and So, S. (2002), 'Evolutionary algorithms for allocating data in distributed database systems', *Distributed and Parallel Databases* **11**(1), 5–32.
- Alavizadeh, F. and Sabouri, H. (2006), 'Fragment Allocation using Genetic Algorithms in Distributed Database'.
- Ameri, F. and Dutta, D. (2005), 'Product lifecycle management : closing the knowledge loops', *Computer-Aided Design and Applications* **2**(5), 577–590.
- Anagnostopoulos, K. and Petalas, C. (2011), 'A fuzzy multicriteria benefit–cost approach for irrigation projects evaluation', *Agricultural Water Management* **98**(9), 1409–1416.
- Anderson, C. and Zeithaml, C. (1984), 'Stage of the product life cycle, business strategy, and business performance', *Academy of Management Journal* **1**(27), 5–24.
- Anon (2000), '"money well spent", frontline solutions', **59**(6), 30–31.
- Apers, P. (1988), 'Data allocation in distributed database systems', *ACM Transactions on Database Systems* **13**(3), 263–304.
- Ashraf, A., Dominic, P., Durai, D. and Hamidah, I. (2010), 'A New Optimistic Replication Strategy for Large-scale Mobile Distributed Database Systems', *International Journal of database management systems* **2**(4), 86–105.
- Atkins, J. E., Boman, E. G. and Hendrickson, B. (1998), 'A spectral algorithm for seriation and the consecutive ones problem', *Journal on Computing, SIAM* **28**, 297–310.
- Augusto, J. (2007), 'Ambient Intelligence : The confluence of ubiquitous/pervasive computing and artificial intelligence', *Intelligent Computing Everywhere* pp. 213–234.

- Aydina, S. and Kahramanb, C. (2012), 'Multiattribute supplier selection using fuzzy analytic hierarchy process', *International Journal of Computational Intelligence Systems* **3**(5), 553–565.
- Azadeh, A., Nazari-Shirkouhi, S., Hatami-Shirkouhi, L. and Ansarinejad, A. (2011), 'A unique fuzzy multi-criteria decision making : computer simulation approach for productive operators' assignment in cellular manufacturing systems with uncertainty and vagueness', *The International Journal of Advanced Manufacturing Technology* **56**, 329–343.
- Baião, F. and Mattoso, M. (1998), A mixed fragmentation algorithm for distributed object oriented databases, in 'International Conference Computing and Information', pp. 141–148.
- Baião, F., Mattoso, M. and Zaverucha, G. (2002), A framework for the design of distributed databases, in 'Workshop on Distributed Data & Structures'.
- Banaszak, Z. and Zaremba, M., eds (2003), *Special issue on internet-based distributed intelligent manufacturing systems*, Vol. 14, Journal of Intelligent Manufacturing.
- Banâtre, M., Bryce, C., Couderc, P. and Weis, F. (2007), *Informatique diffuse : des concepts à la réalité*, Lavoisier.
- Barbará, D. (1999), 'Mobile computing and databases-a survey', *IEEE Transactions on Knowledge and Data Engineering*, **11**(1), 108–117.
- Barbará, D., DuMouchel, W., Faloutsos, C., Haas, P., Hellerstein, J., Ioannidis, Y., Jagadish, H., Johnson, T., Ng, R., Poosala, V. et al. (1997), The new jersey data reduction report, in 'IEEE Data Engineering Bulletin : Special Issue on Data Reduction Techniques', Vol. 20, pp. 3–45.
- Bargagli, R., Wynn-Williams, D., Bersan, F., Cavacini, P., Ertz, S., Frati, F., Freckman, D., Lewis-Smith, R., Russell, N. and Smith, A. (1997), 'Field report, Biotex 1 : first BIOTAS expedition', *Newsletter of the Italian Biological Research in Antarctica* **1**(1995-96), 42–58.
- Barney, H. and Low, G. (2008), 'Object Allocation with Replication in Distributed Systems', *International Journal of Information Technology* **4**, 28–36.
- Basseda, R. and Tasharofi, S. (2005), Data Allocation in Distributed Database Systems, Technical report, Technical Report No. DBRG. RB-ST. A50715.
- Baumer, C., Breugst, M., Choy, S. and Magedanz, T. (1999), Grasshopper : A universal agent platform based on omg masif and fipa standards, in '1st Workshop on Mobile Agents for Telecommunication Applications (MATA99)', pp. 1–18.
- Bellatreche, L., Karlapalem, K. and Simonet, A. (1997), Horizontal class partitioning in object-oriented databases, in 'Database and Expert Systems Applications', pp. 58–67.
- Belton, V. and Gear, T. (1983), 'On a short-coming of saaty's method of analytic hierarchies', *Omega* **11**(3), 228–230.
- Beulens, A., Jansen, M. and Wortmann, J. (2000), The information de-coupling point, in 'International conference on Advances in production management systems', pp. 50–58.

- Bhar, S. and Barker, K. (1995), 'Static allocation in distributed objectbase systems : A graphical approach', *Information Systems and Data Management* pp. 92–114.
- Blankinship, R., Hevner, A. and Yao, S. (1991), An iterative method for distributed database design, in '17th International Conference on Very Large Data Bases', pp. 389–400.
- Bobineau, C., Bouganim, L., Pucheral, P. and Valduriez, P. (2000), PicoDBMS : Scaling down database techniques for the smartcard, in '26th International Conference on Very Large Data Bases', pp. 11–20.
- Bohn, J., Coroamă, V., Langheinrich, M., Mattern, F. and Rohs, M. (2005), 'Social, economic, and ethical implications of ambient intelligence and ubiquitous computing', *Ambient intelligence* pp. 5–29.
- Bouchon-Meunier, B. (2007), *La logique floue*, Presses universitaires de France.
- Boxall, G. (2000), 'The use of rfid for retail supply chain logistics', *Tag 2000* **24**.
- Bozbura, F. T. and Beskese, A. (2007), 'Prioritization of organizational capital measurement indicators using fuzzy ahp', *International Journal of Approximate Reasoning* **44**(2), 124–147.
- Brock, D. (2001), 'The electronic product code (epc)', *A Naming Scheme for Physical Objects* .
- Brun, J., Vicard, D., Mourey, B., Lépine, B. and Frassati, F. (2009), 'Packaging and wired interconnections for insertion of miniaturized chips in smart fabrics', *Microelectronics and Packaging Conference* pp. 1–5.
- Brunstrom, A., Leutenegger, S. and Simha, R. (1995), Experimental evaluation of dynamic data allocation strategies in a distributed database with changing workloads, in '4th International conference on Information and knowledge management', pp. 395–402.
- Buckley, J. (1985), 'Fuzzy hierarchical analysis', *Fuzzy Sets and Systems* **17**, 233–247.
- Bussmann, S. (1998), An agent-oriented architecture for holonic manufacturing control, in '1st International Workshop on IMS, Lausanne', pp. 1–12.
- Büyükoçkan, G., Cifçi and Güleriyüz, S. (2011), 'Strategic analysis of healthcare service quality using fuzzy ahp methodology', *Expert Systems with Applications* **38**(8), 9407–9424.
- Büyükoçkan, G. and Çifçi, G. (2012), 'A combined fuzzy ahp and fuzzy topsis based strategic analysis of electronic service quality in healthcare industry', *Expert Systems with Applications* **39**(3), 2341–2354.
- Catrysse, M., Puers, R., Hertleer, C., Van Langenhove, L., Van Egmond, H. and Matthys, D. (2004), 'Towards the integration of textile sensors in a wireless monitoring suit', *Sensors and Actuators A : Physical* **114**(2-3), 302–311.
- Celik, M., Er, I. D. and Ozok, A. F. (2009), 'Application of fuzzy extended ahp methodology on shipping registry selection : The case of turkish maritime industry', *Expert Systems with Applications* **36**(1), 190–198.

- Chakravarthy, S., Muthuraj, J., Varadarajan, R. and Navathe, S. (1994), 'An objective function for vertically partitioning relations in distributed databases and its analysis', *Distributed and parallel databases* **2**(2), 183–207.
- Chan, D. and Roddick, J. (2003), Context-sensitive mobile database summarisation, in '26th Australasian computer science conference', Vol. 16, pp. 139–149.
- Chan, D. and Roddick, J. (2005), 'Summarisation for Mobile Databases', *Journal of Research and Practice in Information Technology* **37**(3), 267.
- Chang, D.-Y. (1996), 'Applications of the extent analysis method on fuzzy ahp', *European Journal of Operational Research* **95**(3), 649–655.
- Chao, C., Yang, J. and Jen, W. (2007), 'Determining technology trends and forecasts of RFID by a historical review and bibliometric analysis from 1991 to 2005', *Technovation* **27**(5), 268–279.
- Chaturvedi, A., Choubey, A. and Roan, J. (1994), 'Scheduling the allocation of data fragments in a distributed database environment : A machine learning approach', *IEEE Transactions on Engineering Management* **41**(2), 194–207.
- Chen, C. and Klein, C. (1997), 'A simple approach to ranking a group of aggregated fuzzy utilities', *IEEE Transactions on Systems, Man, and Cybernetics, Part B : Cybernetics* **27**(1), 26–35.
- Chen, R. and Tu, M. (2009), 'Development of an agent-based system for manufacturing control and coordination with ontology and RFID technology', *Expert Systems with Applications* **36**(4), 7581–7593.
- Chen, S. and Chen, S. (2003), 'A new method for handling multicriteria fuzzy decision-making problems using fn-iowa operators', *Cybernetics & Systems* **34**(2), 109–137.
- Chen, S. and Chen, S. (2007), 'Fuzzy risk analysis based on the ranking of generalized trapezoidal fuzzy numbers', *Applied Intelligence* **26**(1), 1–11.
- Cheng, C. (1998), 'A new approach for ranking fuzzy numbers by distance method', *Fuzzy sets and systems* **95**(3), 307–317.
- Cheng, C.-H. (1997), 'Evaluating naval tactical missile systems by fuzzy ahp based on the grade value of membership function', *European Journal of Operational Research* **96**(2), 343–350.
- Cheng, C., Lee, W. and Wong, K. (2002), 'A genetic algorithm-based clustering approach for database partitioning', *IEEE Transactions on Systems, Man, and Cybernetics, Part C : Applications and Reviews* **32**(3), 215–230.
- Chin, A. (2002), Incremental data allocation and reallocation in distributed database systems, in 'Data warehousing and web engineering', pp. 137–160.
- Cho, D. W., Lee, Y. H., Ahn, S. H. and Hwang, M. K. (2012), 'A framework for measuring the performance of service supply chain management', *Computers & Industrial Engineering* **62**(3), 801–818.

- Choo, E., Schoner, B. and Wedley, W. (1999), 'Interpretation of criteria weights in multicriteria decision making', *Computers & Industrial Engineering* **37**(3), 527–541.
- Chou, S.-Y., Chang, Y.-H. and Shen, C.-Y. (2008), 'A fuzzy simple additive weighting system under group decision-making for facility location selection with objective/subjective attributes', *European Journal of Operational Research* **189**(1), 132–145.
- Chu, T. and Tsao, C. (2002), 'Ranking fuzzy numbers with an area between the centroid point and original point', *Computers & Mathematics with Applications* **43**(1-2), 111–117.
- Cochran, J. and Chen, H. (2005), 'Fuzzy multi-criteria selection of object-oriented simulation software for production system analysis', *Computers & operations research* **32**(1), 153–168.
- Corcoran, A. and Hale, J. (1994), A genetic algorithm for fragment allocation in a distributed database system, in 'ACM Symposium on Applied computing', pp. 247–250.
- Darabant, A., Campan, A. and Navroschi-Szasz, A. (2004), 'Optimal Class Fragmentation Ordering in Object Oriented Databases', *Studia Universitatis Babes Bolyai Informatica* **49**(1), 45–54.
- Das, P. (2011), 'Selection of business strategies for quality improvement using fuzzy analytical hierarchy process', *International Journal for Quality research* **4**(4), 283–289.
- Day, G. (1981), 'The product life cycle : analysis and applications issues', *The Journal of Marketing* **45**(1), 60–67.
- Demers, A., Petersen, K., Spreitzer, M., Ferry, D., Theimer, M. and Welch, B. (2002), The Bayou Architecture : support for data sharing among mobile users, in 'Mobile Computing Systems and Applications', IEEE, pp. 2–7.
- Dhalla, N. and Yuspeh, S. (1976), 'Forget the product life cycle concept !', *Harvard Business Review* **54**(1), 102–112.
- Dong, W. and Wong, F. (1987), 'Fuzzy weighted averages and implementation of the extension principle', *Fuzzy Sets and Systems* **21**(2), 183–199.
- Dunham, M. (2006), *Data mining : Introductory and advanced topics*, Pearson Education India.
- Durbach, I. N. and Stewart, T. J. (2012), 'Modelling uncertainty in multi-criteria decision analysis', *European Journal of Operational Research* .
- Eradus, W. and Jansen, M. (1999), 'Animal identification and monitoring', *Computers and Electronics in Agriculture* **24**(1–2), 91–98.
- Ezeife, C. and Barker, K. (1998), 'Distributed object based design : Vertical fragmentation of classes', *Distributed and Parallel Databases* **6**(4), 317–350.
- Ezeife, C. and Zheng, J. (1999a), 'Dynamic Database Object Horizontal Fragmentation', *Systems development methods for databases, enterprise modeling, and workflow management* p. 51.

- Ezeife, C. and Zheng, J. (1999b), Measuring the performance of database object horizontal fragmentation schemes, in 'Database Engineering and Applications', IEEE, pp. 408–414.
- Fiedler, M. (1975a), 'A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory', *Czechoslovak Mathematical Journal* **25**(100), 619–633.
- Fiedler, M. (1975b), 'Eigenvectors of acyclic matrices', *Czechoslovak Mathematical Journal* **100**(100), 607–618.
- Fife, L. and Gruenwald, L. (2003), 'Research issues for data communication in mobile ad-hoc network database systems', *ACM SIGMOD Record* **32**(2), 42–47.
- Fox, M., Barbuceanu, M. and Teigen, R. (2000), 'Agent-oriented supply-chain management', *International Journal of Flexible Manufacturing Systems* **12**(2), 165–188.
- Främling, K., Ala-Risku, T., Kärkkäinen, M. and Holmström, J. (2006), 'Agent-based model for managing composite product information', *Computers in Industry* **57**(1), 72–81.
- Främling, K., Harrison, M., Brusey, J. and Petrow, J. (2007), 'Requirements on unique identifiers for managing product lifecycle information : comparison of alternative approaches', *International Journal of Computer Integrated Manufacturing* **20**(7), 715–726.
- Främling, K., Holmström, J., Ala-Risku, T. and Kärkkäinen, M. (2003), 'Product agents for handling information about physical objects', *Report of Laboratory of information processing science series B, TKO-B* **153**(03).
- Fu, H.-P., Chao, P., Chang, T.-H. and Chang, Y.-S. (2008), 'The impact of market freedom on the adoption of third-party electronic marketplaces : A fuzzy ahp analysis', *Industrial Marketing Management* **37**(6), 698–712.
- Fung, C., Karlapalem, K. and Li, Q. (2002), 'An evaluation of vertical class partitioning for query processing in object-oriented databases', *IEEE Transactions on Knowledge and Data Engineering* pp. 1095–1118.
- Gao, T., Jin, R. C., Song, J. Y., Xu, T. B. and Wang, L. D. (2012), 'Energy-efficient cluster head selection scheme based on multiple criteria decision making for wireless sensor networks', *Wireless Personal Communications* **63**(4), 871–894.
- Garcia-Molina, H., Ullman, J. and Widom, J. (2008), *Database systems : the complete book*, Prentice Hall Press.
- Gardner, D. (1986), *The Product Life Cycle : It's Role in Marketing Strategy/some Evolving Observations about the Life Cycle*, BEBR faculty working paper, University of Illinois at Urbana-Champaign.
- Garetti, M., Macchi, M. and Van De Berg, R. (2003), 'Digitally supported engineering of industrial systems in the globally scaled manufacturing', *Network of Excellence on Intelligent Manufacturing Systems (IMS-NoE SIG)* **1**.

- Garey, M. and Johnson, D. (1979), *Computers and Intractability : A Guide to the Theory of NP-completeness*, WH Freeman & Co. New York, NY, USA.
- Grabisch, M. and Perny, P. (1999), 'Agrégation multicritère', *Utilisations de la logique floue. Hermes*.
- Gray, J., Helland, P., O'Neil, P. and Shasha, D. (1996), 'The dangers of replication and a solution', *SIGMOD Rec.* **25**, 173–182.
- Grebla, H., Moldovan, G., Darabant, S. and Câmpan, A. (2004), Data allocation in distributed database systems performed by mobile intelligent agents, in 'Journal, Proc. Intern. Conf. on Theory and Applications of Mathematics and Informatics', pp. 164–173.
- Gruenwald, L., Javed, M. and Gu, M. (2002), Energy-efficient data broadcasting in mobile ad-hoc networks, in 'Database Engineering and Applications Symposium, Proceedings. International', IEEE, pp. 64–73.
- Gu, X. and Zhu, Q. (2006), 'Fuzzy multi-attribute decision-making method based on eigenvector of fuzzy attribute evaluation space', *Decision Support Systems* **41**(2), 400–410.
- Hababeh, I. (2010), 'Improving network systems performance by clustering distributed database sites', *The Journal of Supercomputing* pp. 1–19.
- Hababeh, I., Bowring, N. and Ramachandran, M. (2005), A method for fragment allocation design in the distributed database systems, in 'The 6th Annual UAE University Research Conference', Citeseer.
- Hamideh Sabouri, Fatemeh Alavizadeh, M. R., ed. (2008), *Applying Genetic Algorithms to Fragment Allocation in Distributed Database Systems, Proc. of the 3rd ICSOFT 2008*.
- Hara, T. and Madria, S. (2006), 'Data replication for improving data accessibility in ad hoc networks', *IEEE Transactions on Mobile Computing* pp. 1515–1532.
- Harker, P. and Vargas, L. (1987), 'The theory of ratio scale estimation : Saaty's analytic hierarchy process', *Management Science* **33**(11), 1383–1403.
- Hauglid, J., Ryeng, N. and Nørnvåg, K. (2010), 'DYFRAM : dynamic fragmentation and replica management in distributed database systems', *Distributed and Parallel Databases* pp. 1–29.
- Holmström, J., Främling, K., Kärkkäinen, M. and Ala-Risku, T. (2002), 'Implementing collaboration process networks', *The International Journal of Logistics Management* **13**(2), 39–50.
- Huang, Y. and Chen, J. (2001), 'Fragment allocation in distributed database design', *Journal of Information Science and Engineering* **17**(3), 491–506.
- Huang, Y., Sistla, P. and Wolfson, O. (1994), 'Data replication for mobile computers', *ACM SIGMOD Record* **23**(2), 13–24.
- Huvio, E., Grönvall, J. and Främling, K. (2002), Tracking and tracing parcels using a distributed computing approach, in 'Proceedings of the 14th Annual conference for Nordic researchers in logistics', pp. 29–43.

- ISO/IEC (2004), 'Information technology aidc techniques – rfid for item man- agement – air interface, part 1 – generic parameters for air interface communication', ISO/IEC 18000-1.
- Ives, B. and Vitale, M. (1988), 'After the sale : leveraging maintenance with information technology', *MIS Quarterly* **12**(1), 7–21.
- Jaber, M. (2009), Architecture de Système d'Information Distribu   pour la Gestion de la Cha  ne Logis- tique : Une approche Orient  e Services, PhD thesis, Institut National des Sciences Appliqu  es de Lyon.
- Javanbarg, M. B., Scawthorn, C., Kiyono, J. and Shahbodaghkhan, B. (2012), 'Fuzzy ahp-based multi- criteria decision making systems using particle swarm optimization', *Expert Systems with Applica- tions* **39**(1), 960–966.
- Jennings, N. and Wooldridge, M. (1998), 'Applications of intelligent agents', *Agent technology : Foun- dations, applications and markets* pp. 3–28.
- Jiang, L., Liu, D. and Yang, B. (2004), Smart home research, in 'Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on', Vol. 2, IEEE, pp. 659–663.
- Johnson, D. (2002), 'Rfid tags improve tracking, quality on ford line in mexico', *Control Engineering* **49**(11), 16.
- Kangasharju, J., Roberts, J. and Ross, K. (2002), 'Object replication strategies in content distribution networks', *Computer Communications* **25**(4), 376–383.
- K  rkk  inen, M. (2003), 'Increasing efficiency in the supply chain for short shelf life goods using RFID tagging', *International Journal of Retail & Distribution Management* **31**(10), 529–536.
- K  rkk  inen, M., Ala-Risku, T. and Fr  mling, K. (2003), 'The product centric approach : a solution to supply network information management problems ?', *Computers in Industry* **52**(2), 147–159.
- K  rkk  inen, M., Ala-Risku, T. and Fr  mling, K. (2004), 'Efficient tracking for short-term multi-company networks', *International Journal of Physical Distribution & Logistics Management* **34**(7), 545–564.
- K  rkk  inen, M., Holmstr  m, J., Fr  mling, K. and Arto, K. (2003), 'Intelligent products—a step towards a more effective project delivery chain', *Computers in Industry* **50**(2), 141–151.
- Karlapalem, K. and Li, Q. (1995), Partitioning schemes for object oriented databases, in '5th Internatio- nal Workshop on Research Issues in Data Engineering-Distributed Object Management', pp. 42–50.
- Karlapalem, K., Navathe, S. and Morsi, M. (1994), 'Issues in distribution design of object-oriented data- bases', *Distributed object management* pp. 148–164.
- Karnik, N. and Tripathi, A. (2001), 'Security in the ajanta mobile agent system', *Software : Practice and Experience* **31**(4), 301–329.
- Kaufmann, A. and Gupta, M. M. (1991), *Introduction to Fuzzy Arithmetic : Theory and Application*, Van Nostrand Reinhold.

- Kayaa, T. and Kahramanb, E. (2011), 'A fuzzy approach to e-banking website quality assessment based on an integrated ahp-electre method', *Technological and Economic Development of Economy* **17**(2), 313–334.
- Khan, S. and Hoque, A. (2010), 'A New Technique for Database Fragmentation in Distributed Systems', *International Journal of Computer Applications, IJCA* **5**(9), 20–24.
- Kiritsis, D. (2011), 'Closed-loop plm for intelligent products in the era of the internet of things', *Computer-Aided Design* **43**(5), 479–501.
- Kiritsis, D., Bufardi, A. and Xirouchakis, P. (2003), 'Research issues on product lifecycle management and information tracking using smart embedded systems', *Advanced Engineering Informatics* **17**(3), 189–202.
- Koestler, A. (1967), *The ghost in the machine*, Hutchinson.
- Koronios, A., Nastasie, D., Chanana, V. and Haider, A. (2007), Integration Through Standards—An Overview Of International Standards For Engineering Asset Management, in '4th International Conference on Condition Monitoring', pp. 11–14.
- Kreng, V. B. and Wu, C. Y. (2007), 'Evaluation of knowledge portal development tools using a fuzzy ahp approach : The case of taiwanese stone industry', *European Journal of Operational Research* **176**(3), 1795 – 1810.
- Kubler, S., Derigent, W., Thomas, A. and Rondeau, É. (2010), Problem definition methodology for the "Communicating Material" paradigm, in '10th IFAC Workshop on Intelligent Manufacturing Systems'.
- Kubler, S., Derigent, W., Thomas, A. and Rondeau, É. (2011a), Key factors for information dissemination on communicating products and fixed databases, in Springer, ed., 'Service Orientation in Holonic and Multi-Agent Manufacturing Control', Vol. 402, pp. 89–102.
- Kubler, S., Derigent, W., Thomas, A. and Rondeau, É. (2011b), Prototyping of a communicating textile, in 'International Conference on Industrial Engineering and Systems Management', pp. 1333–1342.
- Kubler, S., Derigent, W., Thomas, A. and Rondeau, É. (2012a), Information dissemination process for context-aware products, in '14th Symposium of Information Control Problems in Manufacturing'.
- Kubler, S., Derigent, W., Thomas, A. and Rondeau, É. (2012b), Storage of information on manufactured products using "communicating materials", in 'AI Problems and Approaches for Intelligent Environments (AI@IE)'.
- Kubler, S., Robert, J., Georges, J.-P. and Rondeau, É. (2012), 'Dual path communications over multiple spanning trees for networked control systems', *Engineering Applications of Artificial Intelligence* **25**(7), 1460–1470.
- Kuo, R., Chi, S. and Kao, S. (1999), 'A decision support system for locating convenience store through fuzzy ahp', *Computers & Industrial Engineering* **1-2**(37), 323–326.

- Kwong, C. and Bai, H. (2002), 'Determining the importance weights for the customer requirements in qfd using a fuzzy ahp with an extent analysis approach', *Journal of Intelligent Manufacturing* **35**(7), 619–626.
- Labib, A., Williams, G. and O'Connor, R. (1998), 'An intelligent maintenance model (system) : an application of the analytic hierarchy process and a fuzzy logic rule-based controller', *Journal of the Operational Research Society* pp. 745–757.
- Lee, C. and Kwak, N. (1999), 'Information resource planning for a health-care system using an ahp-based goal programming method', *Journal of the operational research society* pp. 1191–1198.
- Lee, K. (1989), *Automatic speech recognition : the development of the SPHINX system*, Kluwer Academic Pub.
- Lee, L., Fiedler, K. and Smith, J. (2008), 'Radio frequency identification (RFID) implementation in the service sector : A customer-facing diffusion model', *International Journal of Production Economics* **112**(2), 587–600.
- Leung, L. and Cao, D. (2000), 'On consistency and ranking of alternatives in fuzzy ahp', *European Journal of Operational Research* **124**(1), 102–113.
- Levitt, T. (1965), *Exploit the product life cycle*, Vol. 43, Graduate School of Business Administration, Harvard University.
- Ley, D. (2007), 'Ubiquitous computing', *Ubiquitous Computing, emerging technologie* **2**, 64–79.
- Lin, L. (2009), 'An integrated framework for the development of radio frequency identification technology in the logistics and supply chain management', *Computers & Industrial Engineering* **57**(3), 832–842.
- Lootsma, F. A. (1988), Numerical scaling of human judgement in pairwise-comparison methods for fuzzy multi-criteria decision analysis, in G. Mitra, ed., 'Mathematical models for decision support', Springer-Verlag New York, Inc., chapter Numerical scaling of human judgement in pairwise-comparison methods for fuzzy multi-criteria decision analysis, pp. 57–88.
- Loukopoulos, T. and Ahmad, I. (2004), 'Static and adaptive distributed data replication using genetic algorithms', *Journal of Parallel and Distributed Computing* **64**(11), 1270–1285.
- Lubinski, A. (2000), Small database answers for small mobile resources, in 'International Conference on Intelligent Interactive Assistance and Mobile Multimedia Computing', pp. 9–10.
- Lubinski, A. and Heuer, A. (2000), Configured replication for mobile applications, in 'Rostocker informatik berichte'.
- Madria, S., Mohania, M. and Roddick, J. (1998), A query processing model for mobile computing using concept hierarchies and summary databases, in '5th International Conference on Foundation for Data Organization'.

- Malinowski, E. and Chakravarthy, S. (1997), Fragmentation Techniques for Distributing Object-Oriented Databases, in '16th International Conference on Conceptual Modeling', pp. 347–360.
- McCarthy, J., Minsky, M., Rochester, N. and Shannon, C. (2006), 'A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955', *AI Magazine* **27**(4), 12.
- McDonough, W. and Braungart, M. (2002), *Cradle to cradle : Remaking the way we make things*, North Point Pr.
- McFarlane, D., Giannikas, V., Wong, A. C. and Harrison, M. (2012), Intelligent products in the supply chain - 10 years on, in '14th Symposium of Information Control Problems in Manufacturing'.
- McFarlane, D., Sarma, S., Chirn, J. L., Wong, C. Y. and Ashton, K. (2003), 'Auto ID systems and intelligent manufacturing control', *Engineering Applications of Artificial Intelligence* **16**(4), 365–376.
- McFarlane, D., Sarma, S., Chirn, J., Wong, C. and Ashton, K. (2002), The intelligent product in manufacturing control and management, in '15th Triennial World Congress'.
- McFarlane, D. and Sheffi, Y. (2003), 'The impact of automatic identification on supply chain operations', *International Journal of Logistics Management* **14**(1), 1–17.
- Melton, J., Simpson, A. and Gray, J. (2001), *SQL : 1999 - Understanding Relational Language Components*, Morgan Kaufmann.
- Meyer, G., Främling, K. and Holmström, J. (2009), 'Intelligent products : A survey', *Computers in Industry* **60**(3), 137–148.
- Mikhailov, L. (2003), 'Deriving priorities from fuzzy pairwise comparison judgements', *Fuzzy Sets and Systems* **134**(3), 365–385.
- Moiz, S. A. and Rajamani, L. (2009), 'Replication Strategies in Mobile Environments', *International Journal of Information Technology, BVICAM'S*.
- Monitto, M., Pappalardo, P. and Tolio, T. (2002), 'A new fuzzy ahp method for the evaluation of automated manufacturing systems', *CIRP Annals - Manufacturing Technology* **51**(1), 395–398.
- Morel, G. and Grabot, B. (2003), 'Editorial of special issue', *Engineering Applications of Artificial Intelligence* **16**(4), 271–275.
- Morel, G., Valckenaers, P., Faure, J., Pereira, C. and Diedrich, C. (2007), 'Manufacturing plant control challenges and issues', *Control Engineering Practice* **15**(11), 1321–1331.
- Muhs, W. F. (1985), The product life cycle concept : Origin and early antecedents, in 'Proceedings of the 2nd Workshop on Historical Research in Marketing, Stanley Hollander et al., eds., Michigan State University', pp. 413–419.
- Murakami, S. and Maeda, M. (1984), 'Fuzzy decision analysis in the development of centralized regional energy control systems', *Energy Dev. Jpn. ;(United States)* **6**(4).

- Navathe, S., Ceri, S., Wiederhold, G. and Dou, J. (1984), 'Vertical partitioning algorithms for database design', *ACM Transactions on Database Systems (TODS)* **9**(4), 680–710.
- Navathe, S., Karlapalem, K. and Ra, M. (1995), 'A mixed fragmentation methodology for initial distributed database design', *Journal of Computer and Software Engineering* **3**(4), 395–426.
- Navathe, S. and Ra, M. (1989), Vertical partitioning for database design : a graphical algorithm, in 'ACM SIGMOD Record', Vol. 18, pp. 440–450.
- Noci, G. and Toletti, G. (2000), 'Selecting quality-based programmes in small firms : A comparison between the fuzzy linguistic approach and the analytic hierarchy process', *International Journal of Production Economics* **67**(2), 113–133.
- Nof, S., Morel, G., Monostori, L., Molina, A. and Filip, F. (2006), 'From plant and logistics control to multi-enterprise collaboration', *Annual Reviews in Control* **30**(1), 55–68.
- Ollero, A., Morel, G., Bernus, P., Nof, S. Y., Sasiadek, J., Boverie, S., Erbe, H. and Goodall, R. (2002), 'Milestone report of the manufacturing and instrumentation coordinating committee : From MEMS to enterprise systems', *Annual Reviews in Control* **26**(2), 151–162.
- Ollivier, M. (1995), 'Rfid enhances materials handling', *Sensor Review* **15**(1), 36–39.
- Olson, D. L., Fliedner, G. and Currie, K. (1995), 'Comparison of the rembrandt system with analytic hierarchy process', *European Journal of Operational Research* **82**(3), 522–539.
- Otey, M. (2005), *Microsoft SQL Server 2005 - new features*, McGraw-Hill.
- Oussalah, M. (2002), 'On the compatibility between defuzzification and fuzzy arithmetic operations', *Elsevier* **128**(2), 247–260.
- Ozsu, M. and Valduriez, P. (1991), 'Principles of Distributed Database Systems', *Prentice Hall* .
- Pabmanabhan, P. and Gruenwald, L. (2006), DREAM : A Data Replication Technique for Real-Time Mobile Ad-hoc Network Databases, in '22nd International Conference on Data Engineering, IC-DE'06', pp. 134–134.
- Padmanabhan, P., Gruenwald, L., Vallur, A. and Atiquzzaman, M. (2008), 'A survey of data replication techniques for mobile ad hoc network databases', *The VLDB Journal* **17**(5), 1143–1164.
- Park, S., Mackenzie, K. and Jayaraman, S. (2002), The wearable motherboard : a framework for personalized mobile information processing (PMIP), in '39th Annual Design Automation Conference', ACM, pp. 170–174.
- Parlikad, A. and McFarlane, D. (2007), 'RFID-based product information in end-of-life decision making', *Control Engineering Practice* **15**(11), 1348–1363.
- Peters, L. (2006), 'Nabaztag wireless communicator', *Personal Computer World* **2**.
- Petersen, K., Spreitzer, M., Terry, D., Theimer, M. and Demers, A. (1997), Flexible update propagation for weakly consistent replication, in 'Operating Systems Review, SIGOPS', Vol. 31, pp. 288–301.

- Petrescu, M. H. (2010), Dynamic Relocation of data fragments in a distributed database, PhD thesis, Babes-Bolyai University, Faculty of Mathematics and Computer Science.
- Phatak, S. and Badrinath, B. (1999), Data partitioning for disconnected client server databases, in '1st International Workshop on Data Engineering for Wireless and Mobile Access', pp. 102–109.
- Pinto, D. and Torres, G. (2002), 'On dynamic fragmentation of distributed databases using partial replication', *Advances in systems theory, mathematical methods and applications* pp. 208 – 211.
- Polli, R. and Cook, V. (1969), 'Validity of the product life cycle', *The Journal of Business* **42**(4), 385–400.
- Potdar, M., Sharif, A., Potdar, V. and Chang, E. (2009), Applications of Wireless Sensor Networks in Pharmaceutical Industry, in 'International Conference on Advanced Information Networking and Applications Workshops', pp. 642–647.
- Rahmani, S., Torkzaban, V. and Haghghat, A. (2009), A New Method of Genetic Algorithm for Data Allocation in Distributed Database Systems, in '1st International Workshop on Education Technology and Computer Science', pp. 1037–1041.
- Ramli, N. and Mohamad, D. (2009), 'A comparative analysis of centroid methods in ranking fuzzy numbers', *European Journal of Scientific Research* **28**(3), 492–501.
- Ramparany, F. and Boissier, O. (2002), Smart devices embedding multi-agent technologies for a proactive world, in 'Proc. of the Ubiquitous Computing Workshop'.
- Rink, D. and Swan, J. (1979), 'Product life cycle research : A literature review', *Journal of business Research* **7**(3), 219–242.
- Runceanu, A., ed. (2007), *Towards vertical fragmentation in distributed databases. International Symposium on Systems Theory, Automation, Robotics, Computers, Informatics, Electronics and Instrumentation - SINTES 13*, Vol. 2.
- Saaty, T. (1980), *The Analytic Hierarchy Process*, New York : McGraw-Hill.
- Saaty, T. (1983), 'Conflict resolution and the falkland islands invasions', *Interfaces* pp. 68–83.
- Saaty, T. (2008), 'Decision making with the analytic hierarchy process', *J. Services. Sciences* **1**(1), 83–98.
- Sadeh, N., Hildum, D. and Kjenstad, D. (2003), 'Agent-based e-supply chain decision support', *Journal of Organizational Computing and Electronic Commerce* **13**(3-4), 225–241.
- Sallez, Y., Berger, T. and Trentesaux, D. (2009), 'A stigmergic approach for dynamic routing of active products in fms', *Computers in Industry* **60**(3), 204–216.
- Santucci, G. (2009), 'From internet of data to internet of things', *Information Society and Media, Directorate General, European Commission, Brussels* .

- Sarhan, A. (2009), 'A new allocation technique for methods and attributes in distributed object-oriented databases using genetic algorithms', *Int. Arab J. Inf. Technol.* **6**(1), 17–26.
- Schoner, B. and Wedley, W. (1989), 'Ambiguous criteria weights in ahp : Consequences and solutions', *Decision Sciences* **20**(3), 462–475.
- Schwarz, A. and Van Langenhove, L. (2006), Report on State-of-the-art on intelligent textiles, "Development of a strategic Master plan for the transformation of the traditional textile and clothing into a knowledge driven industrial sector by 2015", Technical report, Clevertex.
- Sharma, P., Kant, K. and Chauhan, N. (2010), 'A Comparative Study of Cluster-based Data Replication Techniques for MANETs', *International Journal of Information Technology* **2**(2), 665–667.
- Shen, W. and Norrie, D. (1999), 'Agent-based systems for intelligent manufacturing : A state-of-the-art survey', *Knowledge and Information Systems, an International Journal* **1**, 129–156.
- Singh, A. and Kahlon, K. (2009), 'Non-replicated Dynamic Data Allocation in Distributed Database Systems', *IJCSNS* **9**(9), 176.
- Sleit, A., AlMobaideen, W., Al-Areqi, S. and Yahya, A. (2007), 'A Dynamic Object Fragmentation and Replication Algorithm In Distributed Database Systems', *American Journal of Applied Sciences* **4**(8), 613–618.
- Son, J. and Kim, M. (2004), 'An adaptable vertical partitioning method in distributed systems', *Journal of Systems and Software* **73**(3), 551–561.
- Streitz, N. and Nixon, P. (2005), 'The disappearing computer', *Communications of the ACM* **48**(3), 32–35.
- Sundmaeker, H., Guillemin, P., Friess, P. and Woelfflé, S. (2010), 'Vision and challenges for realising the Internet of Things', *Cluster of European Research Projects on the Internet of Things, European Commission* .
- Tadisina, S., Troutt, M. and Bhasin, V. (1991), 'Selecting a doctoral programme using the analytic hierarchy process : The importance of perspective', *Journal of the Operational Research Society* **42**(8), 631–638.
- Tajima, M. (2007), 'Strategic value of RFID in supply chain management', *Journal of purchasing and supply management* **13**(4), 261–273.
- Tamhankar, A. and Ram, S. (1998), 'Database fragmentation and allocation : an integrated methodology and case study', *Transactions on Systems, Man and Cybernetics, Part A : Systems and Humans* **28**(3), 288–305.
- Tan, L. and Wang, N. (2010), Future internet : The internet of things, in '3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)', Vol. 5, pp. V5–376.
- Tasharofi, S. and Basseda, R. (2006), Near Neighborhood Allocation (NNA) : A novel dynamic data allocation algorithm in DDB, in '11th Computer Society of Iran Computer Conference' .

- Terzi, S., Bouras, A., Dutta, D. and Garetti, M. (2010), 'Product lifecycle management—from its history to its new role', *International Journal of Product Lifecycle Management* **4**(4), 360–389.
- Thomas, A. (2009), Rfid et nouvelles technologies de communication ; enjeux économiques incontournables et problèmes d'éthique, *in* '6ème Conférence Internationale Conception et Production Intégrées (CPI'2009)'.
- Timm, I., Woelk, P., Knirsch, P., Tönshoff, H. and Herzog, O. (2001), Flexible mass customisation : Managing its information logistics using adaptive cooperative multi agent systems, *in* 'Logistics and the Digital Economy. Proceedings of the 6th International Symposium on Logistics', pp. 227–232.
- Tolga, E., Demircan, M. L. and Kahraman, C. (2005), 'Operating system selection using fuzzy replacement analysis and analytic hierarchy process', *International Journal of Production Economics* **97**(1), 89–117.
- Triantaphyllou, E. and Lin, C. (1996), 'Development and evaluation of five fuzzy multi-attribute decision-making methods', *Approximate Reasoning* **14**(4), 281–310.
- Tursi, A., Panetto, H., Morel, G. and Dassisti, M. (2009), 'Ontological approach for products-centric information system interoperability in networked manufacturing enterprises', *Annual Reviews in Control* **33**(2), 238–245.
- Ulus, T. and Uysal, M. (2003), 'Heuristic approach to dynamic data allocation in distributed database systems', *Pakistan Journal of Information and Technology* **2**(3), 231–239.
- Usami, M., Tanabe, H., Sato, A., Sakama, I., Maki, Y., Iwamatsu, T., Ipposhi, T., Inoue, Y. and Hitachi, T. (2007), A 0.05×0.05 mm² RFID Chip with Easily Scaled-Down ID-Memory, *in* 'International Solid-State Circuits Conference', pp. 482–483.
- Uysal, M. and Ulus, T. (2007), 'A Threshold Based Dynamic Data Allocation Algorithm-A Markov Chain Model Approach', *Journal of Applied Sciences* **7**, 165–174.
- Vaidya, O. and Kumar, S. (2006), 'Analytic hierarchy process : An overview of applications', *European Journal of operational research* **169**(1), 1–29.
- Valckenaers, P. (1998), 'Special issue on intelligent manufacturing systems', *Computers in Industry* **37**(3).
- Valckenaers, P., Kollingbaum, M., Van Brussel, H. et al. (2004), 'Multi-agent coordination and control using stigmergy', *Computers in Industry* **53**(1), 75–96.
- Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L. and Peeters, P. (1998), 'Reference architecture for holonic manufacturing systems : PROSA', *Computers in Industry* **37**(3), 255–274.
- van Laarhoven, P. and Pedrycz, W. (1983), 'A fuzzy extension of saaty's priority theory', *Fuzzy Sets and Systems* **11**(1), 199–227.

- van Wesemael, S. (2005), 'L'ère du vide', *Rilune* **1**, 1–13.
- Ventä, O. (2007), 'Intelligent products and systems', *Technology Theme-Final Report. VTT, Espoo : VTT Publications* p. 304.
- Vildan Özkir, V. and Demirel, T. (2012), 'A fuzzy assessment framework to select among transportation investment projects in turkey', *Expert Systems with Applications* **39**(1), 74–80.
- Voisin, A. (1999), Contribution à l'évaluation subjective : du concept à l'application, PhD thesis, Université Nancy 1, Henri Poincaré.
- Wang, Y. and Lee, H. (2008), 'The revised method of ranking fuzzy numbers with an area between the centroid and original points', *Computers & Mathematics with Applications* **9**(55), 2033–2042.
- Wang, Y.-M. and Chin, K.-S. (2011), 'Fuzzy analytic hierarchy process : A logarithmic fuzzy preference programming methodology', *International Journal of Approximate Reasoning* **52**(4), 541–553.
- Weck, M., Klocke, F., Schell, H. and Rüenauer, E. (1997), 'Evaluating alternative production cycles using the extended fuzzy ahp method', *European Journal of Operational Research* **100**(2), 351–366.
- Weiser, M. (1991), 'The computer for the 21st century', *Scientific American* **265**(3), 94–104.
- Wiesmann, M., Pedone, F., Schiper, A., Kemme, B. and Alonso, G. (2000), Understanding replication in databases and distributed systems, in 'icdcs', pp. 464–487.
- Wikipedia (2003), 'Chord (geometry)'.
- Wong, C. Y., Mcfarlane, D., Zaharudin, A. A. and Agarwal, V. (2002), The Intelligent Product Driven Supply Chain, in 'International Conference on Systems, Man and Cybernetics', pp. 4–6.
- Wooldridge, M. and Jennings, N. (1995), 'Intelligent agents : Theory and practice', *Knowledge engineering review* **10**(2), 115–152.
- Wujuan, L. and Veeravalli, L. (2003), An adaptive object allocation and replication algorithm in distributed databases, in '23rd International Conference on Distributed Computing Systems Workshops', pp. 132–137.
- Xia, M. and Xu, Z. (2011), 'Methods for fuzzy complementary preference relations based on multiplicative consistency', *Computers & amp ; Industrial Engineering* **61**(4), 930–935.
- Yager, R. (1980), 'On a general class of fuzzy connectives', *Fuzzy sets and Systems* **4**(3), 235–242.
- Yang, M., Khan, F. I. and Sadiq, R. (2011), 'Prioritization of environmental issues in offshore oil and gas operations : A hybrid approach using fuzzy inference system and fuzzy analytic hierarchy process', *Process Safety and Environmental Protection* **89**(1), 22–34.
- Yazdani-Chamzini, A. and Yakhchali, S. H. (2012), 'Tunnel boring machine (tbn) selection using fuzzy multicriteria decision making methods', *Tunnelling and Underground Space Technology* .

- Yeh, C.-H., Deng, H. and Chang, Y.-H. (2000), 'Fuzzy multicriteria analysis for performance evaluation of bus companies fuzzy multicriteria analysis for performance evaluation of bus companies fuzzy multicriteria analysis for performance evaluation of bus companies', *European Journal of Operational Research* **126**(3), 459–473.
- Young, M. (2008), Extensible supply-chain discovery service concepts, Technical report, IETF Internet-Draft, Available at <http://www.ietf.org/internet-drafts/draft-young-esds-concepts-00.txt>.
- Zábovský, K. (2008), 'A data replication for mobile environment', *Journal of Electrical Engineering* **59**(5), 277–280.
- Zadeh, L. (1965), 'Fuzzy sets', *Information and Control* **8**, 338–353.
- Zadeh, L. (1971), 'Similarity relations and fuzzy ordering', *Information sciences* **3**(2), 177–200.
- Zaïdi, A. (1990), *QFD : Quality Function Deployment une introduction*, Technique et Documentation Lavoisier.
- Zheng, G., Zhu, N., Tian, Z., Chen, Y. and Sun, B. (2012), 'Application of a trapezoidal fuzzy ahp method for work safety evaluation and early warning rating of hot and humid environments', *Safety Science* **50**(2), 228–240.

Résumé

Depuis de nombreuses années, plusieurs communautés telles que IMS (*Intelligent Manufacturing Systems*), HMS (*Holonic Manufacturing System*) ont suggéré l'utilisation de produits intelligents pour rendre les systèmes adaptables et adaptatifs et ont montré les bénéfices pouvant être réalisés, tant au niveau économique, qu'au niveau de la traçabilité des produits, qu'au niveau du partage des informations ou encore de l'optimisation des procédés de fabrication. Cependant, un grand nombre de questions restent ouvertes comme la collecte des informations liées au produit, leur stockage à travers la chaîne logistique, ou encore la dissémination et la gestion de ces informations tout au long de leur cycle de vie.

La contribution de cette thèse consiste en la définition d'un cadre de dissémination des informations relatives au produit durant l'ensemble de son cycle de vie. Ce cadre de dissémination est associé à un nouveau paradigme qui change radicalement la manière de voir le produit et la matière. Ce nouveau concept consiste à donner la faculté au produit d'être intrinsèquement et intégralement « communicant ». Le cadre de dissémination des informations proposé offre la possibilité à l'utilisateur d'embarquer des informations sensibles au contexte d'utilisation du produit communicant.

Outre la définition du processus de dissémination des informations, cette thèse offre un aperçu des champs de recherche, tant scientifiques que technologiques, à investiguer à l'avenir concernant le concept de « matière communicante ».

Mots clés

Gestion du cycle de vie du produit, produits intelligents, dissémination des données, bases de données distribuées, Internet des objets, informatique ubiquitaire

Abstract

Over the last decade, communities involved with intelligent-manufacturing systems (IMS - *Intelligent Manufacturing Systems*, HMS - *Holonic Manufacturing System*) have demonstrated that systems that integrate intelligent products can be more efficient, flexible and adaptable. Intelligent products may prove to be beneficial economically, to deal with product traceability and information sharing along the product lifecycle. Nevertheless, there are still some open questions such as the specification of what information should be gathered, stored and distributed and how it should be managed during the lifecycle of the product.

The contribution of this thesis is to define a process for disseminating information related to the product over its lifecycle. This process is combined with a new paradigm, which changes drastically the way we view the material. This concept aims to give the ability for the material to be intrinsically and wholly « communicating ». The data dissemination process allow users to store context-sensitive information on communicating product.

In addition to the data dissemination process, this thesis gives insight into the technological and scientific research fields inherent to the concept of « communicating material », which remain to be explored.

Keywords

Product life cycle management, intelligent products, data dissemination, distributed database, Internet of things, ubiquitous computing