HAL
archives-ouvertes.fr

Methode d'analyse de donnees pour le diagnostic a posteriori de defauts de production - Application au secteur de la microelectronique

Hasna Yahyaoui

To cite this version:

Hasna Yahyaoui. Methode d'analyse de donnees pour le diagnostic a posteriori de defauts de production - Application au secteur de la microelectronique. Autre. Ecole Nationale Supérieure des Mines de Saint-Etienne, 2015. Français. <NNT : 2015EMSE0795>. <tel-01282255>

HAL Id: tel-01282255

https://tel.archives-ouvertes.fr/tel-01282255

Submitted on 3 Mar 2016

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.

NNT : *2015 EMSE 0795*

THÈSE

présentée par

Hasna BARKIA Ep YAHYAOUI

pour obtenir le grade de
Docteur de l'École Nationale Supérieure des Mines de Saint-Étienne

Spécialité : Informatique

Méthode d'analyse de données pour le diagnostic *a posteriori* de défauts de production

Application au secteur de la microélectronique

soutenue à Saint-Étienne, le 21 Octobre 2015

Membres du jury

Président :	D. A. ZIGHED	Professeur, Université Lyon II, Lyon
Rapporteurs :	P. LENCA	Professeur, Telecom Bretagne, Brest
	E. BONJOUR	Professeur, Université de Lorraine, Nancy
Examinateur(s) :	H. ELGHAZEL	Maître de conférences, Université Lyon 1, Lyon
Directeur(s) de thèse :	X. BOUCHER	Professeur, Ecole des Mines de St-Etienne, St-Etienne
	R. LE RICHE	Directeur de Recherche, CNRS, St-Etienne
	P. BEAUNE	Maitre-Assistant, Ecole des Mines de St-Etienne, St-Etienne
	H.DUVERNEUIL	Encadrant industriel, STMicroelectronics, Crolles
Invité(s) éventuel(s):	M.A.GIRARD	Maître-Assistant, Ecole des Mines de St-Etienne, St-Etienne
	D. ROZIER	Xerox Research Centre Europe, Meylan

à mes parents, Abderrazak et Nawel

à mon frère Aziz et ma sœur Rahma

à mon Cœur Wael

à tous ceux qui me sont chers

« Le savoir acquis en exil est une patrie et l'ignorance en patrie est un exil »

Ibn Rushd, Averroés 1126-1198

"Tous peuvent entendre mais seuls les êtres sensibles comprennent "

khalil Gibran

REMERCIEMENTS

Ces travaux de thèse ont été réalisés dans le cadre d'une Convention Industrielle de Formation à la Recherche en Entreprise (C.I.F.R.E.) avec l'institut Fayol de l'école des mines de Saint-Étienne (l'ENSMSE) et la société STMicroelectronics.

Je remercie Philippe Lenca Professeur à Telecom Bretagne, et Eric Bonjour Professeur à l'Université de Lorraine, d'avoir accepté de rapporter mes travaux et d'y avoir porté tant d'intérêt. Je tiens à remercier Djamel Abdelkader Zighed Professeur à l'Université Lyon II d'avoir accepté d'être président du jury. Je remercie également Haytham Elghazel Maître de conférences à l'Université Lyon I d'avoir accepté de faire partie de ce jury en tant qu'examinateur.

Je tiens à remercier mes tuteurs scientifiques, Xavier Boucher Professeur à l'école des mines de Saint-Étienne, Rodolphe Le Riche Chercheur au CNRS et à l'école des mines de Saint-Étienne, et Philippe Beaune Maître assistant à l'école des mines de Saint-Étienne pour m'avoir encadré et conseillé et pour le temps qu'ils ont consacré à diriger mes travaux. Je remercie, également, Marie Agnès Girard pour ses conseils tout au long de la thèse.

Je remercie Hugues Duverneuil, Responsable de la section IT – Automation et Advanced Process Control chez STMicroelectronics, pour m'avoir permis de continuer et terminer mes travaux de recherche. Je remercie David Rozier et Guillaume Chezaud, avec qui j'ai commencé cette thèse. Merci de m'avoir fait confiance et de m'avoir permis de commencer ces travaux de recherche ainsi que mon parcours professionnel.

Cette thèse n'aurait pas été aussi enrichissante sans la présence et l'expérience de certaines personnes. Je pense en particulier à François Pasqualini, Stéphane Hubac, Steddy Lagin et Philippe Vialletelle.

Enfin, je tiens à exprimer ma sympathie à tous ceux qui ont contribué de près ou de loin au bon déroulement de cette thèse, et tout particulièrement, toute l'équipe de l'Institut Fayol, les secrétaires et l'équipe informatique, ainsi que toute l'équipe de STMicroelectronics, le site de Crolles.

Merci à mes parents, Nawel et Abderazzak Barkia. Les plus formidables que l'on puisse rêver. Merci pour votre soutien inconditionnel, votre amour. Merci d'avoir toujours cru en moi. Merci à ma sœur Rahma et mon frère Aziz ainsi que toute la famille pour leur soutien constant et leur amour.

Je remercie mon mari Wael Yahyaoui pour son soutien et ses encouragements dans les instants les plus difficiles, qui m'a donné l'équilibre et la force nécessaire pour mener à bien mes recherches.

Nombreux sont ceux que je n'ai pas cités, qu'ils m'en excusent et trouvent toute ma reconnaissance dans ces quelques lignes.

Merci à toutes et à tous !

SOMMAIRE

LISTE DES FIGURES

LISTE DES TABLEAUX

LISTE DES ALGORITHMES

ABREVIATIONS

Symbole	Signification
{}	un ensemble d'éléments
MAJUSCULE	un ensemble d'éléments
DoE	Design of Experiments
#	nombre de (utilisé dans les tables de contingence)
CLARIF	*CL*ustering and *A*ssociation *R*ules *I*dentiFier
ARCI	*Association Rules based on Clusters Identifier*
P	L'ensemble des étapes de production. Une étape de production est notée p
Q	L'ensemble des étapes de contrôle qualité. Une étape de contrôle qualité est notée q
e	une étape
EC	ensemble d'étapes causes
EE	ensemble d'étapes effets
Y	le phénomène à expliquer = l'ensemble des produits à expliquer
$\bar{Y}$	Ensemble de produits complémentaires à Y, représente tout phénomène autre que Y = l'ensemble des produits à ne pas expliquer
Y^e	Y relatif à une étape e
$\bar{Y}^e$	$\bar{Y}$ relatif à une étape e
OK	un ensemble de produits de bonne qualité, respectant les limites de contrôles
OOC	« *Out Of Control* » un ensemble de produits problématiques, ne respectant pas les limites de contrôles
D^e	Les données collectées lors d'une étape e
v_i	la variable d'indice i des données D^e, la $i^{éme}$ colonnes de D^e.
W^e	L'ensemble des produits qui ont connu une étape e
M	Ensemble des produits *Mesuré* aux étapes Q Plus généralement, M est l'ensemble des produits qui ont connu les étapes effets *EE*
$partition_M$	Partition de M en sous-ensembles de produits Y et $\bar{Y}$. Cette partition est basée sur l'expertise. C'est un vecteur de taille $\|M\|$ dont chaque composante est le label du sous-ensemble auquel un produit $w \in M$ appartient.
$partition_M(w)$	label du sous-ensemble (ou groupe) du produit w au sein de la $partition_M$
$partition'_M$	La partition détaillée des produits de M en sous-ensembles $\{y\}$ et $\bar{Y}$. Elle est obtenue en divisant $partition_M$ par analyse des données. C'est un vecteur de taille $\|M\|$ dont chaque composante est le nouveau label du sous-ensemble auquel un produit $w \in M$ appartient.

$partition^x$	La partition qui contient le sous-ensemble de produits, ou mode, *x*.
$< e, k_k m_x >$	le sous-ensemble de produits labélisés *x*, aussi appelé *mode descriptif* labélisé *x*, généré à l'étape *e* sur les données D^e par partition en *k* modes.
$< e, t, k_k m_x >$	le mode descriptif labélisé *x*, généré sur la machine *t* à l'étape *e* par partition en *k* modes.
mode descriptif	une des parties (un des sous-groupes) d'une partition des produits qui, dans notre contexte, sert à construire des règles d'associations. Ces partitions sont réalisées aux différentes étapes du segment analysé (*EC* et *EE*)
cluster, groupe	synonymes de mode descriptifs
itemset	un ensemble d'items, par exemple dans notre cas, un ensemble de modes
k	Sauf indication contraire, *k* est le nombre de modes à distinguer à partir des données à travers l'application d'une méthode de *clustering*
$\|x\|$	Cardinalité de l'ensemble *x*
$r1 \succ r2$	La règle *r1* domine la règle *r2* au sens de Pareto
weight()	un vecteur de pondérations réelles et positives ou nulles de taille $\|W^e\|$. Utilisé pour l'induction d'arbres de décision avec des classes minoritaires
Tools	L'ensemble des machines traitant les plaques analysées
$Tools^e$	Le sous-ensemble des machines traitant les plaques analysées à l'étape *e*
S^e	espace des mesures de l'étape $e \in EE$. Par exemple, si *e* consiste en *3* mesures de nombres réels, $S^e = \mathbb{R}^3$.
e(w)	vecteur des mesures du produit w à l'étape e. $e(w) \in S^e$
label(w)	En induction d'arbre de décision, label(w) renvoie le label de la classe à laquelle *w* appartient.
r(x→y)	Une règle qui associe deux modes descriptifs, l'un venant des étapes causes *EC*, noté *x*, l'autre mode étant associé à *EE*, *i.e.*, *y*. Dans ce cas, nous noterons la règle *r(x→y)*, dont le sens est "les produits de *y* étaient dans *x*"
k_x	complexité de la règle *r(x→y)*
T	données regroupant l'ensemble des modes associés à chaque produit $w \in M$

PUBLICATIONS EFFECTUEES DURANT LA THESE

Semiconductor Yield Loss' Causes Identification: A Data Mining Approach.
Hasna Barkia, Xavier Boucher, Rodolphe Le Riche, Marie-Agnès Girard, Philippe Beaune.
Proceedings of the 2013 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM2013), article no. IEEM13-P-0503- *2013 IEEE International Conference on Industrial Engineering and Engineering Management* ***(IEEM2013),*** December 2013, Bangkok, Thailand.

Extraction de connaissances pour une maîtrise du rendement en industrie du semi-conducteur

Barkia H., Boucher X., Le Riche R., Girard M.-A., Beaune P., Rozier D. Dans **Actes du 10ième Congrès International en Génie Industriel (CIGI 2013)** - *10ième Congrès International en Génie Industriel (CIGI 2013)*, La Rochelle : France (2013)

Proposition d'une méthode hybride d'analyse de donnée pour remonter aux origines d'un défaut
Hasna BARKIA-YAHYAOUI
Présentation durant les 20èmes journées STP du GdR MACS, le 6 février 2015 à Troyes, France.

Local quality loss causes identification in the semiconductor industries
BARKIA-YAHYAOUI Hasna, Xavier BOUCHER, Rodolphe LE RICHE, Hugues DUVERNEUIL.
Présentation de poster à la journée de la recherche EDSIS à l'ENSISE, le 12 juin 2014 Saint-Etienne, France.

A Data Mining Approach for Yield Loss' causes Identification in Semi-conductor Industry.
Hasna Barkia, Xavier Boucher.
Abstract presentation on 13th Annual Conference of ENBIS, Sep 2013, Ankara, Turkey. 2013.

Introduction générale

Un des défis auquel se confronte l'industrie du semi-conducteur est l'évolution drastique des investissements et des coûts de production [1]. Afin de rentabiliser ces investissements, il est important d'améliorer l'efficacité du processus de fabrication. Pour cela, en industrie du semi-conducteur, un processus de contrôle est appliqué en parallèle au processus de production global. Un processus de contrôle est un ensemble d'étapes de contrôle qualité effectuées après des étapes de production critiques, sur un échantillon des plaques produites.

Cette thèse CIFRE est le résultat d'une collaboration entre *l'Institut Fayol* de l'Ecole des Mines de Saint-Étienne et STMicroelectronics, le site de fabrication de puces microélectroniques *Crolles 300mm*. Afin de garantir des mesures de bonne qualité, les sites de production, tels que *STMicroelectronics Crolles 300mm*, continuent de plus en plus à améliorer les équipements de mesures et d'inspections pour améliorer la sensibilité et le débit des données remontées [2]. Cette grande masse de données collectées est remontée dans de larges bases de données, caractérisant l'état des plaques à différentes étapes de contrôle qualité. En plus de ces données, des systèmes, tels que la FDC ("Fault Detection and Classification") utilisés pour garantir le bon fonctionnement des équipements de production, remontent des quantités importantes de données relatives aux conditions de fonctionnement des équipements de production grâce aux différents capteurs intégrés dans une grande majorité des équipements de production.

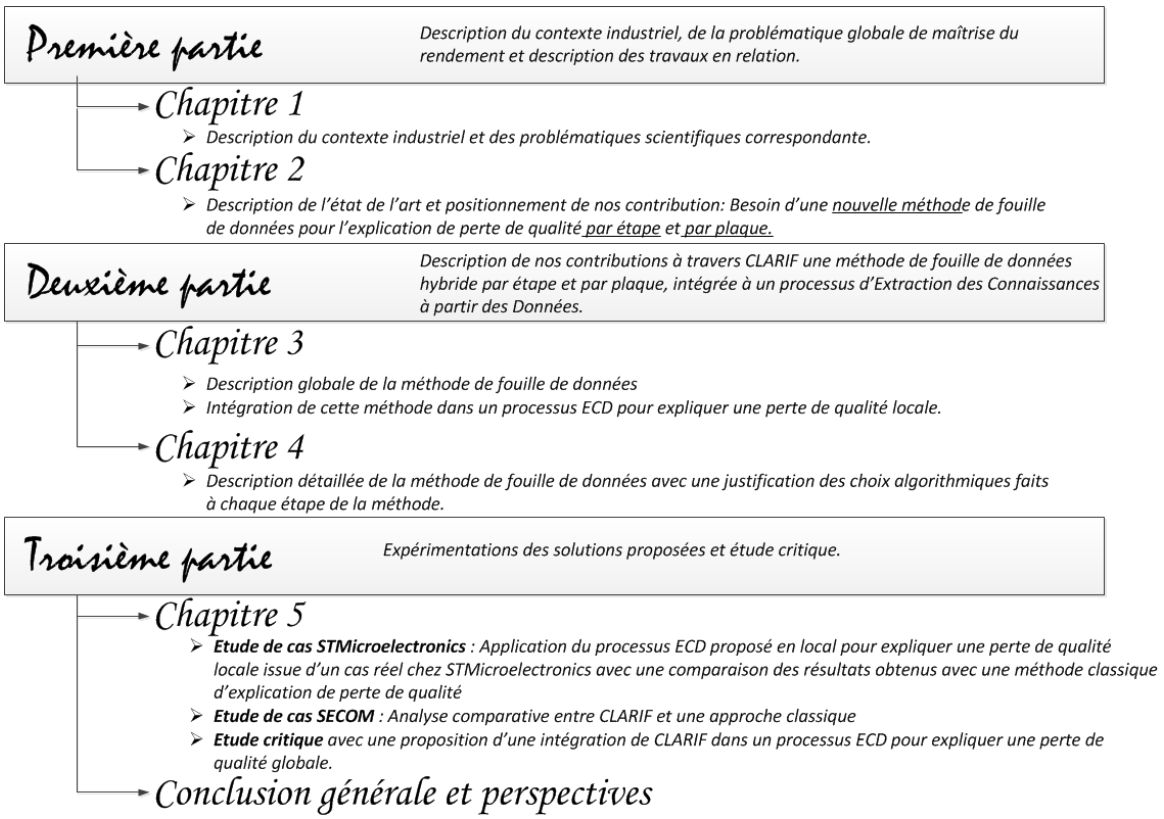
La quantité importante et la diversité des données collectées fait du système d'information des industries du semi-conducteur un des plus riches [3]. La gestion, l'exploration et la valorisation de ces collections de données reste une question clé, à *STMicroelectronics Crolles 300mm*. Plusieurs travaux ont déjà été réalisés, notamment des travaux de thèses. Nous citons, à titre d'exemple, les travaux de *Cyril Alegret* qui se sont intéressés au développement de méthodes statistiques de corrélation entre les mesures électriques et physiques et les étapes de fabrication, ceux de *Ali Hajj Hassan* qui se sont intéressés à la détection multidimensionnelle au test paramétrique (PT) ou encore ceux de la thèse de Michel Lutz pour l'étude de la gestion du système d'information d'un site de fabrication micro-électronique.

L'amélioration de rendement, en industrie du semi-conducteur, est devenue un des objectifs majeurs, car même une légère amélioration induit un important avantage financier. Une des méthodes d'amélioration de rendement, est l'identification rapide des causes de perte de qualité, afin de pouvoir appliquer, rapidement, des actions correctrices, permettant l'amélioration du processus de fabrication futur. Le travail élaboré durant cette thèse s'inscrit

dans cette perspective. Pour améliorer le rendement, on s'intéresse à expliquer des cas de perte de qualité.

L'objectif est de proposer une nouvelle approche, basée sur des techniques de fouille de données, pour l'identification des causes explicatives de perte de qualité, dans un objectif d'amélioration globale de la performance des procédés de production. On s'intéresse à combiner des données complémentaires issues de différents systèmes de contrôle, à savoir : les données de production, collectées à travers le système *FDC* et décrivant l'état des équipements lors du traitement des plaques, ainsi que les données de mesures qualité, décrivant les résultats de mesures des étapes de contrôle qualité, telles que les données collectées à travers les systèmes de *métrologie, PT,* et/ou *EWS*. On cherche à identifier des causes consistantes décrivant des conditions de production potentiellement responsables des plaques de mauvaise qualité.

Ainsi, la question centrale de cette thèse vient adresser cette préoccupation: « *Comment expliquer rapidement une nouvelle perte de qualité (dont la cause est inconnue), en combinant les données issues des différents systèmes de contrôle* ». Autrement dit, « *Comment identifier des patterns de disfonctionnements d'équipements, responsables d'une perte de qualité* ». Autour de cette question, ce manuscrit est organisé en cinq chapitres, groupés en trois grandes parties, illustrées ci-dessous.



Dans la première partie, et à travers le premier chapitre, nous donnons une description globale du contexte industriel, de la problématique globale de maîtrise du rendement et des problématiques scientifiques correspondantes. Dans le deuxième chapitre, nous donnons une description de l'état de l'art des travaux existant en relation avec cette problématique, avec un positionnement de notre principale contribution, à savoir *« la définition d'une nouvelle méthode de fouille de données pour l'explication d'une perte de qualité* ». Pour clore ce deuxième chapitre, nous donnerons une introduction aux méthodes de fouille de données qui seront utilisées par la méthode proposée.

La deuxième partie de ce manuscrit sera consacrée à la présentation des principales contributions de ces travaux de thèse. Ainsi, dans le troisième chapitre, nous proposons une généralisation de la *problématique d'explication de perte de qualité détectée à des étapes de contrôle qualité par des causes concernant des étapes de production* par la problématique *d'explication d'un phénomène Y, détecté à une ou plusieurs étapes effet, notées EE, par des conditions particulières sur des étapes causes EC*. Pour cela, nous proposons *CLARIF* pour *CL*ustering and *A*ssociation *R*ules *I*denti*F*ier, une méthode hybride, combinant différents techniques de fouille de données, à savoir du *clustering*, de la recherche de *règles d'associations* et d'*induction d'arbre de décision*. Par la suite, nous proposons une définition d'un processus d'*Extraction des Connaissances à partir des Données,* utilisant *CLARIF* pour la remontée aux causes d'une perte de qualité locale. Dans le quatrième chapitre, nous donnons une présentation détaillée de *CLARIF* en détaillant le fonctionnement, et en justifiant les choix des méthodes de fouille pour chaque phase de la démarche. Par ailleurs, nous définissons la méthode *ARCI*, une adaptation de l'algorithme *APRIORI* de fouille de règle d'association pour prendre en comptes les contraintes liées à notre problématique.

La dernière partie de ce manuscrit de thèse est consacrée aux expérimentations mettant en application les contributions méthodologiques pour expliquer des cas de perte de qualité sur un cas issue du site de fabrication de *STMicroelectronics,* ainsi que pour expliquer un cas d'étude issu d'un autre site de fabrication de semi-conducteur dont les données sont disponibles sur le répertoire d'apprentissage automatique *UCI Machine Learning Repository*. Pour clore le cinquième chapitre, nous donnons une étude critique du processus *ECD* proposé et nous proposons une extension, à travers une définition d'un processus *ECD* qui permet d'expliquer des pertes de qualité plus globales, à travers une analyse récursive par étape. Finalement, nous proposons une conclusion générale à ce manuscrit avec une présentation des perspectives ouvertes à ce travail.

Chapitre 1 : Contexte industriel et problématiques

Ce premier chapitre représente une entrée à ce manuscrit de thèse permettant de présenter de manière générale le domaine du semi-conducteur ainsi que la problématique traitée dans le cadre de cette thèse. Pour cela nous commençons par décrire globalement l'industrie du semi-conducteur, avec un focus sur notre partenaire *STMicroelectronics* le site de fabrication de *Crolles 300mm*, ainsi que les secteurs d'intégration des produits issus de cette industrie. Par la suite, nous présentons le processus de fabrication correspondant, ainsi que l'environnement de production, afin d'introduite plus tard les enjeux auxquels les sites de fabrication tel que celui de *STMicroelectronics Crolles 300mm* sont confrontés. Pour finir cette première partie, nous décrivons les outils industriels adoptés pour faire face à ces enjeux.

Dans une deuxième section, nous décrivons la problématique dans laquelle ces travaux s'intègrent, avec un bref positionnement de nos travaux par rapport à l'existant. Et à partir de la description des données disponibles dans le système d'information d'un site de fabrication, nous présentons les défis scientifiques et industriels implicites que nous devons traiter.

Pour clore ce chapitre, nous proposons de mettre en évidence les éléments forts de la réponse apportée par cette thèse, avec une description des contributions majeures qu'apporte ce travail, sur les deux plans scientifique et industriel.

TABLE DES MATIERES

1.1 L'INDUSTRIE DES SEMI-CONDUCTEURS

1.1.1 Description générale de l'industrie du semi-conducteur

L'industrie des semi-conducteurs[1] est un secteur industriel qui regroupe les activités de conception, de fabrication et de commercialisation des composants électroniques et des Circuits Intégrés (*Figure 1.2*). Un composant électronique est un système semi-conducteur qui n'exécute qu'une seule fonction électronique élémentaire. Par exemple, un transistor est un composant à 3 bornes, à l'intérieur duquel le courant, circulant entre 2 bornes, est commandé par l'application du courant à la 3éme borne. Il est généralement utilisé pour stabiliser une tension, moduler un signal… Connectés entre eux sur une même plaque de silicium[2], les composants électroniques, tels que les transistors, les diodes, les résistances et/ou de condensateurs, constituent un circuit intégré, noté *CI*, permettant de réaliser des fonctions complexes. Un *CI* est composé de deux parties principales : une partie carrée en silicium de quelques millimètres de côté, fragile et mince, appelé la *puce*. Et une autre partie représentant un boitier de protection pour la puce de son environnement. Ce boitier est muni de pattes qui assurent la connexion dans les applications des systèmes électroniques.

Ces circuits intégrés sont présents dans quasiment tous les objets de notre quotidien. On estime que chaque personne utilise, chaque jour, environ 250 circuits électroniques (*Figure 1.1*) : de nos téléphones portables, à nos voitures en passant par les outils de divertissement comme les télévisions et les stations de jeu.

Figure 1.1 : Les domaines d'intégration des CI

[1] Le semi-conducteur est un corps cristallin dont les propriétés de conductibilité électrique sont intermédiaires entre celle des métaux et celle des isolants, et sont accentuées par des opérations de dopage.

[2] Le *silicium* est un élément du groupe IV de symbole *Si* et de numéro atomique 14, il est utilisé pour fabriquer des diodes, des transistors et d'autres circuits intégrés.

Traditionnellement, les sociétés de fabrication de circuits intégrés possèdent leurs propres usines de fabrication, généralement appelés *fab* en référence au terme anglais « *fabrication plant* ». Ces sociétés font les trois activités de conception, fabrication et de commercialisation des circuits intégrés. Ces sociétés sont appelées des sociétés *IDM (Integrated Device Manufacturer).*

Figure 1.2 : Les étapes principales de création de Circuits Intégrés

De nos jours, et dans un objectif de faire face aux coûts continuellement en augmentation de ces *fab*, on voit apparaître de nouveaux types de sociétés de *CI* avec, d'un côté des sociétés spécialisées dans la conception et la vente, dites des sociétés *fabless* et, d'un autre côté, des sociétés spécialisées dans la fabrication des *CI* sur des plaques, appelées *fonderie* (*Foundry* en anglais). Les sociétés *fabless* n'ont pas d'usine de fabrication, mais sous-traitent la fabrication de leurs produits aux entreprises *fonderie*.

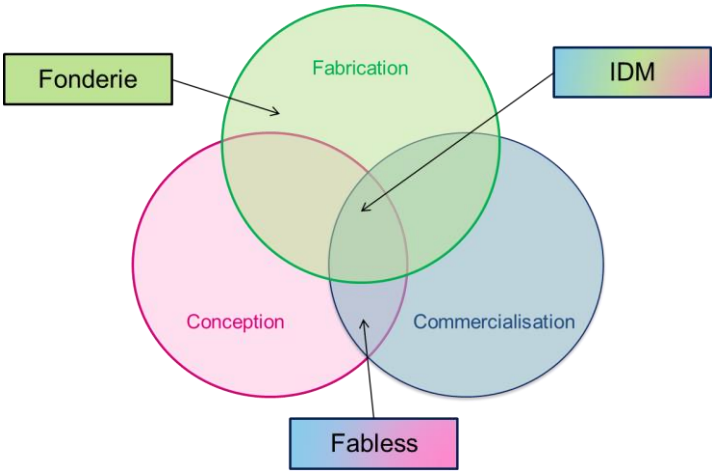


Figure 1.3 : Les trois grands types de sociétés de semi-conducteur

STMicroelectronics est une société *IDM*, puisqu'elle conçoit développe, fabrique et commercialise des composants électroniques et des circuits intégrés. *ST* dispose de 15 sites de fabrication dans le monde, 16 centres de recherche et développement *R&D*, 39 centres de conception et d'applications et 78 bureaux de vente directe dans 36 pays dans le monde. *ST* est l'un des principaux fabricants de semi-conducteurs au monde. Il est classé 10éme au monde et premier en Europe (*Figure 1.4*).

2014F Top 20 Semiconductor Sales Leaders ($M)

2014F Rank	2013 Rank	Company	Headquarters	2013 Total	2014 Total	2014/2013 % Change
1	1	Intel	U.S.	48,321	51,368	6%
2	2	Samsung	South Korea	34,378	37,259	8%
3	3	TSMC*	Taiwan	19,935	25,088	26%
4	4	Qualcomm**	U.S.	17,211	19,100	11%
5	5	Micron + Elpida	U.S.	14,294	16,614	16%
6	6	SK Hynix	South Korea	12,970	15,838	22%
7	8	TI	U.S.	11,474	12,179	6%
8	7	Toshiba	Japan	11,958	11,216	-6%
9	9	Broadcom**	U.S.	8,219	8,360	2%
10	10	ST	Europe	8,014	7,374	-8%
11	11	Renesas	Japan	7,975	7,372	-8%
12	12	MediaTek + MStar**	Taiwan	5,723	7,142	25%
13	14	Infineon	Europe	5,260	6,151	17%
14	16	NXP	Europe	4,815	5,625	17%
15	13	AMD**	U.S.	5,299	5,512	4%
16	17	Sony	Japan	4,739	5,192	10%
17	15	Avago + LSI**	Singapore	4,979	5,087	2%
18	19	Freescale	U.S.	3,977	4,548	14%
19	20	UMC*	Taiwan	3,940	4,300	9%
20	21	Nvidia**	U.S.	3,898	4,237	9%
Top 20 Suppliers				237,379	259,562	9%
Top 20 Suppliers Excluding Foundries				213,504	230,174	8%

*Foundry **Fabless

Source: IC Insights' Strategic Reviews Database

Figure 1.4 : Classement 2014 des Top 20 des ventes en semi-conducteur

Dans le prochain paragraphe, nous donnons une description du processus de fabrication des *Circuits Intégrés* dans l'industrie du semi-conducteur, plus précisément chez notre partenaire *STMicroeletronics*, le site de fabrication de *Crolles 2*, aussi connu sous le nom de *Crolles 300mn*, en référence au diamètre des plaques en production sur ses lignes. Cette usine, construite en 2002 pour un investissement de 2,2 Milliard d'euros, dispose d'un parc de 300 équipements, et a une capacité de production de 3500 plaquettes par semaine, avec une finesse de gravure entre 14 nm et 90 nm.

1.1.2 Processus de fabrication des circuits intégrés

Nous nous intéressons au processus de transformation d'une plaque de silicium en Circuits Intégrés. Les *CI* produits de nos jours sont composés de plusieurs millions de composantes élémentaires (transistors, résistances…). Ainsi, et comme illustré dans la *Figure 1.5*, deux sous-processus de fabrication peuvent être distingués. D'un côté, on a le processus "*Front-End*", qu'on notera "*FE*". Durant ce premier processus, des plaques de silicium vierges sont traitées pour créer des puces individuelles sur chacune d'elles. Après ce premier processus composé d'une centaine d'opérations et qui dure environ deux mois de production, vient le deuxième sous processus "*Back-End*", qu'on notera "*BE*". Durant ce processus, les plaques traitées en "*FE*" sont découpées en puces individuelles, afin d'être assemblées et emballées pour construire les produits finaux.

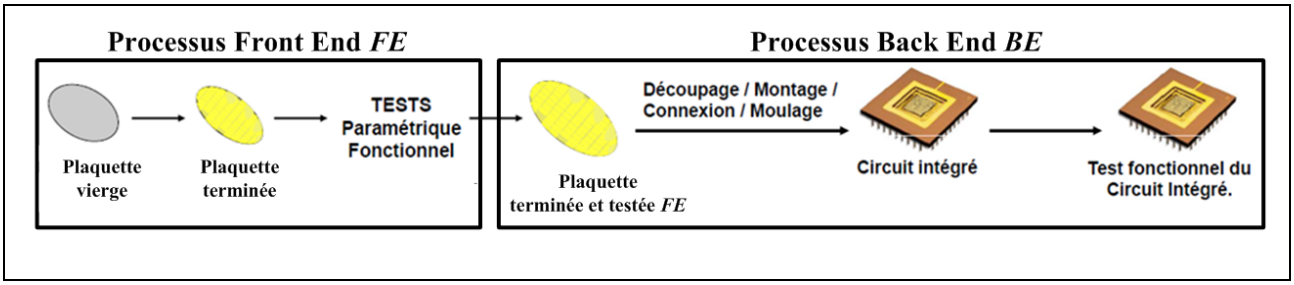


Figure 1.5 : Représentation du processus global de transformation d'une plaquette de silicium en un circuit intégré

Dans un environnement compétitif et avec des domaines d'application sensibles tel quel l'automobile, la qualité et la fiabilité d'un produit en sortie sont primordiales pour un client. Ces tests paramétriques et fonctionnels en fin du processus *FE*, ainsi qu'aux tests fonctionnels des circuits intégrés *CI* en fin du processus de fabrication *BE*, permettent d'assurer qu'uniquement les produits sans défauts sont livrés, et ainsi d'éliminer les produits défectueux.

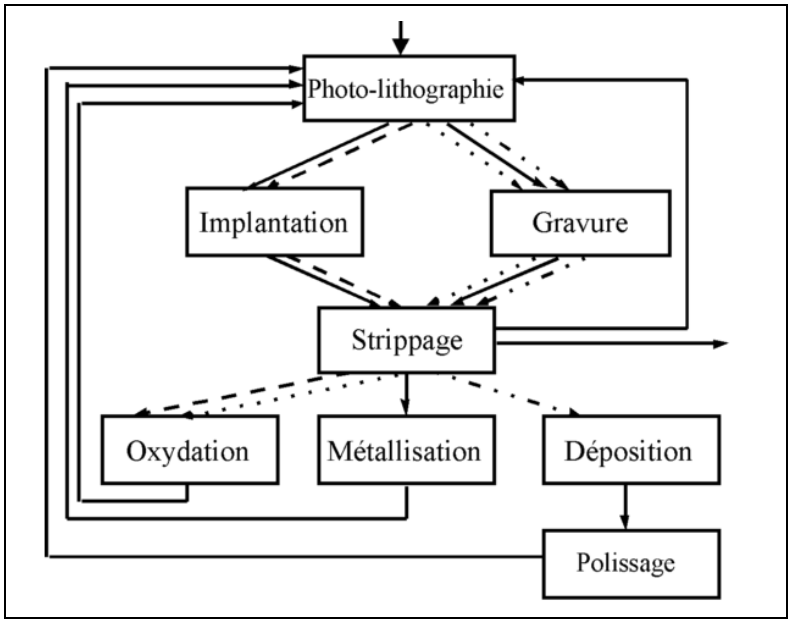


Figure 1.6 : Séquence des étapes de la fabrication des wafers [4]

Sur le site de fabrication de *Crolles 300mm*, est fait le processus *FE*. Durant ce processus, et pour arriver à une puce électronique qui fonctionne, il y a de nombreuses étapes intermédiaires. Le point de départ est une plaque de silicium, sur laquelle sont appliquées diverses opérations (*Figure 1.*6), comme la lithographie, la gravure, l'implantation, la métallisation, le polissage … Ces opérations sont répétées plusieurs fois selon le processus de fabrication du produit final.

Les plaques sont fabriquées par lots qui contiennent généralement 25 plaques, regroupées dans un *foup (Figure 1.7)*. Au bout de 500 opérations environ, on arrive à une plaque finie qui contient des milliers de puces prêtes à être découpées.

Figure 1.7 : Un foup en 300mm

Comme représenté dans *Figure 1.8*, le processus *FE* est lui aussi décomposé en deux sous processus, notés *FEOL* et *BEOL*. La première partie *FEOL*, pour *Front End Of the Line*, permet la création des composants élémentaires du circuit intégré. Par la suite durant la deuxième partie *BEOL*, pour *Back End Of the Line*, des interconnexions sont créées entre ces composants, notamment à travers des étapes de métallisation.

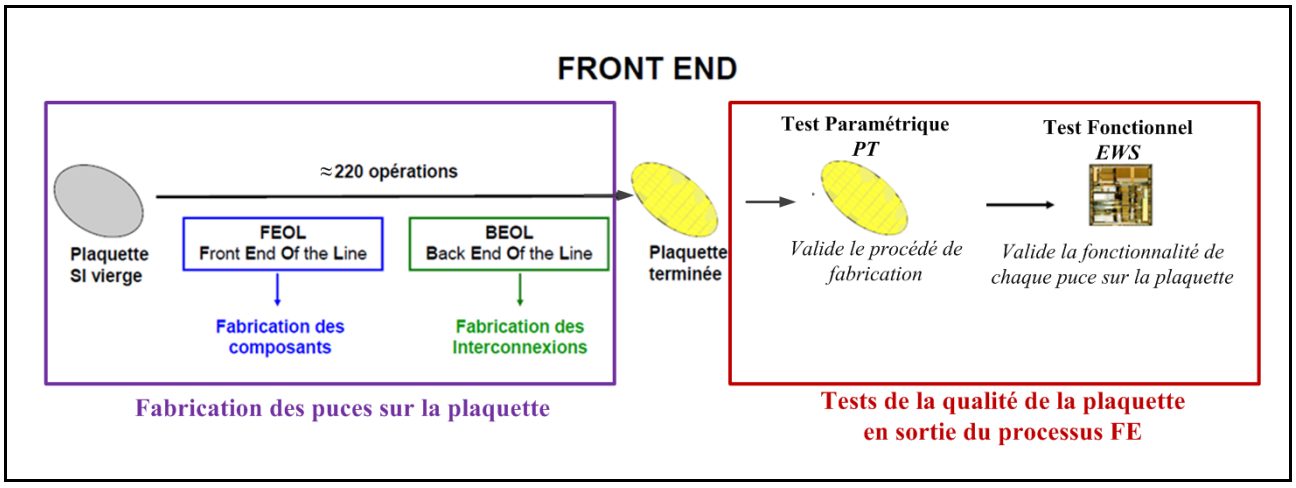


Figure 1.8 : Décomposition du processus de production Front End FE

Durant ces deux sous processus, une plaque subit plusieurs allers et retours dans les différents ateliers avant d'arriver à une plaque finie composée de plusieurs circuits intégrés. Afin d'éviter des contaminations par particules, ces différents ateliers sont intégrés dans un environnement où le nombre de particules par m^3 est maîtrisé, appelé *une salle blanche* et les plaques d'un même lot transportées dans un *foup* (*Figure 1.7*).

Pour le site de *Crolles 300mm*, la salle blanche est de taille 10 000m^2 et est classée, selon la norme *ISO 14644-1* en classe *ISO 4* alors que les *foups* sont classés en classe *ISO 2* (moins de 4 particules de plus d'1 micromètre de diamètre au pied cube). Pour assurer un tel niveau de propreté, différents facteurs sont à maîtriser, notamment, la température et l'humidité de la salle, ainsi que la tenue des opérateurs. On distingue huit ateliers de production sur le site de fabrication *Crolles 300mm*.

- **L'atelier de traitement thermique *TT* :** Deux grands types d'activités sont réalisés dans cet atelier, à savoir (1) *la fabrication de couches* d'isolants ou de semi-conducteur et (2) *la réalisation de recuits.* La fabrication de couches est faite par exemple par oxydation sèche, qui consiste à faire croître, à très haute température (900-1000°C), une couche d'Oxyde de Silicium à partir du Silicium de la plaque. Par ailleurs, les recuits servent à activer les dopants et à corriger les défauts introduits dans le silicium après les étapes d'implantation ionique.

- **L'atelier *WET* :** Dans cet atelier, on réalise *le nettoyage* de la surface des plaques afin d'éviter toute contamination pouvant affecter les circuits créés et/ou les machines des étapes de production suivantes. Par ailleurs, cet atelier est aussi utilisé pour réaliser des opérations de *gravure humide*. Ce type de gravure est dit sélectif et isotrope, *i.e.* il grave de la même façon dans toutes les directions. Dû à cette propriété, l'objectif de la gravure humide est l'élimination uniforme d'une couche entière sur une plaque, *i.e.*, sans utilisation préalable de masque, parfois sur des plaques de production, mais surtout pour le recyclage des plaques *NPW*[3].

- **L'atelier de photolithographie *LITHO* :** Cet atelier représente le cœur de la *fab.* Il a comme objectif de *dessiner des motifs*, afin de définir les zones sur la plaque à traiter durant les étapes suivantes, typiquement de gravure ou d'implantation ionique. Ceci est réalisé en transférant des motifs géométriques d'un masque dans une résine. Cette opération est faite en trois étapes chronologiques : (1) l'étalement d'une résine photosensible ; (2) l'exposition de la résine ; (3) le développement de la résine. Cette dernière étape permet de dissoudre la partie exposée, laissant apparaitre la sous couche correspondante à la zone qui sera traitée par la suite, alors que la zone cachée par la résine, non insolée, ne sera pas traitée.

- **L'atelier de gravure *ETCH* :** Dans cet atelier, deux activités sont faites : la *gravure sèche* et *l'élimination de résine*, cette dernière connue sous le nom de *Stripping*. La *gravure sèche* est celle utilisée après l'étape de photolithographie, permettant ainsi de graver les motifs dessinés par la résine. Par ailleurs, le *stripping*, consiste à enlever la résine restante sur les zones non insolées, après le traitement, par gravure ou implantation ionique, des zones insolées.

- **L'atelier d'implantation ionique *IMPLANT* :** Cet atelier a comme objectif de *doper* le silicium en lui apportant des électrons de type N ou des trous supplémentaires de type P.

[3] NPW *(Non Productive Wafers*) aussi connues sous le nom de *plaques témoins*, elles circulent dans la *fab* avec les plaques de production. Elles servent à contrôler, entre autre, l'état des équipements. Elles peuvent aussi être utilisées pour compléter un batch dans certains équipements, et au conditionnement ou préchauffage dans d'autres cas.

Ceci est réalisé en bombardant la surface de la plaque avec un faisceau d'ions, permettant, ainsi, d'améliorer la conductivité. Pour cela, une étape d'IMPLANT est toujours précédée par une étape de photolithographie afin de définir les zones à doper et celles à protéger.

- **L'atelier Diélectrique *DIEL :*** Cet atelier a comme objectif de *déposer des couches diélectriques*, *i.e.* d'isolants électriques, sur la surface de la plaque permettant *(1)* l'isolation électrique des différentes zones actives durant le processus *FEOL*, *(2)* celle des différentes bandes métalliques en *BEOL* et finalement, *(3)* en fin du processus de fabrication du CI, en le protégeant de son environnement externe.

- **L'atelier de polissage *CMP (Chemical Mechanical Polishing)* :** Dans cet atelier, l'objectif est de *polir la surface de la plaque*, les couches d'oxyde et/ou de métaux déposés, permettant la réduction de l'épaisseur, ainsi que la planarisation des couches. En semi-conducteur le polissage adopté est mécano chimique, *i.e.* utilisant l'action combinée de forces mécaniques et chimiques.

- **L'atelier *Métal* :** Il permet de *déposer des couches conductrices* à la surface de la plaque. Ces couches sont de trois types *(1)* les *contacts* permettant l'accès aux composants, *(2)* Les *lignes métalliques et Vias* pour relier les composants entre eux, et *(3)* les *pads* aussi appelés *plots* pour connecter le *CI* à son boitier durant le processus *BE*.

Finalement, les plaques terminées en sortie du processus *BEOL*, contenant plusieurs milliers de puces, sont testées et contrôlées individuellement avant d'être envoyées à un autre site de fabrication pour la découpe et la mise en boitier, *i.e.* pour le processus *BE*. Ce contrôle est réalisé en deux étapes. Premièrement, à travers les tests paramétriques, notés *PT*, et deuxièmement à travers le test fonctionnel des puces, noté *EWS*. Une description de ces deux ateliers est donnée dans ce qui suit, compte tenu du lien à la suite de nos travaux.

- **L'atelier de test paramétrique *PT* :** Dans cet atelier, chaque plaque est mesurée en neufs sites. Ces neufs sites sont généralement répartis comme suit : un site au centre de la plaque, et huit autres répartis sur la plaque avec au moins cinq sites à proximité du bord de la plaque. Ces sites sont situés sur les zones de découpes d'une plaque. Les zones de découpes ont une double fonction : elles permettent d'un côté de délimiter les puces pour la découpe ultérieure, en début du processus *BE* et d'un autre côté de fournir des indicateurs de qualité des puces qu'elles entourent.

- **L'atelier de tri électrique des plaques *EWS* :** Cet atelier est le dernier par lequel la plaque passe avant de finir son processus *FE*. L'objectif est de tester la fonctionnalité de chaque puce présente sur chaque plaque, de chaque lot, afin de vérifier sa conformité par rapport aux spécifications demandées. Cette étape est connue sous le nom du tri électrique des plaques *EWS (Electrical Wafer Sorting)*. Les puces à défauts sont marquées afin d'être mises en rebut (*scapées*). Grâce à ces tests, on mesure, pour chaque plaque, son indicateur de rendement, qui est le pourcentage de puces bonnes sur celle-ci, appelé *Die yield*. Selon,

un seuil minimum de rendement, les plaques sont sélectionnées pour être découpées en puces individuelles, pour construire les produits finaux, durant le processus *BE*.

1.1.3 Enjeux du site de fabrication *ST Crolles 300mm*

L'évolution de nos besoins en produits électroniques de plus en plus compacts, de plus en plus performants et de moins en moins chers résulte dans un système de production en industrie microélectronique caractérisé par des rythmes de changements et de transformations très élevés, classiquement illustrés par la loi de Moore [5]. Gordon Moore avait postulé que la complexité des semi-conducteurs proposés en entrée de gamme doublait tous les ans à coût constant depuis 1959. Vérifiant cette loi, les machines électroniques sont devenues de plus en plus puissantes et de moins en moins coûteuses.

L'évolution de la complexité des semi-conducteurs est représentée par l'évolution des nœuds technologiques. Celui-ci est utilisé en industrie du semi-conducteur pour identifier la largeur de grille du transistor le plus fin présent sur la puce. Actuellement, on est en transition vers une technologie en 14 nanomètres ($14 * 10^{-9}$ mètres).

→ *Cette course à la finesse de gravure représente une des contraintes majeures du secteur du semi-conducteur.*

Par ailleurs, ce rythme élevé d'évolution de produits et de technologies de production résulte en un processus de fabrication des *CI* très complexe, caractérisé par :

- un *temps de cycle important* (environ 2 mois),
- un *nombre important d'opérations* élémentaires (environ 700 en moyenne).
- des *équipements* de production, de contrôle et de transport *de plus en plus onéreux*. Par exemple, un équipement de lithographie coûte plusieurs dizaines de millions d'euros.
- des lignes de fabrication caractérisées par une *forte variété des produits* induisant une forte variabilité des *techniques de production*, associée à de *faibles volumes*. Ce problème est connu sous le nom « *High Mix, Low Volume problem* ».
- des produits de très *courte durée* (environ 2 ans) résultant en *une forte rentabilité mais de très courte durée* (entre 6 et 12 semaines) [6].

Ces différentes contraintes liées à la production des *CI*, résultent dans (1) d'importants investissements en équipements de production, de contrôle qualité et de transport (2) d'importants investissements pour maintenir une production dans une salle blanche en norme, (3) d'important investissements dans les travaux de recherches pour suivre cette évolution constante des nœuds technologiques.

→ *Il est important d'assurer un retour sur investissement.*

Ainsi, il est important d'assurer la qualité des circuits en sortie. Ceci revient à contrôler la qualité d'un site de fabrication, en mesurant son rendement. En industrie du semi-

conducteur, on distingue trois types de rendement, relatifs à trois niveaux dans le processus de production global des *CI*. Premièrement, on peut mesurer le rendement au niveau de la ligne, le *« Line yield »*, qui est le pourcentage de plaques qui arrivent à la fin du cycle de production *BEOL*, juste avant les tests finaux (*PT* et *EWS*). Deuxièmement, on peut mesurer, le *« Die yield »* qui est le pourcentage de puces fonctionnelles sur une plaque, mesuré à la suite des tests *EWS*, *i.e.* en sortie du processus *FE*. Finalement, le rendement au niveau du test final, le *« final test yield »*, qui est le pourcentage de produits dans lesquels des puces ont été intégrées et qui, après le test au niveau du produit, sont jugés avoir le bon niveau de qualité requis pour l'envoi au client final, *i.e.* à la fin du processus *BE*. Parmi ces trois types de rendement, le plus critique est le *« Die yield »* [7].

Sur le site de production *Crolles 300mm*, où le processus *FE* est fait, on s'intéresse à mesurer la qualité des étapes de ce processus *FE*. On fait, ainsi, référence par le terme rendement, à celui évalué au niveau de la puce, le *Die yield*. Sa valeur est un indicateur majeur sur la santé d'un site de fabrication : même un léger gain au niveau du rendement peut induire un important profit financier.

L'amélioration du rendement est divisé en deux grandes catégories de méthodes : (1) l'amélioration de la qualité du processus de fabrication, et (2) l'analyse et le diagnostic des cas de rendement faible, *i.e.* de perte de qualité [8]. La première catégorie est basée sur les réglages du processus pour améliorer les performances et réduire les défauts, alors que la deuxième catégorie implique le diagnostic des défaillances causées par des événements anormaux tels que des anomalies de fonctionnement, des problèmes sur des équipements de production ou une contamination par particules.

Ainsi, l'objectif est d'assurer une production à capacité maximale de puces atteignant un niveau de qualité suffisamment bon, et ce le plus rapidement possible. Autrement dit, on cherche à atteindre *un bon taux* de rendement *rapidement*, notamment en phase d'implémentation de nouvelles technologies de production (problématique de courbe d'apprentissage). Pour cela, différentes techniques de contrôle des procédés de fabrication existent. Une présentation des techniques les plus répondues pour les sites de fabrication en *FE* est donnée dans la section suivante.

1.1.4 Le contrôle en temps réel

Comme le montre la Figure 1.9, en parallèle au processus de fabrication, des étapes de contrôle qualité sur les plaques sont définies pour contrôler la variabilité des étapes de production critiques, permettant, ainsi, de détecter les dérives qualité, dans un objectif global de maitriser la qualité des plaques en production. Notons, que pour des mesures d'optimisation du temps de production et de gestion des coûts, et selon un plan de contrôle défini, seul un échantillon, noté *M*, sur l'ensemble des plaques produites est contrôlé aux étapes de contrôle qualité intermédiaires durant le processus *FE*.

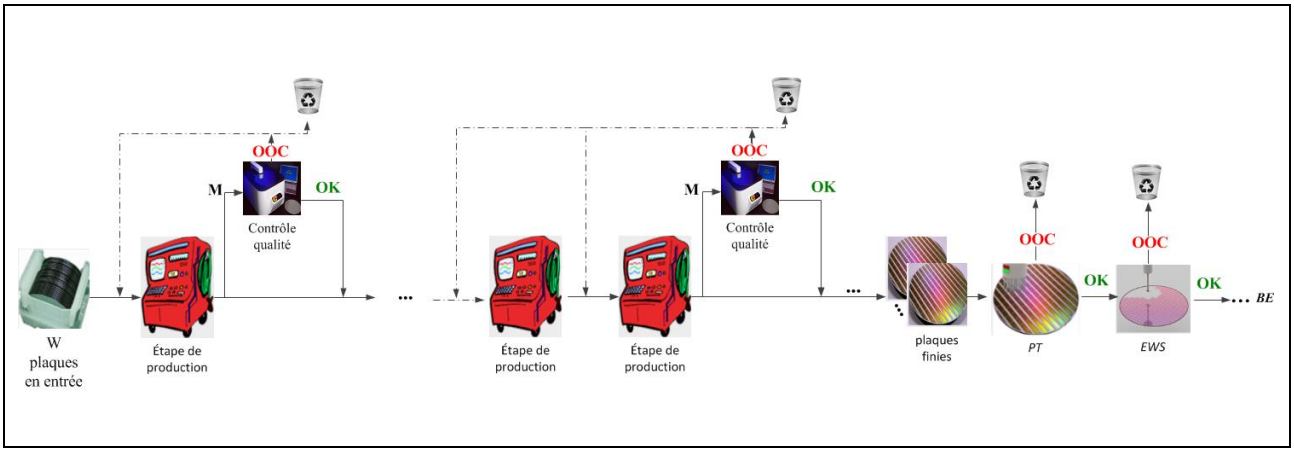


Figure 1.9: Description du processus de production FE et de contrôle qualité

Durant le processus de production des CI, on distingue trois types de contrôle qualité (1) Les mesures physiques dans l'atelier de *métrologie*, collectant *les mesures de métrologie* (2) Les tests paramétriques des plaques dans l'atelier *PT*. On fera référence aux données collectées à ces contrôles par *les mesures paramétriques*, et finalement (3) le test des fonctionnalités de chaque puce d'une plaque dans l'atelier *EWS*. Ce type de contrôle collecte *des mesures électriques*.

Ces différentes étapes de contrôle de qualité, de *métrologie*, de *PT*, et de tests *EWS*, font que les plaques mesurées et ne respectant pas les spécifications requises à un niveau du contrôle sont soit retraitées, si le problème identifié peut être corrigé, soit rejetées. Seules les plaques bonnes ou non contrôlées poursuivent le processus de fabrication normal. Ainsi, durant le processus de fabrication, les lots, initialement lancés avec 25 plaques, peuvent perdre des plaques, être divisés en plusieurs lots fils, ou fusionnés avec d'autres lots… En plus des contrôles sur les plaques, les machines de production sont elles aussi contrôlées.

L'objectif des méthodes en temps réel est de contrôler le bon déroulement des étapes de production critiques, afin de détecter rapidement les dérives, *i.e.* les anomalies. Une fois une anomalie détectée, on s'intéresse à la corriger permettant ainsi une maîtrise du rendement final. Ces techniques surveillent un ensemble de paramètres critiques durant le processus de production, et vérifient que leurs valeurs respectent les limites de contrôle correspondantes. Si une valeur est hors contrôle, une alarme est déclenchée et un ensemble d'actions correctrices est appliqué pour assurer que les paramètres contrôlés restent dans leurs limites définies. Ces approches sont issues du domaine de la maîtrise statistique des procédés MSP, connue souvent sous son nom anglais, *SPC Statistical Process Control* [9] [10] [11].

On propose de présenter deux des outils de contrôle des procédés, les plus répondus en industrie du semi-conducteur et plus particulièrement sur le site de *Crolles 300mm* de *STMicroeletronics* tout au long des processus *FEOL* et *BEOL*. Ces techniques sont détaillées dans les sous-sections suivantes.

1.1.4.1 SPC

Depuis le début des années 1990, l'application de techniques SPC est devenue un critère de sélection des fournisseurs pour un fabricant, généralisant son application dans la plupart des industries. Par exemple, Ford exigeait déjà un score de 90% d'application des techniques SPC par ses fournisseurs pour avoir le prix de l'excellence qualité et faire ainsi, partie de la liste des fournisseurs préférés [12].

Les cartes de contrôles SPC sont utilisées comme un outil pour détecter les sources de variabilité indésirables et ainsi les éliminer. Traditionnellement, les techniques SPC se basent sur les cartes de contrôle de Shewhart [13], introduites depuis les années 1930. Les limites basses et hautes d'une carte de contrôle sont calculées statistiquement et si un point sort de ces limites, le processus de fabrication s'arrête et une action correctrice est déclenchée avant que le processus ne puisse reprendre.

Par ailleurs, souvent, pour un processus de fabrication critique, différents paramètres sont mesurés pour contrôler sa variabilité. Par exemple, un processus de gravure sèche en semi-conducteur est souvent contrôlé à travers des paramètres tels que le taux et l'uniformité de la gravure, ainsi que la sélectivité par rapport à la résine photosensible et à l'oxyde. Contrôler ce processus à l'aide d'une carte de contrôle pour chaque paramètre indépendamment serait insuffisant, car même si ces mesures contiennent d'importantes informations sur le processus, des cartes individuelles ne prennent pas en compte les corrélations qui peuvent exister entre ces mesures. Pour faire face à ce problème, plusieurs méthodes basées sur les statistiques multivariées ont vu le jour, telles que « les cartes de contrôle multivariées de T^2 de Hotellings » proposées durant les années 1947 ou encore « les cartes de contrôle basées sur ACP (Analyse en composantes principales).

1.1.4.2 FDC

Le système *FDC* (Fault Detection and Classification) permet une maîtrise en temps réel des procédés de fabrication et est dérivé des techniques *SPC. FDC* est un outil qui permet de suivre les paramètres d'un équipement tels que la température d'un four [14]. Cette méthode permet de détecter les dérives équipements: si un paramètre dépasse ses limites de contrôles, des actions correctives sont appliquées pour le réintégrer dans ses limites de contrôles [15]. Le concept *FDC* constitue une boucle de contrôle permanente sur l'équipement de production.

Selon le type de l'équipement contrôlé, les valeurs de ses paramètres sont collectées pour chaque plaque, si il s'agit d'un équipement mono plaque, ou par batch de plaques (groupe de plaques), lorsqu'il s'agit d'un équipement multi plaques. Une collecte de données est connue sous le nom de *contexte*, et permet, donc, de décrire l'état de l'équipement lors du traitement d'une plaque ou d'un batch de plaques.

Ces données sont collectées à travers des stratégies FDC définies par des ingénieurs pour répondre à des problèmes spécifiques. Une stratégie FDC est une méthode de contrôle du bon fonctionnement d'un équipement, définie par les ingénieurs comme suit :

1. Identification des paramètres clés pour le problème à surveiller sur l'équipement
2. Pour chaque paramètre :
    a. On identifie la **fenêtre temporelle** à étudier. En effet, l'objectif de cette étape est de réduire l'information temporelle pour ne conserver que la période intéressante. Par la suite, ceci permet aux ingénieurs de faire des calculs efficaces. Par exemple, en étudiant la température d'un four, la première et dernière phase relatives à la montée et à la descente en température peuvent ne pas être intéressantes, mais plutôt la phase où la température est stabilisée.
    b. Il peut être intéressant de **transformer le flux temporel** à travers par exemple une combinaison linéaire de différents paramètres, ou un changement d'échelle,…
    c. **Résumer** le flux temporel en une seule valeur, utilisée comme indicateur, et permettant de synthétiser le fonctionnement de l'équipement pour la fenêtre temporelle choisie.
    d. Définir **des limites de contrôles** pour cet indicateur constituant une carte de contrôle, ainsi que les actions correctrices à appliquer en cas de violation de ces limites.

Ainsi, selon les stratégies définies, le système *FDC* permet de détecter les défauts au moment où ils se produisent, en se basant sur une analyse des paramètres ***process*** et ***équipement*** capturé sur un équipement donné. Ceci est fait en trois étapes principales *Figure 1.10* : (1) le système *FDC* demande à l'équipement la collecte des valeurs des paramètres selon les stratégies définies. Comme illustrée, une passerelle gère les communications de l'équipement avec l'extérieur notamment le système *FDC*, mais aussi le système d'automatisation qui contrôle la production en salle. Ainsi, souvent pour des raisons de priorité, la collecte de données *FDC* peut ne pas aboutir à cause d'une communication en cours entre l'équipement et le système d'automation. (2) le système *FDC* calcule les indicateurs résumés à partir de ces valeurs temporelles et vérifie si les valeurs de ces indicateurs respectent les limites de contrôle définies. Si une ou plusieurs cartes de contrôle ne sont pas respectées, des alarmes sont déclenchées avec un code qui identifie à quelle carte de contrôle cette alarme correspond. (3) Les données temporelles ainsi que les données résumées sont stockées dans la base correspondante, nommée *centric*.

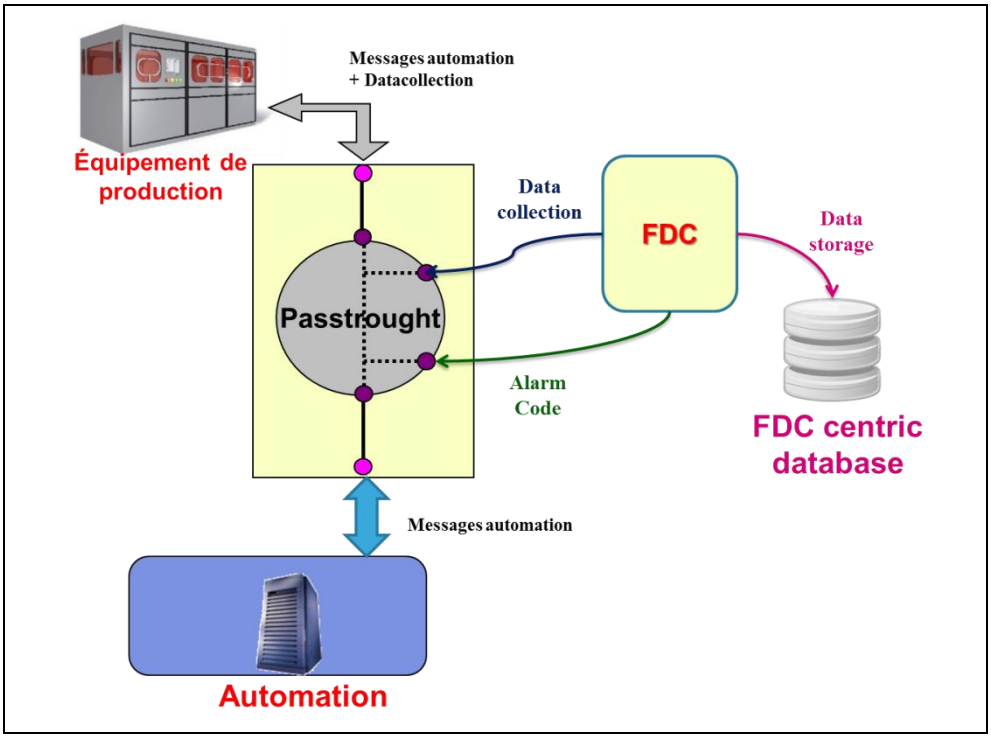


Figure 1.10 : Fonctionnement du système FDC

1.1.4.3 Les techniques de fouille de données pour le contrôle en temps réel

Les progrès connus par les capteurs et les technologies de base de données ont permis de collecter d'énormes quantités de données décrivant le processus de fabrication en industrie. À partir de cette masse importante de données, plusieurs travaux ont cherché à améliorer le contrôle en temps réel en identifiant les anomalies. Les trois familles principales de méthodes appliquées pour traiter cette problématique sont décrites ci-dessous.

Premièrement, on distingue les méthodes de classification pour la détection d'anomalies. À partir d'un ensemble d'exemples labélisés, ce type de méthodes cherche à construire un modèle qui permettrait de classer les nouvelles observations, *i.e.* à détecter les cas anormaux. Un des algorithmes de classification les plus utilisés est les machines à vecteurs supports (SVM) [16]. Initialement développé pour la classification binaire, un algorithme SVM identifie un hyperplan qui sépare au mieux les données appartenant à deux classes différentes. Par exemple, les auteurs de [17] proposent d'appliquer l'algorithme SVM dans le cadre des contrôles FDC. D'autres travaux, comme en [18], ont adapté la méthode SVM classique pour construire un modèle en se basant uniquement sur un échantillon de données de la classe normale. Ce type d'algorithmes est appelé « machines à vecteur support à une classe » (1-SVM). En plus des SVM, les réseaux de neurones ont aussi été souvent utilisés pour construire un modèle à partir des données pour la détection des anomalies et ainsi assurer un contrôle en temps réel de la qualité du procédé de fabrication. Par exemple, les auteurs de [19] proposent de construire un modèle à travers l'apprentissage de réseaux de

neurones modulaires (*MNN*) pour détecter les anomalies de fonctionnement d'un équipement de gravure plasma pour l'industrie du semi-conducteur. Dans un réseau de neurones modulaires, plusieurs réseaux de neurones élémentaires coopèrent pour classifier une nouvelle observation en *normale* ou *anomalie*. Par ailleurs, d'autres travaux ont exploré les algorithmes classiques d'induction d'arbres de décision, comme ceux de [20], qui ont utilisé l'algorithme CART pour détecter les courbes de température anormales d'un processus de lithographie.

Deuxièmement, on distingue les méthodes de clustering, *i.e.* d'apprentissage non supervisé, pour la détection des anomalies. Contrairement à la première famille de méthodes, les données d'apprentissage ne sont pas labélisées, et ces méthodes ne cherchent pas construire un modèle à partir de ces données. En effet les techniques de clustering cherchent identifier des groupes aussi appelés clusters, tels que les observations au sein d'un même cluster sont plus similaires les unes aux autres qu'à celles de clusters différents. Pour la détection d'anomalies, l'idée principale est que les observations appartenant à la classe normale constitueront des clusters denses. Par exemple, les auteurs de [21] proposent une nouvelle méthode pour la détection d'anomalies locales.

Finalement, on distingue les méthodes des plus proches voisins. Comme les techniques de clustering, on ne cherche pas à construire un modèle à partir des données d'apprentissage. Par contre, contrairement aux méthodes de clustering qui ont une vision globale des données, ces méthodes analysent chaque observation par rapport à son voisinage local. Par exemple, les travaux de [22] ont adapté la méthode k-NN (k-Nearest Neighbor) pour identifier les défauts à partir d'un ensemble d'apprentissage représentant les cas normaux uniquement, et ce en se basant sur une idée simple qui consiste à considérer un nouveau cas comme normal si il suit une trajectoire similaire aux trajectoires des exemples normaux, sinon, *i.e.* si sa trajectoire présente une déviation par rapport aux trajectoires des exemples normaux, il est considéré comme problématique.

1.1.4.4 Premiers constats

L'objectif des méthodes « de contrôle en temps réel » est de contrôler la variabilité du procédé de fabrication en détectant les anomalies le plus rapidement. Par ailleurs quand une anomalie est détectée, il est tout aussi important de remonter à sa cause, si elle n'est pas déjà connue. Ainsi, malgré que ces outils de contrôle en temps réel permettent de définir une base solide pour des procédés de fabrication bien réglés, l'identification des causes de perte de qualité reste difficile et coûteuse [23] : elle présente des défis industriels et scientifiques importants.

En industrie du semi-conducteur, l'identification des causes de perte de qualité fait partie de la problématique globale de maîtrise du processus de production pour l'amélioration du rendement. En plus de la première famille de techniques en temps réel, décrite précédemment, une deuxième famille de méthodes existe [24]. Celles-ci sont appelées des techniques à posteriori ou « post-hoc ». L'objectif de cette deuxième famille de méthodes consiste à expliquer à posteriori les causes d'un problème, *i.e.* identifier la cause du problème

de perte de qualité, afin d'appliquer les actions correctives pour empêcher ce problème de se reproduire dans le futur.

Nos travaux de thèse s'inscrivent dans cette deuxième famille d'approches. La problématique correspondante sera d'avantage détaillée dans la section suivante.

1.2 PROBLEMATIQUE DE LA THESE

Dans un contexte industriel concurrentiel et caractérisé par des domaines d'application critiques, la maîtrise de la qualité des puces en production est la clé de succès des sites de fabrication microélectronique. La maîtrise de qualité est assurée grâce aux différentes techniques de Maîtrise statistique des procédés (MSP), notamment à travers la définition de plans de contrôle. Mais malgré ces différentes techniques de contrôle, de nouvelles pertes de qualité sont détectées assez fréquemment et il est important de remonter à leurs origines.

On distingue, principalement, deux types de produits en production ; des produits dont la technologie est mature, et ceux dont la technologie est en cours de maturation. Malgré la maturité des processus de contrôle définis pour les produits de technologies matures, des pertes de qualité occasionnelles surviennent en cours de production. Pour les produits de technologies en cours de maturation, on est face à une situation contraire, caractérisée, selon le degré de maturation, par un nombre important de plaques problématiques face à relativement peu de bonnes plaques.

Face à ces situations de perte de qualité, on s'intéresse à leur donner une explication, à travers l'identification de causes de production qui en sont probablement responsables. Cette problématique s'intègre dans l'objectif global d'une meilleure maîtrise du processus de fabrication, *i.e.* une maîtrise du rendement du site de fabrication.

Un exemple typique de cette problématique concerne l'atelier de défectivité. La mission de cet atelier est de détecter, analyser et réduire le nombre de défauts sur les plaques. L'inspection de défectivité, sur l'échantillon de plaques à contrôler, passe par 3 étapes : *(1)* la détection sur les plaques des problèmes par des outils d'inspection optiques ou de microscopie électrique ; *(2)* l'analyse de ces problèmes (qui est elle-même composée des 3 sous étapes de *(2.i)* revue des défauts, *(2.ii)* classification des défauts et *(2.iii)* détermination de leurs origines) ; et finalement *(3)*, une dernière étape de réduction du nombre de défauts. Ainsi, les ingénieurs, une fois un défaut détecté, essaient de l'expliquer, dans l'étape *(2.iii)*, en identifiant la cause correspondante, si elle est non connue.

Traditionnellement, les méthodes « post-hoc », utilisées par les ingénieurs de l'atelier *« defectivity »* ou dans toute autre situation où l'objectif est l'explication d'un problème, notamment de perte de rendement, se basent sur l'expertise d'ingénieurs métiers qui, à partir d'un défaut détecté, identifient un ensemble d'hypothèses représentant des causes potentiellement explicatrices du problème. À partir de cette sélection d'hypothèses, ils isolent

celle qui a causé ce défaut. Ces techniques, basées sur l'expertise métier seront détaillées dans le prochain chapitre.

Par ailleurs, l'évolution continue de la complexité des CI en production et la finesse de gravure de leurs composants fait que les procédés de fabrication correspondants sont de plus en plus complexes et ainsi, l'explication des défauts, en se basant uniquement sur l'expertise humaine, devient insuffisante. Ainsi, d'autres techniques se basant sur l'exploration des données, en plus de l'expertise humaine, sont de plus en plus utilisées pour répondre à cette problématique industrielle.

En effet, durant les différentes étapes, de production et de contrôle qualité, et grâce aux capteurs intégrés dans la plupart des équipements, des données sont collectées. Ces données représentent une mine d'information à explorer pour résoudre différentes problématiques industrielles. Par exemple, dans la section 1.1.4.3, on a présenté une sélection de travaux qui se sont basés sur des méthodes de fouille de données pour le contrôle en temps réel, *i.e.* pour la détection des anomalies.

De même, les données disponibles peuvent être explorées à travers les méthodes de fouille de données pour répondre à notre problématique d'explication de cas de perte de qualité. Nos travaux se positionnent dans ce type de méthodes, où on s'intéresse à l'explication de cas de perte de qualité en industrie du semi-conducteur à travers une exploitation de la richesse du système d'information des sites de fabrication. Ainsi, l'objectif est bien l'explication des phénomènes de perte de qualité passés et non pas la définition d'un modèle de prédiction pour le futur, et ce en nous appuyant sur l'utilisation de nouvelles méthodes d'analyse et en explorant de nouvelles sources de données.

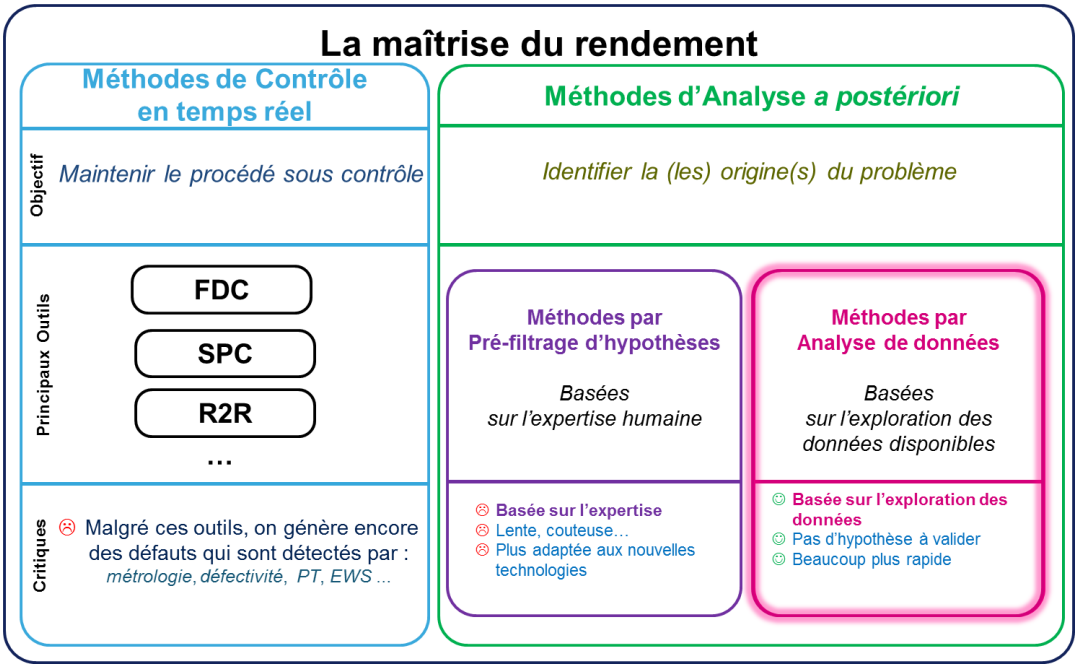


Figure 1.11 : Positionnement de nos travaux de thèse

1.2.1 Positionnement des travaux en référence au contexte industriel

Les travaux relatifs à notre problématique sont présentés en détails dans le prochain chapitre. Dans cette section, nous proposons une brève description pour les justifier et les positionner par rapport à l'existant. Les systèmes d'analyse traditionnels se basent généralement sur des techniques d'identification de corrélations à travers une analyse globale, en cherchant à identifier l'étape de production responsable, ainsi que l'équipement problématique, sans donner de détails sur les conditions propres à cet équipement.

→ *Nos travaux chercheront à répondre au besoin d'expliquer les pertes de qualité de manière plus approfondie, en explorant d'autres sources de données caractérisant les paramètres de fonctionnement interne des équipements, telles que la base relative aux données collectées lors des contrôles équipements, FDC. Ceci permettrait d'identifier de manière plus précise et plus efficiente des causes détaillées potentielles, qui descendent au niveau des paramètres équipement.*

Pour cela, on a besoin d'explorer les techniques d'analyse de données afin de proposer une nouvelle méthode destinée à répondre aux contraintes industrielles du semi-conducteur et qui puisse être ultérieurement généralisable dans différents autres domaines.

→ *L'enjeu central de la thèse consiste à proposer de nouvelles méthodes d'analyse des données.*

Par ailleurs, pour faciliter l'adoption de la méthode d'analyse proposée, il serait intéressant de proposer une approche d'intégration de la méthode d'analyse, permettant de guider l'ingénieur dès l'extraction des données d'analyse et ce jusqu'à l'exploration des connaissances identifiées.

→ *Il s'agira d'inscrire la méthode d'analyse proposée au sein d'une approche globale décrivant les différentes étapes de préparation, analyse des données et exploration des connaissances extraites.*

1.2.2 Description des données disponibles dans le système d'information d'un site de fabrication *FE*

Les données, qui résument l'historique de fabrication d'une plaque de silicium en cours de production, en *FE*, peuvent être classées en deux grands types selon l'objet qu'elles caractérisent. Pour une plaque donnée, on a d'un côté les données relatives à des caractéristiques de la plaque en cours de production, qu'on nomme « *données plaques* », et, d'un autre côté, les données relatives à l'état des équipements qui l'ont traitée, qu'on nomme « *données équipement* ». Par ailleurs, il faut noter qu'un troisième type de données existe, relatif à l'environnement et au contexte de production, tel que le fournisseur des matières

premières par exemple, ou encore les données décrivant les étapes de maintenance des équipements de production. Dans le cadre de cette section, on choisit de présenter les deux types de données qui nous seront utiles dans la suite du manuscrit de thèse, à savoir les « *données plaques* » et les « *données équipement* ».

1.2.2.1 Les « données équipement »

Les « données équipement » sont principalement collectées par l'outil FDC. Elles permettent d'avoir un historique du fonctionnement des équipements. En effet, grâce aux capteurs disponibles sur les équipements, pour chaque équipement de production, d'importantes quantités de données sont collectées décrivant l'état de l'équipement, lors du traitement des plaques ou même quand il est en repos. Par exemple, pour une machine de type four, un exemple de paramètres collectés est la température et, ce, à différents emplacements du four. Grâce aux stratégies de contrôle *FDC*, les données collectées par le système FDC et stockées dans la base *centric* du système d'information de *ST Crolles 300mm* sont disponibles en deux formats :

→ **Temporel** : Selon un *« sampling rate »* donné, définissant la fréquence de collecte, les valeurs d'un paramètre, *p*, sont remontées pour un contexte donné, *i.e.* pour une plaque dans le cas d'un équipement mono plaque, ou plusieurs sinon. La *Figure 1.12* représente le flux temporel du paramètre de pression *p* pour le contexte n°*9763958* relatif à la plaque « *Q445782.07*».

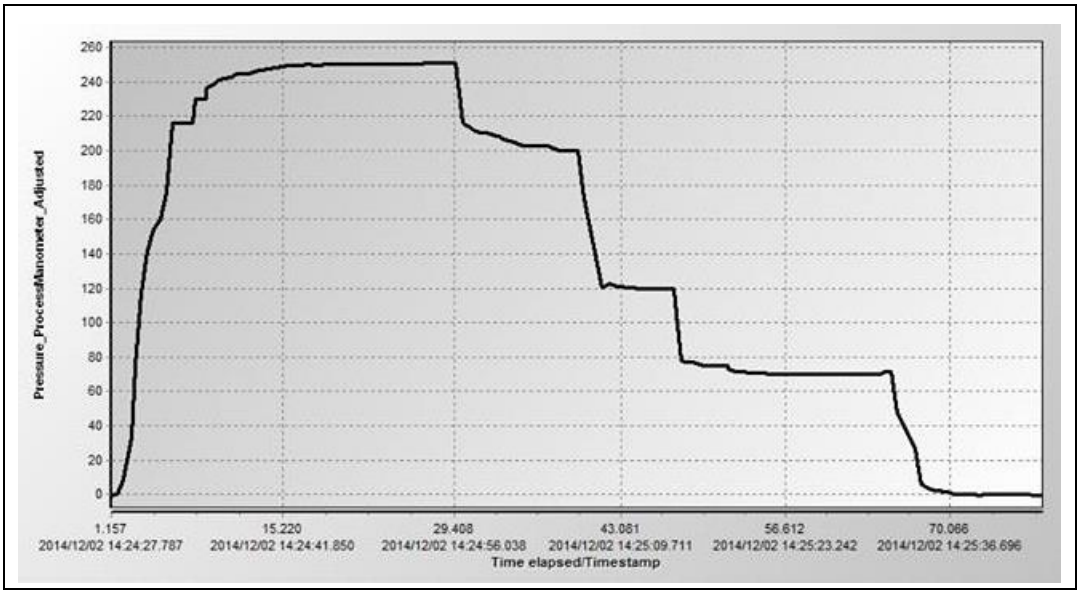


Figure 1.12 : Flux temporel du paramètre p pour le contexte 9763958 (la plaque Q445782.07)

→ **Indicateur résumé** : Le flux temporel pour un contexte est résumé selon la définition de la stratégie *FDC* en une seule valeur représentant l'indicateur résumé. L'évolution de cet indicateur permet de représenter l'historique de l'état de l'équipement, pour les différents contextes considérés. La *Figure 1.13* représente

la courbe d'évolution de l'indicateur résumant le paramètre temporel *p* représenté dans le paragraphe précédent pour 25 contextes. Cet indicateur est construit à partir de la valeur moyenne du flux temporel du paramètre *p* sur la fenêtre temporelle identifiée par *s(3)*. Chaque point noir de cette courbe représente la valeur de cet indicateur, résumé pour le contexte correspondant.

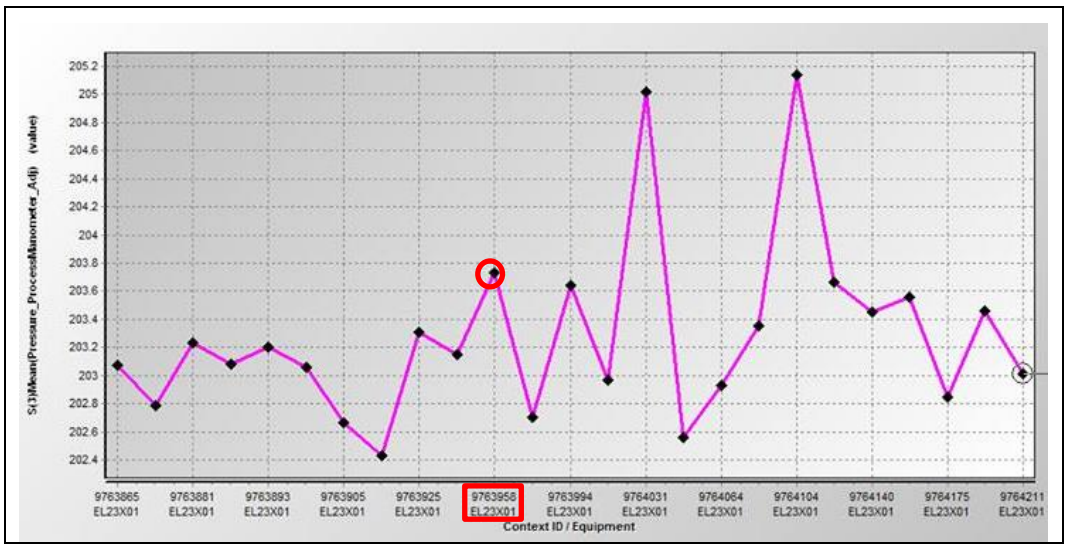


Figure 1.13 : Courbe d'un paramètre résumé FDC

Les données temporelles sont trop importantes. Par conséquence, il est quasiment impossible de les analyser telles qu'elles sont. Ainsi, on se base généralement sur l'analyse des données résumées. Pour enrichir cet ensemble, généralement, on fusionne les données relatives aux différents indicateurs résumés construits par les différentes stratégies actives sur un équipement. Ceci suffit pour décrire l'état d'un équipement à un moment donné.

1.2.2.2 Les « données plaques »

Par ailleurs, les « données plaques » sont de quatre types et sont collectées principalement par les outils de contrôle de qualité relatifs aux quatre ateliers de contrôle qualité présentés précédemment :

- Les données de ***métrologie*** représentant les résultats des étapes de mesure de métrologie qui viennent valider les étapes critiques de fabrication.
- Les données relatives à l'étape **d*'inspection de défectivité*** des plaques représentant si un défaut a été détecté ou non, et dans l'affirmative, de quel type.
- Les données relatives aux ***tests paramétriques PT*** représentant les résultats des différents tests pour les différents sites.
- Finalement, les données relatives au ***test*** **EWS,** représentant les résultats de ces tests pour chaque puce de la plaque.

Comme pour les données équipement, les données plaques sont disponibles en deux formats : en *mesure individuelle*, et en *indicateur résumé*. Les mesures individuelles représentent les mesures telles qu'elles ont été collectées, *i.e.* pour chaque site de la plaque mesurée. À partir de ces mesures au niveau site, sont calculés les indicateurs résumés, à travers des fonctions statistiques de base telles que la moyenne, l'écart type … On se base sur ces indicateurs résumés et, selon les cartes de contrôles définies pour une étape de contrôle qualité donnée, une plaque sera considérée comme bonne, bonne, notée *OK* ou pas, notée *OOC.*

Ces données, décrivant l'état des plaques et celui des équipements, sont stockées dans des bases de données dispersées et non reliées. Ce système d'information représente une mine à explorer afin d'identifier des connaissances cachées. Par contre, leur forte volumétrie requiert des méthodes d'analyse très structurées et rigoureuses. Il est, ainsi, crucial d'assister les ingénieurs avec des méthodes et outils les aidant à mieux explorer ces sources de données pour répondre à leurs besoins, notamment pour répondre à la problématique globale d'explication de cas de perte de qualité.

1.2.3 Défis relevés

L'exploration de ces sources de données hétérogènes pose des défis scientifiques et industriels importants pour répondre à la problématique globale d'explication de cas de perte de qualité. Les défis industriels et scientifiques traités par nos travaux sont précisés dans les paragraphes suivants.

- **<u>Défis industriels</u>**

La problématique abordée met en évidence des défis relatifs au contexte général d'identification de corrélations en industrie du semi-conducteur.

- Premièrement (DI1), un lot peut être divisé en plusieurs sous-lots, à n'importe quel moment de sa vie et ceci *N* fois. On parle, alors, d'opération de *« split »*. De plus des sous-lots peuvent être groupés afin de créer un nouveau lot. On parle, alors, d'opération de *« merge »*. Comme précédemment décrit, ces changements de la composition des lots sont principalement les résultats des étapes de contrôle qualité, qui font que des plaques ne respectant pas les cartes de contrôle définies sont soient retraitées soient rejetées.

  → Ainsi, la méthode d'identification des causes explicatives d'une perte de qualité devrait gérer les modifications de composition au sein d'un lot.

- Deuxièmement (DI2), l'ordre théorique de passage des plaques d'un même lot n'est pas forcément respecté, et ce suite à des interventions manuelles ou pour des raisons d'optimisation du traitement des plaques d'un lot par un équipement.

Prenons l'exemple d'un lot composé de 25 plaques, *w1, …, w25*, en entrée sur un équipement de production *t*. Ces plaques peuvent être traitées dans l'ordre, *w1, w2…w25*, mais aussi dans un ordre inversé, ou même dans un ordre aléatoire. Par ailleurs, pour les équipements qui traitent plusieurs plaques en même temps, comme c'est le cas pour les équipements de type four, des plaques de lots différents peuvent être traitées en même temps et, inversement, des plaques d'un même lot peuvent être traitées séparément.

→ Ainsi, la méthode d'identification des causes explicatives d'une perte de qualité devrait gérer les différences d'ordre de passage des plaques d'un même lot sur un même équipement à une étape de production donnée.

- Troisièmement (DI3), les coordonnées de mesures entre les sites des différentes opérations de contrôle qualité ne correspondent pas forcément entre eux. Par exemple, les mesures de métrologie et des tests paramétriques *PT* sont à un niveau site de mesure, alors que les tests de l'*EWS* sont au niveau puce. Et même pour les résultats de métrologie et des tests *PT*, on ne contrôle pas les mêmes sites.

  → Ainsi, la méthode d'identification des causes explicatives d'une perte de qualité devrait gérer les différences de cordonnées des emplacements contrôlés sur une même plaque aux différentes étapes de métrologie, PT et EWS.

- Quatrièmement (DI4), comme seul des échantillons de plaques sont mesurés durant les étapes de contrôle qualité et pour des problèmes de collecte, en confrontant les différentes données issues de sources différentes, des données manquantes pour des plaques à certaines étapes peuvent apparaître. Par ailleurs, sur une même étape de production ou de contrôle qualité, des données manquantes peuvent apparaître et ceci pour diverses raisons, tel que l'arrêt momentané de la connexion entre l'équipement (de production ou de contrôle qualité) et le système de collecte et de sauvegarde des données correspondantes.

  → Ainsi, la méthodologie devrait gérer les données manquantes entre différentes étapes, ou bien au sein d'une même étape.

- Finalement (DI5), les causes identifiées doivent être *valides* en décrivant des connaissances avec un degré de confiance acceptable, *nécessairement compréhensibles* par les ingénieurs et *potentiellement utiles, i.e.*, représentant un intérêt pour les ingénieurs.

  → Ainsi, la méthode d'analyse devrait proposer des indicateurs pour mesurer ces aspects de validité, compréhensibilité et utilité des causes explicatives identifiées pour expliquer un cas de perte de qualité donné.

- **Défis scientifiques**

Face à la problématique globale d'explication de cas de perte de qualité, et dans une perspective d'analyser différentes sources de données, des défis scientifiques sont définis.

- Premièrement (DS1), des défis reliés à la spécificité des sources hétérogènes des données. Ces problématiques concernent principalement la préparation des données, notamment pour l'alignement (relatif aux problématiques DI1, DI2 et DI3) et la gestion des valeurs manquantes (relative à DI4).
- Deuxièmement (DS2), on identifie une problématique relative à l'analyse des données, notamment celle de la forte volumétrie des données disponibles, notamment concernant celles collectées à travers le système FDC durant le temps de traitement des plaques par un équipement.
- Finalement (DS3), pour garantir l'identification de connaissances *valides*, *nécessairement compréhensibles* et *potentiellement utiles* (relatives à DI5), on identifie des problématiques de gestion de la qualité des connaissances extraites.

Suite à cette première mise en exergue des verrous scientifiques, nous nous proposons de les approfondir dans les chapitres qui suivent : en référence à l'état de l'art développé au chapitre 2, les problématiques scientifiques de recherche seront exprimées de manière plus formelle dans le cadre du chapitre 3.

1.3 NOS CONTRIBUTIONS

Dans ce travail de thèse, nous adressons le problème d'explication de cas de perte de qualité, une problématique difficile et coûteuse [23] en industrie du semi-conducteur. Pour répondre aux différents défis cités dans le paragraphe précédent, les travaux réalisées dans le cadre de cette thèse ont permis de proposer trois contributions, détaillées dans ce qui suit :

- *C 1 : Proposition de CLARIF : une méthode hybride de fouille de données « par étape » et « par plaque »*

L'idée générale de la méthode proposée *CLARIF*, pour *« CLustering and Association Rules based IdentiFier »*, consiste à adopter *:*

- *une analyse par plaque*, permettant ainsi d'éviter les problématiques relatives aux *changements structurels* des compositions des lots ainsi qu'à la *différence d'ordre de passage* des plaques sur un équipement. Pour les données des machines de production, nous analyserons les indicateurs résumés décrivant l'état de fonctionnement de cette machine durant le traitement d'une plaque donnée. Par ailleurs, pour les étapes de contrôle qualité, nous analyserons les mesures individuelles, relatives à chaque paramètre contrôlé séparément, et ce au niveau des sites de mesures de la plaque correspondante.
- *une analyse par étape*, nous permettant d'expliquer les pertes de qualité relatives, d'un côté, à une étape, un équipement de production et plus précisément à des conditions particulières correspondantes, mais aussi

d'identifier des causes relatives à différentes étapes, différents équipements et différentes conditions, d'un autre côté.

Par ailleurs, afin d'éviter le problème de "boite noire", et proposer, ainsi, une méthode de fouille de données compréhensive et transparente à l'utilisateur, on propose d'utiliser la représentation en règles pour expliquer un phénomène *Y*. Les règles forment une représentation naturelle des relations de dépendances causales. Une règle, *r*, représente une association d'un sous ensemble de conditions, *X*, vers un résultat, *Y*. On pourra noter une règle *r* comme suit $r(X \rightarrow Y)$. Par exemple, dans le cadre d'identification de causes de perte de qualité locale, une règle $r(X \rightarrow Y)$ est interprétée comme suit ; pour un cas de perte de qualité *Y*, identifié à partir des mesures de contrôle qualité, cette règle identifie *X* comme une cause potentielle, susceptible de contribuer à expliquer *Y*. *X* décrit des conditions particulières sur une ou plusieurs étapes cause, *i.e. P*, résultant dans le phénomène *Y* identifié sur les étapes effets, *i.e. Q*.

Nous proposons une combinaison de trois méthodes afin de tirer avantage de chacune d'entre elles : D'une part, l'utilisation de *techniques de clustering* est sélectionnée pour leurs capacités à construire des groupes en se basant sur la similarité des données. D'autre part, les outils de fouille de *règles d'association* sont sélectionnés pour l'identification des causes explicatives. Enfin, nous préconisons des techniques *d'induction d'arbre de décision* pour accroitre la lisibilité et l'utilisabilité des connaissances extraites. Ainsi, *CLARIF* est une méthode hybride de fouille de données, qui combine ces trois techniques de fouille de données complémentaires, à savoir *la recherche de règles d'association*, *le clustering* et *l'induction d'arbre de décision*.

- *C 2 : Proposition d'un mécanisme de gestion de la qualité des connaissances extraites*

Une règle $r(X \rightarrow Y)$ identifie une cause *X* pour expliquer *Y*. Comme décrit par le défi scientifique *DS3*, la qualité de cette règle doit être mesurée selon les trois critères de *validité*, de *compréhensibilité* et *d'utilité*. Le critère de compréhensibilité est traité en partie grâce à la formulation de cette connaissance selon des règles. Par ailleurs, il serait important de proposer des indicateurs afin de comparer deux règles, par exemple.

Traditionnellement, la qualité d'une règle d'association est mesurée par des indicateurs tels que *la confiance*, ou le *support*. À partir des indicateurs déjà disponibles en littérature, nous chercherons à définir l'ensemble des indicateurs les plus pertinents à notre problématique, en réutilisant certain et en proposant de nouveaux si besoin.

La combinaison des indicateurs sélectionnés constituera un outil pour mesurer la qualité des causes explicatives identifiées, et ainsi, de les comparer entre elles. On propose donc, de sélectionner les règles les plus intéressantes, pour ne garder que celles qui apportent le plus de valeur ajoutée à l'ingénieur. Pour cela, on propose une combinaison de méthodes de sélection : d'un côté, on peut se baser sur des seuils fixés par les ingénieurs sur un ou

plusieurs indicateurs de qualité, d'un autre côté, on renforce cette sélection à travers un *Front de Pareto* pour ne garder que les règles les plus dominantes selon ces indicateurs.

- *C 3: Proposition d'une définition d'un processus ECD pour appliquer la méthode CLARIF proposée*

Finalement, nous proposons une définition d'un processus d'Extraction des Connaissances à partir des Données, *ECD*, permettant de préciser les données à collecter, la manière d'appliquer la méthode de fouille proposée *CLARIF*, ainsi que les exploitations possibles de ses résultats. Ce processus *ECD* sera présenté comme un processus itératif et récursif, composé de trois étapes clés [25]: *(1)* la formulation du problème, l'extraction et la préparation des données, *(2)* l'analyse des données et (3) l'intégration des connaissances identifiées. Il sera détaillé dans le chapitre 3.

1.4 CONCLUSION

Dans ce chapitre, nous avons brièvement présenté le contexte de notre partenaire industriel, avec une présentation de la problématique à traiter. On s'intéresse à la problématique d'amélioration du rendement, en général, et ce à travers l'explication des causes de perte de qualité qui peuvent affecter le rendement d'un site de fabrication tel que celui de *STMicroelectronics Crolles 300mm*. Un bref positionnement par rapport aux techniques existantes pour répondre à cette problématique a été introduit. Finalement, les contributions majeures proposées par ce travail ont été présentées.

Dans le prochain chapitre, on propose de fournir un état de l'art des techniques existantes pour répondre à la problématique d'explication des pertes de qualité avec un focus sur les techniques basées sur l'analyse des données, ainsi qu'une introduction au processus ECD ainsi qu'aux techniques de fouille de données utilisées, afin de faciliter l'interprétation de leurs utilisations.

Chapitre 2 : Approches *a posteriori* pour l'explication de défauts en semi-conducteur

Dans une optique globale d'une maîtrise du rendement en industrie, on distingue deux types d'approches. Les approches en temps réel où l'objectif est le maintien du procédé de fabrication sous contrôle et la détection des anomalies. Et les approches *a posteriori* où l'objectif est l'identification de la ou les origines du problème de perte de qualité. La première famille de méthodes est présentée dans le chapitre précédent. Dans ce chapitre, on propose un état de l'art sur les travaux existants en relation avec les problématiques de la deuxième famille de méthodes, avec une étude critique et un positionnement de nos travaux proposés dans le cadre de cette thèse.

TABLE DES MATIERES

2.1 CARACTERISATION DES DEFAUTS EN INDUSTRIE SEMI-CONDUCTEURS

Comme introduit dans le précédent chapitre, la problématique traitée dans cette thèse s'intègre dans un objectif global d'amélioration du rendement final d'un site de fabrication et plus précisément de réduction des pertes de qualité qui interviennent tout au long du processus de production. En plus des méthodes dites « en temps réel » précédemment introduites, la maîtrise de la qualité passe aussi par des méthodes dites « *a posteriori* » ou « *post-hoc* ». L'objectif de celles-ci est d'identifier la cause d'un cas de perte de qualité détecté, et d'appliquer les actions correctrices pour empêcher ce défaut de se reproduire dans le futur. Ces techniques se basent sur une méthodologie en trois étapes principales : *(i)* comparer les résultats finaux, *(ii)* analyser la « root-cause », *(iii)* ajuster les paramètres de production pour garantir la qualité des plaques futures.

Il est important de préciser que l'objectif n'est pas de déduire un modèle de prédiction, mais plutôt un outil permettant d'identifier les caractéristiques qui séparent une ou plusieurs populations par rapport à d'autres. Comme introduit dans [26], si l'objectif était de faire des prédictions, le problème serait un problème traditionnel de régression (on cherche à estimer une valeur précise) ou celui de classification (on cherche à prédire la classe résultante, plaque bonne ou pas, par exemple), et plusieurs méthodes pourraient être alors appliquées. Notre objectif ultime est le diagnostic, et non pas la prédiction d'une valeur ou d'une classe future.

Dans la suite de cette première section de ce chapitre, on propose une description des différents types de défauts en industrie du semi-conducteur, suivie par une description des différents types de sources de défauts.

2.1.1 Les types de défauts sur une plaque en semi-conducteur

Les défauts détectés à des étapes de contrôle qualité et plus précisément aux tests EWS peuvent décrire trois catégories [27] [28]:

- **Défauts aléatoires** : cette catégorie de défauts est caractérisée par des puces défectueuses distribuées généralement de façon aléatoire sur la plaque. Il n'y a donc pas de signatures spatiales particulières (*i.e.* un regroupement spatial particulier). Ils sont dus à des facteurs relatifs à l'environnement de fabrication, car même dans un environnement quasi-stérile, les particules ne peuvent être complétement supprimées. La représentation (a) de la Figure 2.1décrit une plaque présentant des défauts aléatoires.
- **Défauts systématiques** : ce type de défauts est caractérisé par la présence de signatures spatiales particulières (*i.e.* un regroupement spatial particulier) sur la

surface de la plaque. Par exemple, sur la Figure 2.1 les représentations (b) et (c) donnent deux exemples de défauts systématiques, avec une signature de damier, pour la première et une signature représentant un effet de bord pour la deuxième.

- **Défauts mixtes** : cette dernière catégorie mixe les deux types de défauts, en présentant d'un côté une signature particulière et d'un autre côté des puces défectueuses disposées aléatoirement sur cette plaque. Plusieurs travaux, comme [8] et [29], proposent d'isoler les deux catégories de défauts sur la puce, afin de mieux expliquer les défauts systématiques à travers des problèmes de production particuliers. La représentation (d) de la Figure 2.1 décrit une plaque présentant une combinaison de défauts aléatoires, et de défaut systématiques.

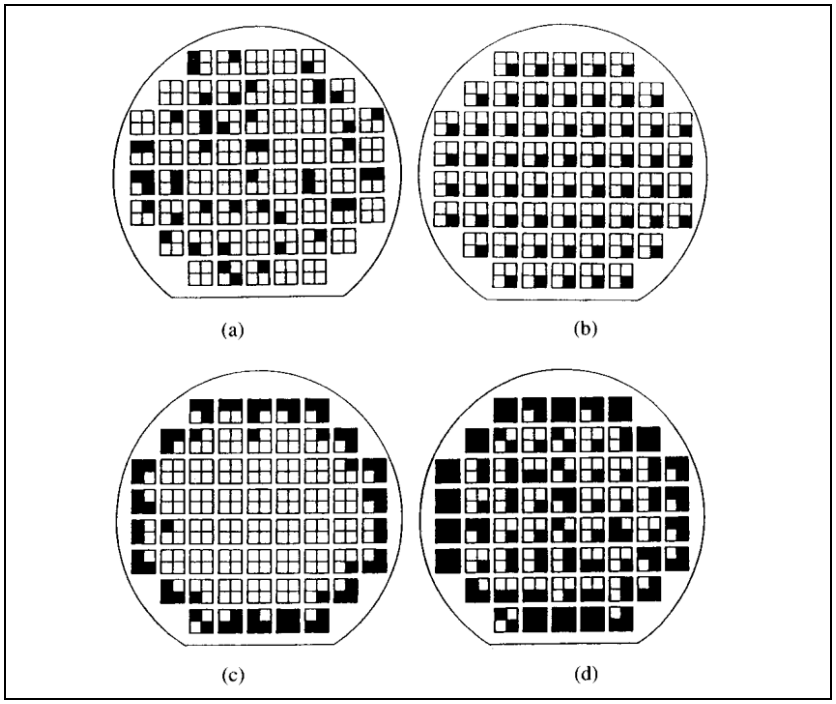


Figure 2.1: Exemples de signatures spatiales de défauts aléatoires (a), systématiques (b, c) et mixtes (d) en semi-conducteur ([30]).

La réduction des défauts aléatoires nécessite un travail à long terme sur la propreté de la salle blanche, alors que, la réduction des défauts systématiques nécessite une détection, une classification et une correction rapide des défauts [31]. Il est ainsi courant, en semi-conducteur de travailler à court terme pour expliquer et éviter ces défauts systématiques. Nos travaux dans le cadre de cette thèse s'inscrivent dans le cadre de l'explication des défauts systématiques.

Plusieurs travaux, comme [32] et [8], proposent d'analyser les données collectées au niveau de chaque puce lors des tests EWS pour identifier des signatures spatiales permettant ainsi de simplifier le diagnostic et l'explication des défauts relatifs à des pertes de rendement.

2.1.2 Les différentes sources d'un défaut en semi-conducteur

Un défaut en semi-conducteur peut être dû à différentes raisons. On peut distinguer deux types de sources explicatives d'un défaut.

- **Les sources externes** au procédé de fabrication. Cette première catégorie de sources de défauts est indépendante du fonctionnement interne de la salle blanche. Un exemple de sources externes est la qualité des *matières premières* utilisées, ou encore le *design* du produit. Dans [33], les auteurs ont pu expliquer une perte de rendement par une utilisation d'un substrat de silicium non standard (*i.e.* la plaque en entrée de la ligne de production).
- **Les sources internes** au procédé de fabrication. Ces sources de défauts sont relatives au fonctionnement interne à la salle blanche, comme *des équipements*, *des étapes*, ou *des recettes de production*. Des travaux comme [8], permettent aussi de définir la source d'un défaut comme une fausse manipulation faite par *un opérateur* à une étape de production.

Dans la littérature, on trouve des travaux qui s'intéressent à identifier et corriger les sources externes au fonctionnement de la salle, et plus précisément à corriger le design pour améliorer le rendement. Ces travaux s'intègrent dans le domaine *DFM* (*Design For Manufacturing*) [34]. Par ailleurs, d'autres travaux s'intéressent à identifier les sources internes au processus de fabrication, afin de mieux contrôler le fonctionnement interne à la salle. On s'intéresse donc, à identifier *l'étape de production*, *l'équipement* et/ou *l'opérateu*r problématique(s) pour expliquer un défaut détecté soit à une étape de contrôle qualité intermédiaire, soit au test final EWS.

Dans la section suivante, on propose une présentation des techniques traditionnelles qui se basent sur l'expertise humaine pour l'identification des sources de perte de qualité. Cette section est suivie par une autre section donnant une présentation détaillée d'un autre type de techniques basées sur l'exploration des données de production pour expliquer les défauts.

2.2 TECHNIQUES D'ANALYSE A POSTERIORI

2.2.1 Approches qualitatives : méthodes basées sur l'expertise

Un premier type d'approches d'analyse *a posteriori*, *i.e.* pour remonter aux causes d'un problème, est défini par les approches dites qualitatives se fondant sur l'expertise métier. . Ces méthodes sont aussi appelées « méthodes par pré-filtrage ». Comme illustré dans la

Figure 2.2, les méthodes *« a posteriori »* se fondent sur l'expertise d'ingénieurs métiers qui, à partir d'un défaut détecté, identifient un ensemble d'hypothèses représentant des causes potentiellement explicatives du problème. À partir de cette sélection d'hypothèses, les ingénieurs isolent celle(s) qui a (ont) causé ce défaut, pour finalement ajuster les paramètres de production futurs.

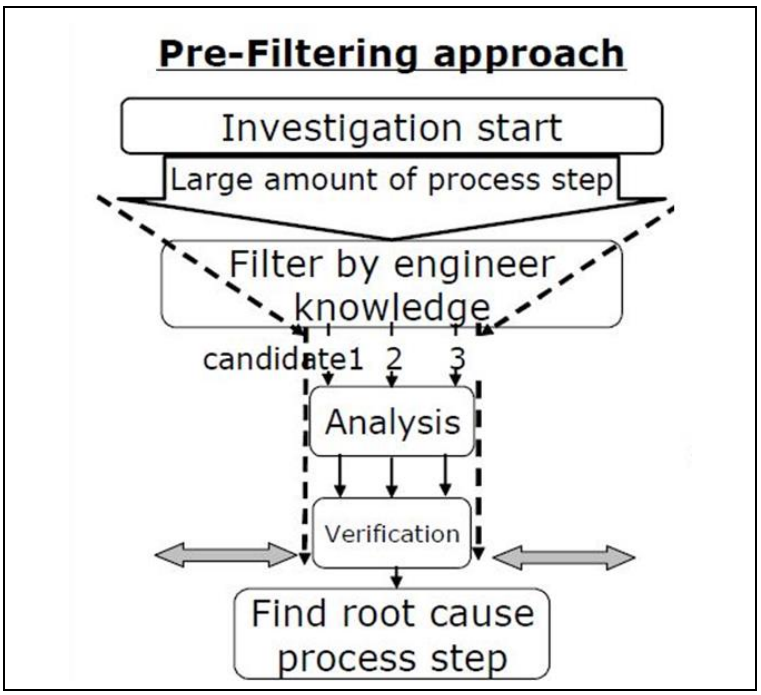


Figure 2.2: Schématisation du fonctionnement des méthodes par pré-filtrage [35]

Ce processus est généralement basé sur les outils de type plan d'expériences *D.O.E* (*Design Of Experiments*) [36], initialement proposé par les travaux de Fisher en 1925 dans l'agriculture. Un plan d'expériences sert, la plupart du temps, à construire un polynôme qui modélise la performance ciblée d'un système. Par exemple, les auteurs de [37] ont utilisé un plan d'expérience, i.e. *D.O.E*, pour améliorer un procédé de lithographie.

Dans le cadre de notre problématique d'identification de causes d'un défaut détecté à une étape *e*, ceci consiste à modéliser le résultat de cette étape *e* en fonction d'un ensemble de variables en entrée. Autrement dit, ceci consiste à faire varier, itérativement, les variables en entrée, et observer le résultat en sortie, jusqu'à identifier la combinaison de valeurs des variables en entrée qui décrivent cette condition de défaut sur le résultat.

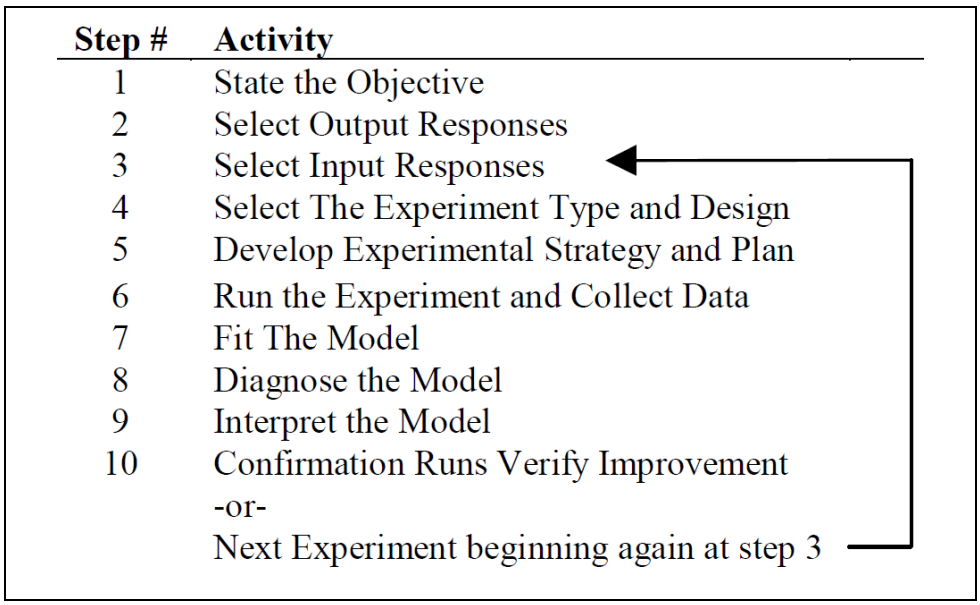


Figure 2.3: Les étapes d'un plan de contrôle D.O.E [38]

Comme illustré sur *Figure 2.3*, une fois l'objectif et la variable de sortie définie, les ingénieurs identifient, à l'étape *3*, *en se basant sur leurs connaissances réunies*, les causes potentielles de ce défaut, *i.e.* les éventuels facteurs causaux de ce défaut. Ces facteurs constituent les hypothèses à tester dans la suite du *D.O.E.* Durant les étapes *4*, *5* et *6*, les ingénieurs réalisent les expérimentations tout en contrôlant les facteurs causaux sélectionnés. Ceci consiste à fabriquer des plaques sous les conditions de l'expérimentation en cours. Par la suite, durant les étapes *7*, *8* et *9*, des analyses statistiques sont appliquées pour identifier le ou les facteurs causaux qui expliquent réellement ce défaut. Finalement, à l'étape *10*, si un facteur causal est identifié comme étant une cause valide, des plaques de tests sont fabriquées pour essayer les actions correctrices proposées. Par contre, si aucun facteur n'est identifié comme cause, une nouvelle itération du processus commence à partir de l'étape *3*.

En pratique, ces méthodes d'analyse par pré-filtrage sont connues pour être très lentes, couteuses et fortement dépendantes des connaissances des ingénieurs, et de leur degré d'expertise [35]. En effet, la réussite d'un plan d'expériences repose en grande partie sur la réussite de sa troisième étape, puisque si l'ensemble des facteurs causaux, initialement sélectionnés, ne contient pas la cause du défaut étudié, toutes les étapes suivantes du *D.O.E* sont inutiles et représenteront des pertes en termes de temps et d'argent. La qualité des résultats et le temps d'aboutissement dépendent, ainsi, de l'ensemble d'hypothèses sélectionnées au départ, qui dépend de la connaissance et la maîtrise de l'ingénieur du processus de fabrication. Avec la complexité croissante des procédés de fabrication en semi-conducteur, il est devenu de plus en plus difficile d'identifier dès le départ, un ensemble pertinent d'hypothèses potentiellement explicatives d'une perte de qualité.

Face à cette complexité des procédés de fabrication et au besoin d'une maîtrise des coûts et des temps de cycle, il est maintenant crucial d'avoir un générateur d'hypothèses qui

permet d'accroitre les chances d'une sélection rapide des bons facteurs causaux dès le départ [33].

Par ailleurs, les outils de contrôle des procédés de fabrication, introduits dans le chapitre précédent, tels que le système *SPC, FDC…* avec les mesures sur les plaques, ou sur les équipement de production, (i.e. en *métrologie*, en inspection de *défectivité*, en *PT*, et *EWS*) fournissent une masse importante de données qui décrit différents aspects du cycle de production, et de l'historique des plaques, et qui constitue donc une mine d'information à explorer pouvant aider les ingénieurs à identifier les causes de défauts détectés.

Ainsi, face aux techniques présentées dans cette section et qui sont basées sur principalement l'expertise humaine pour l'explication d'une défaillance, des techniques se basant sur l'exploration des données sont de plus en plus utilisées pour répondre à des problématiques industrielles. Dans la prochaine section, nous donnons des exemples de travaux appliquant les outils de fouille de données pour répondre à notre problématique d'identification de l'origine d'un défaut.

2.2.2 Approches exploratoires : méthodes basées sur l'exploration des données

Les avancées des techniques de fouille de données, les immenses masses de données collectées grâce aux capteurs disponibles sur presque tous les équipements de production, ainsi que les progrès des performances de calcul des ordinateurs, permettent de proposer un *« générateur automatique d'hypothèses»* [33] pour expliquer un défaut détecté. Contrairement aux techniques traditionnelles de *pré filtrage*, ces méthodes basées sur des outils de fouille de données explorent tout l'espace d'hypothèses, afin d'identifier avec précision la cause potentiellement explicative d'un défaut.

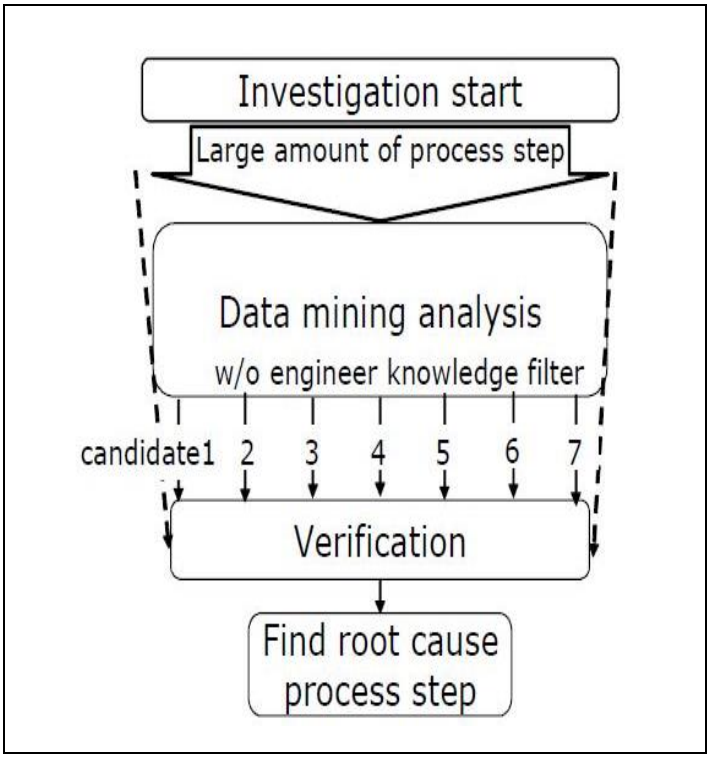


Figure 2.4: Schématisation du fonctionnement des méthodes par fouille de données

Plusieurs méthodes de différents domaines ont été appliquées pour explorer les données et ainsi identifier les causes d'un défaut. Plusieurs travaux reposent sur des techniques d'analyse statistique pour expliquer un défaut. Par exemple, les travaux présentés dans [39] proposent une *analyse statistique par corrélation* pour identifier les paramètres significatifs qui influencent le rendement. Par ailleurs, d'autres travaux ont exploré le domaine de la *théorie des ensembles* pour identifier les équipements problématiques responsables d'une perte de qualité en semi-conducteur [40].

À cause du nombre important de facteurs potentiellement explicatifs et de leurs interactions non linéaires, les méthodes classiques notamment celles qui se basent sur l'analyse statistique sont aujourd'hui insuffisantes. Par exemple, les outils d'analyse disponibles, initialement développés pour permettre aux ingénieurs d'analyser les données collectées et ainsi d'identifier rapidement les causes d'un défaut, comme les équipements de production, génèrent un nombre très important de diagrammes et d'index qui ne peuvent pas être toujours facilement jugés et intégrés par les ingénieurs [23]. Ainsi, à partir des années 2000 on a vu le développement de l'application des méthodes de fouille de données (appelées souvent de *Data Mining* ou *DM*) pour l'explication des défauts.

2.2.2.1 Introduction au domaine de la fouille de données

La naissance des méthodes de fouille de données a été motivée principalement par l'accroissement exponentiel des données collectées dans les entreprises. Ces données disponibles représentent une mine d'informations qui peut assister les utilisateurs pour la prise

de décision. Ces outils ont comme objectif la valorisation des données déjà disponibles dans les systèmes d'informations des entreprises.

Contrairement aux approches qualitatives précédemment présentées, les approches exploratoires se basent sur l'analyse et l'exploration des données pour remonter à la cause d'un problème, et ce en appliquant des méthodes de *fouille de données*. La fouille de données (*Data Mining DM* en anglais) est un ensemble de techniques issues de différents domaines tels que l'intelligence artificielle, les bases de données, et les statistiques [41]. L'objectif principal de ce domaine de recherche est l'exploitation des bases de données de plus en plus répandues et vastes dans différents champs d'activités, comme c'est le cas pour l'industrie du semi-conducteur. En effet, jusqu'aux années 1990, les recherches en bases de données portaient sur l'organisation, la représentation, le stockage… des données. Autrement dit, ces technologies ne permettaient que de retrouver une information stockée, et non pas de découvrir éventuellement des tendances, des régularités, implicitement contenues dans les bases de données [42]. Les techniques automatiques permettant la découverte de cette connaissance à partir des données sont souvent désignées par *l'Extraction des Connaissances à partir des Données ECD* (*Knowledge Discovery from Databases KDD* en anglais) ou encore par la *fouille de données* en référence à un maillon d'un processus plus long qui est l'*ECD*, indépendamment des étapes de prétraitement et de post-traitement. Dans ce manuscrit, nous faisons le choix de distinguer ces deux termes en utilisant *ECD* pour faire référence au processus en totalité, et *DM* pour l'étape centrale de ce processus.

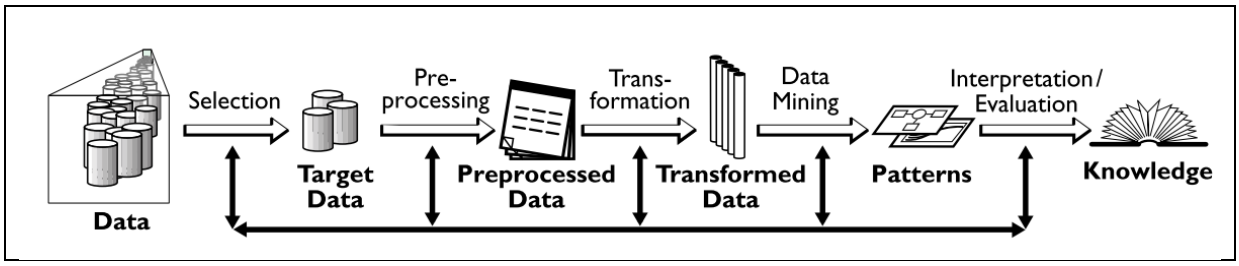


Figure 2.5 : Le processus d'Extraction des Connaissances à partir des données, ECD, proposé par Fayyad en 1996 [43]

Comme illustrée dans la Figure 2.5, le processus *ECD* est une suite d'opérations qui vont de la *sélection des données* jusqu'à *l'évaluation des connaissances extraites*, en passant par la *fouille des données* qui est le cœur du processus. L'*ECD* a été introduit dans [43], comme « *un processus non-trivial d'identification de structures inconnues, valides et potentiellement exploitables dans les bases de données* ».

Comme défini dans [43], les étapes composant un processus *ECD* sont :

1. Compréhension du domaine d'application
2. Création du fichier cible (target data set)
3. Traitement des données brutes (data cleaning and preprocessing)
4. Réduction des données (data reduction and projection)
5. Définition des tâches de fouille de données

6. Choix des algorithmes appropriés de fouille de données
7. Fouille de données (data mining)
8. Interprétation des formes extraites (mined patterns)
9. Validation des connaissances extraites

Ces étapes peuvent être distinguées en trois principales phases :

- ***Les opérations de prétraitement***

Celles-ci concernent l'identification des données qui constitueront les éléments d'information analysés par la suite. Pour cela, il est souvent nécessaire de procéder *(1)* au nettoyage des données afin d'éliminer le bruit et les données incohérentes, *(2)* l'intégration des données si celles-ci proviennent de sources différentes, *(3)* la sélection des données les plus pertinentes pour l'analyse, à savoir la sélection des attributs et des instances qui constitueront la population d'analyse, ainsi que le traitement des données manquantes, et *(4)* leur transformation afin de les mettre en forme pour l'étape suivante d'analyse. Notons que ces opérations sont cruciales pour le bon déroulement du processus global, car la qualité des connaissances extraites à la fin de ce processus dépend de cette phase.

- ***La fouille de données***

Cette étape représente le cœur d'un processus *ECD*, elle permet d'extraire à partir des données des motifs (des régularités) intéressants. On peut distinguer trois familles principales de méthodes de fouille de données selon le type des données analysées, à savoir l'*apprentissage supervisé*, l'*apprentissage non supervisé* et l'*apprentissage semi-supervisé*. Ces familles de méthodes seront présentées, respectivement, dans les sections 2.2.2.1.1, 2.2.2.1.2 et 2.2.2.1.3. L'objectif est d'extraire une nouvelle connaissance potentiellement utiles, qui sera par la suite proposée à la validation.

- ***Les opérations de post-traitement***

Les résultats obtenus grâce à la fouille de données, quel que soit le type de méthode appliquée, ne peuvent être considérés comme une connaissance qu'après avoir été évaluées et validées. Ceci est l'objectif de ces opérations de post-traitement. On s'intéresse, donc, à valider ces résultats, ainsi qu'à les rendre intelligibles à l'humain s'il est l'utilisateur final, ou à les reformuler pour être intégrés à d'autres systèmes automatisés.

2.2.2.1.1 L'apprentissage « supervisé »

Ce type de méthode analyse des données ayant été étiquetées par un expert, (« *un oracle* »). Ainsi chaque instance des données d'apprentissage prend la forme d'un couple

(*observation, étiquette*), avec d'un côté la description d'une situation qu'on appelle *observation* et d'un autre côté une réponse, aussi appelée *label* ou *étiquette* fournie par un *oracle*. Cette réponse peut être qualitative[4] ou numérique. Les algorithmes de fouille de données pour l'apprentissage supervisé cherchent à estimer la réponse, i.e. l'étiquette de nouvelles observations à partir des données d'apprentissage. On parle de *régression* quand le résultat à estimer est numérique, et on parle de *classification* quand le résultat à estimer est qualitatif. Plus précisément, si le résultat est restreint à deux valeurs {vrai, faux} par exemple, on parle alors d'*apprentissage supervisé de concepts*. L'induction d'arbres de décision est une méthode typique pour l'apprentissage supervisé. Ces méthodes sont utilisées dans différents domaines comme la détection de fraudes, le diagnostic médical ou encore l'accord de crédits.

2.2.2.1.2 L'exploration des données ou l'apprentissage « non supervisé »

Elle consiste à analyser des données non étiquetées. Ainsi, les instances sont représentées uniquement par les observations qui les décrivent. Un premier objectif de ces méthodes est *la réduction de la dimension* du point de vue du nombre de variables. Par exemple, l'analyse en composantes principales, notée *ACP* est une technique de projection orthogonale sur des sous-espaces. Un deuxième objectif de cette famille de méthodes est *la distinction de groupes homogènes d'instances* (aussi appelés *clusters*) homogènes d'instances. Ce qu'on appelle *clustering*. Finalement, le troisième sous-type de méthodes d'apprentissage non supervisé est la *recherche de règles d'association*. On cherche alors à identifier des conjonctions significatives d'évènements à partir de la base d'apprentissage. Les premières applications ont concerné l'analyse du panier de la ménagère, où l'objectif est d'identifier les achats concomitants et qui correspondent à des fréquences importantes. Plus de détails sur cette famille de méthodes seront donnés dans la section 2.4. Ce type de méthodes peut être appliqué, par exemple, en marketing pour la segmentation du marché dans le but de la découverte de profils de clients.

2.2.2.1.3 L'apprentissage « semi- supervisé »

Ce troisième type de méthodes se situe entre les deux familles d'apprentissage précédemment citées. En effet, les techniques d'apprentissage jusqu'ici présentées, supposent que les données sont soit étiquetées (pour l'apprentissage supervisé), soit non étiquetées (pour l'apprentissage non supervisé). Cependant, pour diverses applications les données sont un mélange de données étiquetées et de données non étiquetées. Considérer seulement une partie des données (les données étiquetées ou les autres) reviendrait à perdre la connaissance contenue dans la partie des données non sélectionnées. L'objectif de l'apprentissage semi-supervisé est de tirer parti des deux types de données pour résoudre un problème donné. On

[4] « Categorical » en anglais, parfois traduit aussi par « catégoriel ».

parle alors d'exemples supervisés pour les instances étiquetées et d'exemples non supervisés pour les instances non étiquetées [42].

Pour traiter ce problème tout en combinant les deux types de données disponibles, deux approches sont possibles :

(1) D'un point de vue supervisé, on cherchera à exploiter les données supervisées pour étiqueter les données non supervisées. Ceci peut être vu comme une induction supervisée complétée par les données non supervisées. Dans cette catégorie, on peut citer les algorithmes dits *d'auto-apprentissage*, qui fonctionnent comme suit : on commence par apprendre une règle de classification à partir des données supervisées. Ensuite, cette règle est utilisée pour étiqueter un point non supervisé. À partir des données initialement supervisées et du point qu'on vient d'étiqueter, une nouvelle règle est construite. Ceci est fait autant de fois que nécessaire afin d'étiqueter toutes les données non supervisées.

(2) D'un point de vue non supervisé, on cherchera à enrichir une partition construite de façon non supervisée à travers les données étiquetées. Ainsi, ceci peut être vu comme une classification non supervisée facilitée par des contraintes exprimées sous la forme des exemples étiquetés. Dans cette catégorie, on peut citer la méthode « *cluster and label* » qui, comme son nom l'indique, est composée de deux étapes. Premièrement, un algorithme d'apprentissage non supervisé, i.e. *clustering*, est appliqué sur l'ensemble des données disponibles, indépendamment qu'elles soient étiquetées ou pas. Par la suite, les clusters identifiés sont étiquetés selon un vote majoritaire parmi les données étiquetées qui se trouvent dans ce cluster.

2.2.2.2 État de l'art des travaux exploitant des techniques *DM* pour la recherche de causes de pertes de qualité en production industrielle

L'application des techniques de *DM* a permis d'identifier l'origine de la perte de qualité pour des cas industriels et ce de 6 [44] à 10 fois [38] plus rapidement qu'en se basant sur les méthodes de *pré-filtrage*. Ceci est principalement dû à la complexité croissante des technologies de fabrication qui rend les techniques par pré-filtrage moins efficaces. Dans les sous-sections suivantes nous proposons un état de l'art des travaux exploitant des méthodes de *DM* pour répondre à la problématique de remontée aux causes d'une perte de qualité. Cet état de l'art est classé par familles de méthodes.

2.2.2.2.1 Par induction d'arbre de décision

Parmi les outils de fouilles de données les plus utilisés pour l'identification des sources du défaut, on identifie celles *d'induction d'arbre de décision*. L'induction d'arbre de décision est une méthode de fouille de données supervisée, *i.e.* où une étiquette est donnée pour chaque instance analysée. Par exemple, les auteurs de [45] ont utilisé une méthode

d'induction d'arbre de décision pour expliquer des pannes d'équipements à travers l'analyse des données de maintenance et en choisissant comme valeur cible, l'occurrence d'une panne sur un équipement ou pas.

Par ailleurs, dans une optique d'expliquer une perte de qualité, l'étiquette représente généralement le résultat de l'étape de contrôle dans laquelle le défaut qu'on cherche à expliquer est détecté, *i.e.* si *oui* ou *non* les mesures collectées pour la plaque (ou le lot) respectent les limites de contrôle à cette étape.

L'induction d'arbre de décision est un algorithme de classification, *i.e.* d'apprentissage supervisé. Il peut être utilisé (1) pour prédire les tendances futures (outil de classification) ou (2) pour extraire des motifs qui décrivent les classes (outil d'exploration). Dans le cadre de l'explication de défauts, cette méthode est utilisée selon leur deuxième aspect, *i.e.* comme *un outil d'exploration* des données et non comme un outil de classification.

Son succès est dû à sa facilité d'interprétation, puisque le résultat de ces méthodes est une arborescence (c'est à dire un graphe orienté, simple, connexe et sans circuits). Cette structure est orientée du haut vers le bas, et chaque nœud représente un attribut sur lequel une décision est prise. Les feuilles de cet arbre représentent le résultat, *i.e.* une valeur de l'attribut choisi comme étiquette.

L'auteur de [46] propose d'appliquer l'algorithme d'induction d'arbre de décision CART pour l'identification des *étapes de production* et *des équipements* responsables des pertes de rendement. La Figure 2.6 donne une description des données utilisées. La première colonne de cette matrice stocke l'identifiant du lot ; la deuxième contient l'étiquette qui indique si le lot est finalement accepté ou non, suite aux tests EWS. Les colonnes restantes représentent les *N* étapes de production. Pour chaque étape, l'équipement utilisé pour traiter les plaques d'un même lot est marqué dans la cellule correspondante. Par exemple, les plaques du lot '*AAAAAAA'* ont été traitées par l'équipement '*RCAU-501*' pour la première étape '*STEP1*'. L'analyse de ces données par induction d'arbre de décision a permis d'identifier une *étape de lithographie* réalisée par *l'équipement STEPPER-507* comme étant la cause explicative des défauts de ce cas.

LOT-ID	RESPONSE	STEP1	STEP2	.....	STEPN
AAAAAAA	good	RCAU-501	NITD-501		MTRP-501
BBBBBBB	good	RCAU-501	NITD-503		MTRP-502
CCCCCCC	good	RCAU-502	NITD-501		MTRP-501
DDDDDDD	good	RCAU-502	NITD-501		MTRP-503
EEEEEEE	bad	RCAU-501	NITD-501		MTRP-503
FFFFFFF	good	RCAU-501	NITD-501		MTRP-503
GGGGGGG	bad	RCAU-501	NITD-501		MTRP-501
HHHHHHH	good	RCAU-502	NITD-501		MTRP-503

Figure 2.6 : Description des données d'analyse pour identifier l'étape et l'équipement potentiellement problématiques [46]

D'autres applications d'inductions d'arbre de décision ont cherché à améliorer localement un processus de fabrication. Par exemple, les auteurs de [47] appliquent un autre

algorithme d'induction d'arbres de décision permettant la prise en compte de variables catégoriques et continues, dénommé C4.5, pour améliorer le processus de nettoyage, ou encore de lithographie en [48]. La méthode proposée dans ces deux articles, a permis aux auteurs de modéliser les données, et ainsi d'identifier *les paramètres* qui caractérisent le fonctionnement idéal, et ceux qui caractérisent le fonctionnement problématique, pour mieux maîtriser leurs processus.

2.2.2.2.2 Par induction d'arbre de régression

Un autre type de méthodes d'induction d'arbre de décision est celles d'*induction d'arbre de régression*. À la différence des arbres de décision classiques où l'étiquette est qualitative, l'étiquette des arbres de régression prend ses valeurs dans un domaine continu, par exemple le rendement résultant des mesures des tests EWS. Ainsi, les feuilles de l'arbre de décision ne sont pas des classes mais des valeurs de rendement. Les auteurs de [44] ont appliqué une méthode d'induction d'arbre de régression pour modéliser le rendement. L'analyse, par cette méthode, de l'historique de 58 lots, composé chacun de 50 plaques leur a permis d'induire un arbre de régression indiquant *l'étape de production* problématique et *l'équipement* correspondant au faible rendement.

Toujours en se basant sur l'application des algorithmes d'induction d'arbre de régression, et au lieu de chercher à construire un arbre de régression sur les valeurs d'une mesure numérique de la performance comme le rendement, les auteurs de [26] proposent de prendre l'écart par rapport à la moyenne globale comme valeur cible. Les données à analyser tracent l'historique des plaques, en indiquant, pour chaque étape de production, l'équipement qui l'a traitée. En plus de cet historique, la mesure de la performance donnée par le test final est notée comme la cible de l'analyse, l'étiquette. L'outil proposé permet d'induire un arbre de régression afin de construire des règles de décision avec au plus 2 conditions. Ces règles sont filtrées selon des critères de sélection définis par les ingénieurs (un nombre minimal de plaques / lots couverts, l'écart minimal par rapport à la moyenne, …). L'application de cette méthode a permis d'expliquer des pertes de qualité par des règles qui identifient un ou deux *équipements de production* intervenus à une ou deux *étapes différentes* comme une cause explicative.

Par ailleurs d'autres applications d'algorithmes d'induction d'arbre de régression ont permis par exemple aux auteurs de [49] d'identifier les paramètres du test paramétrique PT qui influent le plus sur le rendement.

2.2.2.2.3 Par combinaison d'induction d'arbres et d'autres méthodes

D'autres travaux ont essayé d'enrichir l'utilisation d'induction d'arbre de décision, en combinant son utilisation avec d'autres méthodes. Par exemple les auteurs de [33] proposent une méthode de fouille de données hybride [50] qui combine l'utilisation *des cartes de Kohonen*, aussi appelée *SOM NN (Self Organized Map Neural Networks)* [51], avec *l'induction de règles* au moyen de l'outil *C4.5* de construction d'arbres de décision [52] pour

identifier les origines de pertes de rendement. Les auteurs utilisent pour cela un outil appelé *CorDex* [53].

Premièrement, les auteurs [33] proposent d'identifier des groupements naturels sur les données, à travers l'application de la méthode *SOM NN*. Pour cela, ils proposent d'analyser l'historique sur deux mois de 2500 plaques, décrites par 133 paramètres de trois types de données :

(1) **Les données de tests** sur les plaques, qui indiquent le nombre de puces qui ont réussies (ou pas) les tests sur la plaque.
(2) **Les données du processus de contrôle**, qui indiquent les mesures collectées lors des étapes de contrôle qualité intermédiaires et ce sur 8 sites de mesure sur chaque plaque.
(3) **Les données des étapes de production** qui concernent le fournisseur de plaque vierge, la position d'une plaque dans son batch, la machine qui a traité la plaque à une étape de production donnée, ...

La méthode *SOM NN* permet de faire une régression non linéaire multivariée, dont le résultat, illustré en *Figure 2.7*, représente *« une carte de clusters »*, qui est une représentation en deux dimensions des données analysées et qui maintient au mieux les relations internes de celles-ci sur les 133 paramètres initiaux.

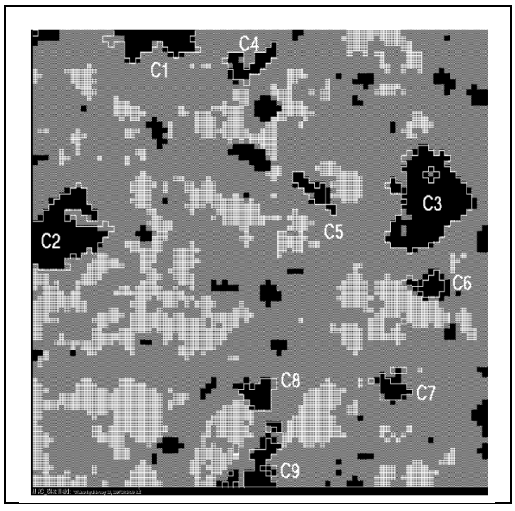


Figure 2.7: La carte des clusters obtenue par SOM NN pour l'explication des pertes de rendement [33]

Sur cette carte, on peut distinguer des groupes, appelés *clusters*, avec différents niveaux de gris, relatifs aux niveaux de rendement des plaques le composant. Les clusters de couleur noire représentent des plaques avec un niveau faible de rendement (les plaques appartenant aux 12.5% des plaques avec le plus bas rendement), alors que ceux en clair représentent les plaques ayant des taux de rendement plus élevés. La taille d'un cluster

indique la force de la relation que le cluster représente. Le nombre de clusters différents indique le nombre de relations statistiquement différentes qui existent. Finalement, la relation spatiale entre les clusters indique leur degré de similarité.

Ensuite, à travers *l'induction de règle*s, les auteurs proposent d'expliquer chacun de ces *clusters* en identifiant les paramètres les plus représentatifs de ces clusters. Cette deuxième étape permet de donner une signification directe aux clusters identifiés précédemment et qui peuvent, ainsi, être utilisés pour la suite d'un *D.O.E*, par exemple, pour valider ces règles et mettre en œuvre les actions correctrices correspondantes.

Les règles induites ont permis aux auteurs de [33] d'identifier les causes de leur problème : d'un côté, des *matériaux* utilisés pour le *substrat de silicium*, et d'un autre côté, *des étapes du procédé d'épitaxie*. Notons que les règles identifiées ont permis d'expliquer 50% des cas de perte de rendement. Permettant, ainsi, de réaliser des améliorations en termes de taux de rendement, et de temps de résolution du problème, et donc un bénéfice financier.

2.2.2.2.4 Par d'autres méthodes de fouille de données

Bien que les méthodes basées sur l'induction d'arbre de décision soient les plus fréquemment utilisées pour remonter à la cause d'un défaut, d'autres méthodes de fouille de données ont été explorées pour répondre à cette problématique. Par exemple, les auteurs de [54] proposent l'utilisation des réseaux bayésiens, alors que les auteurs de [55] proposent l'application des techniques de programmation génétique (GP) [56], ou encore des méthodes de fouille de séquences [57] en [58].

Par ailleurs, les techniques de recherche de règles d'association restent les moins explorées pour la problématique d'amélioration du rendement. Ces techniques ont été initialement proposées dans le domaine du marketing, pour identifier les produits les mieux vendus et identifier des règles caractérisant les achats des clients [59] [60]. Depuis, ces méthodes ont été appliquées sur plusieurs autres problématiques et ont montré leur efficacité dans divers domaines : secteur banquier [61], enseignement [62], analyse des comportements animaliers [63], sécurité des réseaux informatiques [64] [65] ou encore prévention des crimes [66] [67].

En microélectronique, les auteurs de [68] ont utilisé les règles d'association pour identifier des paramètres intéressants pour l'amélioration du rendement. Les auteurs de [23] ont aussi exploré cette famille de méthodes en proposant une adaptation de l'algorithme classique *APRIORI* [60] pour intégrer des critères de sélection de règles. Les règles identifiées permettent de mettre en exergue le(s) équipement(s) de production problématique(s).

2.3 CRITIQUES ET POSITIONNEMENT

2.3.1 Choix d'une variable à expliquer

Pour répondre à la problématique d'identification d'une cause explicative d'un cas de perte de qualité, les travaux traditionnels ont souvent considéré le résultat du contrôle comme variable à expliquer en ignorant les mesures individuelles, et en ne considérant que le label *OK* ou *OOC (Out Of Control* pour faire référence aux produits de qualité problématique*)* pour faire référence à un produit respectivement de bonne ou de mauvaise qualité. Ces étiquettes sont obtenues sur la base de limites de contrôle définies par les ingénieurs métiers. Souvent ces limites ne sont pas exhaustives, et sont souvent révisées pour intégrer de nouvelles contraintes et/ou connaissances.

Par ailleurs, d'autres travaux ont choisi de ne pas se limiter à cet étiquetage basé sur l'expertise, et d'exploiter les mesures collectées aux étapes de contrôle, stockées dans les systèmes d'information, pour identifier les classes à expliquer, à travers ***une analyse spatiale des mesures de contrôle***. À titre d'exemple, les travaux de [8] ont montré l'intérêt d'une telle stratégie, afin de mieux caractériser le problème et ainsi de mieux l'expliquer par la suite.

→ Pour nos travaux, nous proposons de combiner ces deux approches en enrichissant la partition des produits basée sur l'expertise métier (deux groupes de produits *OK* ou *OOC*) grâce à une analyse des mesures de contrôle des plaques correspondantes, afin de mieux définir la classe à expliquer.

2.3.2 Prise en compte des modes de production

De plus, les approches traditionnelles ont souvent situé cette problématique dans un modèle de production ne prenant en compte que des ensembles d'équipements, comme illustré dans la Figure 2.8. Un ensemble de matériaux est en entrée d'un cycle de production. Ces matériaux sont traités en différentes étapes de production par différents équipements. À la fin du processus de fabrication, la qualité des produits est mesurée. On cherche alors à identifier un ou plusieurs équipements problématiques, pour expliquer les produits défectueux. Souvent les auteurs, comme [23], adoptent cette approche en y faisant référence par le terme de problème d'identification d'ensemble d'équipements problématiques ou, en anglais par le terme « r*oot-cause machineset identifiation problem* ».

Par ailleurs, dans un site de production, comme celui de *Crolles 300mm* de *STMicroelectronics*, caractérisé par un grand nombre de type de produits différents et par un volume faible, les équipements sont partagés pour faire des traitements différents avec des réglages qui varient assez fréquemment en fonction du type du produit en cours. Ainsi, selon les conditions de production sur un même équipement de production, une plaque peut être de bonne ou de mauvaise qualité. Identifier uniquement l'équipement problématique ne suffit

donc plus à expliquer précisément un cas de perte de qualité dans un contexte de production complexe.

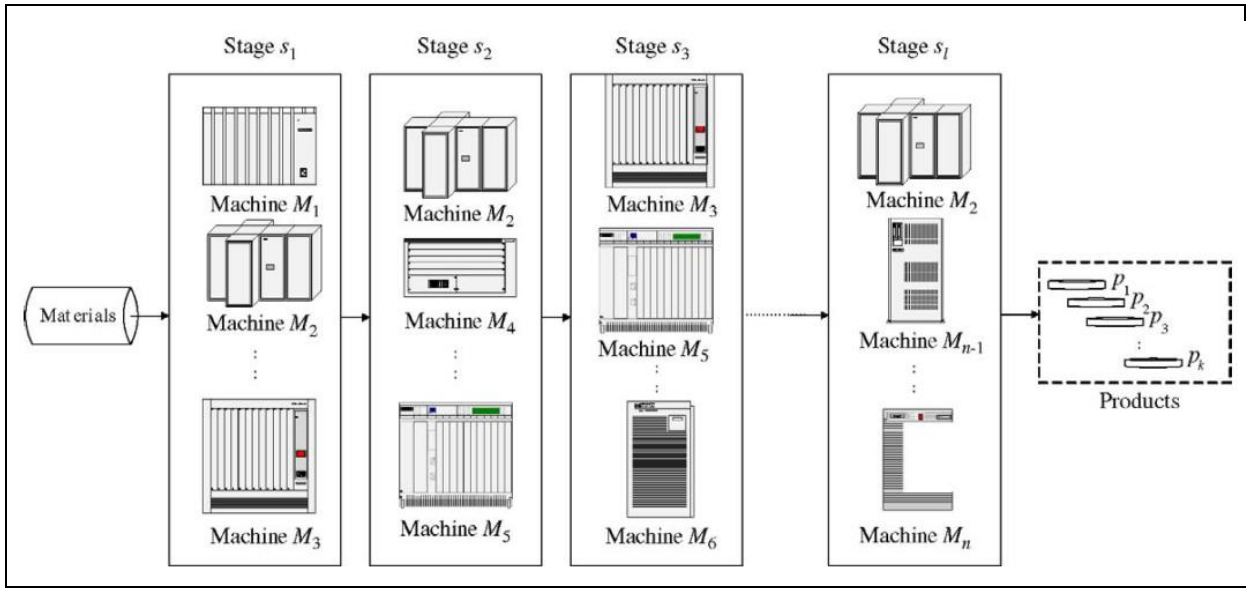


Figure 2.8 : Représentation d'un processus de fabrication [23]

Les travaux de [23] ont montré que généralement les fonctions d'un équipement de production sont indépendantes, i.e. un équipement peut produire des plaques problématiques à cause d'une fonction particulière dans une étape de production, et en même temps être utilisé plus tard, à travers une autre fonction et produire de bonnes plaques. Pour approfondir cette idée, nous proposons de descendre au niveau de la chambre d'un équipement et plus particulièrement à ses paramètres qui décrivent son fonctionnement interne lors du traitement d'une plaque, collectés en temps réel à travers l'outil de contrôle des équipements, FDC.

→ Pour nos travaux nous proposons d'analyser les données *FDC* de chaque équipement afin de pouvoir expliquer un défaut à travers des **modes de production**, où chaque mode de production est une combinaison de paramètres de production avec des intervalles de valeurs spécifiques.

Notons que quelques travaux ont essayé d'expliquer des défauts à travers des conditions particulières sur les paramètres de production d'un équipement mais ils se sont focalisés sur l'analyse d'une seule étape de production, comme dans [47] et [48], notamment à cause de la forte dimensionnalité des données même à une seule étape, ce qui rend leur combinaison sur différentes étapes impossibles avec les outils traditionnels, tels l'induction d'arbres de décision.

2.3.3 L'exploitation des données relatives aux produits non contrôlés

Par ailleurs, les travaux traditionnels ont souvent ignoré les données disponibles et relatives aux plaques présentant des mesures manquantes. Par exemple, dans l'explication de défauts locaux, comme dans [47], les plaques non mesurées à l'étape de contrôle qualité correspondante sont supprimées des données d'analyse. Rappelons que pour des raisons d'optimisation de coût et de temps de cycle, seul un échantillon réduit de plaques d'un échantillon de lots est mesuré durant les étapes de contrôle intermédiaire. Ainsi, supprimer les historiques des produits n'appartenant pas à l'échantillon sélectionné représente une perte importante d'informations potentiellement utiles.

Des méthodes de « métrologie virtuelle » proposent d'estimer les mesures manquantes. Comme défini par [69], la métrologie virtuelle est ‘‘ *une nouvelle technique de prédiction de la performance de la plaque à partir des données d'état d'un équipement''*. Par exemple, plusieurs travaux proposent d'appliquer des outils de régression pour estimer ces valeurs [70] [71]. Notons que les estimations des mesures manquantes doivent être intégrées avec précaution dans la démarche d'identification des causes explicatives.

→ Dans le cadre de cette thèse, nous avons choisi de ne pas estimer les mesures manquantes afin d'éviter d'introduire du bruit dans l'analyse. Pour autant, nous choisissons de ne pas supprimer tout l'historique des plaques présentant des données manquantes à des étapes de contrôle qualité, puisque les données décrivant l'état de fonctionnement des équipements sur lesquels ces plaques non mesurées sont passées, sont normalement disponibles. On propose de **les analyser afin d'enrichir la construction des modes de production** candidats à l'explication d'un défaut.

2.3.4 Le type de méthodes de fouille de données

De plus, les méthodes traditionnelles ont toutes abordé le problème de façon supervisée, en cherchant à identifier directement les paramètres qui séparent les plaques problématiques des autres, en utilisant directement le résultat de contrôle. Même si ce type de méthodes reste le plus évident à utiliser, d'autres familles de méthodes peuvent être utilisées pour répondre à cette problématique.

→ Dans le cadre de cette thèse, nous explorons des méthodes d'apprentissage non supervisé. Pour cela, nous proposons de *générer de façon non supervisée*, pour chaque étape et pour chaque équipement, *des modes de fonctionnement* (caractérisés par un ensemble de paramètres propre à l'équipement), *i.e.* en nous basant sur l'exploration des données décrivant cette étape *indépendamment du*

résultat de contrôle, afin d'identifier des groupements naturels de produits selon des similarités sur les conditions de production qu'elles ont connues.

2.3.5 Vers une analyse non supervisée des modes de production

La méthode d'analyse que nous proposons, nommée *CLARIF* pour « *CL*ustering and *A*ssociation *R*ules *I*dentiFier », s'inscrit dans la famille de méthodes d'analyse *a posteriori*, « *post-hoc* », et propose une contribution pour mieux exploiter les outils de fouille de données pour répondre à notre problématique d'identification des causes de perte de qualité, en industrie du semi-conducteur. Nos travaux sont ainsi guidés par les orientations suivantes :

- Identifier des causes explicatives d'une perte de qualité qui descendent au niveau des paramètres des équipements de production, i.e. des modes de production.
- Proposer une analyse spatiale des données de contrôle afin de mieux définir le problème à expliquer.
- Utiliser des données de production des plaques non mesurées pour enrichir la génération des modes de production candidats pour expliquer les défauts.
- Utiliser des méthodes d'apprentissage non supervisé.

Pour cela, nous proposons une méthode hybride de fouille de données, qui combine trois techniques différentes de fouille de données pour expliquer un cas de perte de qualité : les techniques d'*apprentissage non supervisé (i.e. clustering)*, *l'identification de règles d'association* et *l'induction d'arbres de décision*. Les trois techniques proposées sont choisies pour leur complémentarité et leur efficacité. Pour autant que nous sachions, il n'existe pas de méthode dans la littérature qui se propose de résoudre ce problème en combinant ces techniques d'exploration de données.

Premièrement, les techniques d'apprentissage non supervisé sont sélectionnées pour leurs efficacités à réduire la complexité et la dimension des données en entrée. Ceci nous permettra (1) d'identifier *des modes de fonctionnement sur les équipements* de production qui seront considérés plus tard comme des causes candidates pour expliquer les défauts, et (2) de faire une analyse spatiale des données des contrôles comme ceux du test *EWS*.

Deuxièmement, les outils d'identification de règles d'association sont sélectionnés pour l'identification de relations causales naturellement très compréhensibles, entre les modes de fonctionnement identifiés et la classe de défaut. Les premiers travaux appliquant les techniques de fouille de règles d'association dans le secteur de la microélectronique [68] [23] ont montré la pertinence de l'utilisation des méthodes de recherche d'association, et le besoin de travaux complémentaires dans ce domaine.

Finalement, on propose d'induire des arbres de décision, afin de faciliter l'explication des causalités identifiées. Cet algorithme de classification supervisée est, ici, utilisé comme un

outil d'interprétation des connaissances précédemment extraites. Il est choisi pour sa facilité d'interprétation et d'utilisation par les utilisateurs.

Les trois méthodes de fouille de données composant notre proposition seront présentées en détail dans la section suivante.

2.4 APPROFONDISSEMENT DE L'ETAT DE L'ART SUR LES METHODES UTILISEES DANS NOTRE APPROCHE

2.4.1 Clustering

L'objectif du *Clustering*, aussi appelé *classification* (*non supervisée*), *regroupement*, ou encore *segmentation* est de distinguer des *groupes*, *clusters*, les plus homogènes possibles, à partir d'un ensemble d'observations disponibles, sans connaissance *a priori*. Un cluster peut donc être défini comme étant un ensemble d'objets tel que les objets d'un même cluster sont similaires et ceux de clusters différents sont dissimilaires. La similarité entre des observations est définie à l'aide d'une fonction appelée « distance ». Ainsi, un cluster est une agrégation de points tels que la distance entre deux points quelconques d'un cluster est inférieure à la distance entre un point quelconque du cluster et tout point extérieur. Notons qu'une autre caractérisation des clusters, à l'aide d'une fonction « densité », peut aussi être utilisée. Dans ce cas, un cluster est une région d'un espace multidimensionnel contenant une forte densité de points, et qui est séparée des autres clusters par des régions possédant une faible densité de points.

Soit $S = \{x_1, \dots x_m\}$ l'ensemble fini des m objets à partir desquels on cherche à construire des clusters d'objets similaires. On cherche, donc, à construire une *partition* de *S*. Une partition *P* est définie *comme un ensemble de parties de S, non vides et disjointes deux à deux, et dont l'union est S. Si s désigne un élément de S, il existe donc un unique élément de P comprenant s.* On notera P_k une partition qui identifie *k* clusters à partir des données.

Cette problématique est abordée en littérature, principalement, de deux façons différentes : soit en construisant une hiérarchie de classes ; soit en cherchant directement un nombre *k* de classes. Ces deux approches sont détaillées dans les sous-sections suivantes.

2.4.1.1 Clustering hiérarchique

Un algorithme de clustering hiérarchique fournit en sortie, à partir de l'ensemble de données, une arborescence de clusters, où les feuilles représentent les objets et les nœuds intermédiaires représentent les clusters. Cette structure, appelée un *dendrogramme*, est illustrée dans la Figure 2.9.

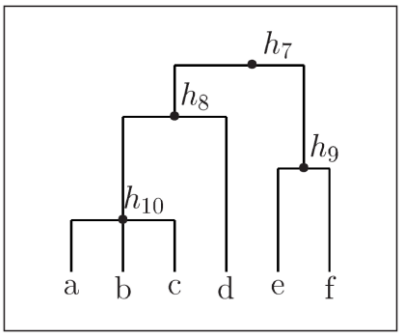


Figure 2.9 : Un dendrogramme sur S = {a, b, c, d, e, f}. [72]

Un *dendrogramme* peut être vu comme une suite de partitions emboitées, allant de la plus fine P_m, dans laquelle chaque individu constitue une classe, à la plus grossière P_1, dans laquelle tous les individus constituent une seule et unique classe. Une suite de partitions emboitée est appelée *chaine*. Plus formellement, une chaine dans l'espace des partitions possibles de *S* est un ensemble de partition { P_1,…, P_r } tel que pour *i* variant de *1* à *r-1*, on a P_i est plus fine que P_{i+1}. Notons qu'une partition P_i est dite plus fine qu'une partition P_j si et seulement si tout cluster de P_j est un cluster de P_i ou l'union de plusieurs clusters de P_i.

Ainsi, construire une hiérarchie, un *dendrogramme*, sur un ensemble d'exemple est donc équivalent à trouver une chaine de partitions sur cet ensemble.

Une hiérarchie, un dendrogramme, noté *H*, sur *S* est un sous-ensemble de toutes les partitions possibles de S tel que :

- pour tout élément s de S, $\{x\} \in H$;
- pour tout couple d'éléments (clusters) *h* et *h'* de *H* avec $h \neq h'$, on a :
  - soit $h \cap h' = \emptyset$,
  - *soit* $h \cap h' \neq \emptyset$, *alors* $h \subset h'$, ou $h' \subset h$

Si on reprend l'exemple donné dans [72], dont le dendrogramme résultant est représenté dans la Figure 2.9. À partir de l'ensemble des observations *S={a, b, c, d, e, f}*, l'algorithme a ditingué une hiérarchie avec 5 partitions différentes à partir des données, définie comme la chaine de partitions suivante :

(a, b, c, d, e, f)
(a, b, c, d), (e, f)
(a, b, c),(d), (e, f)
(a, b, c),(d), (e), (f)
(a), (b), (c),(d), (e), (f)

L'identification de cette chaine de partitions peut se faire de deux façons différentes : *(1)* **de façon descendante**, ce sont les algorithmes divisibles. On commence donc à partir de la partition P_1 (regroupant toutes les observations dans un seul *cluster*) et on cherchera à identifier des sous-groupes, et ainsi de suite jusqu'à arriver à la partition la plus fine P_m. *(2)*

de façon ascendante, ce sont les algorithmes agglomératifs. Pour ce deuxième type, on part de la partition la plus fine P_m et on cherche à créer des groupes moins fins, *i.e.* en regroupant des clusters les plus similaires dans un même cluster, jusqu'à l'obtention la partition P_1.

Indépendamment de l'approche ascendante ou descendante choisie, pour grouper ou diviser des clusters, les algorithmes peuvent se baser sur des critères différents à partir d'une mesure de dissimilarité *d*.

- **Le critère du lien minimum**

Ce critère aussi appelé, « *single-linkage* », lien simple, ou du plus proche voisin, définit la distance entre deux clusters *A* et *B* par :

$$d(A,B) = \min\{\, d(a,b) | a \in A \; et \; b \in B\}$$

Ce critère permet la détection de clusters courbés, ainsi que la détection des *outliers*, mais présente un problème de chaînage, où des clusters distincts peuvent être groupés dans un même cluster s'il existe un ensemble de points construisant une chaine qui relie les deux clusters.

- **Le critère du lien maximal**

Ce critère aussi appelé, « *complete-linkage* » considère, inversement au premier critère, la distance avec le voisin le plus éloigné. Ainsi, il est défini comme suit :

$$d(A,B) = \max\{\, d(a,b) | a \in A \; et \; b \in B\}$$

Ce critère permet l'identification de clusters compacts avec des diamètres égaux, et évite le problème de chaînage, mais est sensible aux outliers.

- **Le critère de la distance moyenne**

Aussi appelé « *average-linkage* », ce critère considère la moyenne des distances entre les éléments des clusters. Il est définit comme suit :

$$d(A,B) = \frac{1}{|A| * |B|} * \sum_{\forall a \in A} \sum_{\forall b \in B} d(a,b)$$

- **La méthode par centroïde**

Pour mesurer la distance entre deux clusters *A* et *B*, ce critère considère la distance entre les centroïdes respectifs, c_A et c_B. Il est définit comme suit :

$$d(A,B) = d(c_A, c_B)$$

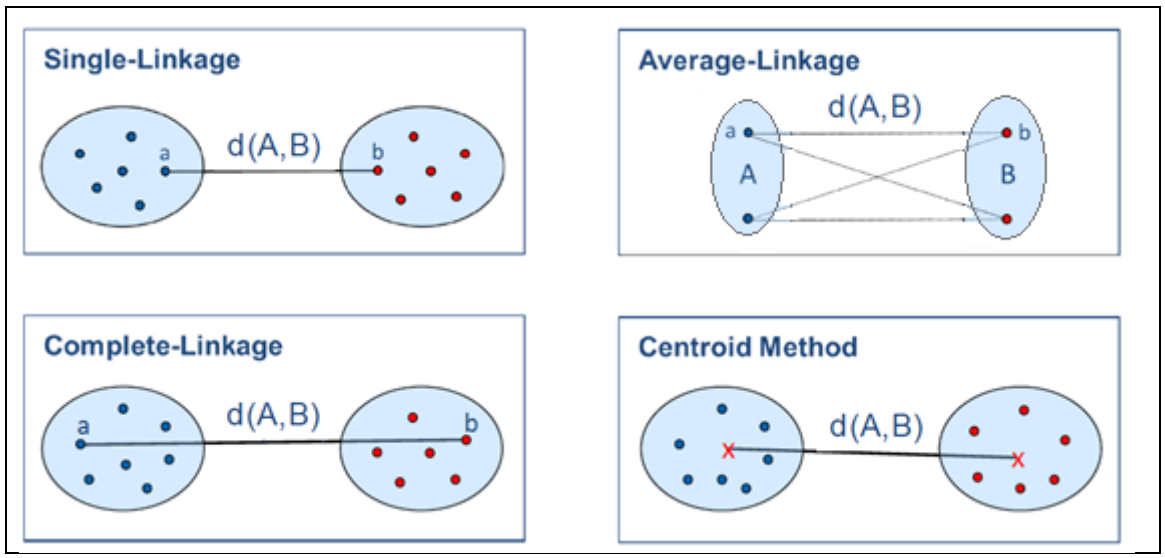


Figure 2.10 : Quatre exemples de critére de dissimilarité

2.4.1.2 Clustering partitionnel

Le résultat d'un algorithme de clustering partitionnel est une seule et unique partition, à la différence du clustering hiérarchique qui propose une chaine de plusieurs partitions. Ainsi, à partir des données en entrée, on cherche à identifier une partition P_k qui identifie k clusters (k étant un paramètre fixé *a priori*) à partir des m observations disponibles. L'algorithme *k-means*, ou *k-moyennes* est un exemple d'algorithme de clustering partitionnel.

À partir d'une partition initiale, distinguant k clusters, celle-ci est repartitionnée itérativement, afin d'obtenir une partition optimisant un objectif spécifié. Le principe est la réallocation autour de centres mobiles. L'algorithme commence par sélectionner k points représentatifs des k clusters, et chaque observation est attribuée au point le plus proche parmi les k points choisis initialement. Lors de l'itération suivante, les points représentatifs sont recalculés en fonction de la composition des clusters correspondants, et les observations sont réattribuées aux nouveaux clusters selon leur proximité avec les nouveaux points représentatifs.

L'avantage de ce type de méthode est la rapidité ainsi que la facilité de mise en œuvre, permettant le traitement de grandes bases de données. Notons qu'un inconvénient de ce type de méthode est que la qualité de la partition finale dépend fortement de la partition initiale, *i.e.* les k points représentatifs sélectionnés au départ. Ainsi, la partition obtenue n'est pas forcément la meilleure puisqu'il peut s'agir d'un optimum local, résultant de la partition initiale. Par ailleurs, un inconvénient pratique est le choix *a priori* du nombre k de clusters à définir.

Afin de remédier à ce problème, des travaux comme [73] proposent de générer plusieurs partitions en faisant varier k, et en mesurant pour chaque partition un indice de

validité de cluster, « *Cluster Validity Index* » *CVI*, afin de choisir finalement la partition qui optimise cet indicateur. Cet indice permet d'identifier la meilleure partition, celle qui optimise à la fois la compacité et la séparation des clusters identifiés, c'est à dire la partition qui minimise la distance intra-cluster, entre les observations d'un même cluster, et qui maximise la distance inter-clusters, entre les observations de différents clusters.

Ce type de méthodes de clustering attribue une observation à un seul cluster à la fois, c'est ce qu'on appelle en anglais *hard-clustering*. Ceci est pertinent quand les observations peuvent être bien séparées. Par contre, si les observations ne peuvent être bien séparées, les points aux bords des clusters sont généralement attribués presque au hasard.

Par ailleurs, d'autres méthodes autorisent une observation à appartenir à plusieurs clusters, permettant ainsi de mieux gérer les observations qui ne sont pas parfaitement séparables. C'est ce qu'on appelle le clustering flou, *fuzzy clustering*, qui se base sur la théorie des ensembles flous introduite par Lotfi Zadeh en 1965 [74]. Ainsi, en *Fuzzy-Clustering*, il n'y a pas une division stricte des clusters, puisqu'un objet appartient à tous les clusters d'une partition avec des degrés d'appartenance allant de 0 à 1.

D'autres méthodes de clustering existent. Par exemple, les algorithmes de clustering à base de graphe, où les données sont transformées en un graphe connecté, tel que les observations sont représentées par des nœuds du graphe. Les nœuds sont connectés entre eux à travers des arrêtes, et chaque arrête est caractérisée par un poids, qui est la distance entre les nœuds qu'elle relie. Le principe de ces méthodes est d'éliminer des branches, à partir de l'arbre recouvrant minimal[5] du graphe de départ, afin de distinguer des clusters.

D'autres méthodes de clustering se basent sur la densité, appelées « *density based clustering* ». Le principe est qu'un cluster est composé d'un ensemble dense d'observations, alors que les points bruits, les *outliers*, sont dans des zones hors clusters et de faible densité. Ainsi, une approche naïve consiste à définir un nombre minimal de « voisins » pour composer un cluster. Les algorithmes comme *DBSCAN* [75] approfondissent cette idée afin de mieux gérer les observations qui sont aux extrémités des clusters et qui par définition ont moins de voisins que des observations au centre. Ce type de méthode a l'avantage d'être résistante aux observations bruitées.

[5] Un arbre recouvrant minimal d'un graphe connexe G=(V,E) est un sous graphe connexe, ayant un poids minimal, contenant tous les nœuds de G et n'ayant aucun cycle. Il peut être calculé avec des algorithmes comme *Prism*, *Kruskal*…

2.4.2 La recherche de règles d'association

Le but des algorithmes de recherche de règles d'association est de trouver toutes les règles d'association intéressantes. Les règles d'associations expriment des corrélations présentes dans les données. Les données sont décrites par des attributs qu'on appelle des *items*. Une règle d'association est une application de la forme $X \rightarrow Y$ *où* X *et* Y sont des items.

Les données d'analyse sont décrites à travers des transactions (lignes), où des variables booléennes dénotent la présence (vrai) ou pas (faux) d'items. Par exemple, les achats d'un client peuvent être décrits par les types d'items qu'ils contiennent. Dans ce contexte, on peut être intéressé à découvrir les combinaisons de produits achetés simultanément.

Pour cela, les algorithmes de recherche de règles d'association passent généralement par deux phases : *(1)* la recherche de tous les ensembles d'*items* fréquents aussi appelés *motif* ou *itemset fréquent*, et *(2)* la génération des règles à partir des différents *itemsets fréquents* identifiés. Plus de détails sur ces deux étapes seront donnés dans les sous-sections suivantes.

2.4.2.1 La recherche de motifs fréquents

Un *motif* est un ensemble d'items, i.e. d'attributs dont la valeur doit être vraie. On appelle la *couverture d'un motif* l'ensemble des exemples couverts par tous les items du motif. Par ailleurs, le *support d'un motif* est le cardinal de la couverture de ce motif. La *fréquence d'un motif* est égale à son support divisé par le nombre total d'exemples. Notons que souvent la fréquence est appelée support dans certains ouvrages. Dans la suite de ce manuscrit, nous ferons référence par le *support* d'un motif à sa *fréquence*. Un motif est considéré comme fréquent si son support est supérieur à un seuil minimal défini *a priori*.

L'identification de l'ensemble des itemsets fréquents, aussi appelés *FIS* pour « *Frequent ItemsSets* », est la tâche centrale d'un algorithme de recherche de règles d'association. Elle est très couteuse en calculs puisque les données peuvent être en très grand nombre et impliquer un important nombre d'itemsets parmi lesquels il faut identifier ceux qui sont fréquents. Soit A l'ensemble des attributs décrivant une base de données, et D la taille de A. L'ensemble des *motifs* qu'on peut construire à partir de A est égal à l'ensemble des parties de A de taille 1, + l'ensemble des parties de A de taille 2, … + l'ensemble des parties de A de taille D. Ce qui revient à $2^D - 1$ motifs possibles.

C'est pourquoi les travaux de recherche se sont focalisés sur l'optimisation de cette phase. Deux type d'approches sont disponibles : *(1)* réduire le nombre des itemsets candidats dont il faut vérifier le support, typiquement à travers l'application de la propriété *d'antimonotonie* qui stipule que tous les sous-ensembles des itemsets fréquents doivent aussi être fréquents, et *(2)* rendre le comptage des candidats plus efficace, notamment à travers des tables de hachage permettant d'accéder aux itemsets candidats présents pour un exemple de la base.

La propriété *d'antimonotonie* est souvent utilisée pour organiser la recherche de motifs fréquents de manière efficace. Puisqu'un motif ne peut être fréquent que si tous ses sous-motifs sont fréquents, on peut réduire l'espace de recherche et en n'examinant pas les sur-motifs d'un motif non fréquent, *i.e.* ne considérer que les sur-motifs dont tous les sous-motifs sont fréquents. Cette propriété est la base de l'algorithme *APRIORI* [60] qui procède par une génération ascendante *(bottom-up)*, en identifiant en premier lieu les motifs fréquents de taille 1, puis, à partir de ceux-ci, les motifs fréquents de taille 2, et ainsi de suite.

2.4.2.2 Les règles d'association

Une règle d'association est une représentation populaire de corrélations locales en fouille de données. Ce type de méthodes génère un très grand nombre de règles, avec des degrés d'intérêt très variant, qu'il est impossible de tous valider par un expert. Ainsi, afin de sélectionner et d'ordonner les règles générées, des critères d'évaluation sont disponibles. Une synthèse sur les principaux critères d'évaluation des mesures de qualité des règles d'association est disponible dans [76]. Parmi ces critères, nous distinguons le *support* et la *confiance*. L'un des gros avantages de ces deux critères est leur clarté et facilité d'interprétation pour l'utilisateur non expert.

Si on revient à l'exemple de la base de données stockant les achats des clients, supposons que l'analyse correspondante a permis l'identification de la règle suivante :

ordinateur → antivirus [*support* = 3%, *confiance* = 60%]

Pour cette règle, le support de 3% signifie que dans 3% des transactions analysées, les clients ont acheté simultanément un ordinateur ainsi qu'un antivirus. Par ailleurs, une confiance à 60% signifie que 60% des clients qui ont achetés un ordinateur ont aussi acheté un anti-virus.

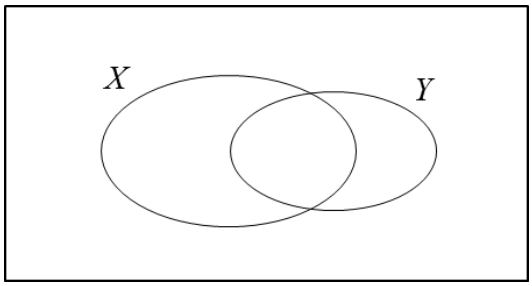


Figure 2.11 Support et confiance d'une règle X→ Y. Le support correspond à la proportion d'exemples contenants à la fois les items de X et ceux de Y dans l'ensemble de tous les exemples. La confiance correspond à la proportion des items de X qui contiennent aussi les items de Y. [42]

Le support d'une règle $X \rightarrow Y$ est la probabilité P(X, Y) que a et b soient VRAI en même temps. Notons que le support de la règle $X \rightarrow Y$ est le même que celui de la règle $Y \rightarrow X$.

La confiance d'une règle $X \rightarrow Y$ est la probabilité P(X|Y) que Y soit vérifiée quand X l'est.

$$\text{confiance } (X \rightarrow Y\ ) \ = \ \mathrm{P}(X|Y) \ = \frac{support(X \cup Y)}{support(X)}$$

À partir de l'ensemble des *itemsets* ou *motifs fréquents*, *i.e.* dépassant le seuil minimal de *support*, l'objectif est d'identifier des règles d'association dites fortes, *i.e.* qui vérifient un support et une confiance qui dépassent les seuils minimaux fixés *a priori*. Ainsi, pour chaque *itemset fréquent*, noté l, nous commençons par identifier les sous-ensembles non vide de l. Ensuite, pour un sous-ensemble noté s de l, nous produisons en sortie une règle de la forme *« $s \rightarrow (l \setminus s)$ »* si la confiance de cette règle générée dépasse le seuil minimal, *i.e. :*

$$\frac{support\ (l)}{support(s)} \geq \min\_confiance$$

Notons que toutes les règles générées satisfont automatiquement le seuil de support, puisque les motifs qui les définissent sont fréquents.

2.4.3 L'induction d'arbres de décision

L'induction d'arbre de décision est une famille de méthodes d'apprentissage supervisé. Rappelons qu'en apprentissage supervisé, nous cherchons à apprendre un modèle qui permet de prédire la classe d'un nouvel individu à partir d'un ensemble d'exemples représentés d'un côté par des variables descriptives dans l'espace de représentation, et d'un autre côté par l'étiquette de classe, à savoir le résultat donné par un expert. Par exemple pour la base d'apprentissage *IRIS* [77], les observations représentent des instances de fleurs et les quatre variables descriptives sont la largeur et la longueur des pétales et des sépales, et le label est le type de fleur auquel l'observation appartient. On cherche pour ce cas à définir un modèle qui permet de décrire les différentes classes de fleurs afin de prédire la classe correspondante à une nouvelle instance de fleur dont on ne connait pas la classe.

L'induction d'arbre de décision se base sur la technique de « *diviser pour régner* » (*divide and conquer*). Elle se résume à identifier des sous-problèmes, à leur trouver une solution, puis à combiner ces solutions pour résoudre le problème général. Suivant ce principe, les algorithmes d'apprentissage par arbres de décision apprennent à identifier les sous-espaces de l'espace d'entrée pour lesquels la solution (la classe) est identique, et lorsqu'un nouveau cas est soumis au système, celui-ci identifie le sous-espace correspondant et retourne la réponse (la classe) associée [42].

La classification d'un attribut se fait par une suite de tests sur les attributs qui le décrivent. Ces tests sont organisés de façon à ce que la réponse à un test indique le prochain test à vérifier. Ces tests sont ainsi organisés en un arbre de décision, où une feuille indique

une des classes, et un nœud intermédiaire correspondant à un test portant sur un attribut de l'espace de représentation, aussi appelé un *sélecteur*.

Un algorithme d'induction d'arbre de décision passe généralement par trois étapes décrites dans les sous-sections suivantes à savoir : *(1)* l'apprentissage d'un arbre de décision, *(2)* l'élagage de l'arbre obtenu et *(3)* l'extraction de règles à partir de l'arbre élagué.

2.4.3.1 L'apprentissage d'un arbre de décision

En partant de la racine de l'arbre qui regroupe toutes les observations, l'objectif est d'identifier le paramètre qui permet de créer une division plus pure que celle du nœud parent. On cherche, ainsi, à construire de façon récursive un arbre de décision dont l'erreur apparente est nulle.

Il est important de noter qu'il est impossible d'explorer de manière exhaustive l'ensemble des arbres de décision possibles afin de sélectionner le plus performant pour la problématique de classification, puisque le nombre d'arbres possibles croit exponentiellement avec le nombre d'attributs et le nombre moyen de valeurs possibles par attribut. Ainsi, une sélection intelligente est nécessaire pour identifier le premier *sélecteur*, le deuxième … définissant l'arbre de décision. Pour cela, nous nous basons sur une mesure, telle que l'*entropie*, l'*indice de GINI* ou encore le test de χ^2.

Ainsi, quand les données d'apprentissage entrent dans le nœud racine d'un arbre de décision, un test est appliqué afin d'explorer toutes les divisions possibles selon les variables explicatives, et choisir celle qui est la plus discriminante selon la variable cible, i.e. le label. Ceci est répété récursivement pour diviser les nœuds fils, jusqu'à l'obtention d'un nœud aussi pur que possible, i.e. correspondant à des exemples idéalement appartenant à une même classe.

En suivant une séquence de nœuds et de branches depuis la racine de l'arbre jusqu'à une feuille, on raffine progressivement la description des exemples concernés jusqu'à obtenir une description correspondant, si tout va bien aux objets d'une classe. Chaque chemin correspond à une conjonction de conditions sur des attributs décrivant les exemples.

2.4.3.2 L'élagage de l'arbre

Même si ce n'est pas toujours possible, les algorithmes d'induction d'arbre de décision cherchent à ramifier l'arbre autant de fois qu'il le faut pour obtenir des nœuds feuilles purs. Ceci engendre une problématique de sur-apprentissage pouvant refléter des erreurs dans l'échantillon analysé. En effet, comme le montre la Figure 2.12, le risque empirique continue de diminuer avec la prise en compte d'informations et ce grâce à l'accroissement des données d'apprentissage, ou encore par la répétition des exemples d'apprentissage. Par contre, le risque réel, d'abord décroissant, commence à augmenter après un certain stade. À partir de ce moment, on observe le phénomène de *sur-apprentissage* aussi appelé *sur-ajustement*, « *over-*

fitting » en anglais. Pour contourner ce problème et pour avoir une meilleure capacité de généralisation, on cherchera à élaguer l'arbre appris afin de réduire sa complexité. Ceci est l'objectif de cette étape.

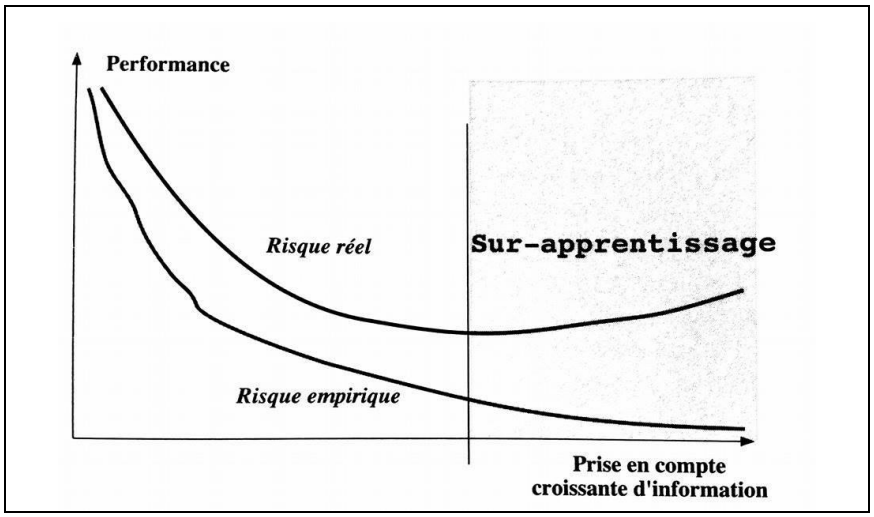


Figure 2.12 : Représentation du phénomène de sur-apprentissage [42]

Une fois un arbre de décision maximal appris à partir des données, deux façons pour l'élaguer sont possibles en fonction de la taille des données disponibles. La première, si le nombre d'exemples le permet, consiste à diviser les exemples disponibles en trois parties, *(1)* un ensemble d'apprentissage noté *A* qui sera utilisé lors de la première étape pour la construction d'un arbre maximal, noté T_{max}, dont toutes les feuilles sont aussi pures que possibles, *(2)* un ensemble de validation noté *V*, pour l'élagage de T_{max} et *(3)* un ensemble de test *T* qui sert à évaluer le risque réel de l'arbre construit.

Par ailleurs, si les données disponibles sont peu nombreuses, on se base sur une technique de validation croisée pour élaguer l'arbre maximal T_{max}. Ces types d'élagage sont dits des post-élagages puisqu'ils interviennent après la génération d'un arbre maximal. Un autre type d'élagage peut être fait en même temps que la génération de l'arbre et qui consiste à arrêter la division des nœuds dès que la pureté des exemples correspondants est suffisante, sans être forcément maximale comme c'est le cas pour la construction de T_{max}. Ce deuxième type d'élagage est moins utilisé car il peut empêcher le développement d'un arbre qui serait excellent. Plus de détails sur l'opération d'élagage sont disponible dans [42].

2.4.3.3 **L'extraction des règles :**

L'arbre de décision induit peut être transformé en un ensemble de règles de généralisation, tel que chaque chemin est transformé en une conjonction de tests associés à une classe. Ainsi, l'ensemble des chemins peut donc être reformulé comme une disjonction de conjonctions. [78] Une règle est créée pour chaque chemin de l'arbre qui part de la racine et s'arrête à un nœud feuille, de la forme « Si… Alors … ». Chaque condition, représentée par une variable et sa valeur, composant un arc du chemin sélectionné représente une conjonction

dans l'antécédent de la règle, i.e. la partie « Si ». En outre, le nœud feuille représente la classe et constitue la conclusion de la règle, *i.e.* la partie « Alors ». De telles règles sont assez faciles à comprendre par les utilisateurs.

Plusieurs algorithmes ont été développés pour l'induction d'arbres de décision, qui diffèrent par l'implémentation de la première étape et du type des données traitées. Un des premiers algorithmes est *CHAID*. Il permet l'induction d'arbre de décision non-binaire (un nœud peut avoir plus que deux nœuds fils), et qui détermine la meilleure division du nœud courant sur la base de tests de signification (Kass, 1980). *CHAID* ne peut être appliqué que sur des variables qualitatives. Un autre algorithme très souvent utilisé est *CART* (i.e., Classification And Regression Tree). Cet algorithme permet la construction d'arbres de décision binaires avec l'Indice de Gini comme critère de fractionnement [79]. *CART* peut traiter les variables qualitatives et continues. Finalement, C4.5 qui est une variante de ID3 [52] se base sur une mesure d'entropie comme critère de découpage. Il permet d'analyser des variables qualitatives et continues.

2.5 CONCLUSION

Dans ce chapitre, on a proposé une revue de littérature des travaux sur la problématique d'identification des causes d'un défaut en industrie et plus précisément en industrie du semi-conducteur, suivie par une analyse critique montrant les points qu'on cherche à prendre en compte dans nos travaux de thèse, notamment une analyse qui descend au niveau des paramètres équipements pour identifier une cause explicative d'un défaut, ainsi que l'analyse spatiale des données de contrôle.

Dans les deux prochains chapitres, on donne une description de la méthode d'analyse hybride, nommée *CLARIF*, que nous proposons pour expliquer un défaut à travers l'identification d'un ou plusieurs modes de production sur un ou plusieurs équipements à une ou plusieurs étapes du processus global de fabrication. *CLARIF* combine des techniques de *clustering*, de *recherche de règles d'association,* ainsi que *d'induction d'arbre de décision*. Dans les chapitres suivants, nous donnerons plus de détails sur cette combinaison de méthodes.

Chapitre 3 : Vers une analyse par étape et par plaque

Dans ce chapitre, on propose une solution pour traiter les problématiques décrites dans les précédents chapitres. Premièrement, on commence par introduire les notions clés de l'approche que nous proposons. Deuxièmement, la problématique d'explication de perte de qualité est généralisée en une problématique d'explication d'un phénomène *Y*. Troisièmement, on donne une description détaillée des quatre problématiques de recherche qui devront être traitées pour construire la méthode de fouille de données attendue. La quatrième section est consacrée à la description générale des étapes composant cette méthode d'analyse pour expliquer un phénomène *Y*. Plus de détails sur cette méthode feront l'objet du chapitre suivant. Finalement, on propose l'intégration de cette méthode d'analyse dans un processus d'Extraction des Connaissances à partir des Données, *ECD*, pour expliquer des cas de perte de qualité locale.

Table des matières

3.1 NOTIONS DE BASE DE L'APPROCHE PROPOSEE

Dans le cadre de cette thèse, on s'intéresse à la problématique d'explication de cas de perte de qualité. Une perte de qualité est représentée par un ensemble de plaques ne réussissant pas une ou plusieurs étapes de contrôle qualité, dans le contexte de l'industrie du semi-conducteur. Comme décrit dans le précédent chapitre, on s'intéresse à expliquer un cas de perte de qualité en identifiant les conditions particulières de fonctionnement d'un ou plusieurs équipements.

Pour adresser cette problématique et comme discuté dans les précédents chapitres, on propose une méthode d'analyse *''par étape''* et ''*par plaque''*. Cette approche dite *''par étape''* est, ainsi, basée sur une notion fondamentale de décomposition du processus de fabrication en étapes nous permettant de remonter à des causes relatives, d'un côté, à une seule étape, et d'un autre côté, à plusieurs étapes. Par ailleurs, cette approche, dite *''par plaque'',* est basée sur une analyse de l'historique informationnel d'un ensemble de plaques étudiées. Ces deux notions feront l'objet des deux sous-sections suivantes.

3.1.1 La décomposition du processus de fabrication

Le processus de fabrication en semi-conducteur est composé d'étapes de production, durant lesquelles les plaques sont traitées, et d'étapes de contrôle qualité permettant le contrôle des plaques en traitement. Comme précédemment énoncé, on propose une analyse par étape. Ainsi, la méthode proposée concernera un segment du processus, noté *s*, qui est un ensemble d'étapes de production successives, noté *P*, suivis par une ou plusieurs étapes de contrôle qualité qui leur sont associées *Q*.

$$s = \{P, Q\}$$

Une perte de qualité est détectée à travers une ou plusieurs étapes de *Q*. Par exemple, après des étapes de gravure, un ensemble d'étapes de contrôle qualité viennent vérifier le bon déroulement de ces étapes de gravure, en mesurant par exemple la hauteur restante d'une couche récemment gravée.

La *Figure 3.1* donne une schématisation d'un segment du processus de production. Pour un intervalle de temps donné, un ensemble de plaques, noté *W*, sont traitées durant un segment *s*. Toutes ces plaques sont traitées pour chaque étape de production $p \in P$. Notons que différents équipements de production peuvent intervenir pour traiter les plaques *W*, à une même étape *p*. Par exemple, pour la première étape *p'*, les équipements qui ont traité les plaques *W* sont les équipements *t'1*, *t'2* ... Notons qu'un même équipement peut traiter des plaques pour différentes étapes de production *p* de *P*. Par ailleurs l'ensemble des plaques traitées par un équipement *t* lors d'une étape de production *p* est noté W^{pt}. Ainsi, $W^{p't'1}$ représente les plaques traitées par la machine *t'1* lors de la première étape *p'*. On notera *Tools* l'ensemble des machines intervenues dans le traitement des plaques *W* durant les étapes de

production P du segment d'analyse s. Par ailleurs, on notera $Tools^p$ l'ensemble des machines intervenant dans le traitement des produits durant l'étape p.

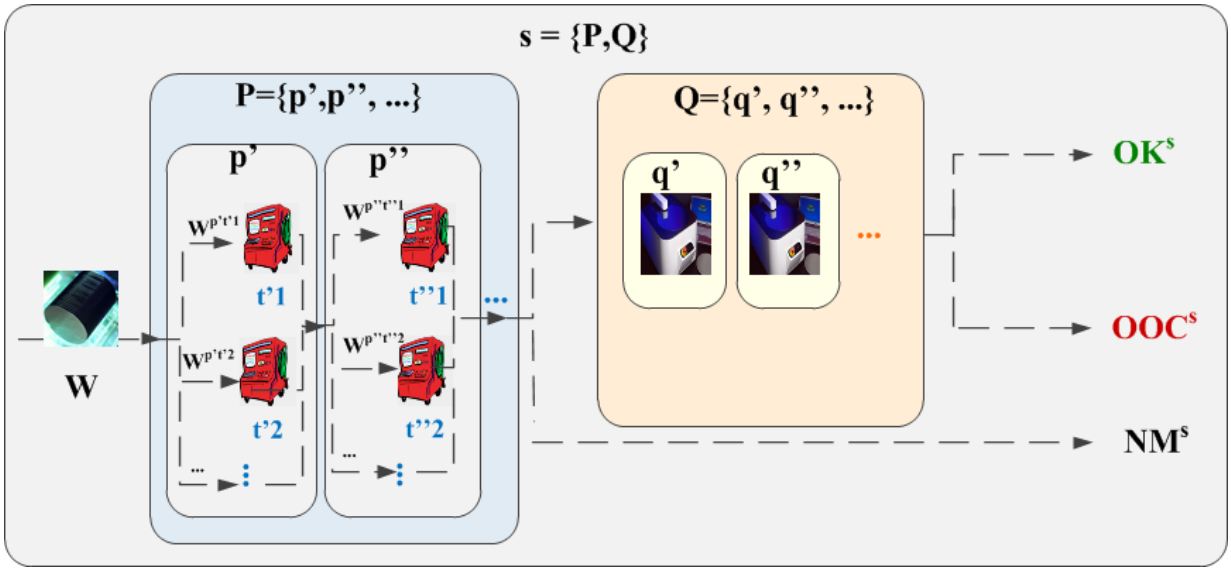


Figure 3.1: Schématisation d'un segment du processus de fabrication.

À la suite de ces étapes de production P, un ensemble d'étapes de contrôle qualité viennent vérifier le bon déroulement de celles-ci. Comme introduit dans le premier chapitre de ce manuscrit, souvent pour des mesures d'optimisation de coût et de temps de production, seul un échantillon des plaques W est mesuré, qu'on note M. Ainsi,

$$M \subseteq W.$$

Pour chaque étape $q \in Q$, des mesures physiques et/ou électriques sont collectées sur les plaques sélectionnées. Les plaques avec des mesures acceptables, *i.e.* dans les limite se contrôles définies par l'expert, sont considérées comme bonnes, notées OK^q, alors que celles qui ne le sont pas sont considérées comme problématiques, notées, OOC^q.

Ainsi, une perte de qualité est définie par l'ensemble des plaques détectées problématiques par au moins une étape de contrôle qualité $q \in Q$ d'un segment s, *i.e.* cet ensemble représente l'union des différents OOC^q

$$OOC^s = \bigcup_{\forall q \in Q} OOC^q$$

Les plaques bonnes, quant à elles, notées OK, représente l'ensemble des plaques réussissant toutes les étapes de contrôle $q \in Q$, *i.e.* l'intersection des différents OK^q.

$$OK^s = \bigcap_{\forall q \in Q} OK^q$$

Par ailleurs, l'ensemble des plaques non sélectionnées pour être mesurées aux étapes Q, noté NM^s, continuent normalement leur processus de fabrication. Pour simplifier les

notations, nous noterons *NM*, *OK*, et *OOC* pour faire référence respectivement aux produits non contrôlés, contrôlés et bons et finalement les produits contrôlés et problématiques.

3.1.2 Analyse de l'historique d'une plaque

Pour expliquer une perte de qualité, on propose une analyse de l'historique des *W* plaques. L'historique d'une plaque, illustré dans *Figure 3.2*, est représenté par l'ensemble des données collectées durant son traitement au cours des étapes de production et/ou de contrôle qualité sélectionnées.

D'un côté, l'historique d'une plaque $w \in W$, est constitué, dans le système d'information, par l'ensemble des données décrivant les mesures collectées sur un équipement de production $t \in Tools$ lors de son traitement à une étape de production $p \in P$. On note $p^t(w)$, le vecteur de données décrivant ces données pour une plaque *w*. Les données de production sont collectées grâce aux outils de contrôle équipements le système FDC (Fault Detection and Classification).

D'un autre côté, l'historique d'une plaque *w* est représenté par ses mesures physiques ou électriques collectées à une étape de contrôle qualité $q \in Q$. Ce vecteur de données est noté *q(w)*. Ces données sont collectées à travers les systèmes de contrôles qualité, relatifs notamment aux étapes de métrologie, de défectivité, des tests paramétriques *PT* ou encore du test final de tri électrique des plaques EWS.

Ces vecteurs $p^t(w)$ et *q(w)* appartiennent, respectivement, à S^{pt} et S^q, les espaces des mesures de production et de qualité considérées. Par exemple, S^{p1t1} représente l'espace des mesures collectées sur l'équipement *t1* lors de l'étape de production *p1*. Le vecteur $p1^{t1}(w)$ représenté, dans la *Figure 3.2*, par un point sur ce plan à deux dimensions de l'espace S^{p1t1}, décrit les données de production de la plaque *w* à l'étape *p1* sur l'équipement *t1*.

$$historique\_informationnel(w) = \{p1^{t1}(w), p2^{t4}(w), q1(w), p3^{t6}(w), q2(w), \dots\}$$

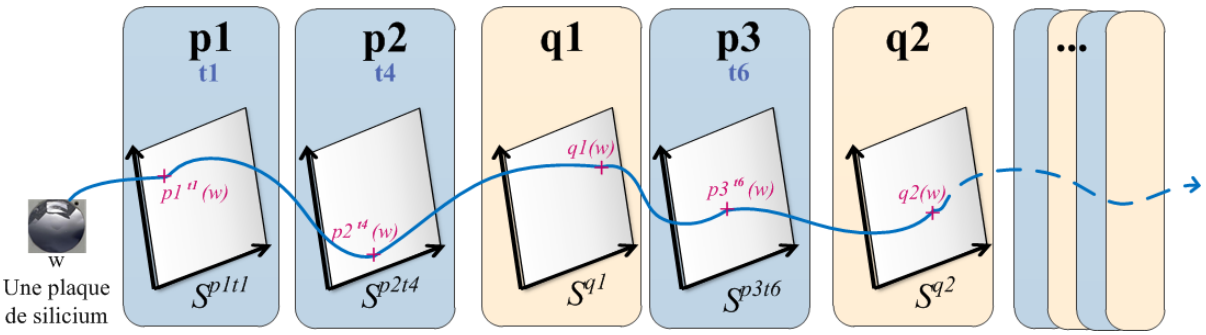


Figure 3.2 : Représentation de l'historique informationnel d'une plaque w.

Au niveau d'un segment, l'historique d'une plaque représente l'ensemble des données collectées aux étapes de production *P*, et éventuellement aux étapes de contrôle qualité *Q*. À partir de l'ensemble des historiques informationnels, représentant notre support pour la recherche de causes de perte de qualité, deux types de données sont disponibles *Figure 3.3*:

- D'un côté, pour chaque étape de production $p \in P$, et pour chaque équipement $t \in Tools^p$, des données décrivant l'état d'un équipement *t* lors du traitement des plaques sont collectées dans différents fichiers. Par exemple, nous notons $D^{p't'1}$, $D^{p't'2}$, … les données collectées lors de la première étape *p'*, respectivement à partir de l'équipement *t'1*, *t'2*, … lors du traitement des plaques respectives $W^{p't'1}$, $W^{p't'2}$, …
- D'un autre côté, les données collectées à partir des étapes de contrôle qualité *Q* sont stockées, elles aussi, dans d'autres fichiers. Nous notons $D^{q'}$, $D^{q''}$, … les données de contrôle qualité collectées, respectivement, aux étapes *q*', *q*''… $D^{q'}$, $D^{q''}$ …

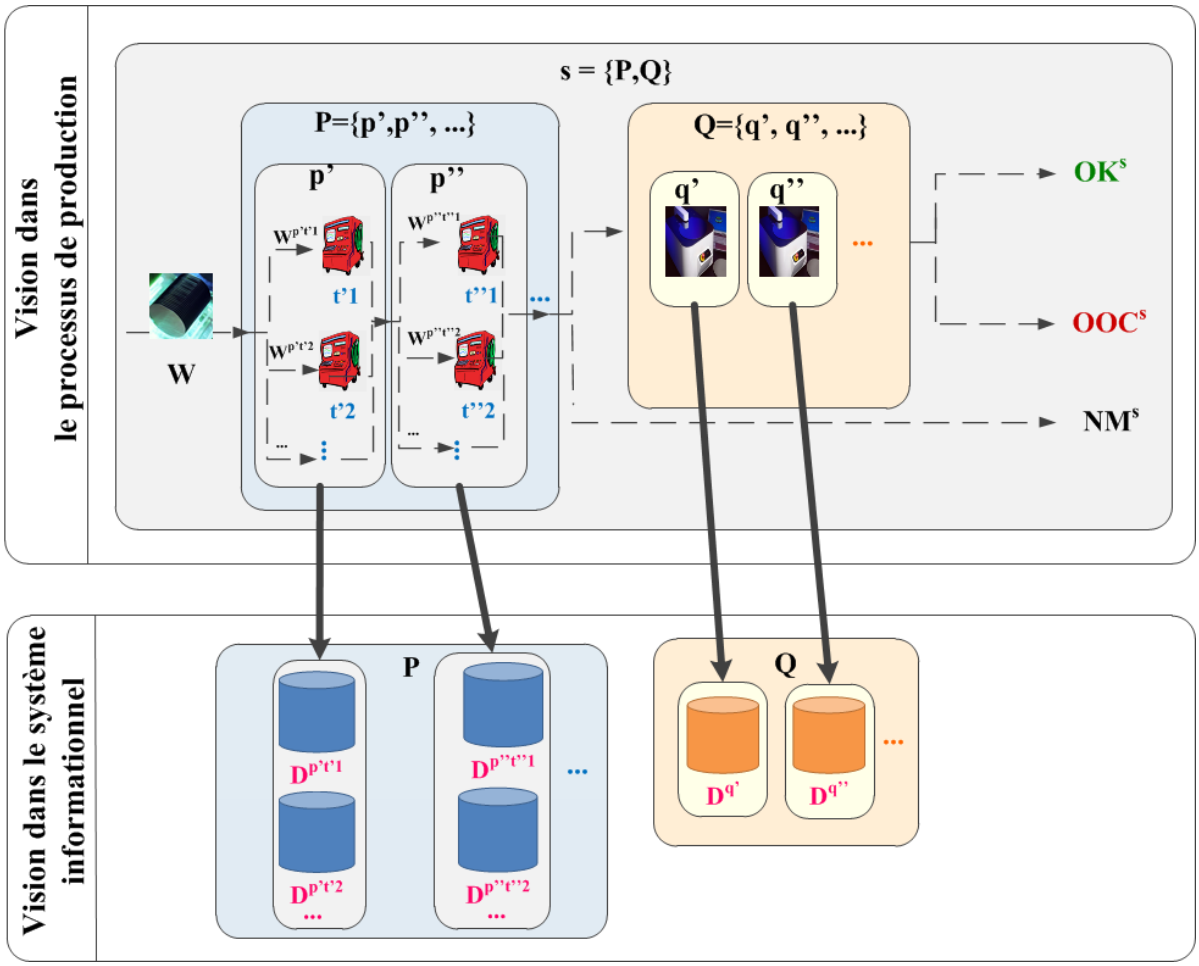


Figure 3.3: Représentation des données collectées au niveau d'un segment.

Expliquer une perte de qualité, en se basant sur ces données disponibles au niveau d'un segment, revient à corréler les données décrivant les différentes étapes de production *P*, aux données décrivant les étapes de contrôle qualité *Q*. Cette analyse permettra d'identifier *les équipements de production potentiellement responsables* de ce défaut, ainsi que *leurs conditions de production particulières problématiques*.

3.1.3 Généralisation en un problème d'explication d'un phénomène *Y*

Le problème d'explication de cas de perte de qualité, où on cherche à *corréler un phénomène* de perte de qualité *détecté à une ou plusieurs étapes de contrôle qualité $q \in Q$, à des conditions de production sur une ou plusieurs étapes de production $p \in P$*, peut être généralisé en une problématique d'explication d'un phénomène observé à une ou plusieurs étapes par des conditions particulières sur des étapes précédentes.

Cette généralisation permet de poser la problématique générale d'explication de phénomène, et de proposer ainsi une méthode d'analyse générique pour la résoudre. Celle-ci pourra être utilisée pour expliquer une perte de qualité, notée OOC^s, mais aussi tout autre phénomène, noté *Y*, détecté par un ensemble des étapes, que nous notons *EE* pour étapes effets, et qu'on cherche à comprendre à travers des conditions particulières sur des étapes, que nous notons *EC*, pour étapes causes. Pour une mesure de simplicité, on utilise la même notation *Y*, pour faire référence au phénomène à expliquer, ainsi qu'aux plaques le composant.

Durant les étapes *EE*, d'autres plaques sont contrôlées et le phénomène *Y* ne leur a pas été associé. Ces plaques sont notées $\overline{Y}$. L'union des deux ensembles *Y* et $\bar{Y}$ définit l'ensemble des plaques mesurées *M*. Par ailleurs, durant les étapes *EC*, les plaques *M* ont été traitées avec d'autres plaques qui n'ont pas été contrôlées aux étapes *EE*. Ces plaques constituent l'ensemble *NM*. Ainsi, l'ensemble de plaques à étudier est noté *W*.

$$W = \{M, NM\} = \{Y, \bar{Y}, NM\}$$

Par analogie, nous notons D^{ec1}, D^{ec2} … les données collectées, respectivement à partir des étapes causes *ec1*, *ec2*, … lors du traitement des plaques respectives W^{ec1}, W^{ec2}… , et D^{ee1}, D^{ee2} … les données des étapes effets *ee*1, *ee2*…(*Figure 3.4*).

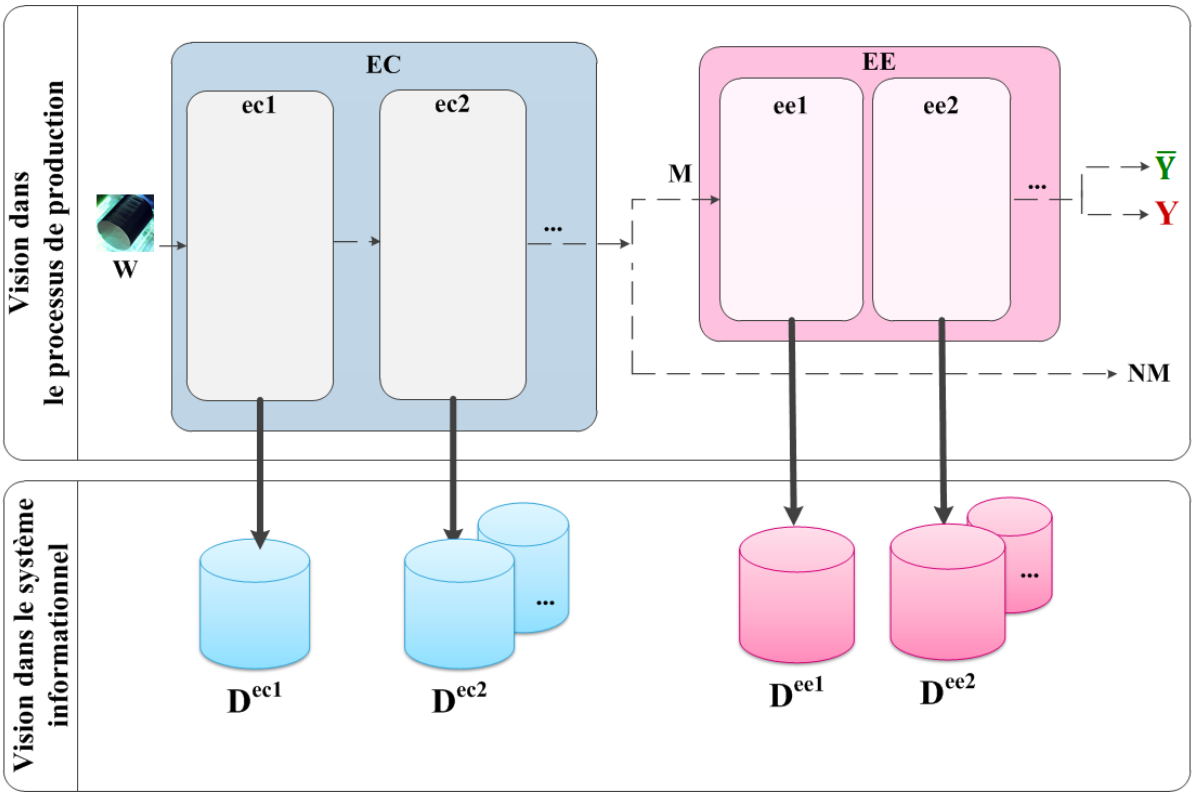


Figure 3.4: Description d'un cas d'explication d'un phénomène Y

À partir de cette formulation générale du problème, on cherchera à *expliquer un phénomène détecté sur une ou plusieurs étapes EE,* à travers l'identification *de conditions particulières sur une ou plusieurs étapes EC* et ce en analysant les fichiers de données disponibles, d'un côté pour les étapes causes *EC*, et d'un autre côté pour les étapes effets *EE*. Dans la section suivante, on propose une formalisation des différentes problématiques de recherche à traiter pour proposer une méthode d'explication d'un phénomène *Y*.

3.2 FORMALISATION DES PROBLEMATIQUES DE RECHERCHE

Pour proposer une méthode d'analyse répondant à la problématique d'explication d'un phénomène *Y*, on définit, tout d'abord, quatre grandes problématiques de recherche à traiter par cette méthode d'analyse. Celles-ci sont présentées dans les sous-sections suivantes.

3.2.1 PR1 : La distinction de différents sous-phénomènes composant *Y*

Les instances constituant un phénomène *Y* à expliquer ne sont pas nécessairement toutes du même type. Elles peuvent, donc, être dues à des causes différentes. Si on prend l'exemple d'explication d'un phénomène *Y*, relatif à des plaques détectées comme problématiques, *OOC*, toutes les plaques détectées à un moment donné comme problématiques ne sont pas forcément toutes dues à la même cause. Il est, ainsi, important de distinguer les différents sous-phénomènes qui composent le problème *Y*, avant de chercher leurs causes explicatives. Cela permettrait de se focaliser sur chaque sous phénomène, individuellement, afin de trouver la cause qui l'explique au mieux, sans être biaisé par les autres phénomènes, qui peuvent posséder d'autres causes.

→ Ainsi, la méthode d'analyse devrait, lors d'une étape de prétraitement, distinguer les différents sous-phénomènes composant *Y*, s'ils existent, en analysant les données décrivant les étapes *EE* où *Y* est détecté.

3.2.2 PR2 : La gestion de la forte volumétrie des données

Afin d'expliquer un phénomène *Y*, on propose d'analyser un ensemble de fichiers de données qui décrivent l'historique des produits à analyser, à différentes étapes, d'un côté des étapes où *Y* a été détecté, *i.e.* les étapes *EE*, et d'un autre côté, les données relatives aux étapes qui contiennent les potentielles causes explicatives, *i.e.* les étapes *EC*.

Pour chaque étape *e* de *EE* et de *EC*, d'énormes quantités de données peuvent être collectées. Les étapes de production sont typiquement des étapes où on collecte le plus de données. Comme décrit dans les chapitres précédents, les données de production sont collectées grâce au système de contrôle des équipements, nommé *FDC*. Un produit *w* traité par un équipement *t*, à une étape de production *p*, peut être décrit par une centaine voire plusieurs centaines de paramètres résumés, et plusieurs milliers de paramètre non résumés.

Face à cette grande volumétrie des données, il est important de proposer une méthode qui permet de les gérer, permettant ainsi, le passage à l'échelle, notamment quand on s'intéresse à l'explication de phénomènes relatifs à plusieurs étapes *EE* par plusieurs étapes *EC*.

On s'intéresse ainsi à identifier une méthode, qui par sa définition, permet de réduire la forte dimensionnalité des données afin de permettre l'exploitation des richesses du système d'information microélectronique.

→ Ainsi, la deuxième problématique de recherche à traiter concerne la gestion des données de *forte dimensionnalité*.

3.2.3 PR3 : La gestion de la qualité des causes explicatives identifiées

En plus des deux problématiques précédemment citées, et pour répondre aux défis industriels et scientifiques présentés dans le chapitre 1, on s'intéresse à identifier des causes qui soient *nécessairement compréhensibles, valides* et *potentiellement utiles*. Ainsi, la méthode à proposer doit gérer ces trois aspects.

Pour assurer l'identification de causes compréhensibles, deux points sont à valider. D'un côté, le choix de la formulation des causes explicatives est important puisque une cause formulée d'une façon complexe ne sera pas utilisée même si elle représente une connaissance valide et utile pour expliquer le phénomène *Y*. D'un autre côté, la méthode proposée doit, elle aussi, gérer la compréhensibilité des causes identifiées. Par exemple, si pour gérer la volumétrie des données, des données ont été transformées, il faut revenir à l'espace de départ pour présenter une cause directement compréhensible et utilisable par les ingénieurs métiers.

Par ailleurs, pour assurer les aspects de validité et d'utilité des causes identifiées, la méthode d'analyse doit proposer des indicateurs pour mesurer ces aspects de qualité des causes potentielles. À titre d'exemple, la validité d'une cause pourrait être mesurée par une notion telle que le degré de confiance, avec lequel on peut dire que la cause identifiée explique réellement le phénomène *Y*. En utilisant ce type d'indicateurs, et selon le risque que décrit *Y*, différents seuils de mesures pourraient être définis par les ingénieurs pour estimer si une cause est valide ou pas.

Finalement, l'utilité d'une cause peut être mesurée par des notions telles que le degré de généralisation de celle-ci. Ceci est un problème classique en analyse et apprentissage des données. En effet, une connaissance qui décrit trop parfaitement les données d'analyse n'est pas forcément utile car ce qu'elle décrit peut ne pas être généralisable, et par conséquence, peut ne pas être considéré comme une cause explicative du phénomène *Y*.

→ Ainsi, la troisième problématique de recherche que la méthode à proposer devrait traiter concerne la gestion de la qualité des causes explicatives, à travers les trois aspects précédemment cités, *la compréhensibilité, la validité* et *l'utilité* de celles-ci pour expliquer le phénomène *Y*.

3.2.4 PR4 : L'identification de causes explicatives de différents types

Finalement, afin de tirer profit de la décomposition du processus de fabrication, on propose d'expliquer un phénomène *Y*, par des explications qui peuvent être relatives à une ou plusieurs étapes causes $e \in EC$. Les analyses locales au niveau de chaque étape élémentaire devront être ainsi reliées afin d'identifier d'autres causes explicatives de *Y*. Ainsi, les causes

recherchées pourront se révéler de différents types : des causes simples, où la cause sera relative à une seule étape, et des causes composées, où la cause sera relative à la combinaison de conditions particulières sur différentes étapes élémentaires.

→ Ainsi, on définit la quatrième problématique de recherche comme l'identification de relations de dépendance entre des conditions particulières sur différentes étapes élémentaires *EC*, qui pourront être interprétées en termes de causalités plus globales expliquant le phénomène à analyser.

3.3 DESCRIPTION GENERALE DE LA METHODE D'ANALYSE PROPOSEE

Dans cette section, on donne une description générale de la méthode d'analyse que nous proposons pour l'explication d'un phénomène *Y*. Rappelons que la méthode qu'on propose permet de fournir un outil d'aide à l'interprétation et à la capitalisation de l'historique d'un ensemble de produits *W*. Cette méthode d'analyse, que nous nommons *CLARIF*, permet d'expliquer un phénomène *Y* détecté par des étapes *EE*, à travers des conditions particulières sur une ou plusieurs étapes de *EC*.

Comme décrit dans *Figure 3.5*, *CLARIF* a besoin d'un ensemble d'informations en entrée :

- Premièrement, donner le contexte d'analyse à travers la sélection des produits dont l'historique est à analyser, *W*. Trois types de produits sont à spécifier : les produits à expliquer, *Y*, les produits qui ont été contrôlés et sur lesquels le phénomène *Y* n'a pas été observé, notés, $\bar{Y}$. Et finalement, les produits non contrôlés, *NM*.
- Deuxièmement, donner les étapes où le phénomène *Y* a été identifié, *i.e. EE*. Il faut, aussi, définir précisément à quel étape *e* de *EE* les sous-ensembles de produits *Y* ont été détecté. Il faut donc préciser pour chaque étape, les ensembles Y^e, et $\bar{Y}^e$. Notons que Y^e est le sous-ensemble de Y, *i.e.* l'ensemble des produits détectés comme décrivant le phénomène *Y* lors de l'étape *e*. L'ensemble $\bar{Y}^e$ représentent les autres produits mesurés en *e* et ne décrivant pas le phénomène *Y*.
- Troisièmement, spécifier les étapes qui peuvent contenir la cause de ce phénomène, *i.e. EC*.
- Finalement, pour chaque étape *e* des étapes *EE* et *EC*, extraire les données décrivant l'historique des produits à analyser. D^e est le fichier de données décrivant les données représentant l'historique des produits W^e lors de l'étape *e*.

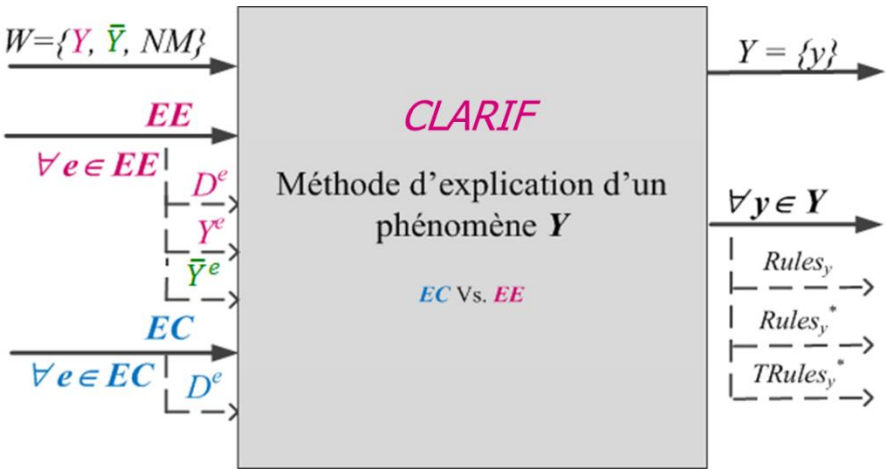


Figure 3.5 : Schématisation des entrées et sorties de CLARIF, la méthode proposée pour l'explication d'un phénomène Y.

Pour analyser ces données en entrée, et fournir les différents résultats ($Y = \{y\}$; $Rules_y$; $Rules_y^*$; $TRules_y^*$), comme illustré dans la *Figure 3.5*, *CLARIF*, qui sera spécifiée puis validée dans les chapitres à venir, est composée de quatre sous étapes, décrites dans les sous-sections suivantes. L'ensemble de ces étapes est synthétisé dans *Figure 3.6*. Cette section 4 se limite à une explication générale de la structuration de la méthode : les choix des techniques de traitement des données nécessaires à la réalisation de chacune des 5 étapes seront exposés et justifiés dans le chapitre suivant consacré à la description détaillée de *CLARIF*.

3.3.1 Séparation des sous-phénomènes *y* composant *Y*

La première étape de *CLARIF* traite la première problématique de recherche, *PR1*. Dans cette étape, on s'intéresse à analyser les données des différentes étapes *EE*, afin d'identifier différents sous phénomènes composant le phénomène à expliquer *Y*.

On cherche alors à identifier des sous-groupes de *Y*, où chaque sous-groupe, noté *y*, est construit de manière à identifier des sous-groupes de produits en se basant sur un critère de similarité. On propose d'appeler ces sous-groupes des modes descriptifs du phénomène *Y*, ou aussi, des sous-phénomènes.

$$Y = \{y\}$$

L'objectif de cette première étape est d'enrichir la partition des produits contrôlés *M*, notée $partition_M$, initialement composée de deux groupes, *Y* et $\bar{Y}$, à travers l'analyse des données décrivant les étapes *EE*. Cette analyse permettrait d'identifier une nouvelle partition des produits *M*, notée $partition'_M$, composée, d'un côté, du groupe des produits qui n'appartiennent pas au phénomène *Y*, *i.e.* $\bar{Y}$, et d'un autre côté, un ou plusieurs groupes de produits qui appartiennent à *Y*.

$$partition_M = \{\ \bar{Y},\ \boldsymbol{Y}\ \} \quad \Rightarrow \quad partition'_M = \{\ \bar{Y},\ \{\boldsymbol{y}\}\ \}$$

Pour cela, nous proposons une analyse en deux sous-étapes :

- Premièrement, pour chaque étape du processus industriel $e \in EE$, on identifie à partir des données qui lui sont relatives, D^e, une partition qui sépare au mieux les produits Y^e des produits $\overline{Y}^e$. Cela permet ainsi, de synthétiser les données initialement en p colonnes de l'étape $e \in EE$ sous la forme d'un vecteur catégorique, affectant chaque produit $w \in W^e$, à un groupe indiquant si le produit w appartient ou pas au phénomène analysé Y^e, si oui à quel mode descriptif y^e en particulier de Y^e. Y^e peut donc être noté : $Y^e = \{ y^e \}$.
- Deuxièmement, à partir de ces partitions obtenues, une partition finale est construite pour définir les sous-phénomènes de Y. Ainsi, un mode descriptif y est défini par l'ensemble des produits qui sont similaires selon toutes les partitions individuelles des étapes $e \in EE$. Ainsi, deux produits sont groupés dans un même y si toutes les partitions élémentaires les ont identifiés dans un même groupe, sinon ces deux produits sont attribués à des groupes différents. On obtient, ainsi, la décomposition de Y qui peut ainsi être noté : $Y = \{y\}$.

Les différents sous-phénomènes, $y \in Y$, représentent le premier résultat de la méthode proposée. Cela permet d'indiquer que le phénomène à expliquer est potentiellement composé de différents sous-phénomènes, où chacun est potentiellement dû à des causes différentes. Cette première étape est synthétisée au sein de la *Figure 3.6*.

Par ailleurs, notons que cette étape permet de traiter partiellement la deuxième problématique *PR2*, en remplaçant les variables initiales des différentes étapes *EE* en une seule variable catégorique, $partition'_M$.

3.3.2 Génération non supervisée des modes descriptifs candidats pour chaque étape *EC*

Au niveau de chaque étape cause $e \in EC$, on propose d'analyser les données D^e pour distinguer des groupements des produits basés sur leurs similarités. Ainsi, nous réduisons la forte dimension des données numériques en entrée relatives D^e, en les remplaçant par une dimension catégorique, qui est la partition décrivant les groupements obtenus. Cette étape nous permet, ainsi, de traiter la problématique *PR2*. Les groupements obtenus, appelés, *modes descriptifs*, seront considérés dans l'étape suivante comme des causes potentielles des sous-phénomènes composant Y.

Par ailleurs, afin d'enrichir cet ensemble de causes potentielles, nous proposons de générer, non pas une seule partition à partir de D^e, mais plusieurs partitions distinguant des modes descriptifs avec différents niveaux de complexité (voir section suivante), candidats à expliquer les phénomènes $y \in Y$.

Ainsi, pour chaque étape $e \in EC$, on s'intéresse à générer des modes descriptifs candidats à expliquer chaque $y \in Y$. À partir de D^e, le fichier de données décrivant l'historique de l'ensemble des produits W^e traités par l'étape $e \in EC$, on propose de construire plusieurs partitions des produits W^e, composant le fichier résultant noté D'^e.

Par exemple, dans un contexte d'explication de phénomène de perte de qualité locale, un mode descriptif candidat représente un groupe de produits qui ont connus des conditions de production similaires, sur un même équipement de production, à une même étape de production $p \in P$. Ces modes de production sont utilisés lors de la prochaine étape pour identifier des explications potentielles de chacun des *modes descriptifs de perte de qualité y*.

3.3.3 Identification de règles explicatives

Après avoir identifié, d'un côté un ensemble de modes descriptifs du phénomène à analyser $Y=\{y\}$, et de l'autre côté, un ensemble de modes descriptifs décrivant des sources d'explications potentielles, on propose de rechercher des corrélations entre ces deux ensembles, constituant des explications possibles du phénomène *Y*. Un produit est, ainsi décrit d'un côté par l'ensemble des labels des *modes descriptifs candidats* pour chaque étape de *EC*, et d'un autre côté par le *label du mode descriptif y*, s'il fait partie du phénomène à expliquer, sinon par *le label de non appartenance* , $\bar{Y}$. À partir de cette représentation de l'historique des différents produits analysés, successivement, on cherche à expliquer un mode descriptif *y*.

Comme décrit dans la problématique *PB3*, le choix de la formulation des causes explicatives à identifier est important pour assurer l'identification d'une connaissance *nécessairement compréhensible*. Pour cela, on a choisi d'identifier ces corrélations sous la forme de règles notées $r(X \rightarrow y)$. La justification de l'utilisation de règles sera développée au prochain chapitre. Cette formulation représente une relation de causalité intuitive où *X* représente la cause d'un effet *y*. Cette connaissance permet d'identifier une cause potentielle de *y*, sous forme des prémices *X* de *r*, constitués d'un ou plusieurs *modes descriptifs candidats* générés à partir des données des étapes *EC*.

Par ailleurs, pour assurer l'identification de connaissances *valides* et *potentiellement utiles*, un ensemble d'indicateurs de qualité a été mis au point : *confiance*, *complexité* et *contribution*. Le choix et la justification de ces indicateurs sera également approfondi au prochain chapitre. L'indicateur de *confiance* mesure la validité avec laquelle la prémisse, notée *X*, d'une règle permet d'expliquer le phénomène *y*. L'indicateur de *complexité*, quant à lui, permet d'assurer l'identification de connaissances potentiellement utiles à l'ingénieur et ce en maîtrisant la complexité des règles identifiées. L'indicateur de *contribution* permet de mesurer le pourcentage d'instances décrites par la règle par rapport à l'ensemble définissant le phénomène *y*. Finalement, une règle trop complexe, même si elle est valide (bonne confiance), compréhensible et possède une bonne contribution, ne peut pas être utilisée puisqu'elle ne permet pas d'être généralisée. Cet aspect de généralisation peut être vu, ici, comme une mesure de robustesse par rapport à l'ensemble de produits analysés, *W*. Dans le

prochain chapitre, les formules de calcul proposées pour ces trois indicateurs de qualité seront présentées.

De plus, pour traiter la problématique de recherche *PB4*, c'est à dire fournir des causes explicatives de différents niveaux, *CLARIF* permet l'identification de deux types de règles pour expliquer un sous-phénomène *y* ; *des règles simples* et *des règles de trajectoires*.

- Les *règles simples* identifient un mode descriptif candidat identifié à une étape $e \in EC$ comme une cause explicative de *y*.

On propose d'enrichir ce premier ensemble de règles simples, à travers la combinaison de modes descriptifs élémentaires pour identifier de nouvelles règles, potentiellement, plus intéressantes, comme décrit par *PB4*.

- Ainsi, on obtient les *règles de trajectoires*, qui représentent des combinaisons de modes descriptifs candidats identifiés à différentes étapes $e \in EC$.

Ces deux types de règles représentent l'ensemble $Rules_y$, avec les différentes causes potentiellement explicatives du sous phénomène *y*.

3.3.4 Sélection des règles les plus pertinentes

La qualité des règles identifiées est très variable. Pour faciliter, l'appropriation de ces nouvelles connaissances, il est ainsi intéressant de réduire cet ensemble, en sélectionnant les plus intéressantes. L'enjeu est donc l'utilité des connaissances extraites par la méthode de fouille de données pour l'ingénieur métier qui l'applique. Cette analyse est faite pour les règles explicatives de chaque sous-phénomène *y* séparément.

Pour cela, on propose une méthode de sélection progressive. En premier lieu, on propose d'identifier le sous ensemble des meilleures règles expliquant *y* à travers la résolution d'un problème d'optimisation multi critères basé sur les trois indicateurs de qualité proposés. Selon ces indicateurs, une règle est d'autant plus intéressante qu'elle est plus confiante, contribue à décrire le plus de produits problématiques de *y* et qu'elle est le moins complexe possible. Ce qui revient à filtrer l'ensemble résultant de règles d'associations selon la surface de Pareto, i.e., en ne gardant que les règles dominantes selon les trois indicateurs proposés. En deuxième lieu, des seuils sur ces indicateurs peuvent être fixés par les ingénieurs pour filtrer d'avantage l'ensemble final de règles.

On obtient, ainsi, $Rules_y^*$ l'ensemble des règles sélectionnées pour expliquer *y*. Les différents ensembles identifiés représentent le deuxième résultat de la méthode proposée.

3.3.5 Transformation des règles pertinentes

À partir de l'ensemble des règles filtrées, on a réussi à identifier un ensemble de modes descriptifs potentiellement responsables de *Y*. Rappelons que les modes descriptifs identifiés comme causes explicatives représentent un groupement de produits similaires.

Dans un contexte d'explication de perte de qualité locale, soit une règle simple *r(x→y)*, expliquant un *mode de perte de qualité y*, par le *mode de production x*. Cette règle d'association discrète constitue une hypothèse que les produits appartenant à *y* ont aussi appartenus à *x*. Cette règle identifie le mode de production connu par les produits de *x* comme une cause potentielle de la perte de qualité *y*, avec un certain degré de *confiance*, de *contribution* et de *complexité.*

Cette règle, telle que définie à ce niveau de *CLARIF*, ne permet pas une interprétation aisée de la cause de la perte de qualité *y*. En effet, la cause exprimée par le mode de production *x*, est définie de manière exhaustive par l'énumération de l'ensemble discret des produits qui constitue *x*. Afin de garantir l'identification de connaissances *potentiellement utiles* aux ingénieurs, il est ainsi important de revenir à l'espace de départ correspondant aux paramètres physiques manipulés par les ingénieurs. Il s'agit d'identifier, pour chaque mode problématique, des limites sur les paramètres qui le caractérisent le plus dans cet espace. Nous appelons cette étape par « la transformation des règles discrètes en règles continues ».

Par exemple, dans un contexte d'explication de cas de perte de qualité locale, on s'intéresse à identifier les caractéristiques de production représentatives du groupement de produits ayant le label du mode de production discret *x*. On propose, ainsi, de le transformer en un nouveau mode de production continu *x'*. Ce dernier représente une sous-région de S^{pt} avec des limites associées aux paramètres de production. Par exemple, un mode de production problématique *x*, défini, initialement, par les identifiants des produits qui le compose, *x* = {*w22*, *w33*, *w37*}, peut être traduit par

$$x' = \{\text{température moyenne(w)} > \text{seuil minimal}\}.$$

Ainsi, pour toutes les règles identifiées, qu'elles soient simples ou de trajectoire, on propose de transformer chacun de leurs modes descriptifs de production discrets en un mode descriptif continu. On obtient, ainsi, les différents règles continues, $TRules_y^*$. Cela représente le troisième résultat de cette méthode.

Plus de détails sur les méthodes proposées, ainsi que sur la justification des choix d'algorithmes, pour l'identification, la combinaison, la sélection et la transformation des règles sont proposés dans le chapitre 4. Pour clôturer ce chapitre 3, nous proposons, au préalable, d'intégrer la méthode de fouille de données proposée pour l'explication d'un phénomène *Y*, dans une approche d'Extraction de Connaissances à partir des Données pour l'explication de cas de perte de qualité locale.

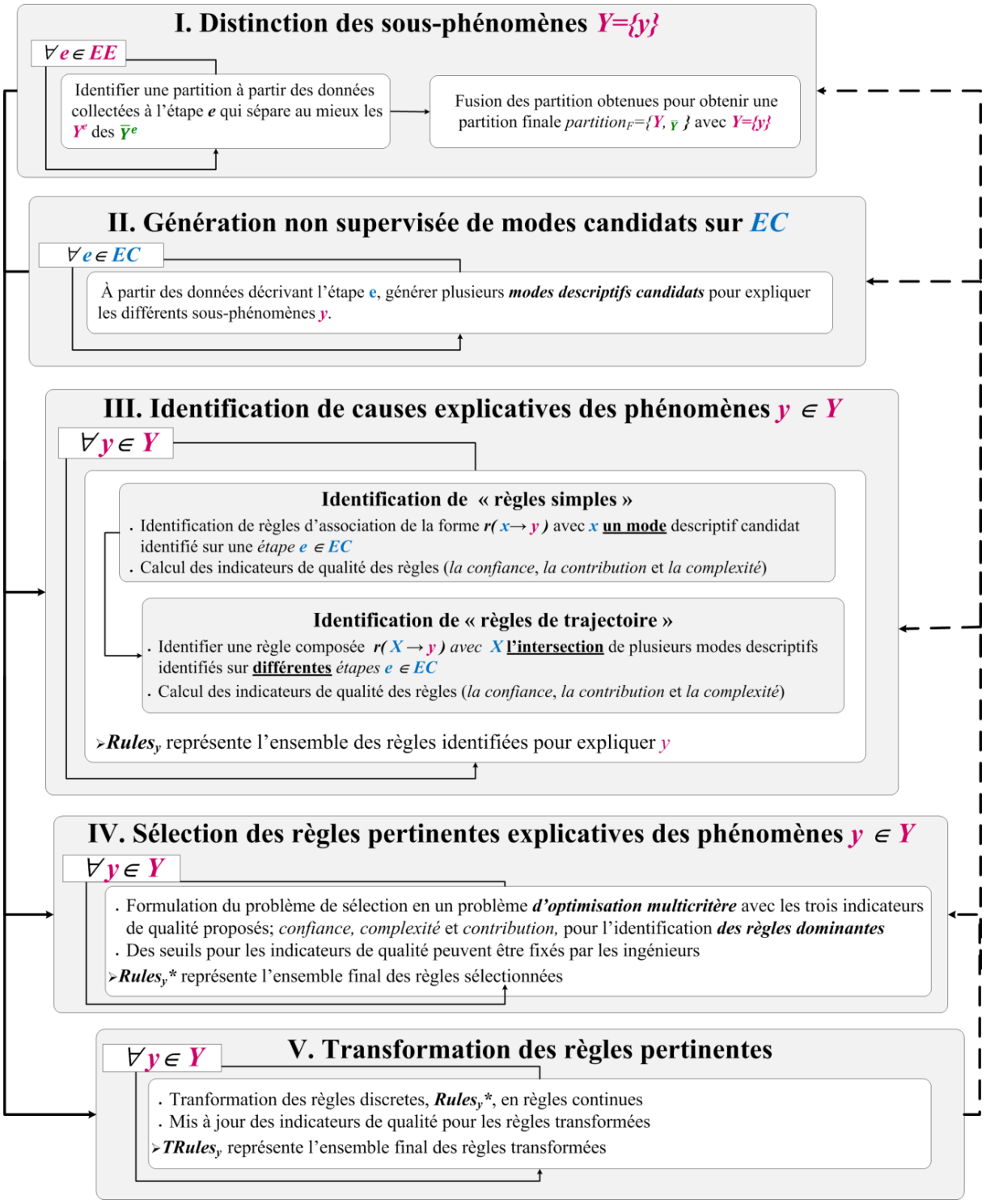


Figure 3.6: Représentation des 4 étapes de la méthode d'analyse proposée pour l'explication d'un phénomène Y

3.4 INTEGRATION DE LA METHODE PROPOSEE DANS UN PROCESSUS D'*ECD*

Nous proposons d'intégrer la méthode d'analyse proposée, *CLARIF*, dans un processus itératif et récursif d'Extraction de Connaissances à partir des Données, *ECD*, composé de trois étapes principales représentées dans la *Figure 3.7* :

- (E1) *la formalisation du problème, extraction et préparation des données*;
- (E2) *l'analyse des données* à travers l'appel de la méthode *CLARIF*.
- Et finalement (E3) *la validation et l'exploitation des connaissances identifiées par l'utilisateur final*.

Le processus d'extraction des connaissances *ECD*, initialement proposé en [25], est générique et destiné à être plus spécifiquement contextualisé, pour s'appliquer à différents cas d'application. En l'occurrence, dans le cadre de nos travaux, nous proposons une définition de chaque étape du processus *ECD* pour expliquer une perte de qualité *OOC* détectée aux étapes de contrôle qualité *Q* relatives à un segment *s*, en exploitant la méthode de fouille de données proposée *CLARIF*. Ainsi, dans les sous-sections suivantes, nous détaillerons ces trois étapes.

3.4.1 Étape 1: Formulation du problème, extraction et préparation des données

3.4.1.1 Formulation du problème

Premièrement, on s'intéresse à la formulation du problème de perte de qualité à analyser. Pour cela, on commence par définir les étapes de contrôle qualité qui ont détecté la perte de qualité. Ces étapes représentent l'ensemble *Q*. Pour chaque étape $q \in Q$, un ensemble de plaques problématiques a été détecté, noté OOC^q, et un autre ensemble de plaques ont été détectée comme bonnes, OK^q.

L'union des plaques localement problématiques, OOC^q, pour chaque étape représente l'ensemble *OOC* du phénomène de perte de qualité à expliquer. Les plaques ne décrivant pas le phénomène à expliquer sont, ainsi, représentées par l'intersection des plaques bonnes, OK^q à ces différentes étapes $\forall q \in Q$.

Un segment d'analyse est définit, en plus des étapes de contrôle qualité *Q*, par les étapes de production qui leurs sont associées. Ainsi, en ayant recours à l'expertise des ingénieurs, on identifie un ensemble d'étapes de production *P*, qui sont associés aux différentes étapes $q \in Q$, pour obtenir la définition du segment d'analyse *s*.

$$s = \{P, Q\}$$

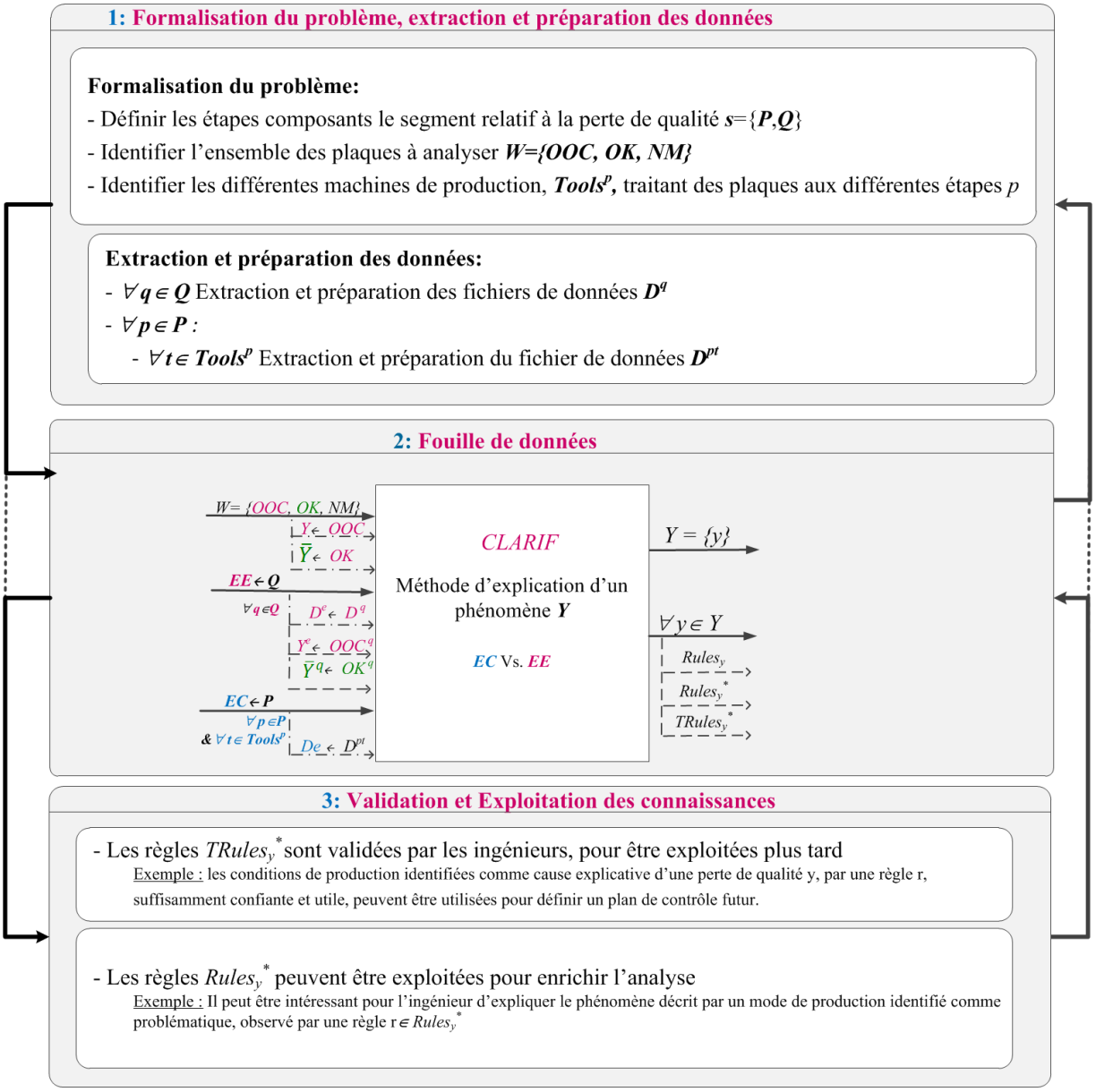


Figure 3.7: Une approche ECD pour l'explication des cas de perte de qualité locale

Durant l'intervalle de temps d'analyse considéré, en plus des plaques mesurées, qu'elles soient bonnes (*OK*) ou problématiques (*OOC*), et pour des raisons d'optimisation du temps de cycle, des plaques, notées *NM*, sont traitées aux étapes $p \in P$, mais ne sont pas mesurées durant les étapes de contrôle qualité *Q*. Pour enrichir l'analyse, la méthode *CLARIF* intègre l'historique de ces plaques dans l'analyse. Ainsi, l'ensemble des plaques dont l'historique est à analyser est noté :

$$W = \{OK \cup OOC \cup NM\}.$$

Finalement, après avoir identifié les plaques à analyser, *W*, les étapes de contrôle qualité *Q*, ainsi que les étapes de production qui leurs sont associées, *P*, on s'intéresse, pour chaque étape $p \in P$, à l'identification des différents machines de production $Tools^p$ traitant l'ensemble des produits W^p.

Pour un segment $s = \{P, Q\}$, différents équipements de production peuvent intervenir pour le traitement des plaques durant les différentes étapes de production $p \in P$. Par ailleurs, une même machine de production peut intervenir dans le traitement des produits à différentes étapes de production.

Une fois la formulation du problème à analyser constituée, on s'intéresse à l'extraction et à la préparation des données à analyser.

3.4.1.2 Extraction et préparation des données d'analyse

D'un côté, pour chaque étape de contrôle qualité $q \in Q$, on extrait le fichier de données D^q qui représente les résultats de mesures $q(w)$ des différentes plaques mesurées w durant cette étape. On note par W^q l'ensemble des plaques mesurées à l'étape q. Ainsi, D^q est définit comme suit :

$$D^q = \{q\ (w)\}\ \forall\ \mathrm{w} \in W^q$$

D'un autre côté, pour chaque équipement de production $t \in Tools$, nous collectons un fichier D^{pt} représentant les différents vecteurs de données $p^t(w)$ pour les différentes plaques w de W^{pt}. Rappelons que ce dernier représente l'ensemble des plaques traitées par l'équipement t au cours de l'étape p, et que $p^t(w)$ est le vecteur de données décrivant le fonctionnement de la machine t durant le traitement de la plaque w. Ainsi, D^{pt} est définit comme suit :

$$D^{pt} = \{p^t\ (w)\}\ \forall\ \mathrm{w} \in W^{pt}$$

À la suite de l'extraction des différents fichiers, on restreint l'ensemble *W* des plaques à analyser à l'ensemble des plaques complétement renseignées, *i.e.* ne présentant pas de données manquantes, pour toutes les étapes de production, ainsi que pour toutes les étapes de contrôle qualité si les plaques font partie de l'échantillon de contrôle sélectionné.

3.4.2 Étape 2: Fouille de données

On propose d'appliquer la méthode de fouille, *CLARIF*, proposée dans la section *3.3*, pour expliquer le phénomène de perte de qualité, *i.e.* les plaques *OOC*, à travers des conditions sur les étapes de production. Pour cela, on donne en entrée de méthode, les différents fichiers de données D^{pt} ainsi que les différents fichiers D^q, en spécifiant le phénomène à expliquer *Y*, comme étant les plaques problématiques *OOC*. Les plaques *OK*, quant à elles, représentent $\bar{Y}$. Par ailleurs, les étapes *EE* et *EC* sont respectivement,

représentées par les étapes *Q* et *P*. Et pour finir, on précise l'ensemble des machines de production $Tools^p$ pour chaque étape *p* de *P*.

L'étape de prétraitement de *CLARIF* permettra la distinction de différents modes descriptifs de perte de qualité *y* définissant le phénomène global de perte de qualité *Y*. Pour expliquer chacun de ces modes, la méthode proposée permettra l'extraction de deux types de règles à savoir :

- Des règles simples, identifiant *un mode de production* sur *un équipement* donné comme une cause explicative d'un mode de perte de qualité *y*.
- Et des règles de trajectoires, identifiant ***une combinaison*** *de modes de production* sur ***différents*** *équipements d'un* ***même chemin*** *de production* comme cause explicative de *y*.

Comme décrit lors de la définition de la méthode, ces différentes règles seront représentées par deux façons différentes. On a d'un côté des règles discrètes $Rules_y^*$, et d'un autre, des règles continues, $TRules_y^*$, pour expliquer chacun des modes de perte de qualité *y*.

3.4.3 Étape 3: Intégration des connaissances identifiées

Le premier objectif de la méthode d'analyse proposée est de fournir une explication au phénomène *Y*, ceci est donné à travers les règles identifiées. Par ailleurs, à partir de l'ensemble final des règles $TRules_y^*$ expliquant un mode de perte de qualité $y \in Y$, et selon leurs indicateurs de qualité, *confiance*, *complexité* et *contribution*, les ingénieurs peuvent intégrer, de différentes façons, ces nouvelles connaissances dans les futures processus de fabrication et de contrôle, afin d'éviter des pertes de qualité similaires. Une intégration possible est de définir un nouveau plan de contrôle à travers une nouvelle stratégie de contrôle FDC, avec les limites sur les paramètres identifiés, caractérisant les modes de production problématiques *X*, afin de détecter ce problème le plutôt possible. D'autres règles, avec un faible degré de confiance, par exemple, peuvent être intégrées avec des stratégies FDC, avec des actions non bloquantes, permettant ainsi, aux ingénieurs de suivre les plaques potentiellement à risque, sans arrêter leur processus de fabrication.

D'autre part, les règles $Rules_y^*$ peuvent être exploitées pour enrichir l'analyse effectuée. Par exemple, il peut être intéressant pour l'ingénieur d'expliquer le phénomène décrit par un mode de production identifié comme problématique, par des causes externes au segment d'analyse, par exemple des actions particulières de maintenance, ou des conditions particulières sur une précédente étape de production. Il serait ainsi intéressant d'identifier une cause plus détaillée de la perte de qualité *y*. Ceci reviendrait à expliquer ce mode de production à travers d'autres sources de données par exemple des données de maintenance, ou d'autres données de production relatives à d'autres étapes précédentes. Cela permettrait, ainsi, de corréler ces modes problématiques de production à d'autres données, et de remonter ainsi,

à la cause qui peut expliquer pourquoi un équipement a fonctionné selon ces conditions particulières potentiellement problématiques.

3.5 RESUME ET CONCLUSION

Ce chapitre a eu pour objectif de réaliser une présentation globale de la méthode proposée, pour identifier statistiquement des explications à un phénomène *Y*. Notre approche se base sur une analyse par plaque et par étape. Ainsi, les deux notions clés de l'approche proposée ont été définies : (1) une décomposition en étapes et (2) une analyse de l'historique des plaques.

Le problème d'explication d'une perte de qualité ***OOC***, détectée à des étapes de contrôle qualité ***Q***, à travers des conditions particulières sur des étapes de productions ***P***, est généralisé en une problématique d'explication d'un phénomène ***Y*** détecté à des étapes effets ***EE***, à travers des conditions particulières sur des étapes causes ***EC***. Cette généralisation nous permettra de proposer une méthode générique qui permet d'expliquer un phénomène de perte de qualité et tout autre nouveau phénomène qu'on cherche à comprendre.

À partir de cette définition générique du problème et des défis industriels et scientifiques présentés dans les précédents chapitres, nous avons exprimé l'ensemble des axes et problématiques de recherches à traiter. Les principaux axes de recherche concernent (PR1) l'identification de sous-phénomènes à expliquer $y \in Y$; (PR2) la gestion de la forte dimensionnalité des données ; (PR3) La gestion de la qualité des causes explicatives à extraire, afin d'identifier des causes *nécessairement compréhensibles, valides* et *potentiellement utiles* ; et finalement, (PR4) l'identification de causes explicatives de différents types, permettant d'exploiter les avantages de la décomposition du processus de fabrication, en identifiant des causes relatives à une seule étape, mais aussi en identifiant des causes qui combinent différentes conditions relatives à plusieurs étapes.

Pour traiter ces quatre axes de recherche, une description synthétique des cinq étapes de la méthode de fouille de données *CLARIF*, a été développée. Cette méthode d'analyse est intégrée dans un processus itératif et récursif d'*Extraction des Connaissances à partir des Données* en trois étapes.

Dans le prochain chapitre, une description détaillée de la méthode *CLARIF*, sera développée en justifiant les choix des techniques utilisées.

Chapitre 4 : *CLARIF* une analyse non supervisée pour l'explication d'un phénomène

Nous nous intéressons à l'explication d'un phénomène *Y*, observé à une ou plusieurs étapes *EE*. La méthode que nous décrivons dans ce chapitre vient proposer un ensemble de causes possibles à travers des conditions particulières sur une ou plusieurs étapes *EC*. Ceci passe par cinq étapes :

- Premièrement, nous construisons des sous-groupes caractérisant le phénomène *Y*.
- Deuxièmement, différents modes descriptifs candidats sont générés de façon non supervisée et ce pour chaque étape cause *ec* de *EC*.
- Troisièmement, les modes descriptifs candidats des étapes *EC* et les modes descriptifs du phénomène *Y* sont confrontés pour identifier des corrélations, représentant les deux types de règles « *simples* » et « *de trajectoire* ».
- Quatrièmement, l'ensemble des règles pertinentes est filtré au moyen de la résolution d'un problème d'optimisation multicritères basé sur trois indicateurs, la *confiance*, la *complexité* et la *contribution*.
- Finalement, les *règles discrètes* les plus pertinentes sont transformées en *règles continues*.

Table des matières

4.1 INTRODUCTION

La méthode proposée, nommée *CLARIF*, pour *CLustering and Association Rules IdentiFier*, combine des techniques de *clustering*, de *fouille de règles d'association* ainsi que *l'induction d'arbres de décision*. Son originalité réside principalement en :

(1) une étape de prétraitement qui permet de distinguer les sous-phénomènes qui composent *Y*, à partir des données des étapes effets *EE*. Cette étape permet de mieux définir le phénomène à expliquer par la suite. Par exemple, dans le cadre de l'explication d'une perte de qualité, les produits composant le phénomène de perte de qualité ne sont pas tous nécessairement dus à la même cause. Il est, ainsi, important de distinguer les différents types de perte de qualité, appelés sous-phénomènes, pour mieux les expliquer.

(2) une génération non supervisée des modes explicatifs sur les étapes causes *EC*. Cette analyse non supervisée permet entre autres d'exploiter les données relatives aux produits non mesurés (*NM*) i.e. qui n'ont pas connus les étapes *EE*. Les modes candidats sont ainsi construits sur la base de la similarité des mesures des produits à chaque étape *e* des étapes *EC*.

Pour cela, nous procédons à l'analyse des données collectées lors de ces étapes illustrées dans la *Figure 4.1*.

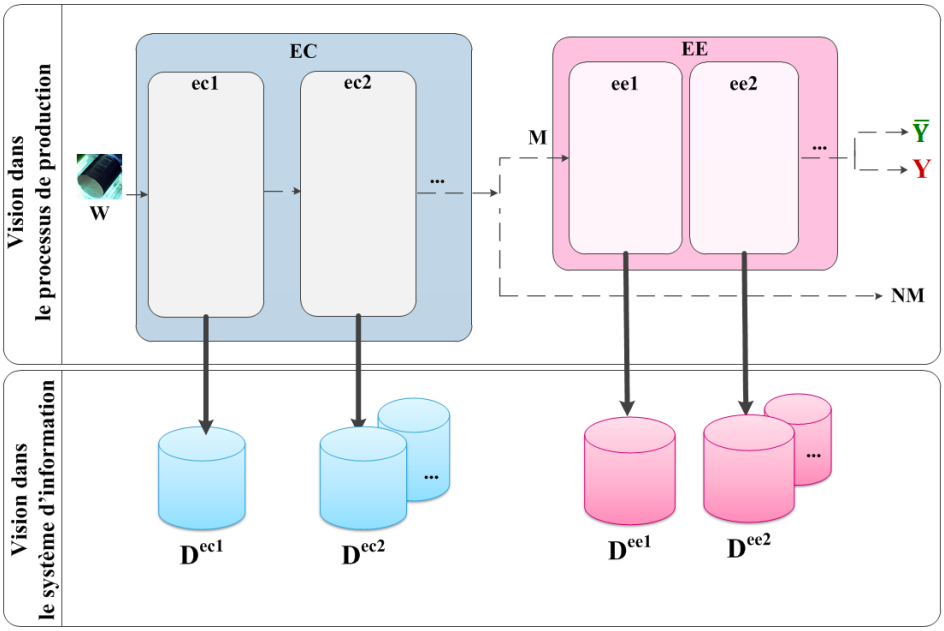


Figure 4.1: Description du contexte d'analyse pour l'explication d'un phénomène Y

Pour ce chapitre, souvent, nous nous baserons sur l'exemple donné dans le Tableau 4.1, pour expliquer *CLARIF*. L'exemple concerne 10 produits, w_1, w_2... w_{10}, qui ont suivi un processus de fabrication composé de 3 étapes de production p_1, p_2 et p_3, et sont traités par

quatre machines t_1, t_2, t_3 et t_4. Notons que la machine t_1 intervient à deux étapes différentes du processus.

Dans le Tableau 4.1, nous décrivons pour chaque produit la machine qui l'a traité durant chaque étape *p1*, *p2* et *p3*. Finalement la dernière colonne, nommée *label* donne le résultat des contrôles qualité appliqués aux produits, *1* si les produits sont défectueux, *0* sinon. Par exemple, la première ligne indique que le produit w_1 est traité par *t1* à l'étape p_1, *t4* à l'étape p_1 et *t2* à la dernière étape *p3*. Finalement, le produit w_1 est considéré comme défectueux.

Tableau 4.1: Exemple illustratif avec 10 produits, 3 étapes de production

ID	p_1	p_2	p_3	label
w_1	t_1	t_4	t_2	1
w_2	t_1	t_1	t_2	0
w_3	t_2	t_1	t_3	-
w_4	t_2	t_4	t_3	0
w_5	t_1	t_4	t_2	1
w_6	t_1	t_4	t_2	1
w_7	t_1	t_1	t_3	-
w_8	t_1	t_1	t_3	1
w_9	t_2	t_1	t_3	1
w_{10}	t_2	t_4	t_2	-

Notons que parmi les 10 produits pour lesquels des données sont disponibles, trois (w_3, w_7 et w_{10}) ne possèdent pas de label de qualité, *i.e.* ceux-ci n'ont pas été contrôlés.

On s'intéresse pour cet exemple à remonter à la cause des défauts détectés, sur les produits w_1, w_5, w_6, w_8 et w_9. Une analyse classique aurait considéré un fichier de données similaire à celui donné dans le Tableau 4.1, mais n'aurait considéré que les produits totalement renseignés, *i.e.* aurait ignoré les données relatives aux produits non mesurés (w_3, w_7 et w_{10}). *CLARIF*, par contre, exploite les données disponibles pour les produits non contrôlés, afin d'enrichir la génération des modes explicatifs sur les étapes *EC*.

Comme représenté dans la Figure 4.1, *CLARIF* se compose de cinq étapes, et prend en entrée :

- → les fichiers D^e décrivant chaque étape cause $e \in EC$,
- → les fichiers D^e décrivant chaque étape effet $e \in EE$,
- → la partition des produits *M*, qui définit le phénomène *Y* à expliquer,
- → des seuils (minimaux/maximaux) des indicateurs de qualité des causes à identifier.

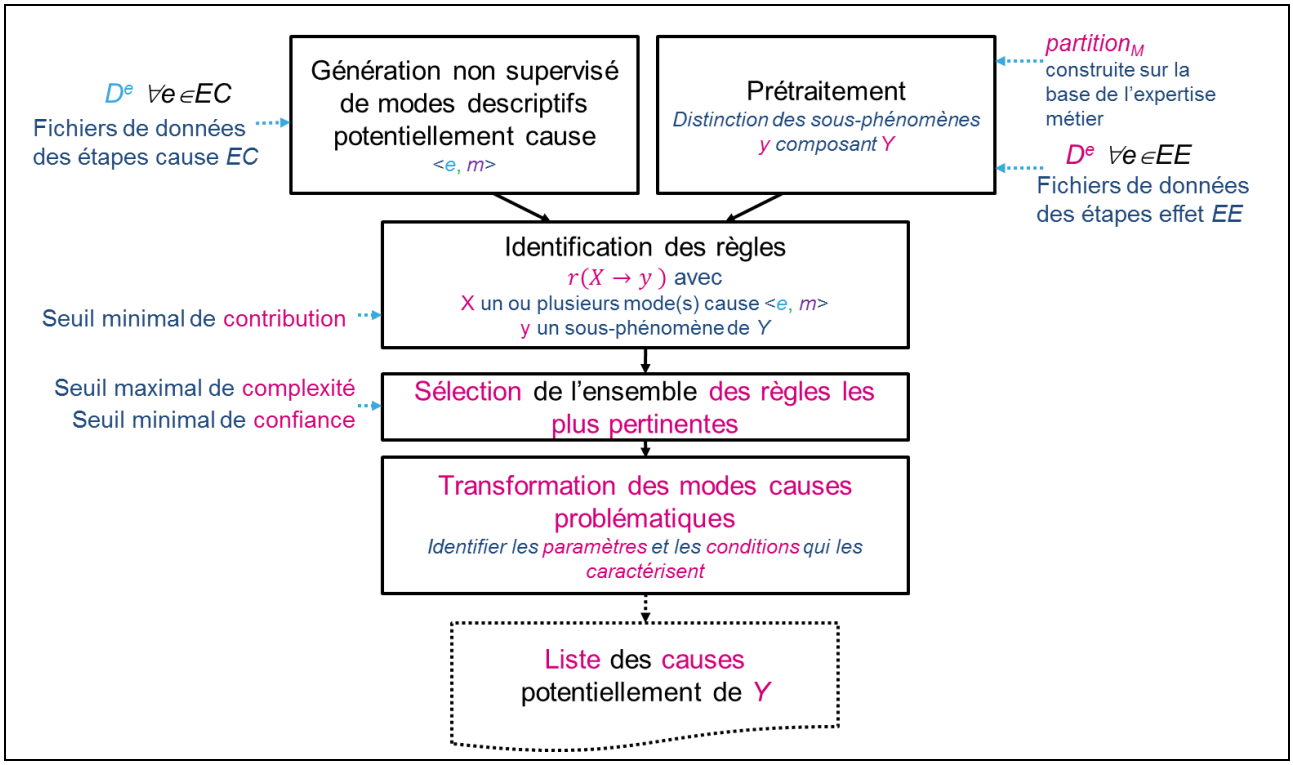


Figure 4.2 Description globale de CLARIF

Les étapes de notre méthode de fouille de données ont été déjà présentées brièvement dans le précédent chapitre. On s'intéresse, dans les sections suivantes, à détailler chacune des cinq étapes en décrivant son algorithme, et en donnant un exemple illustratif quand cela est nécessaire.

4.2 SEPARATION DE SOUS-PHENOMENES *y* COMPOSANT *Y*

L'objectif de cette première étape consiste à construire des groupes qui caractérisent différents sous-phénomènes présents dans les produits de *Y*. En effet, une explication d'une perte de qualité (*Y*) peut avoir plusieurs causes que nous cherchons, par cette première séparation des effets, à isoler les uns des autres. Cela permet de distinguer les différents sous-phénomènes afin de pouvoir les expliquer sans être biaisé par les autres. La partition initiale construite sur la base de l'expertise, $partition_M$, des plaques *M*, *i.e.* ayant connues les étapes *EE*, distingue deux groupes, les plaques constituant le phénomène *Y* et les autres, $\bar{Y}$. Comme décrit dans la *Figure 4.3*, on cherche à détailler cette partition $partition_M$ en proposant une décomposition du phénomène *Y* en plusieurs sous phénomènes *y* et en gardant le groupe des plaques $\bar{Y}$ tel qu'il est. Cette nouvelle partition est notée $partition'_M$.

$$partition_M = \{ \bar{Y}, Y \} \quad \Rightarrow \quad partition'_M = \{ \bar{Y}, \{y\} \}$$

Les nouveaux groupes de produits sont appelés des "*modes descriptifs*", ou aussi des "*clusters* ". Le problème d'explication du phénomène *Y* revient ainsi à expliquer séparément chacun des sous-phénomènes *y*.

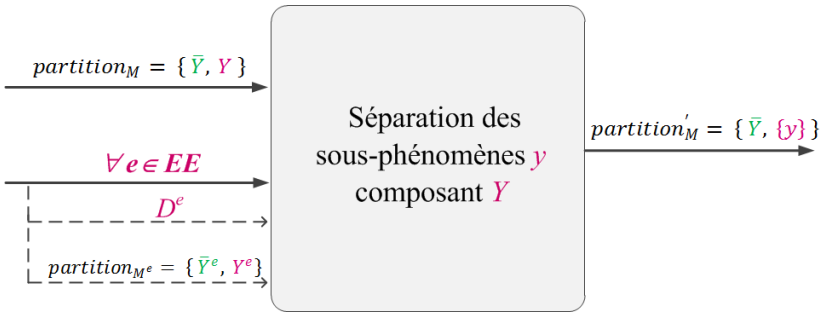


Figure 4.3: Description des entrées et sorties de la méthode de séparation des sous phénomènes y

On s'intéresse donc à analyser les données décrivant chacune des étapes *EE*, *i.e.* les différents fichiers D^e, afin de construire cette nouvelle partition des produits analysés. Dans la première sous-section, nous justifions le choix de l'algorithme de fouille de données utilisé pour l'identification de ces groupes. Puis, dans la deuxième sous-section, nous détaillerons la méthode proposée pour cela.

4.2.1 Choix de l'algorithme de fouille de données

Les techniques permettant l'identification de groupes à partir de données appartiennent à la catégorie de méthodes dites d'*apprentissage* ou de *classification non supervisé*. Ces techniques partitionnent un fichier de données composé de *n* instances et *d* dimensions en *k* groupes appelés "*clusters*". La partition est obtenue à partir des données brutes, sans connaissance à priori des relations qui existent entre les instances, dans notre cas, sans connaissance à priori de l'appartenance ou pas des produits au phénomène à expliquer *Y*. En se basant sur une mesure de similarité, les techniques de "*clustering*" partitionnent les données, tel que *les instances appartenant à un même cluster, groupe, mode, sont plus similaires les unes aux autres qu'aux instances des autres groupes*.

Pour cela, nous utilisons l'algorithme *K-means* [80]. Cet algorithme a été développé dans les années 50. Grâce à sa facilité d'implémentation, sa simplicité et son efficacité prouvée dans différents domaines, il est toujours un des algorithmes de classification non supervisée les plus utilisés [81].

En résumé, *K-means* fonctionne comme suit : (1) Initialisation de la partition à *k* clusters où chaque cluster est décrit par son centre c_i avec *i* variant *de 1* à *k*. (2) Création d'une nouvelle partition en affectant chaque instance au cluster dont le centre est le plus proche. (3) Mise à jour du centre de chaque cluster c_k. Les étapes (2) et (3) sont répétées jusqu'à stabilisation de la partition.

L'algorithme *K*-means nécessite trois réglages, la *mesure de distance* à utiliser, une *partition initiale*, et *k le nombre de groupes* que l'on cherche à identifier. Dans ce travail, nous avons utilisé la distance euclidienne pour mesurer la similarité des instances [81]. Pour initialiser la partition, *k* points sont sélectionnés aléatoirement comme étant les centres des *k* clusters et chaque point du fichier de données est affecté au plus proche centre. Par ailleurs et afin d'éviter le problème des optima locaux, où différentes initialisations peuvent résulter en différentes partitions, on propose d'appliquer l'algorithme 100 fois en variant la partition initiale, et la partition ayant la plus petite distance intra-clusters est sélectionnée. Finalement, le choix de *k* est détaillé dans la prochaine sous-section.

4.2.2 Description de la méthode proposée

La création des différents sous-phénomènes composant Y se fait en deux étapes; Premièrement, l'identification d'une partition locale pour chaque étape $e \in EE$. Deuxièmement, à partir de ces partitions identifiées localement au niveau de chaque étape $e \in EE$, on construit la partition finale, notée $partition'_M$.

Ces deux étapes sont détaillées dans les prochaines sous-sections.

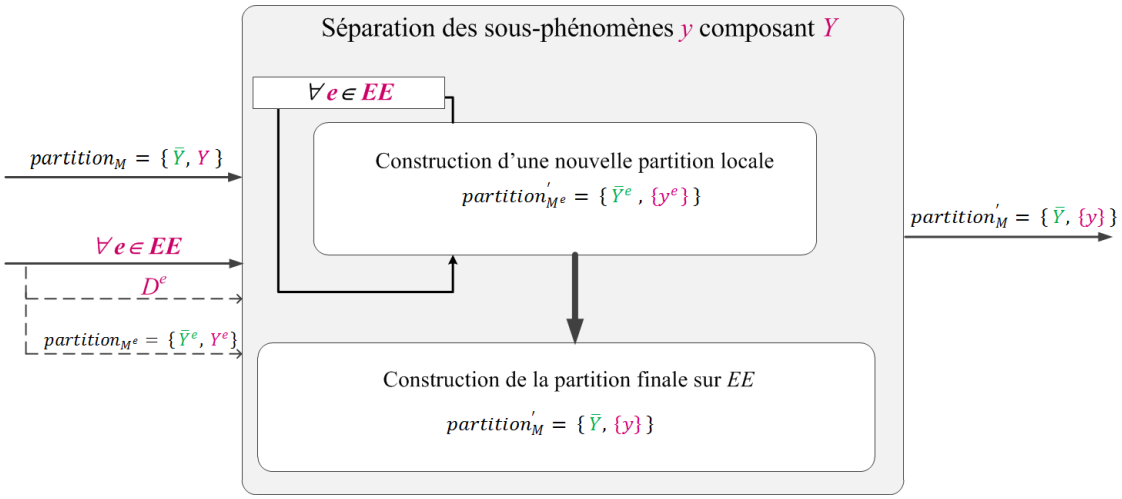


Figure 4.4 : Vision détaillée de la méthode d'identification des sous phénomènes y

4.2.2.1 Construction d'une partition locale

Comme illustré sur la *Figure 4.5*, à partir des données collectées pour une étape $e \in EE$, D^e, on cherche à bâtir une partition qui groupe les produits par similarité de leurs mesures à cette étape et qui permet de séparer au mieux les produits $\bar{Y}^e$ de ceux décrivant le phénomène à expliquer exprimé au niveau de cette étape, *i.e.* Y^e. La méthode proposée pour la construction d'une telle partition se base sur l'algorithme *K-means*.

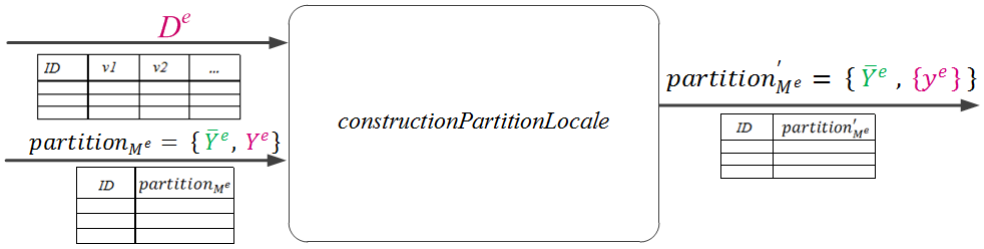


Figure 4.5: Description des entrées/sorties de la méthode d'identification d'une partition locale

Pour choisir la meilleure partition, on propose de générer plusieurs partitions en variant *k*, le nombre de groupes à identifier. On retient à la fin une seule partition, celle qui sépare le mieux les produits de Y^e de ceux de $\bar{Y}^e$.

Ainsi, tel que décrit dans *l'Algorithme 4.1*, en commençant avec *k*=2, on incrémente *k* jusqu'à $n/2$, ou jusqu'à l'obtention d'une partition qui sépare parfaitement les produits $\bar{Y}^e$ des produits Y^e. Sachant que *n* représente le nombre de produits, *i.e.* d'instance dans D^e, le seuil maximal, $k = n/2$, a été choisi pour des raisons de performance algorithmique.

Pour mesurer le degré de séparation entre les groupes des produits $\bar{Y}^e$ et ceux relatifs aux produits Y^e d'une partition, nous proposons l'*Algorithme 4.2*. Lors du calcul de cet indicateur, nous vérifions la composition de chaque cluster de la partition étudiée afin d'en mesurer sa pureté. Dans l'*Algorithme 4.2*, nous comparons le contenu d'un cluster au groupe de produits définissant le phénomène à expliquer *Y* (lignes 5-6). Ainsi, l'indicateur mesuré varie entre 0 et 1, avec 1 la valeur maximale, pour laquelle les produits Y^e, sont affectés à des clusters contenant 100% de produits appartenant à Y^e.

Algorithme 4.1: construction d'une partition locale pour une étape e ∈EE

1. **début** ***constructionPartitionLocale*** (D^e, $partition_{M^e}$)
2. soit *n* le nombre de produits dans D^e
3. $k \leftarrow 2$
4. *degreSeparation* ← *0*
5. **tant que** ($k < \frac{n}{2}$) **&** (*degreSeparation* < *1*) **faire**
6. *temp* ← *kmeans* (D^e, *k*)
7. *k* ← *k* +*1*
8. *degreSeparation_temp* ← *calculDegSeparation* (*temp*, Y^e)
9. **si** (*degreSeparation* < *degreSeparation_temp*) **faire**
10. $partition'_{M^e} \leftarrow temp$
11. *degreSeparation* ← *degreSeparation_temp*
12. **Fin**
13. **Fin**
14. **retourner** $partition'_{M^e}$
15. **Fin**

Algorithme 4.2: calcul du degré de séparation d'une partition entre les produits Y et ceux de $\bar{Y}$

début *calculDegSeparation (partition, Y)*
  degreSeparation ← 0
  n_clusters_Y← 0
  pour tout cluster $c \in partition$ **faire**
    $Y\_c \leftarrow |Y \cap c|$
    $degreSeparation \leftarrow degreSeparation + \frac{Y\_c}{|c|}$
    si ($Y\_c > 0$) **alors**
      $n\_clusters\_Y \leftarrow n\_clusters\_Y + 1$
    Fin
  Fin
  $degreSeparation \leftarrow \frac{degreSeparation}{n\_clusters\_Y}$
  retourner *degreSeparation*
Fin

On obtient, ainsi, une partition locale $partition'_{M^e}$, pour chaque étape $e \in EE$, qui affecte chaque produit w à un groupe particulier, soit un des sous-groupes crées localement, y^e, soit le groupe des produits où le phénomène n'est pas détecté, $\bar{Y}^e$.

$$partition'_{M^e} = \{\bar{Y}^e, \{y^e\}\}$$

Cette étape nous a permis de transformer, pour une étape e, un fichier de données D^e à p colonnes, en un vecteur catégoriel à une seule dimension, $partition'_{M^e}$, décrivant l'appartenance ou pas d'une plaque w à un mode $y^e \in Y^e$. La méthode proposée dans *l'Algorithme 4.1* est appliquée pour chaque étape $e \in EE$. L'étape suivante sera consacrée à utiliser ces différentes partitions localement identifiées pour construire une partition finale qui les résume.

4.2.2.2 Construction de la partition finale de *EE*

La *Figure 4.6* donne une vision globale des entrées et sorties de cette deuxième sous-étape. L'objectif, ici, est d'utiliser, ensemble, les modes trouvés indépendamment sur chaque étape e de EE, pour séparer au mieux les sous-phénomènes à expliquer dans Y. On propose d'utiliser toutes ces partitions locales pour construire une seule partition qui permet de dire si un produit correspond ou pas au phénomène à expliquer Y, et si oui, à quel sous-phénomène y.

Figure 4.6:Description des entrées/sorties de la méthode de construction de la partition finale

Pour cela, on propose la méthode décrite dans l'*Algorithme 4.3*. La partition finale, notée $partition'_M$, représente les produits $\bar{Y}$, par le label 0. Les produits à expliquer Y sont-eux représentés par différents labels : les produits qui appartiennent à un *même groupe* pour toutes partitions localement générés pour les différentes étapes $e \in EE$, ont le même label, sinon ils ont des labels différents.

L'*Algorithme 4.3* initialise la partition finale $partition'_M$ en deux groupes ; un labélisé 0 regroupant les produits $\bar{Y}$ (ligne 4), et un labélisé 1 regroupant les produits du phénomène à expliquer Y (ligne 5). Puis, la $partition'_M$ courante est itérativement comparée à la partition locale de chaque étape $e \in EE$, notée $partition'_{M^e}$. Si celle-ci distingue des sous-groupes à l'intérieur des groupes Y déjà définis dans $partition'_M$, mettre à jour cette dernière pour prendre en compte cette subdivision.

Algorithme 4.3 : constructionPartitionFinale

```
1.  Début constructionPartitionFinale ( partition'_Me1, partition'_Me2, … )
2.    Ȳ ← ⋂_∀ e∈ EE Ȳ^e
3.    Y ← ⋃_∀ e∈ EE Y^e
      // Initialisation de la partition finale partition'_M avec deux groupes Y et Ȳ
4.    partition'_M (w) ← 0 ∀ w ∈ Ȳ
5.    partition'_M (w) ← 1 ∀ w ∈ Y
6.    pour tout e∈ EE
        // Modification de partition'_M en intégrant, partition'_Me, la partition obtenue pour l'étape e
7.      tempF ← partition'_M
8.      nouveauLabel ←1
9.      pour tout y∈Y
          Soit W^y l'ensemble des produits ayant le label courant y dans la partition partition'_M
          Soit Lst_lab la liste des groupes distingués localement par partition'_Me pour les produits w ∈ W^y
10.       k ← | Lst_lab |
11.       si (k==1)
12.         tempF(w) ← nouveauLabel ∀ w∈ ci
13.         nouveauLabel ← nouveauLabel +1
14.       sinon
15.         pour tout lab∈ Lst_lab
              Soit W^lab l'ensemble des produits ayant le label courant lab dans la partition locale partition'_Me
16.           tempF(w) ← nouveauLabel ∀ w∈ W^lab
17.           nouveauLabel ← nouveauLabel +1
18.         Fin
19.       Fin
20.     Fin
21.     partition'_M← tempF
22.   Fin
23.   retourner partition'_M
24. Fin
```

Ainsi, pour chaque étape e, on identifie les labels des groupes, clusters, de la partition $partition'_M$, soit celle que l'on vient d'initialiser soit celle qui résulte de l'itération précédente. Ces différents groupes seront traités séparément (ligne 9). Pour chacun, on identifie l'ensemble des produits qui le forme, noté W^y. Pour cet ensemble de plaque, on vérifie la partition $partition'_{M^e}$ localement identifiée à l'étape e. Si celle-ci les classent tous dans un même groupe, *i.e.* $k=1$ (ligne 11), alors ils formeront un même groupe dans la mise à

jours de la $partition'_M$. Par contre, si $k>1$, autrement dit, la partition localement générée les séparent en deux ou plusieurs groupes, cette subdivision locale sera remontée dans la partition finale pour les distinguer. Ce travail sera ainsi répété pour chaque étape, en comparant à chaque fois la dernière version de $partition'_M$ à la $partition'_{M^e}$ générée précédemment.

Soit un exemple illustratif présenté dans *Figure 4.7*, avec trois étapes *e1*, *e2* et *e3*, et les partitions retenues correspondantes, $partition'_{M^{e1}}$, $partition'_{M^{e2}}$ et $partition'_{M^{e3}}$. Cet exemple traite 12 produits *A, B, C, D, E, F, G, H, I, J, K* et *L*, dont huit représentent le phénomène à expliquer *Y*.

$$Y = \{A, D, E, H, I, J, K, L\}.$$

Rappelons que le label d'un groupe n'a pas d'importance en soit, mais c'est sa composition qui importe. Ainsi, les groupements de produits sur lesquels il y a un accord pour les différentes partitions locales $partition'_{M^e}$ de $e \in EE$, seront conservés, alors que les produits sur lesquels il n'y a pas d'accord seront séparés.

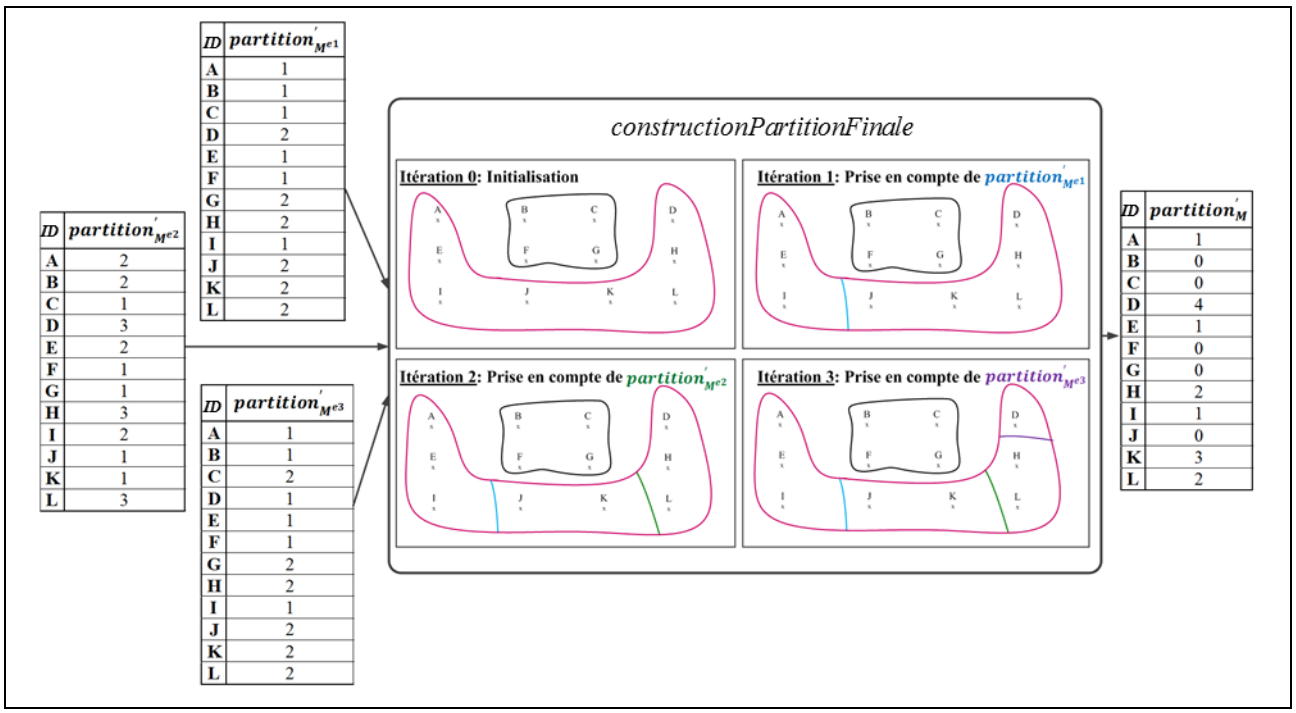


Figure 4.7: Exemple de la création de la partition finale sur EE

Notons qu'il est possible que le phénomène *Y* soit très fractionné. Au pire, chaque produit de *Y* peut être dans des clusters séparés les uns des autres. Ce cas reflète que les produits de *Y* sont différents du point de vue des données *EE*. Dans ce cas, comme nous l'avons déjà dit dans l'introduction à cette section, ces sous-phénomènes sont potentiellement dus à des causes différentes, qu'il est plus approprié de chercher à expliquer séparément.

4.3 GENERATION NON SUPERVISEE DES MODES DESCRIPTIFS CANDIDATS POUR CHAQUE ETAPE DE *EC*

Après avoir distingué les groupes *y* composant le phénomène à expliquer *Y*, la deuxième étape concerne la génération de modes descriptifs candidats pour expliquer ces différents sous-phénomènes. Ceci est fait à travers une analyse locale de chaque étape « cause » $e \in EC$. Cette troisième étape représente le cœur de *CLARIF*, la méthode proposée. Rappelons que contrairement aux méthodes traditionnelles qui se basent sur une analyse supervisée des données, *CLARIF* se base sur une *génération non supervisée* de modes descriptifs sur les étapes causes *EC* qui seront confrontés lors de l'étape suivante pour valider statistiquement la relation de causalité. Grâce à cette analyse non supervisée les modes descriptifs identifiés représentent des groupes de produits décrits par des données similaires à des étapes causes. Par exemple, si on analysait une perte de qualité à expliquer par des données de fonctionnement de machines, on chercherait à cette étape de *CLARIF* à générer, à partir des données machine, des modes descriptifs où les produits d'un même mode auront connu des conditions de fonctionnement similaires, *i.e.* des mesures similaires.

Par ailleurs, la construction de ces modes intègre des données des produits traitées aux étapes *EC* mais pas aux étapes *EE*. Ceci est utile pour les étapes de contrôle qualité intermédiaires où pour des raisons d'optimisation de coûts et de temps de cycle, seul un échantillon de produits est contrôlé. La différence entre cette analyse des signaux cause *EC* et celle des signaux effets des étapes *EE* est que, maintenant, on construit *plusieurs partitions* à partir d'un même fichier de données D^e relatif à une étape $e \in EC$, alors que la *partition* finale, construite à partir de l'analyse des données des étapes *EE*, est unique. Le choix de générer plusieurs partitions est motivé par le besoin de proposer différents modes descriptifs candidats, avec différents degrés de complexité (cf. section suivante), pour expliquer les modes descriptifs, *i.e.* les sous-phénomènes *y*. La méthode proposée, pour cette analyse locale à chaque étape, est décrite dans la sous-section suivante.

4.3.1 Description de la méthode proposée

Comme décrit dans la *Figure 4.8*, nous proposons de transformer les données D^e, relatives à une étape *e*, composées de variables catégorielles et/ou numériques, en un ensemble de $\left(\frac{n}{2} - 1\right)$ variables catégorielles, des partitions, qui indiquent l'appartenance d'un produit à un mode descriptif, composant le fichier résultant D'^{e}. Il s'agit donc de la problématique *PR2*, introduite dans le chapitre précédent, et qui possède de nombreux points communs avec la construction de modes descriptifs de sous-phénomènes $y \in Y$ de l'étape précédente. Ainsi, pour les mêmes raisons, on se base sur l'algorithme *K-means* pour grouper les produits selon la similarité de leurs mesures à une étape $e \in EC$.

Notons qu'une partition distinguant un nombre important de groupes est plus difficile à expliquer et à qualifier qu'une partition avec un faible nombre de groupes. On considère ainsi le nombre de groupes *k*, comme un *indicateur de la complexité* d'une partition. Cet indicateur nous sera particulièrement utile, plus tard dans le manuscrit, lors de la génération et sélection des règles.

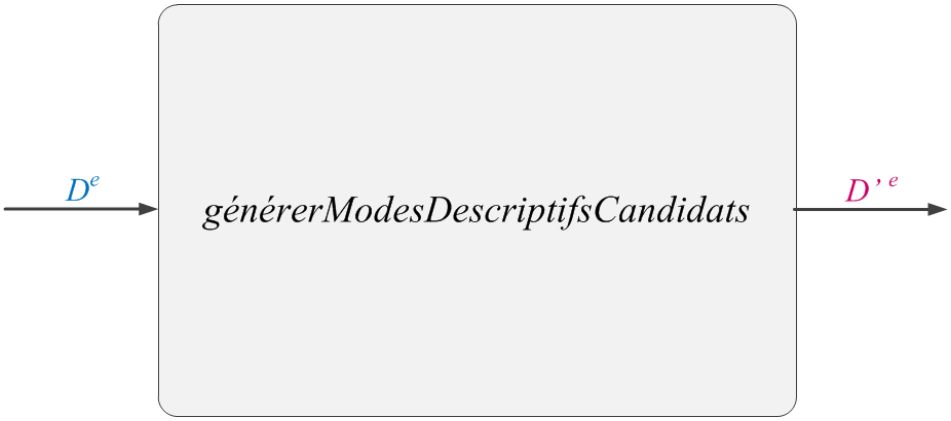


Figure 4.8 : Description des entrées/sorties de l'étape de génération de modes descriptifs candidats sur une étape e de EC

Nous construisons différentes partitions en variant *k* de 2 à $\frac{n}{2}$, avec *n* le nombre de produits W^e dont des données sont collectées à l'étape *e*. Le seuil maximal de $\frac{n}{2}$ est choisi, d'une part, pour des raisons de performance algorithmique, et d'autre part, pour avoir un choix dans la complexité des modes identifiés, et par la suite, dans celle des règles déduites. On obtient donc $\frac{n}{2}$ -1 partitions de ces produits, à chaque étape $e \in EC$, regroupant différents *modes descriptifs* candidats pour expliquer les *modes descriptifs* du phénomène *Y*. Cette méthode est décrite dans *l'Algorithme 4.4*.

Algorithme 4.4 : Identification des modes descriptifs candidats

```
1.  Début identifierModesDescriptifsCandidats(D^e)
2.    ∀ e ∈ EC
3.      Soit n la taille de D^e
4.      k ← 2
5.      Tant que ( k ≤ n/2 ) faire
6.        partition ← kmeans (D^e, k)
7.        k ← k + 1
8.        D'^e [ , k-1] ← partition
9.      Fin
10.     retourner D'^e
11.   Fin
12. Fin
```

On note le mode descriptif labélisé *x*, identifié sur l'étape *e* à travers la partition identifiant *k* modes à partir de D^e comme le couple $< e, k_k m_x >$. Par exemple, le mode numéro *4* identifié avec *k=8* clusters sur l'étape *e2*, est noté $< e_2, k_8 m_4 >$. Par ailleurs, si on choisit de considérer le niveau machine en plus, un mode descriptif est alors décrit par un

triplet de la forme $< e,\ t,\ k_k m_x >$. Ainsi, le mode numéro *4* identifié avec *k=8* clusters sur la machine *t1* à l'étape *e2*, est noté $< e_2, t_1 , k_8 m_4 >$.

4.3.2 Illustration des résultats

On s'intéresse dans cette section à illustrer le résultat de cette étape de génération de modes descriptifs candidats sur l'exemple présenté dans l'introduction de ce chapitre (cf. Tableau 4.1). Pour cela, nous analysons séparément chacun des fichiers de données D^{p1t1}, D^{p2t1}, D^{p2t4}, D^{p3t2}et D^{p3t3} relatifs aux différentes étapes causes.

Lors de la première étape p_1, la machine t_1 a traité $n= |W^{p1t1}| =6$ produits. Ainsi, on génère, à partir du fichier D^{p1t1}, $(\frac{n}{2} - 1) = 2$ partitions : une première qui distingue *2* modes et une deuxième qui distingue *3* modes. Pour les fichiers restants, seule une partition est générée à chaque fois, puisque $|W^{p1t2}| = |W^{p2t1}| = |W^{p2t4}| = |W^{p3t2}|= |W^{p3t3}| =5$ et donc, $\frac{n}{2}$ -1= 1 partition.

Les différentes partitions générées pour les différents fichiers de données seront utilisées pour transformer les vecteurs de données causes des 10 produits analysés. Pour chaque étape cause $e \in EC$, le vecteur cause relatif à un produit sera transformé en un nouveau vecteur catégorique composé des différents modes descriptifs causes représentatifs du produit dans les partitions correspondantes, à cette étape. Notons que quand le nombre de variables analysées, noté *m*, est supérieur à la moitié du nombre d'exemples analysés, *i.e.* $\frac{n}{2}$, la méthode proposée réalise une réduction des données.

Tableau 4.2: D'^{p1t1} la transformation du fichier D^{p1t1}

ID	D'^{p1t1}
w1	$< p_1, t_1, k_2m_1 >$, $< p_1, t_1, k_3m_3 >$
w2	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_1 >$
w5	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_2 >$
w6	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_2 >$
w7	$< p_1, t_1, k_2m_2 >$, $< p_1, t1, k_3m_3 >$
w8	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_3 >$

Tableau 4.3: D'^{p1t2} la transformation du fichier D^{p1t2}

ID	D'^{p1t2}
w3	$< p_1, t_2, k_2m_1 >$
w4	$< p_1, t_2, k_2m_1 >$
w9	$< p_1, t_2, k_2m_2 >$
w10	$< p_1, t_2, k_2m_2 >$

Tableau 4.4: D'^{p2t1} la transformation du fichier D^{p2t1}

ID	D'^{p2t1}
w2	$< p_2, t_1, k_2m_1 >$
w3	$< p_2, t_1, k_2m_1 >$
w7	$< p_2, t_1, k_2m_2 >$
w8	$< p_2, t_1, k_2m_2 >$
w9	$< p_2, t_1, k_2m_1 >$

Tableau 4.5: D'^{p2t4} la transformation du fichier D^{p2t4}

ID	D'^{p2t4}
w1	$< p_2, t_4, k_2m_1 >$
w4	$< p_2, t_4, k_2m_2 >$
w5	$< p_2, t_4, k_2m_2 >$
w6	$< p_2, t_4, k_2m_1 >$
w10	$< p_2, t_4, k_2m_2 >$

Tableau 4.6: D'^{p3t2} la transformation du fichier D^{p3t2}

ID	D'^{p3t2}
w1	$< p_3, t_2, k_2m_2 >$
w2	$< p_3, t_2, k_2m_1 >$
w5	$< p_3, t_2, k_2m_2 >$
w6	$< p_3, t_2, k_2m_2 >$
w10	$< p_3, t_2, k_2m_1 >$

Tableau 4.7: D'^{p3t3} la transformation du fichier D^{p3t3}

ID	D'^{p3t3}
w3	$< p_3, t_3, k_2m_1 >$
w4	$< p_3, t_3, k_2m_1 >$
w7	$< p_3, t_3, k_2m_2 >$
w8	$< p_3, t_3, k_2m_2 >$
w9	$< p_3, t_3, k_2m_2 >$

Par exemple, les trois vecteurs causes, *p1t1(w1)*, *p2t4(w1)* et *p3t2(w1)*, relatifs au produits *w1* sont transformés en trois vecteurs catégoriques :

→ *p1t1'(w1)*= ($< p_1, t_1, k_2m_1 >$, $< p_1, t_1, k_3m_3 >$) pour la première étape *p1* sur la machine *t1 ;*

→ *p2t4'(w1)*=($< p_2, t_4, k_2m_1 >$) pour la deuxième étape *p2* sur la machine *t4 ;*

→ et finalement *p3t2'(w1)* =($< p_3, t_2, k_2m_2 >$) pour la dernière étape *p3* sur la machine *t2*.

Pour cet exemple, le résultat est donné dans les Tableau 4.2 jusqu'au Tableau 4.7. On voit que pour les produits traités par t_1 lors de l'étape p_1, chacun sera décrit par deux modes, un relatif à la partition à $k=2$ et l'autre à $k=3$. Pour les autres fichiers, chaque produit sera décrit par un seul mode descriptif, puisqu'une seule partition est générée à chaque fois.

L'ensemble des modes générés ($< p_1, t_1, k_2m_1 >, < p_1, t_1, k_2m_2 >, \ldots$) représentent des modes potentiellement causes pour expliquer la perte de qualité en cours d'analyse. Ils seront confrontés lors des étapes suivantes aux modes effets qui distinguent les sous-phénomènes *y* de *Y*, afin d'identifier des relations de causalité statistiquement valides.

4.4 IDENTIFICATION DE REGLES EXPLICATIVES

On s'intéresse dans cette troisième étape à expliquer un *mode descriptif* représentant *un sous-phénomène y*, à travers un ou plusieurs des *modes descriptifs causes*. Rappelons que les *modes descriptifs causes* ont été générés de façon non supervisée à partir des données des étapes *EC*, *i.e.* sans utiliser les données des étapes *EE*.

Comme cela est décrit en *Figure 4.9*, on utilise les résultats des deux précédentes étapes, d'une part $partition'_M$, le résultat de la première étape et, d'autre part, les fichiers D'^e générés pour les différentes étapes $e \in EC$ lors de la deuxième étape de *CLARIF*. La confrontation de ces deux résultats proposera des explications possibles pour chaque sous-phénomène $y \in Y$ sous la forme d'un ensemble de règles d'association, noté $Rules_y$. Une règle d'association est une paire de modes descriptifs, l'un venant des étapes causes *EC*, disons *x*, l'autre mode étant associé à *EE*, *i.e.*, *y*. Dans ce cas, nous noterons la règle r(x→y), dont le sens est "les produits de y étaient dans x".

Par ailleurs, on se base sur la valeur du paramètre « *min Contribution* », un seuil minimal de l'indicateur de *contribution* (cf. sous-section 4.4.2) fixé par l'utilisateur final, afin d'optimiser la génération de ces règles.

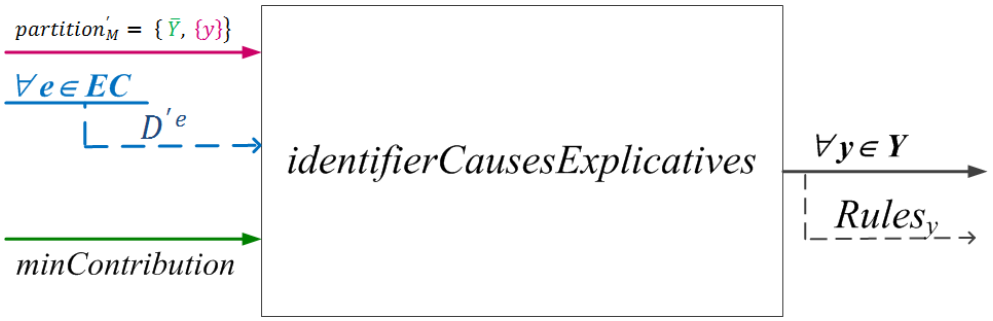


Figure 4.9 : Description des entrées/sorties de l'étape d'identification des causes

4.4.1 Choix de l'algorithme de fouille de données

On cherche à identifier des motifs relationnels qui mettent en lien un *mode y*, caractéristique d'un groupement de produits vus aux étapes *EE*, avec un ou plusieurs *modes X*, décrivant un groupement de produits vus à des étapes *EC*. Pour cela, on propose d'utiliser des techniques d'identification de règles d'associations. Une règle d'association, de type *r(X→y),* peut donc aussi être considérée comme une association entre un sous-ensemble de modes de *EC* et un des modes de *EE*. Cette approche a l'avantage de produire une connaissance intuitive, facile à comprendre puisqu'elle représente un lien entre une cause *X* et un effet *y*, participant ainsi à traiter la troisième problématique, *PR3*.

Les techniques d'identification de règles d'association ont été initialement proposées en 1994 par Agrawal et al. [60]. Leur objectif peut être appréhendé à travers le problème d'analyse du « panier de la ménagère », où il s'agit de déterminer les combinaisons d'articles achetés ensembles. Le fichier d'analyse, noté *T*, représente un ensemble de *n* transactions, où chaque transaction décrit les articles achetés ensemble par un client à un instant donné.

Plus précisément, nous utilisons l'algorithme *APRIORI* introduit dans [60], qui est le premier algorithme et l'un des plus employés pour l'identification de règles d'association. *APRIORI* est composé de deux phases : La première phase consiste à repérer l'ensemble des *itemsets* fréquents. Un *itemset* est, dans notre problématique, un ensemble *X* (ou *Y*) de modes *x* (ou *y*) générés sur une étape *EC* (ou *EE*). Un *itemset* est fréquent s'il respecte le seuil minimal de support, *i.e.* un nombre minimal de transactions dans la base *T*. La deuxième phase consiste à générer des règles, à partir des *itemsets* fréquents.

Alors qu'une énumération simple peut être algorithmiquement coûteuse car s'il existe N_{EC} modes dans *EC* et N_{EE} modes dans *EE*, il y a $N_{EE} \times 2^{N_{EC}} - 1$ règles possibles (*i.e.*, une croissance géométrique du nombre de règles avec le nombre de modes), *APRIORI* permet de générer efficacement des règles, et ce grâce sa propriété d'anti-monotonie, qui stipule que tout sous-ensemble d'un *itemset* fréquent (*i.e.* son support ≥ seuil minimal de support) doit lui aussi être fréquent. *CLARIF* adaptera cette propriété pour enrichir la génération de règles (cf. section 4.4.5).

Plus de détails sur la méthode proposée seront donnés dans une prochaine section. Mais avant, et afin de mieux traiter la problématique *PR3*, nous donnons une définition plus précise des indicateurs de qualité d'une règle, dont le support que nous venons de citer. Par la suite, nous décrivons les différents types de règles que l'on cherche à identifier.

4.4.2 Les indicateurs de qualité d'une règle *r(x→y).*

Afin d'identifier des connaissances *valides*, *nécessairement compréhensibles* et *potentiellement utiles*, il est important de mesurer la qualité des règles construites. Classiquement, la pertinence d'une règle d'association est mesurée selon au moins deux indicateurs à savoir le *support* et la *confiance* [60] (cf. chapitre 2). Notre algorithme se base

sur ces indicateurs pour proposer trois indicateurs de qualité adaptés à la problématique d'explication d'un phénomène $y \in Y$. Le choix de ces indicateurs a été guidé par les préférences et les objectifs de l'utilisateur final : nous proposons des indicateurs qui soient *intelligibles à l'utilisateur*, pour lesquels il est facile de fixer un seuil de sélection. Ces indicateurs mesurent *la validité statistique*, *la compréhensibilité* et *l'utilité des règles extraites*. Nos indicateurs de qualité seront introduits au moyen des notions de table de contingence, présentées dans *Tableau 4.8*.

Soit une règle identifiée de la forme $r(x \rightarrow y)$. On considère comme positif, noté +, les produits appartenant à *y*, le sous-phénomène à expliquer. Les produits négatifs, notées –, sont tous les autres produits, appartenant à un mode différent de *y*. En colonnes, avec la notation ^ (usuellement utilisée en statistiques pour signifier l'estimation), on trouve les produits respectant (^+), ou pas (^-) la cause définie par *x*. La table de contingence *Tableau 4.8* compare les valeurs estimées aux valeurs réelles.

Tableau 4.8: table de contingence pour une règle d'association r

	^+	^-
+	**TP**	**FN**
-	**FP**	**TN**

La première cellule de *Tableau 4.8*, confrontant le résultat de la règle *r*, ^+, aux valeurs réelles, +, représente le nombre de vrais positifs, noté TP (pour « True Positives »), c'est à dire, le nombre de produits ayant connus la condition *x* et qui appartiennent au sous-phénomène *y*. TP représente le nombre de produits correctement décrits par la cause *x*. Notons que le symbole "#" signifie "nombre de".

$$TP = \#(w \in x \ \& \ w \in y) \tag{4.1}$$

FP, représente le nombre de faux positifs : confrontant, ^+ et -. FP est le nombre d'erreurs de classification par la cause *x*, *i.e.*, les produits de *x* qui n'appartiennent pas au mode de perte de qualité *y*.

$$FP = \#(w \in x \ \& \ w \notin y) \tag{4.2}$$

La deuxième colonne du tableau de contingence comptabilise, pour une règle d'association *r*, les produits non décrits par la règle.

FN, faux négatifs, représente le nombre de produits non décrits par la cause *x*, mais qui appartiennent au mode de perte de qualité *y*. Autrement dit, ce sont des produits appartenant au sous-phénomène *y*, que l'on cherche à expliquer mais qui ne sont pas expliqués par la cause *x*.

$$FN = \#(w \notin x \ \& \ w \in y) \tag{4.3}$$

TN, vrai négatifs (« True Negatives »), sont les produits qui n'appartiennent ni à la cause *x* ni au sous-phénomène *y*.

$$TN = \#(\, w \notin x \;\; \& \;\; w \notin y) \qquad (4.4)$$

En se basant sur la table de contingence, nous proposons trois indicateurs de qualité d'une règle *r(x→ y)* à expliquer un sous-phénomène *y*, à savoir, la *contribution*, la *confiance* et la *complexité.*

4.4.2.1 La contribution

La *contribution à l'explication* d'un sous-phénomène *y*, simplement appelé *contribution*, est le premier indicateur que nous proposons. Il est une adaptation d'un indicateur classique qui est le *support*. Rappelons que le *support* d'une règle mesure le nombre de produits concernés par la règle à savoir le mode cause *x* et le mode conséquence *y*.

La contribution permet de mesurer le taux d'explication d'un sous-phénomène *y* par une règle. Il est calculé selon la formule suivante :

$$contribution\ (r(x \to y)) = \frac{\#(\, w \in x \;\; \& \;\; w \in y)}{\#(\, w \in y\,)} = \frac{TP}{TP + FN} \qquad (4.5)$$

Autrement dit, la contribution d'une règle est le pourcentage de plaques appartenant à *y* et qui sont expliquées par la cause *x*. Notons que le dénominateur $TP + FN$ ne dépend pas de la règle *r* et est égal à la cardinalité de *y*, |*y*|.

Notons que l'on trouve dans la littérature cette formulation pour calculer un indicateur nommé « *recall* ». Pour notre travail, nous avons choisi de le nommer "*contribution*" et non pas de garder son nom original, car cette désignation nous paraît plus claire : cet indicateur mesure le degré de contribution d'une règle *r* à expliquer le sous-phénomène *y*. Cet indicateur sera utilisé, dès le départ, par ARCI, tel que décrit dans l'Algorithme 4.4 (cf. section 4.4.4), afin d'optimiser la génération des *itemsets* fréquents et des règles. Cette optimisation est possible grâce à la propriété d'anti-monotonie satisfaite par cet indicateur. Une justification est donnée dans ce qui suit.

La formule de calcul de l'indicateur de contribution pour une règle $r(x \to y)$, donnée ci-dessus, peut être reformulée en fonction de l'indicateur de support comme suit :

$$contribution\ (r(x \to y)) = \frac{support(x, y)}{|y|}$$

A partir de cette écriture et en sachant que |y|, qui est le nombre de lignes contenant y, est fixe, et que l'indicateur de *support* vérifie la propriété d'anti-monotonie, on peut en déduire que la *contribution* vérifie à son tour cette propriété. Nous pouvons ainsi en déduire que :

- Tous les sous-ensembles d'un itemset fréquent au sens de l'indicateur de contribution (contribution>= minContribution) sont fréquents.
- Si un itemset A n'est pas fréquent alors il n'existe pas d'itemset B tel que A⊂B qui soit fréquent.

4.4.2.2 La confiance

Par ailleurs, on ne peut considérer une règle comme une cause explicative d'un sous phénomène *y* que si elle bénéficie de « confiance ». On propose donc de considérer l'indicateur classique de « *confiance* », qui mesure le degré de certitude que *x*, associé à une règle *r*, est une cause explicative de *y*. L'indicateur de confiance est calculé comme suit :

$$confiance(r(x \rightarrow y)) = \frac{\#(\ w \in x\ \&\ w \in y)}{\#(\ w \in x)} = \frac{TP}{TP + FP} \qquad (4.6)$$

Autrement dit, la confiance d'une règle est le pourcentage de produits ayant connus la cause *x* et qui appartiennent finalement à *y* par le nombre total des produits ayant connus la cause *x*, indépendamment de leur label final.

4.4.2.3 La complexité

En ne considérant que les indicateurs de confiance et de contribution, notre problème admet une solution triviale, qui consiste à identifier un ensemble de règles, en fixant, pour chaque règle, *x* comme un mode descriptif représenté par un singleton de plaque appartenant à *y*. L'union de telles règles admet des indicateurs de confiance et de contribution optimaux (égaux à 1). Par contre, elles ne permettent pas d'aider dans la compréhension des causes d'une perte de qualité. En effet, ces règles sont trop complexes pour représenter une connaissance exploitable. Elles reproduisent directement la base de données sans en extraire de connaissance.

Cet aspect est partiellement traité, en fixant un seuil maximal de *k* à *n/2* (inférieur à *n*), lors de la génération non supervisée des modes descriptifs sur les étapes *EC*. Par ailleurs pour comparer deux règles selon ce critère, nous proposons de considérer la complexité de la construction du mode causal descriptif *x*, *i.e.* la complexité de la partition qui l'a distingué. La complexité d'une règle est ainsi définit comme *k* le nombre de groupes, *i.e.* modes, de la partition qui a générée *x*. Cette complexité est notée k_x:

$$complexité\big(r(x \rightarrow y)\big) = k_x \qquad (4.7)$$

4.4.3 Définition des types de règles

Pour expliquer un sous-phénomène $y \in Y$, différents types de règles peuvent être étudiées. D'une part, il y a les règles simples qui identifient un unique mode comme cause de *y*. Ce premier type va être présenté dans la sous-section 4.4.3.1. D'autre part, des modes élémentaires peuvent être combinés pour définir de nouvelles règles. Ces combinaisons peuvent être faites de différentes manières. C'est le sujet de la sous-section 4.4.3.2.

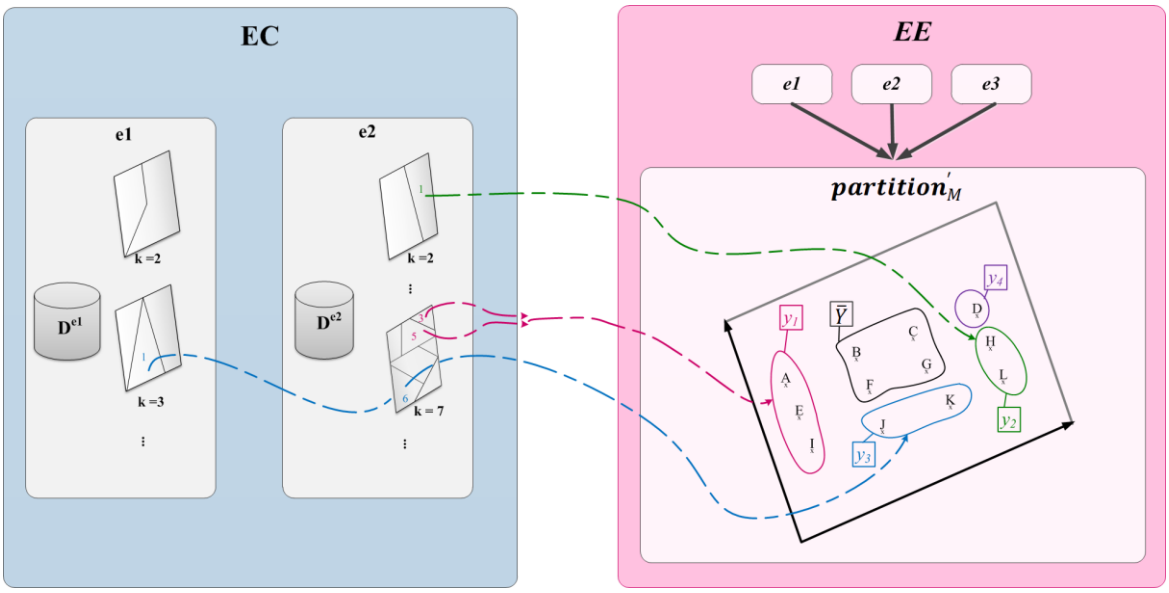


Figure 4.10: Exemple des deux types de règles à identifier. La flèche verte représente une règle simple et la bleue une règle de trajectoire. Les groupes y_1, y_2, y_3 et y_4 sont les sous-phénomènes à expliquer, $\bar{Y}$ contient les autres produits.

4.4.3.1 Les règles simples

Une règle simple donne un unique mode descriptif d'une étape cause $e \in EC$ comme une explication du sous-phénomène *y*. Dans la *Figure 4.10*, la flèche en vert, qui décrit la règle *r1*, associe le sous-phénomène *y2* au mode *1* de l'étape $e2 \in EC$ à travers la partition avec $k=2$.

$$r1 : < e_2, k_2 m_1 > \rightarrow y2$$

Dans un contexte d'explication de perte de qualité, ce type de règle permettra d'identifier l'équipement de production ainsi que son mode de production problématique potentiellement responsable de cette perte de qualité.

La qualité d'une règle simple est mesurée selon les trois formulations suivantes :

$$confiance\left(r(x \rightarrow y)\right) = \frac{TP}{TP + FP} \tag{4.8}$$

$$contribution\ (r(x \rightarrow y)) = \frac{TP}{TP + FN} \tag{4.9}$$

$$\text{complexité}\left(r(x \rightarrow y)\right) = k_x \tag{4.10}$$

4.4.3.2 Les règles composées

Pour enrichir ce premier ensemble de règles, nous créons de nouvelles règles par combinaison de modes élémentaires, cf. le *Tableau 4.9* : les combinaisons peuvent concerner une même étape $e \in EC$, ou différentes étapes $e \in EC$. Par ailleurs, on propose d'étudier deux types de combinaison, la *combinaison par union*, le *« ou »*, de différents modes élémentaires, et la *combinaison par intersection*, le *« et »*, de différents modes élémentaires.

Tableau 4.9 : Description des différents types d'enrichissements des règles simples

Etapes concernées / **Combinaison par**	**une même étape** $e \in EC$	**plusieurs étapes** $e \in EC$
intersection	non considéré	*règle de trajectoire*
union	⇔ plusieurs *règles simples*	

4.4.3.2.1 Combinaison par intersection sur une même étape $e \in EC$

La première combinaison possible est l'intersection de différents modes identifiés sur une même étape $e \in EC$. Cela peut, d'un côté, diminuer l'indicateur de *contribution*, et ainsi être contre-productif par rapport à cet indicateur. D'un autre côté, cela peut aussi augmenter considérablement la complexité de la potentielle règle à identifier. Rappelons que pour des mesures de performances algorithmiques ainsi que pour maîtriser la borne maximale de l'indicateur de confiance, on a choisi de générer des partitions à partir des données des étapes *EC*, en variant k de 2 à $n/2$. Considérer ce type de combinaison reviendrait à créer par combinaison, une partition avec une valeur k, le nombre de groupes la composant, dépassant le seuil *n/2* prédéfinit. Seule la confiance peut s'améliorer par intersection. Néanmoins, le potentiel d'amélioration de la confiance reste faible car on reste au niveau d'une même étape et il n'y a pas d'information supplémentaire.

Ainsi, on choisit de ne pas considérer ce type de combinaison.

4.4.3.2.2 Combinaison par intersection sur différentes étapes e ∈EC

La deuxième combinaison que nous étudions est celle par intersection de plusieurs modes sur différentes étapes causes *e ∈EC*. Les règles obtenues constituent le deuxième type de règles, nommées *« règles de trajectoire »*. Dans la *Figure 4.10*, la flèche en bleu, correspondant à la règle *r3*, caractérise le sous-phénomène *y3* à travers le mode *1* bâtit à l'étape *e1 ∈EC* avec *k=3 clusters*, et le mode *6* de l'étape *e2 ∈EC* à travers la partition avec *k=7*,

$$r3 : (< e_1, k_3 m_1 > \ \&\& \ < e_2, k_7 m_6 >) \rightarrow y3$$

L'objectif de la combinaison de règles simples est d'identifier de nouvelles règles qui sont potentiellement plus intéressantes selon au moins un des trois indicateurs définis. Dans ce sens, les règles de trajectoire permettent d'améliorer l'indicateur de *confiance*.

Par ailleurs, les indicateurs de qualité des *« règles de trajectoire »* sont calculés, selon les formules suivantes.

$$confiance\left(r(X \rightarrow y)\right) = \frac{TP}{TP + FP} \tag{4.11}$$

$$contribution\ (r(X \rightarrow y)) = \frac{TP}{TP + FN} \tag{4.12}$$

$$complexité\left(r(X \rightarrow y)\right) = \sum_{x \in X} k_x \tag{4.13}$$

Notons que les indicateurs de *confiance* et de *contribution* sont calculés en considérant l'intersection des modes de production problématiques $x \in X$ et non pas chaque *x* séparément.

Soit l'exemple illustratif présenté dans la *Figure 4.11*, où l'objectif est l'explication d'un phénomène de perte de qualité locale. Le phénomène *Y* a été divisé en deux sous-phénomènes *y1* et *y2*. L'identification de règles simples a permis l'explication du mode de perte de qualité *y1* avec deux modes de production *x1* et *x2* identifiés respectivement sur les équipements *t1* et *t3*. Ces règles ont des degrés de confiance respectif de $\frac{3}{4}$ et $\frac{3}{5}$.

L'identification de règles de trajectoire permet, pour cet exemple, l'identification d'une nouvelle règle $r(X \rightarrow y1)$, une combinaison de *r1* et *r2*. Ainsi, *X* est l'intersection de x_1 et x_2, autrement dit, c'est la trajectoire dessinée en noir sur la partie droite de la *Figure 4.11*, représentant les produits ayant connu le mode x_1 sur *t1* et le mode x_2 sur *t3*. La confiance de cette nouvelle règle est de 1.

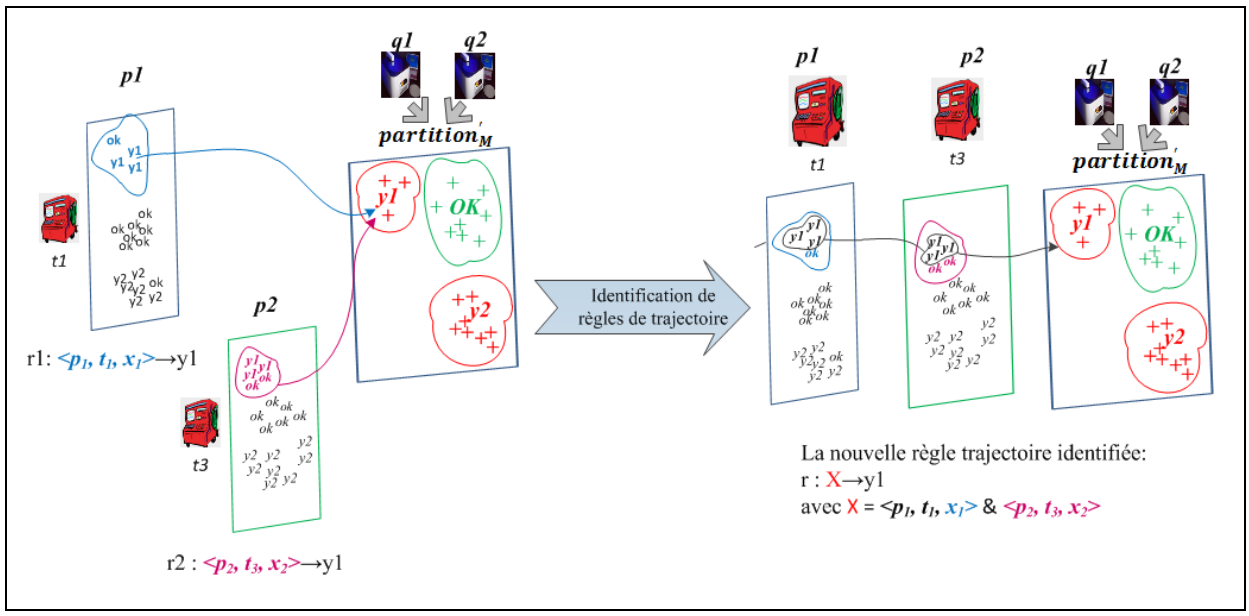


Figure 4.11: Exemple illustratif de règles simples (à gauche) et de règles de trajectoire (à droite)

4.4.3.2.3 Combinaison par union sur une même ou différentes étapes $e \in EC$

La troisième combinaison possible est la *combinaison par union* de différents modes descriptifs identifiés sur *une même étape* cause $e \in EC$. Une règle qui identifie comme cause explicative d'un *y*, une combinaison par union d'un mode *x1* et un mode *x2* sur une même étape *e1*, *i.e.* (x1 OU *x2)*, représente une connaissance déjà existante, en d'autres termes facilement accessible, à travers les deux règle simples : *r1* qui identifie *x1* comme cause de *y* et *r2* qui identifie *x2* comme cause de ce même *y*. De même, la combinaison par union de modes sur différentes étapes $e \in EC$ représente une connaissance déjà existante à travers les deux règles simples correspondantes. Enfin, la combinaison de règles par union ne peut qu'améliorer la contribution, alors que la complexité et la confiance resteront constantes voire se dégraderons.

Ainsi, nous ne considérerons pas d'avantage ce type de combinaison.

4.4.4 Algorithme de génération de règles

Pour cette troisième étape, *CLARIF* suit l'*Algorithme 4.5* afin de générer des causes potentielles d'un phénomène *Y*. Notons que chaque sous-phénomène *y* distingué précédemment sera expliqué séparément (ligne 2). La méthode proposée passe par deux phases :

Premièrement, à partir des fichiers transformés D'^{e} relatifs aux différentes étapes causes, ainsi que la partition transformée $partition'_M$, *CLARIF* extrait les données, notées *T*,

qui sont l'ensemble des modes descriptifs associés à chaque produit mesuré durant les étapes *EE, i.e.* appartenant au groupe *M*. Ainsi, chaque produit sera décrit par l'ensemble des modes descriptifs qui le représente dans chaque partition générée et ce pour chaque étape cause, *EC*, en plus du label final dans $partition'_M$.

Deuxièmement, à partir des données d'analyse, *T*, nous proposons l'algorithme *ARCI, pour Association Rules based on Clusters Identifier*, qui est une adaptation de l'algorithme de fouille de données *APRIORI*. *ARCI* permet *une génération efficace de règles d'associations avec les modes candidats générés*. Comme expliqué précédemment, l'efficacité passe par une génération par niveau des itemsets fréquents qui adapte la propriété d'anti-monotonie du *support*, à travers le nouvel indicateur de *contribution*.

Algorithme 4.5: Description de la méthode de génération de causes de Y

---

1. **Début** *identifierCauses* ($partition'_M$, $\forall e \in EC : D^{*e}$, *mincontrib)*
2. ∀ $y \in Y$ **faire**
3. Soit |*y*| la taille du sous-phénomène à expliquer *y*
4. $T \leftarrow$ ***générerDocAnalyse*** ($partition'_M$, $\forall e \in EC : D^{*e}$)
5. $Rules_y \leftarrow$ *ARCI (EC, T, y, mincontrib)*
6. **Fin**
7. **return** $\bigcup_y Rules_y$
8. **Fin**

L'algorithme *ARCI*, que nous proposons pour cette étape, décrit dans l'Algorithme 4.6, est une adaptation d'*APRIORI* pour résoudre notre problématique. En effet, le problème d'explication d'un sous-phénomène, *y*, s'intègre dans un cas particulier de recherche de règles d'association qui est la recherche de Règle d'Association de Classes *(RAC)*. Pour la recherche de règles d'association normales, tous les éléments peuvent appartenir à la partie condition ou à la conclusion de celles-ci, sans avoir une cible particulière à expliquer. Pour les règles de classes au contraire, la cible est connue. Dans notre problématique, la cible est le sous-phénomène à expliquer *y*. Ainsi, on cherche à identifier des *itemsets*, contenant obligatoirement un item *y*, qui seront transformés plus tard en règles de la forme ($r(X \rightarrow y)$).

Algorithme 4.6 : Description d'ARCI

1. **Début** *ARCI (EC, T, y, mincontrib)*
2. $L_1 \leftarrow$ *ARCI-1stPass (EC, T, y, mincontrib)*
3. **Pour** $(k=2 ; L_{k-1} \neq \emptyset ; k++)$ **faire**
4. $C_k \leftarrow$ *ARCI-gen* **(**L_{k-1}**)**
5. $\forall$ transaction $w \in T$ **faire**
6. $C_w \leftarrow$ subset (C_k, w)
7. $\forall\ c \in C_w$ **do** *c.X_count ++*
8. **si** $(y \in w)$ **faire** $\forall\ c \in C_w$ **do** *c.Xy_count ++*
9. **Fin**
10. $\forall c \in C_k$ **calculer** *c.confiance* **et** *c.contribution* **et** *c.complexité*
11. $L_k \leftarrow \{c \in C_k \mid c.contribution \geq mincontrib\}$
12. $\forall$ itemsets $c \in L_k$ **faire**
13. $\forall$ *(k-1)*-subsets *s* of *c* **faire**
14. **si** *(s.confiance* $\geq$ *c.confiance)* **alors supprimer** *c* **de** L_k
15. **Fin**
16. *rules* est l'écriture des différents *itemsets* identifiés $(\bigcup_k L_k)$ sous la forme de règles $r(X \rightarrow y)$
17. **return** *rules*
18. **Fin**

Algorithme 4.7 : Description d'ARCI-1stPass

1. **Début** *ARCI-1stPass (EC, T, y, mincontrib)*
2. $\forall e \in EC$ **faire**
3. **Soit** $candidats^e$ la liste des *modes fréquents (contribution* $\geq$ *mincontrib)* identifiés sur l'étape *e*
4. $\forall c1 \in candidats^e$ **faire**
5. $\forall\ c2 \in candidats^e \setminus \{c1\}$ **faire**
6. **si** *c1* et *c2* sont identiques **alors supprimer** *c2* **de** $candidats^e$
7. **Fin**
8. **Fin**
9. $L_1 \leftarrow \bigcup_e candidats^e$
10. $\forall c \in L_1$ **calculer** *c.confiance* **et** *c.complexité*
11. **Fin**
12. **return** L_1
13. **Fin**

Algorithme 4.8 : Description d'ARCI-gen

1. **Début** *ARCI-gen* (L_{k-1})
2. $L_{k-1} \leftarrow$ subset $(L_{k-1}$, confiance <1)
3. $C_k \leftarrow L_{k-1} * L_{k-1}$
4. $\forall$ itemsets $c \in C_k$ **faire**
5. **for all** *(k-1)*-subsets *s* of *c* **faire**
6. **si** $(s \notin L_{k-1})$ **alors**
7. **supprimer** *c* **de** C_k
8. $\forall$ itemsets $c \in C_k$ **faire**
9. $\forall$ *(k-1)*-sous-ensemble *s1* **de** *c* **faire**
10. $\forall$ *(k-1)*-sous-ensemble *s2* **de** *c* **faire**
11. **si** $(s1 \neq s2)$ **ET** *(s1.step == s2.step)* **faire**
12. **supprimer** *c* **de** C_k
13. **return** C_k
14. **Fin**

Comme précédemment énoncé, *ARCI* se base sur l'algorithme *APRIORI*. Nous gardons, donc, le fonctionnement général d'*APRIORI* à travers une génération par niveau d'*itemsets* fréquent et un élagage des *itemsets* si au moins un des sous-ensembles n'est pas fréquents. La première différence d'*ARCI* est dans l'identification de L_1 l'ensemble des *1-*

itemsets (ligne 2). Ceci passe par l'appel de la méthode *ARCI-1stPass* décrite dans l'Algorithme 4.7. Alors que dans *APRIORI*, L_1 est composé de tous les *1-items* fréquents, *i.e. support* ≥ *minsup*, nous proposons d'extraire, pour chaque étape cause $e \in EC$, les *1-itemsets* qui, en plus d'être fréquents, coexistent avec le mode *y* de la $partition'_M$. Ceci reviendrait à chercher l'ensemble des *2-itemsets* fréquents composés de deux éléments avec *y* un élément fixe.

Par ailleurs, une fois les *1-itemsets* identifiés, l'objectif est de comparer les compositions de ces modes, pour ne garder que ceux qui sont différents (lignes 4-8 de l'Algorithme 4.7). En effet, puisque des partitions différentes, avec différentes valeurs de *k*, peuvent regrouper le même sous-ensemble de produits, pour une même étape $e \in EC$, les modes candidats sont comparés pour ne garder que des *modes candidats distincts*.

Une autre modification apportée par *ARCI* concerne la génération de l'ensemble des candidats, C_k, à partir des éléments fréquents de l'itération précédente, L_{k-1}, et plus précisément, l'étape d'élagage des candidats non pertinents. Rappelons que, classiquement, l'élagage des candidats se fait en appliquant la propriété d'anti-monotonie du *support*: on supprime un candidat si au moins un des sous-ensembles n'est pas fréquent. Cet élagage est enrichi par *ARCI*, à travers deux autres conditions :

- Premièrement, une règle de trajectoire doit combiner des modes de différentes étapes causes. Ainsi, nous proposons de supprimer les candidats présentant des *items* relatifs à des modes sur les mêmes étapes (ligne 8-12 de l'Algorithme 4.8).
- Deuxièmement, les règles de trajectoires qui sont composées de plusieurs itemsets ($k>1$), servent à améliorer l'indicateur de *confiance*. Ainsi, nous proposons de supprimer un candidat si au moins un de ses sous-ensembles présente un indicateur de confiance plus intéressant. Ceci fait l'objet du post-traitement des lignes 12-14 de l'Algorithme 4.6.

Finalement, la dernière modification apportée par *ARCI* consiste dans le calcul du *support* d'un itemset, (lignes 6-9 de l'Algorithme 4.6), où on ne considère que les transactions de *T* qui correspondent au label du sous-phénomène à expliquer *y*.

Par ailleurs, notons que les indicateurs de qualité sont calculés en se basant sur les indicateurs *Xy_count* et *X_count*, qui représentent le nombre de produit correspondant, respectivement, à *X* et à *y* à la fois et à *X*. Ainsi, les formules de calcul de la *contribution* et la *confiance* deviennent:

$$contribution\ (c) = \frac{c.Xy\_count}{|y|} \tag{4.14}$$

$$confiance(c) = \frac{c.Xy\_count}{c.X\_count} \tag{4.15}$$

Pour illustrer le fonctionnement de la méthode *CLARIF*, nous reprenons l'exemple donné en introduction de ce chapitre. Nous commençons par construire le fichier d'analyse *T*, à partir des transformations, D^{*e}, des fichiers de chaque étape cause $e \in EC$, ainsi que la $partition'_M$ et ce en ne considérant que les produits *M*, ceux analysés aux étapes *EE*. Sachant que pour cet exemple, *M= {w1, w2, w4, w5, w6, w8, w9}*, pour chaque produit, nous construisons un nouveau vecteur qui enregistre les différents modes représentatifs de ce produit dans chaque partition générée pour chaque étape. Par exemple, *w1* est représenté à l'étape *p1*, par deux modes m_1 et m_3 générés respectivement par les partitions avec *k=2* et *k=3*. Pour la deuxième étape *p2*, *w1* est représenté par un seul mode puisqu'une seule partition est générée à cette étape, de même pour la dernière étape *p3*. Pour finir, on ajoute le label de $partition'_M$ identifié pour le produit. Pour le produit *w1*, le label correspondant est *y2*, le sous-phénomène qui lui correspond. Les données résultantes, *T*, sont représentées dans le Tableau 4.10.

Tableau 4.10 : Les données transformées T pour l'identification des règles

ID	items	$partition'_M$
w1	$< p_1, t_1, k_2m_1 >$, $< p_1, t_1, k_3m_3 >$, $< p_2, t_4, k_2m_1 >$, $< p_3, t_2, k_2m_2 >$	y_2
w2	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_1 >$, $< p_2, t_1, k_2m_1 >$, $< p_3, t_2, k_2m_1 >$	0
w4	$< p_1, t_2, k_2m_1 >$, $< p_2, t_4, k_2m_2 >$, $< p_3, t_3, k_2m_1 >$	0
w5	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_2 >$, $< p_2, t_4, k_2m_2 >$, $< p_3, t_2, k_2m_2 >$	y_1
w6	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_2 >$, $< p_2, t_4, k_2m_1 >$, $< p_3, t_2, k_2m_2 >$	y_1
w8	$< p_1, t_1, k_2m_2 >$, $< p_1, t_1, k_3m_3 >$, $< p_2, t_1, k_2m_2 >$, $< p_3, t_3, k_2m_2 >$	y_1
w9	$< p_1, t_2, k_2m_2 >$, $< p_2, t_1, k_2m_1 >$, $< p_3, t_3, k_2m_2 >$	y_2

Nous donnons, dans ce qui suit, une description du fonctionnement de la méthode *ARCI*, pour expliquer le sous-phénomène *y1*. Notons que le même principe sera appliqué pour à *y2*.

Tableau 4.11 : Génération des 1-itemsets pour expliquer le sous-phénomène y1

1-itemsets	produits *y1* impliqués	*X_count*	*Xy_count*	*contribution*	*confiance*
$< p_1, t_1, k_2m_1 >$	-	2	0	0	0
$< p_1, t_2, k_2m_1 >$	-	1	0	0	0
$< p_1, t_1, k_2m_2 >$	*w5, w6, w8*	4	3	1	0.75
$< p_1, t_2, k_2m_2 >$	-	1	0	0	0
$< p_1, t_1, k_3m_3 >$	*w8*	2	1	0,33	0,5
$< p_1, t_1, k_3m_1 >$	-	1	0	0	0
$< p_1, t_1, k_3m_2 >$	*w5, w6*	2	2	0,67	1
$< p_2, t_4, k_2m_1 >$	*w6*	2	1	0,33	0,5
$< p_2, t_1, k_2m_1 >$	-	2	0	0	0
$< p_2, t_4, k_2m_2 >$	*w5*	2	1	0,33	0,5
$< p_2, t_1, k_2m_2 >$	*w8*	1	1	0,33	1
$< p_3, t_2, k_2m_2 >$	*w5, w6*	3	2	0,67	0,67
$< p_3, t_2, k_2m_1 >$	-	1	0	0	0
$< p_3, t_3, k_2m_1 >$	-	1	0	0	0
$< p_3, t_3, k_2m_2 >$	*w8*	2	1	0,33	0,5

À partir du fichier *T*, *ARCI* commence par extraire les *1-itemsets*, notés C_1, représentés dans le Tableau 4.11. Les indicateurs de *contribution* et de *confiance* sont mesurés pour chaque *itemset*. Les itemsets respectant le seuil minimal de *contribution* à savoir pour cet exemple 0.33, noté L_1, sont listés dans le Tableau 4.12.

Tableau 4.12 : Les 1-itemsets retenus, L1, pour expliquer le sous-phénomène y1

1-itemsets	*contribution*	*confiance*
$< p_1, t_1, k_2m_2 >$	1	0.75
$< p_1, t_1, k_3m_3 >$	0,33	0,5
$< p_1, t_1, k_3m_2 >$	0,67	1
$< p_2, t_4, k_2m_1 >$	0,33	0,5
$< p_2, t_4, k_2m_2 >$	0,33	0,5
$< p_2, t_1, k_2m_2 >$	0,33	1
$< p_3, t_2, k_2m_2 >$	0,67	0,67
$< p_3, t_3, k_2m_2 >$	0,33	0,5

Puis *ARCI* se base sur les *1-itemsets* fréquents, L_1, pour extraire les *2-itemsets*, *i.e.* ceux composés de deux items. Ainsi, les *1-itemsets* sont combinés entre eux pour extraire les *2-itemsets*, donnés dans le Tableau 4.13. De même, seuls ceux respectant le seuil minimal de 0.33, relatifs à des étapes différentes et ne détériorant pas l'indicateur de confiance de ses sous-ensembles, composeront L_2 et sont représentés dans le Tableau 4.14. Ceux ayant une contribution faible sont supprimés. Par ailleurs, le *2-itemset "*$< p_1, t_1, k_2m_2 >, < p_1, t_1, k_3m_2 >$*"* a été supprimé car il correspond à une combinaison par intersection sur une même étape *p1*. Notons que, le *2-itemset* "$< p_2, t_4, k_2m_1 >, < p_3, t_2, k_2m_2 >$" n'est pas retenu dans la liste des *2-itemsets* fréquent car cette combinaison détériore l'indicateur de *confiance*, puisque le sous-ensemble, "$< p_3, t_2, k_2m_2 >$" seul propose une confiance à *0.66*, alors qu'elle est de *0.5* avec cette combinaison.

Tableau 4.13: Génération des 2-itemsets pour expliquer le sous-phénomène y1

2-itemsets	produits *y1* impliqués	*X_count*	*Xy_count*	*contribution*	*confiance*
$< p_1, t_1, k_2m_2 >, < p_1, t_1, k_3m_3 >$	*w8*	*1*	*1*	0,33	1
$< p_1, t_1, k_2m_2 >, < p_2, t_4, k_2m_1 >$	*w6*	*1*	*1*	0,33	1
$< p_1, t_1, k_2m_2 >, < p_2, t_4, k_2m_2 >$	*w5*	*1*	*1*	0,33	1
$< p_1, t_1, k_2m_2 >, < p_3, t_2, k_2m_2 >$	*w, w6*	*2*	*2*	0,67	1
$< p_1, t_1, k_2m_2 >, < p_3, t_3, k_2m_2 >$	*w8*	*1*	*1*	0,33	1
$< p_1, t_1, k_3m_3 >, < p_2, t_4, k_2m_1 >$	-	1	0	0	0
$< p_1, t_1, k_3m_3 >, < p_2, t_4, k_2m_2 >$	-	0	0	0	-
$< p_1, t_1, k_3m_3 >, < p_3, t_2, k_2m_2 >$	-	1	0	0	0
$< p_1, t_1, k_3m_3 >, < p_3, t_3, k_2m_2 >$	*w8*	1	1	0,33	1
$< p_2, t_4, k_2m_1 >, < p_2, t_4, k_2m_2 >$	-	0	0	0	-
$< p_2, t_4, k_2m_1 >, < p_3, t_2, k_2m_2 >$	*w6*	2	1	0,33	0,5
$< p_2, t_4, k_2m_1 >, < p_3, t_3, k_2m_2 >$	-	0	0	0	-
$< p_2, t_4, k_2m_2 >, < p_3, t_2, k_2m_2 >$	*w5*	1	1	0,33	1
$< p_2, t_4, k_2m_2 >, < p_3, t_3, k_2m_2 >$	-	0	0	0	-
$< p_3, t_2, k_2m_2 >, < p_3, t_3, k_2m_2 >$	-	0	0	0	-

Tableau 4.14 : Les 2-itemsets retenus, L_2, pour expliquer le sous-phénomène y_1

2-itemsets	*contribution*	*confiance*
$< p_1, t_1, k_2m_2 >, < p_2, t_4, k_2m_1 >$	0,33	1
$< p_1, t_1, k_2m_2 >, < p_2, t_4, k_2m_2 >$	0,33	1
$< p_1, t_1, k_2m_2 >, < p_3, t_2, k_2m_2 >$	0,67	1
$< p_1, t_1, k_2m_2 >, < p_3, t_3, k_2m_2 >$	0,33	1
$< p_1, t_1, k_3m_3 >, < p_3, t_3, k_2m_2 >$	0,33	1
$< p_2, t_4, k_2m_2 >, < p_3, t_2, k_2m_2 >$	0,33	1

Tableau 4.15 : Génération des 3-itemsets pour expliquer le sous-phénomène y_1

3-itemsets	produits *y1* impliqués	*X_count*	*Xy_count*	*contribution*	*confiance*
$< p_1, t_1, k_2m_2 >, < p_2, t_4, k_2m_2 >, < p_3, t_3, k_2m_2 >$	-	*0*	*0*	*0*	-

Finalement, lors de la troisième itération, un seul *3-itemset* peut être généré avec les *2-itemsets* identifiés précédemment, et ce tout en respectant la propriété d'anti-monotonie de la *contribution*, *i.e.* chaque sous-ensemble d'un *3-itemset* fréquent doit l'être aussi. Par exemple, pour considérer "a, b, c" dans C_3, il faut que "a, b", "a, c" ainsi que "b, c" appartiennent à L_2, l'ensemble des *2-itemsets* fréquents. Le *3-itemset* généré ne couvre aucun produit du sous-phénomène *y1*. Il ne respecte donc pas le seuil minimal de *contribution*. Ainsi, *ARCI* s'arrête avec $L_3=Ø$.

Tableau 4.16 : L'ensemble des règles explicatives du sous-phénomène y1

rID	$rules_{y1}$			**contribution**	**confidence**	**complexité**
1	$< p_1, t_1, k_2m_2 >$	$\rightarrow$	y_1	1	0.75	2
2	$< p_1, t_1, k_3m_3 >$	$\rightarrow$	y_1	0,33	0,5	3
3	$< p_1, t_1, k_3m_2 >$	$\rightarrow$	y_1	0,67	1	3
4	$< p_2, t_4, k_2m_1 >$	$\rightarrow$	y_1	0,33	0,5	2
5	$< p_2, t_4, k_2m_2 >$	$\rightarrow$	y_1	0,33	0,5	2
6	$< p_2, t_1, k_2m_2 >$	$\rightarrow$	y_1	0,33	1	2
7	$< p_3, t_2, k_2m_2 >$	$\rightarrow$	y_1	0,67	0,67	2
8	$< p_3, t_3, k_2m_2 >$	$\rightarrow$	y_1	0,33	0,5	2
9	$< p_1, t_1, k_2m_2 >, < p_2, t_4, k_2m_1 >$	$\rightarrow$	y_1	0,33	1	4
10	$< p_1, t_1, k_2m_2 >, < p_2, t_4, k_2m_2 >$	$\rightarrow$	y_1	0,33	1	4
11	$< p_1, t_1, k_2m_2 >, < p_3, t_2, k_2m_2 >$	$\rightarrow$	y_1	0,67	1	4
12	$< p_1, t_1, k_2m_2 >, < p_3, t_3, k_2m_2 >$	$\rightarrow$	y_1	0,33	1	4
13	$< p_1, t_1, k_3m_3 >, < p_3, t_3, k_2m_2 >$	$\rightarrow$	y_1	0,33	1	5
14	$< p_2, t_4, k_2m_2 >, < p_3, t_2, k_2m_2 >$	$\rightarrow$	y_1	0,33	1	4

La dernière étape *d'ARCI* consiste à écrire l'ensemble des *1-itemsets* et des *2-itemsets* fréquents, L_1 et *L2* sélectionnés, sous la forme de règles explicatives du sous-phénomène *y1*, noté $rules_{y1}$. Ces règles sont présentées dans le Tableau 4.16.

4.4.5 Etude de la complexité de l'algorithme proposé

Dans cette section, nous discutons la complexité de l'algorithme de génération de règles proposé. Dans la première sous-section, nous traitons le cas général, indépendamment des optimisations faites par *CLARIF*. Dans la deuxième sous-section, nous montrons comment *CLARIF* optimise la complexité de la génération des règles.

Nous utiliserons les notations suivantes, qui sont aussi disponibles au début du manuscrit :

Symbole	**Signification**
$\|EC\|$	Le nombre d'étapes cause
$n_e = \|W^e\|$	Le nombre de produits traités à l'étape *e*
X^e	Les modes descriptifs générés à l'étape *e* par les différentes partitions.
$\|X^e\|$	Le nombre de modes descriptifs générés à l'étape *e*
X^{EC}	L'ensemble de tous les modes descriptifs générés pour toutes les étapes $e \in EC$
$\|X^{EC}\|$	Le nombre total de modes descriptifs générés pour toutes les étapes $e \in EC$
Rules	Toutes les règles (simples et de trajectoire) identifiées
$\|Rules\|$	Le nombre total de toutes les règles (simples et de trajectoire) identifiées
N_Y	Le nombre de sous-phénomènes distingués et composant *Y*

4.4.5.1 Estimation du nombre de règles à identifier

Nous commençons par estimer le nombre de modes descriptifs générés à une étape cause *e*, par K-means, à partir des données de cette étape D^e. Il est calculé comme la somme des entiers *k* allant de 2 à $\frac{n_e}{2}$:

$$|X^e| = \sum_{k=2}^{\frac{n_e}{2}} k = \frac{\frac{n_e}{2} * (\frac{n_e}{2} + 1)}{2} - 1 = \frac{n_e^2}{8} + \frac{n_e}{4} - 1$$

Comme on peut le constater, ce nombre est <u>*indépendant du nombre de signaux enregistrés à chaque étape*</u> *e* (dimension de S^e) ce qui donne à la méthode un fort potentiel quand le nombre de signaux est grand. La seule dépendance au nombre de signaux enregistrés est celle de K-means mais une fois que K-means a terminé son exécution, cette dépendance a disparu grâce à la labélisation en modes. Par contre, ce nombre dépend du nombre de produits traités à l'étape *e*, puisqu'il est fonction de $\frac{n_e}{2}$, ce qui paraît normal (plus on a de données, plus on crée de partitions, *i.e.*, plus on monte en complexité).

Nous nous intéressons maintenant à estimer le nombre total des modes descriptifs générés à partir des données de toutes les étapes *EC*, $|X^{EC}|$. Il est calculé comme la somme des $|X^e|$ pour toutes les étapes causes :

$$|X^{EC}| = \sum_{e \in EC} |X^e|$$

Pour avoir un ordre de grandeur de ce qu'est $|X^{EC}|$, on peut considérer que n_e est constant et égal à *n*, pour toutes les étapes *EC*. Ainsi,

$$|X^{EC}| \approx O\ (|EC| * n^2)$$

À ce stade, il y a donc $|X^{EC}|$ modes descriptifs candidats pour expliquer les sous-phénomènes problématiques. Pour la génération des règles (simples et de trajectoire), nous considérons le nombre de combinaisons possibles à partir de cet ensemble. Ce qui correspond au nombre de façon d'associer n'importe lesquels de ces modes au sein de *X = {x}* = "un ensemble de modes" :

$$nombre\ de\ combinaison\ possibles\ = 2^{|X^{EC}|} - 1$$

Ainsi, pour expliquer chaque sous-phénomène $y \in Y$, le nombre total de règles (simple et de trajectoire) possible est formulé comme suit :

$$|Rules| = N_Y * \left(2^{|X^{EC}|} - 1\right)$$

Dans cette formule, on voit que chaque mode peut ou peut ne pas être dans la combinaison, soit 2 possibilités par mode, avec $|X^{EC}|$ termes, moins le cas où aucun mode n'est dans la combinaison.

Ainsi, l'ordre de grandeur du nombre total de règles identifiées est :

$$|Rules| \approx O\ (N_Y * 2^{|EC| * n^2})$$

Ce nombre de règles est très grand (géométrique) du fait de la double puissance 2^{n^2}.

4.4.5.2 Limitation du nombre de règles dans *CLARIF*

À travers la méthode *CLARIF*, nous avons choisi de ne considérer que deux types de règles, *(1)* les règles simples qui identifient un mode relatif <u>*à une étape*</u> $e \in EC$ comme une cause d'un sous-phénomène *y*, *(2)* les règles de trajectoire qui identifie une combinaison par intersection de modes descriptifs sur <u>*différentes étapes causes*</u> $e \in EC$. Ces choix nous permettent de réduire les combinaisons possibles entre les modes générés sur les étapes *EC* puisque *(1)* on ne considère que la combinaison par intersection de modes et *(2)* si on combine des modes entre eux (au sein des règles composées dites de trajectoire), ces modes doivent correspondre à différentes étapes $e \in EC$, ce qui permet de supprimer les combinaisons entre les modes d'une même étape (cf. section 4.4.3.2.1).

Ainsi, le nombre total de règles simples et de trajectoire identifiées est le suivant : Pour chaque étape *e*, soit un des modes est dans la combinaison, soit aucun mode de cette étape n'y est, ce qui fait $|X^e| + 1$ possibilités pour chaque étape, donc

$$nombre\ de\ combinaison\ de\ modes\ = \prod_{e \in EC} (|X^e| + 1) - 1$$

$$|Rules| = N_Y * \prod_{e \in EC} (|X^e| + 1) - 1$$

Pour avoir un ordre de grandeur sur ce nombre de règles, on peut dire comme plus haut que le nombre de produits traités à chaque étape est constant et égal à *n*, ainsi $|r^e|$ varie en n^2 et

$$|Rules| \approx O\big(N_Y * n^{2|EC|} \big)$$

Ce nombre de règles est désormais polynomial en fonction du nombre de produits *n*. Ainsi, on voit l'apport de la méthode proposée, qui grâce à la décomposition du problème en étapes, fait passer le nombre de règles à considérer de géométrique à polynomial.

4.5 SELECTION DES REGLES LES PLUS PERTINENTES

À ce stade de l'approche, on a identifié un ensemble de règles, noté $Rules_y$, expliquant chaque sous phénomène $y \in Y$. Ces règles ont des degrés d'intérêt différents pour expliquer *y*. Il est ainsi important de sélectionner le sous-ensemble des règles les plus pertinentes. C'est l'objectif de cette quatrième phase. Ce nouvel ensemble est noté $Rules^*_y$.

Figure 4.12 : Description des Entrées/ Sorties de la méthode de sélection des règles pertinentes

Identifier la meilleure règle selon les critères de qualité proposés précédemment admet en général plusieurs solutions. Dans la littérature, différentes approches existent. Certains travaux, comme ceux décrits dans [82], résument l'ensemble des indicateurs de qualité en un

seul, et sur la base de cet indicateur un tri des règles est effectué. D'autres travaux choisissent de considérer les différents indicateurs en globalité à travers une sélection par Front de Pareto. Dans notre cas, nous optons pour cette dernière solution. Ainsi, nous formalisons ce choix comme un problème d'optimisation multicritères où les trois indicateurs de qualité proposés sont optimisés en faisant varier *X* à travers les règles $rules_y$.

$$\begin{cases} \max(\text{confiance}) \\ \max(\text{contribution}) \\ \min(\text{complexité}) \end{cases} <=> \begin{cases} \max(\text{confiance}) \\ \max(\text{contribution}) \\ \max(-\text{complexité}) \end{cases}$$

Dans l'espace de ces trois critères, pour expliquer un mode de perte de qualité *y*, on propose de garder les règles représentant **l'ensemble de Pareto** [83], *i.e.*, l'ensemble des règles dominantes selon les trois indicateurs. Une règle *r1* domine une règle *r2*, noté $r1 \succ r2$, si les conditions suivantes sont vérifiées :

1. $c\,(r1) \geq c\,(r2)\ \forall c \in \{confiance, -complexité, contribution\}$
2. $\exists c \in \{confiance, -complexité, contribution\}$ tel que $c\,(r1) > c\,(r2)$

L'ensemble final de règles à garder, noté $rules_y^*$, est **l'ensemble de Pareto**, qui est l'ensemble de règles telles qu'aucune autre règle de $rules_y$ ne les domine,

$$Rules_y^* = \{\, r \in Rules_y \mid \nexists\, r' \in Rules_y : r' \succ r \}$$

Par ailleurs, un seuil minimal de *confiance* et/ou un seuil maximal de *complexité* peuvent être fixés par les ingénieurs, pour filtrer d'avantage l'ensemble final de règles.

Prenons une illustration composée initialement de 14 règles, avec des valeurs pour les indicateurs de qualité tel que décrit dans le *Tableau 4.17*.

Tableau 4.17: Exemple pour la sélection de règles

Rule ID	confiance	contribution	complexité
A	1.00	0.50	5
B	1.00	0.33	6
C	0.80	0.33	10
D	0.75	1.00	4
E	0.60	1.00	5
F	0.20	0.50	7
G	0.20	0.50	2
H	0.60	0.66	5
I	0.75	0.33	5
J	0.60	0.50	4
K	0.40	0.60	5
L	0.10	0.50	2
M	0.75	0.40	3
N	0.60	0.50	4

Dans la *Figure 4.13*, les règles dominantes sont dessinées en verts, et celles en bleu sont des règles dominées par au moins une règle verte. Ainsi, on réduit l'ensemble des 14 règles de départ à 4 règles pertinentes selon les trois indicateurs.

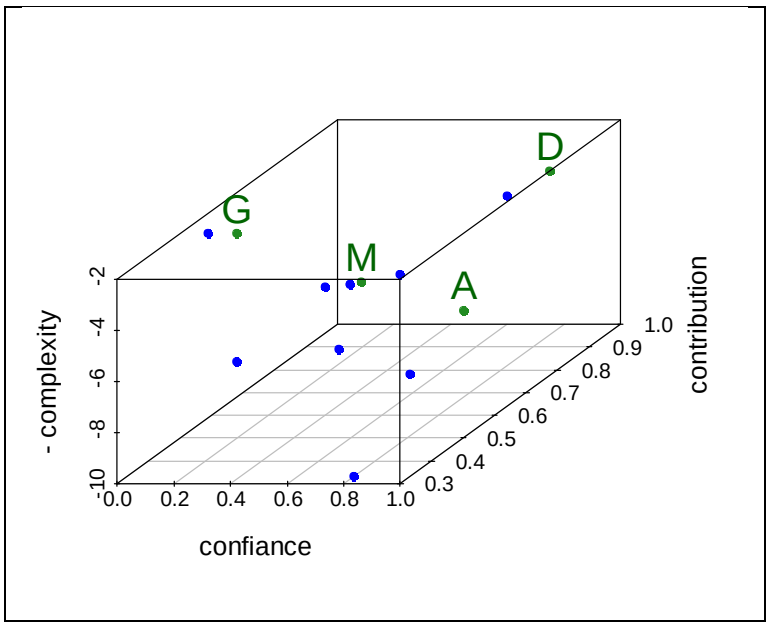


Figure 4.13: Représentation des règles dominantes

Pour revenir à notre exemple théorique, l'ensemble des 14 règles explicatives du sous-phénomène y_1, a été réduit aux trois règles, *r1*, *r3* et *r6*, qui sont dominantes, *i.e.* Pareto-optimales, par rapport aux trois indicateurs de qualité étudiés.

4.6 TRANSFORMATION DES REGLES PERTINENTES

À ce stade de l'approche, nous avons identifié un ensemble de règles, noté $Rules_Y^*$, qui expliquent un sous-phénomène *y*, à travers ses prémisses. Deux types de règles sont disponibles à savoir, des règles simples, $r(x \rightarrow y)$, dont la prémisse est un mode descriptif *x* identifié sur une étape $e \in EC$, et des règles de trajectoire, $r(X \rightarrow y)$, dont la prémisse est une intersection de modes descriptifs $x \in X$, sur différentes étapes $e \in EC$.

Ces règles sont appelées des « *règles discrètes* » car elles s'appuient sur des ensembles discrets de produits : les modes, *x*, qui composent ces règles sont des groupes de produits ayant connus des conditions similaires, et qu'à travers l'identification de règles, on a cherché une relation de causalité entre ces modes et le sous-phénomène *y*.

Par exemple, dans un contexte d'explication d'un cas de perte de qualité locale *y*, un mode de production problématique *x* signifie qu'un groupe de produits ayant connus des conditions de production similaires peut expliquer *y*. Cette première connaissance a besoin d'être enrichie en caractérisant les conditions de production qui ont généré le groupement de cet ensemble de produits dans un même mode de production *x*. Un tel retour aux conditions de production est aussi nécessaire pour étudier une perte de qualité globale, et plus généralement un phénomène *Y*. Ainsi, il est important d'expliquer le mode identifié comme problématique, *x*, à travers des conditions sur les paramètres composant l'espace initial S^e, d'une étape $e \in EC$.

Cette transformation est très utile pour l'ingénieur car elle permet d'exprimer les causes explicatives directement, par des conditions sur les paramètres de l'espace des étapes *EC*. Ainsi, les modes problématiques identifiés précédemment peuvent être expliqués dans des termes qu'un ingénieur pourrait utiliser pour effectuer un plan d'expériences par exemple, ou adapter les stratégies de contrôle en temps réel…

Les « *règles discrètes* » identifiant des modes descriptifs problématiques *x*, sont donc transformées en nouvelles « *règles continues* », notées $TRules^*_y$. La *Figure 4.14* décrit les entrées et sorties de ce traitement.

Figure 4.14 : Description des Entrées / Sorties de la méthode de transformation des règles pertinentes

4.6.1 Description de la méthode de transformation d'une règle discrète en règle continue

Pour transformer les règles discrètes en règles continues, nous cherchons à traduire les modes discrets identifiés, *X*, (un seul mode pour les règle simples, plusieurs pour les règles de trajectoire) par des modes continus notés *X'* :

$r(x' \rightarrow y)$ pour les règles simples

$r(X' \rightarrow y)$, avec $X' = \bigcap_{\forall x \in X} x'$ pour les règles de trajectoire

Les règles continues prendront la forme de branches d'arbres de décision, c'est à dire un ensemble de conditions sur les paramètres enregistrés dans D^e.

La méthode proposée est décrite dans l'Algorithme 4.9. Pour chaque $y \in Y$, chaque règle explicative correspondante *r*, chaque *x* identifié par *r* est traduit en *x'* par la méthode qui sera détaillée dans la section suivante. Dans le cas des règles de trajectoire, la transformation en mode continue est effectuée indépendamment sur chaque mode $x \in X$ et non pas directement sur X. En effet, ceci simplifie les calculs car les règles continues sont apprises dans les sous-espaces D^e plutôt que dans l'espace union. De plus, on utilise ici la

connaissance a priori que chaque x est relatif à une étape cause particulière donc aux signaux associés seulement.

Par ailleurs, et afin d'enrichir l'explication *x'* donnée à chaque mode discret *x*, nous gardons les branches issues des trois premiers arbres de décision induits. Ces trois arbres de décision sont construits à partir du fichier de données correspondant D^e, et ce en supprimant à chaque fois, les paramètres utilisés par la branche, *i.e.* la traduction précédente *x'*. On obtient ainsi, trois interprétations *x'1, x'2* et *x'3* pour caractériser un mode discret *x*. Pour finir, il est important de recalculer les indicateurs de qualité des nouveaux modes continus résultants, car cette transformation peut modifier l'ensemble des produits qui appartiennent à *x* (cf. *section 4.6.2*). Notons que la méthode *parameters*(*X'*) utilisée à la ligne 10 de l'Algorithme 4.9 renvoi l'ensemble des paramètres utilisés pour la traduction *X'*.

Pour le calcul des indicateurs de *confiance* et de *contribution* (le calcul de *TP*, *FP* et *FN)*, on utilise les mêmes formules en considérant *X'* comme la cause de *y* et en se limitant à l'ensemble des produits *M* traitées par les étapes *EE*. Par ailleurs, pour mesurer la *complexité* d'un mode continu, nous proposons une nouvelle formule qui comptabilise le nombre de paramètres intervenant dans la définition de la branche de l'arbre décrivant le ou les transformations *X'*. Ainsi, on considère qu'un mode, et par la suite une règle, est plus complexe si plus de paramètres de l'espace initial sont utilisés pour sa traduction.

Algorithme 4.9 : Transformation des règles discrètes

1. **Début** *rulesTransformation* ($\forall e \in EC : D^e$ et D'^e, $\forall y \in Y : rules_y^*$)
2. ∀ $y \in Y$
3. ∀ $r \in rules_y^*$
4. *usedParam* ← Ø
5. **Pour** i de 1 à 3 **faire**
6. ∀ $x \in X$
7. $x' \leftarrow$ ***transformation*** ($D^e \backslash usedParam$, D'^e, x)
8. $X' \leftarrow \{X' \cap x'\}$
9. **Fin**
10. *usedParam* ← union (*usedParam*, *parameters*(*X'*))
11. *TRules*← {*TRules* ∪ ($X' \rightarrow y$)}
12. **Fin**
13. *TRules* ←{*TRules* ∪ ($X' \rightarrow y$)}
14. **Fin**
15. ∀ $r \in TRules$
16. Calculer les indicateurs de *confiance*, *complexité* et de *contribution* de *r*
17. **Fin**
18. **Fin**
19. **retourner** *TRules*
20. **Fin**

4.6.2 Description de la méthode de transformation d'un mode discret *x* en un mode continu *x'*

Nous décrivons maintenant la méthode adoptée pour transformer un mode descriptif discret *x*, construit à une étape $e \in EC$, en un mode descriptif continu *x'*, à travers l'analyse des données de l'espace initial correspondant. Rappelons que ce mode descriptif *x* est un ensemble de produits ayant des caractéristiques similaires pour l'étape *e*. Pour un ingénieur il est important de connaitre les caractéristiques de ce groupement de produits. Autrement dit, quels paramètres de l'espace S^e, décrivent le mieux les produits représentés par ce mode *x*.

Figure 4.15 : Description des Entrées / Sortie de la méthode de transformation d'un mode discret x en un mode continu x'

Ainsi, et comme décrit dans la *Figure 4.15*, on reprend les données de l'espace S^e, *i.e.*, le fichier contenant les données décrivant l'étape *e*, D^e, auquel on ajoute la partition obtenue à partir de ces données et contenant le mode problématique *x*, notée $partition^x$. Celle-ci est enregistrée dans une colonne de la matrice des données D'^e. Rappelons que comme c'est le cas pour l'identification des modes descriptifs à la deuxième étape de l'approche (génération des modes explicatifs candidats), les produits non mesurés sont aussi considérés, afin d'accroitre le nombre de données disponible et enrichir, ainsi, l'analyse.

Les données pour cette phase de l'analyse, notées *D*, sont composées, pour chaque produit *w*, de deux parties : *e(w)* et *label(w)*. Rappelons que *e(w)* représente le vecteur de données décrivant l'historique du produit *w* à une étape *e*, stocké dans D^e. Par ailleurs, la $partition^x$ peut distinguer, à partir des données, plusieurs modes autres que *x*. Pour la transformation de *x*, nous nous intéressons à connaître ce qui distingue les produits composant le mode *x* du reste. On propose, donc, de labéliser les produits composant *x* par le label *1* et le reste, indépendamment de leurs modes respectifs, par *0*. Ainsi, $\forall w \in x$, label(*w*) =1 et $\forall w \notin x$, label(*w*) =0.

À partir de *D*, nous proposons l'*Algorithme 4.10*, pour transformer un mode discret *x* en un mode continu *x'* défini par une ou plusieurs branches d'un arbre de décision (lignes 2-3). L'idée générale de cette méthode est d'induire un arbre de décision [78]. Celui-ci est construit en apprenant une frontière, exprimée à travers les paramètres de l'espace S^e, qui sépare au mieux les produits labélisés 1 de ceux labélisés 0.

Nous utilisons la méthode *rpart* une implémentation sous *R* de l'algorithme *CART* d'induction d'arbres de décision, initialement proposée dans [79]. Cette opération transforme l'ensemble discret *x* (un ensemble de produits) en une ou plusieurs branches d'un arbre de décision noté *x'* (une description continue de *x*), tel que *x'* caractérise au mieux *x*.

Soit l'exemple de la *Figure 4.16*, relatif à une perte de qualité locale. La partition des données de production a identifié 5 modes de production distincts représentés par la couleur des points sur l'espace S^e. Une règle a identifié le mode représenté par les x en bleu, comme une cause explicative d'une perte de qualité *y*.

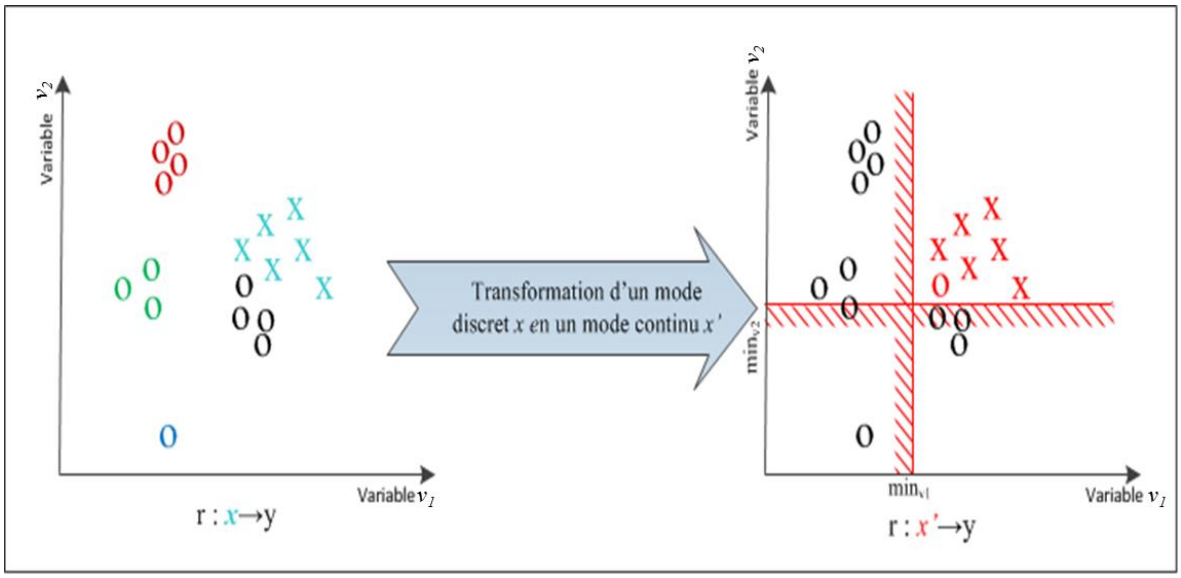


Figure 4.16 : Illustration de la transformation d'un mode de production discret x en un mode continu x'

On s'intéresse donc à donner à ce groupe, représenté initialement par un ensemble de produits, une représentation continue à travers les données de l'espace S^e. Cette représentation continue *x'* est décrite par les limites en rouge sur les deux variables analysées, de la partie droite de la Figure 4.16.

$$x' = \{v1(w) \geq min_{v1} \,\&\, v2(w) \geq min_{v2}\}.$$

On remarque, pour cet exemple, que l'arbre de décision induit a modifié la composition du mode discret *x*, en ajoutant un produit qui n'appartenait pas initialement au mode de production problématique identifié.

En général, l'induction d'arbre de décision pour décrire un mode *x* peut le transformer soit par l'ajout soit par la suppression d'instances. Cette différence signifie que la structure de l'arbre de décision induit peut ne pas permettre une séparation parfaite conformément à *x*. C'est pourquoi les indicateurs de qualité sont recalculés pour les règles à base d'arbre de décision.

Il faut noter que les algorithmes de classification supervisée classiques, comme l'algorithme CART utilisé, tendent à ignorer les classes minoritaires en effectifs, indépendamment de leurs intérêts pour l'analyse de cause. Pour autant, la plupart des règles $r(x \rightarrow y)$, visent à caractériser des modes problématiques x contenant un faible nombre de produits.

Ainsi, nous proposons une amélioration de l'algorithme de génération de règles continues (lignes 4-16 de l'Algorithme 4.10) dans le cas où *x'* est vide : nous appliquons une pondération pour chaque instance *w* du fichier de données. Cette pondération est définie par le vecteur *weight* tel que :

$$\sum_{w \in x}(weight(w)) = \sum_{w \notin x}(weight(w)) = |x| * |\bar{x}|.$$

Cette technique de ré-échantillonnage des classes minoritaires [84] rééquilibre les groupes des produits appartenant à *x* et ceux des produits n'appartenant pas à *x*, et permet ainsi d'identifier les paramètres qui caractérisent les produits de *x*, même si cette classe est minoritaire.

Algorithme 4.10 : Transformation d'un mode de production problématique x

```
1.  Début transformation (D, x)
2.      AD ← rpart (D)
3.      x' est la ou les branches identifiées par AD décrivant le groupe x
4.      Si nul (x') alors   // Rééquilibrage des groupes
5.          n1 ←|x|
6.          n2← n - n1
7.          ∀ w ∈ D
8.              Si (w∈ x) alors
9.                  weight (w)← n2
10.             Sinon
11.                 weight (w)← n1
12.             Fin
13.         Fin ∀ w ∈ D
14.         AD ← rpart (D, weight)
15.         x' est la ou les branches identifiées par AD décrivant le groupe x
16.     Fin
17.     retourner x'
18. Fin
```

4.7 RESUME ET CONCLUSION

Dans ce chapitre, nous nous sommes focalisés sur le cœur de notre approche de l'analyse de données, la méthode *CLARIF* (*CLustering and Association Rules IdentiFier*), à travers une description détaillée des outils et techniques.

La méthode de fouille de données *CLARIF* est une méthode hybride combinant des techniques issues de l'apprentissage non supervisé (*clustering)* pour la création de modes

descriptifs, et des techniques d'apprentissage supervisé telles que les *règles d'association* et l'*induction d'arbres de décision* pour l'identification de motifs relationnels représentant des causes possibles.

CLARIF est composée de cinq phases. Premièrement, on construit à travers une analyse des données décrivant les étapes *EE* où le phénomène *Y* a été observé, un ensemble de modes descriptifs qui sont autant des sous-phénomènes probables de *Y*. L'intérêt de distinguer ces groupes est de pouvoir se focaliser sur chaque sous-phénomène $y \in Y$ sans être biaisé par les autres sous-phénomènes qui n'ont potentiellement pas la même cause.

Deuxièmement, pour chaque étape $e \in EC$, en analysant le fichier de données correspondant, nous générons des modes descriptifs avec différents niveaux de complexité, qui peuvent expliquer les *y* isolés.

Une fois les modes descriptifs construits, on cherche à les corréler à chaque sous-phénomène $y \in Y$. Pour cela, on crée des règles de différents types, ayant la forme $r(X \rightarrow y)$. Pour mesurer la qualité de ces règles, et afin de produire des connaissances *valides*, *nécessairement compréhensibles* et *potentiellement utiles*, on propose trois indicateurs, *la confiance, la complexité* et *la contribution*. Pour cette troisième étape, nous avons proposé un nouvel algorithme, *ARCI* (*Association Rules based on Clusters Identifier*), qui adapte l'algorithme de fouille de règles d'association *APRIORI* pour prendre en compte les spécificités de notre problématique locale à cette étape.

Quatrièmement, en se basant sur ces indicateurs, nous filtrons les règles obtenues, à travers la résolution d'un problème d'optimisation multicritères. On ne garde ainsi que les règles Pareto-optimales selon ces indicateurs. Des seuils fixés par les ingénieurs sur ces indicateurs peuvent renforcer cette sélection.

Finalement, nous proposons de transformer les règles identifiées, qui sont à base de modes discrets, en des représentations continues des modes problématiques à travers une induction d'arbre de décision sur les données de l'espace initial. Notons que les méthodes relatives à ces différentes étapes ont été implémentées sous R [85].

Dans le prochain chapitre, nous donnons des résultats d'expérimentation validant l'approche proposée sur un cas de perte de qualité locale dans un processus de fabrication en industrie du semi-conducteur.

Chapitre 5 : Résultats d'expérimentations

Dans ce chapitre, on s'intéresse à valider la méthode de fouille de données *CLARIF* présentée dans le chapitre précédent, intégrée dans le processus *d'ECD* précédemment défini pour l'explication de cas de perte de qualité locale. Pour cela, on considère un cas d'étude réel issu du contexte industriel d'un site de fabrication de STMicroelectronics. L'objectif est d'expliquer une perte de qualité en identifiant la ou les conditions de production responsables, ainsi que les équipements problématiques correspondants.

On applique, ainsi, l'approche d'*ECD* composée de trois étapes : (E1) La formulation du problème, extraction et préparation des données. (E2) La fouille de données à travers la méthode *CLARIF* et (E3) L'intégration des connaissances identifiées. Ces trois étapes seront détaillées en application pour ce cas dans les sections de ce chapitre.

La deuxième étape de fouille de données consiste à appliquer la méthode *CLARIF* composée de cinq étapes, à savoir (1) la séparation de sous-phénomènes *y* composant *Y*. (2) la génération non supervisée des modes descriptifs candidats sur les étapes *EC*. (3) l'identification de règles explicatives. (4) la sélection des règles les plus pertinentes. Et finalement (5) la transformation des règles pertinentes.

Dans une seconde section, nous proposons de tester les performances de *CLARIF* sur un autre cas d'étude. Ces résultats seront comparés à ceux d'une approche classique par induction.

Pour finir, dans la section 5.3, et afin d'enrichir ces expérimentations, on propose une étude critique avec une proposition d'extensions pour enrichir l'application de *CLARIF* pour expliquer des pertes de qualité relatives à un long processus de production.

Table des matières

5.1 ÉTUDE D'UN CAS STMICROELECTRONICS

5.1.1 Description des données d'analyse

Le cas d'étude analysé dans cette section est extrait du système d'information du site de fabrication *Crolles 300mm* de *STMicroelectronics*. On s'intéresse à un segment du processus de fabrication, noté $s = \{p1, p2, q1, q2\}$ composé de deux étapes de production $P = \{p1, p2\}$ et deux étapes de contrôle qualité $Q = \{q1, q2\}$. *p1* est une étape de gravure, suivi par *p2*, une étape de nettoyage. *q1* et *q2* sont deux étapes de contrôle qualité, utilisées pour vérifier le bon déroulement des étapes de production *p1* et *p2*. Ces deux étapes sont liées entre elles afin de réaliser l'opération de gravure. Sur ce segment de production, les ingénieurs font face à une perte de qualité non connue, qu'ils essaient de l'expliquer afin de mettre en place des actions correctrices futures.

5.1.2 Application de l'approche proposée *CLARIF*

5.1.2.1 Étape 1: Formulation du problème, extraction et préparation des données

Guidée par la connaissance des ingénieurs, on a identifié la période de temps à investiguer, le produit affecté et on a collecté les données de production, ainsi que les mesures de qualité, décrivant ce cas d'étude. Les paramètres bruits, ainsi que les plaques avec des valeurs manquantes sont supprimés des fichiers d'analyse. Durant un mois de production, on s'intéresse à analyser les données relatives à 296 plaques.

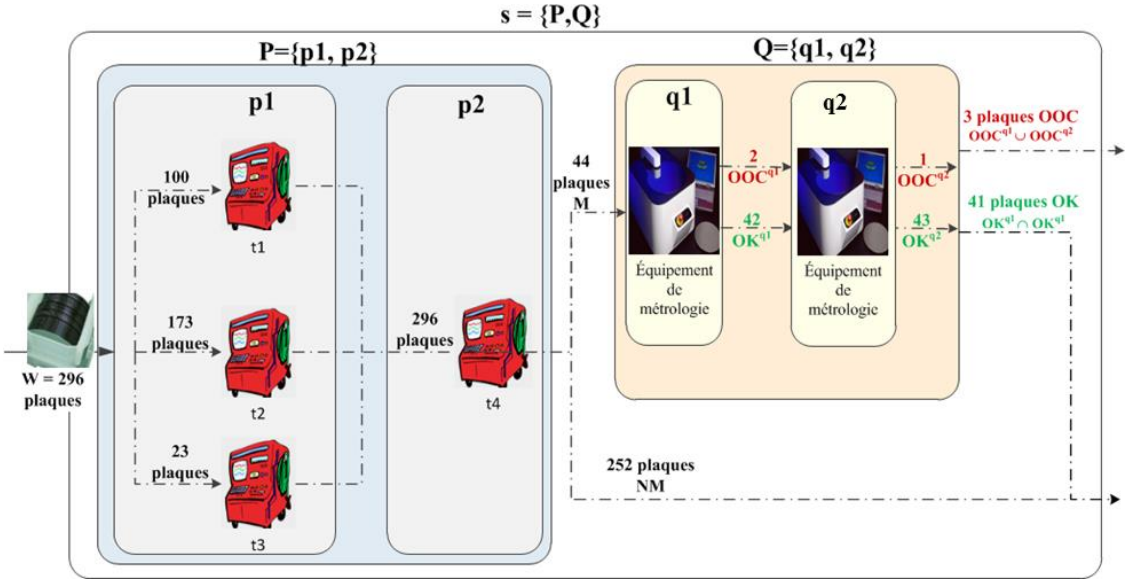


Figure 5.1: Description du cas d'étude analysé

L'historique de ces plaques dans le segment "s" est schématisé dans la Figure 5.*1*. Pour la première étape de production *p1*, trois équipements de production, *t1*, *t2* et *t3*, sont intervenus pour traiter les plaques analysées. Un seul équipement *t4* est intervenu pour l'étape de nettoyage *p2*. Sur les 296 plaques produites aux étapes *p1* et *p2*, seul un échantillon de 44 plaques sont mesurées aux étapes de contrôle qualité *q1* et *q2*. Trois plaques sont détectées comme hors contrôle *OOC* ; Deux *OOC* sont détectées à la première étape de contrôle *q1* et la troisième est détectée à la deuxième étape *q2*. On s'intéresse à identifier les causes de production qui expliquent cette perte locale de qualité.

Tableau 5.1: Description des fichiers d'analyse

Fichiers de données	Nbr de plaques	Nbr de paramètres
D^{p1t1}	100	101
D^{p1t2}	173	78
D^{p1t3}	23	101
D^{p2t4}	296	22
D^{q1}	44	17
D^{q2}	44	17

Pour ce cas d'étude, on a six fichiers de données à analyser, présentés dans *Tableau 5.1*, D^{p1t1}, D^{p1t2}, D^{p1t3}, D^{p2t4}, D^{q1} et D^{q2}, relatifs, respectivement, aux équipements *t1*, *t2*, *t3* pour l'étape *p1*, et l'équipement *t4* pour l'étape *p2* et finalement, aux étapes de contrôle qualité *q1* and *q2*. On considère les indicateurs de production résumés à partir des données collectées durant les étapes de production *p1* et *p2*. Pour les étapes de contrôle qualité, on s'intéresse à l'analyse des données de mesures collectées pour chacun des 17 sites de mesures pour chaque plaque contrôlée.

5.1.2.2 Étape 2: Fouille de données

Après l'extraction et la préparation des données, on s'intéresse, dans cette étape, à expliquer le phénomène de perte de qualité représenté par les trois plaques problématiques *OOC*. Pour cela, on propose d'appliquer la méthode de fouille de données pour l'explication d'un phénomène *Y*, en considérant le phénomène à expliquer *Y* comme étant les plaques problématiques *OOC*, les étapes effets *EE* comme *Q* et les étapes causes, *EC*, comme *P*.

L'application de cette méthode ainsi que les résultats obtenus sont décrits dans les sous-sections suivantes relatives respectivement à chaque étape de la méthode de fouille de données.

5.1.2.2.1 Séparation de sous-phénomènes y composant Y

L'objectif de cette première étape est de transformer la partition initiale $partition_M$, des plaques mesurées *M*, composée de deux groupes : un groupe pour les plaques *OK*, *i.e.* $\bar{Y}$ et

un autre pour les plaques *OOC*, *i.e. Y*, en une nouvelle partition $partition'_M$, qui distingue les différents sous phénomènes de perte de qualité *OOC*.

$$partition_M = \{OK, OOC\}$$

Le segment analysé contient deux étapes de contrôle qualité *q1* et *q2*. Pour l'identification de modes descriptifs composant le phénomène *Y*, les données relatives à chaque étape sont analysées séparément. Premièrement, pour l'étape *q1*, on applique, itérativement, l'algorithme *K-means* sur le fichier de données D_{q1}, en commençant avec *k=2*, et jusqu'à ce qu'on identifie une partition qui sépare parfaitement les 42 plaques bonnes, des 2 plaques problématiques détectées à ce niveau. Dans ce cas, la partition finale est celle identifiée avec *k=4*, illustrée dans *Figure 5.2*. Cette partition a permis de construire une partition séparant les plaques bonnes, OK^{q1}, des plaques mauvaises, OOC^{q1}, avec deux groupes relatifs aux plaques bonnes et deux autres pour les deux plaques problématiques.

À travers l'application de cette étape, on identifie des groupes de plaques ayant des mesures similaires au niveau site. Les trois partitions représentées dans *Figure 5.2* montrent que les deux plaques problématiques 5 et 36, représentées par des croix rouges, sont affectées dans des clusters différents, *i.e.* ces deux plaques ne sont pas similaires entre elles. Il est intéressant donc de les considérer comme deux familles de perte de qualité distinctes.

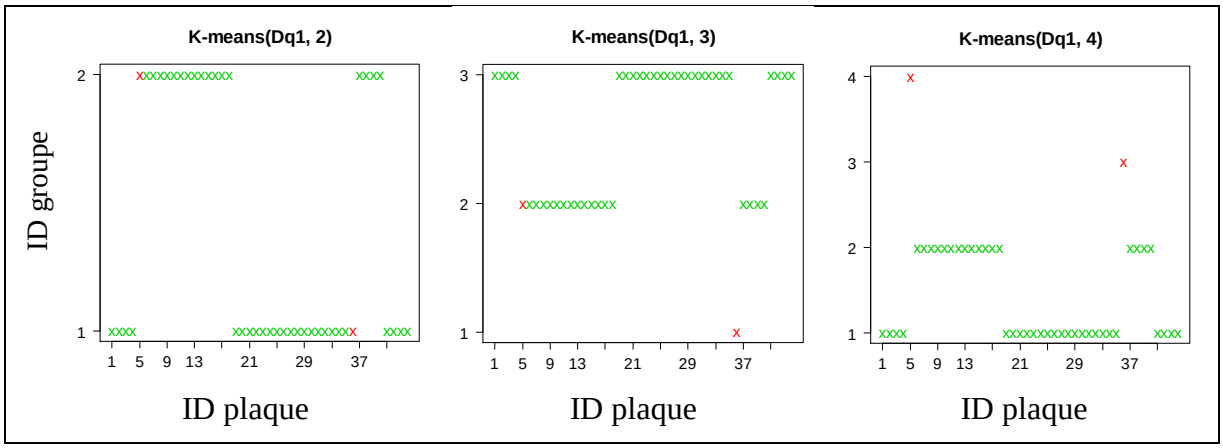


Figure 5.2: Identification de modes de qualité pour la première étape q1, en utilisant Kmeans avec k=2, k=3 et k=4.

Pour la deuxième étape *q2*, comme on est en présence d'une seule plaque *OOC*, la méthode permet de distinguer, d'un côté, un mode de perte de qualité relatif à l'unique plaque problématique détectée à ce niveau de contrôle qualité, avec d'un autre côté un deuxième mode relatif aux plaques respectant les limites de contrôle de l'étape *q2*.

Finalement, on s'intéresse à combiner les partitions obtenues pour chaque étape de contrôle de qualité *q1* et *q2*, afin d'identifier la partition finale $partition'_M$. On obtient ainsi, une partition, schématisée dans *Figure 5.3*, avec quatre modes descriptifs de qualité, avec un premier correspondant aux plaques respectant toutes les limites de contrôle de *q1* et de *q2*, labélisées 1, et trois *modes descriptifs de sous-phénomènes* à expliquer correspondant chacun

à une plaque problématique, labélisés chacun par l'identifiant de la plaque qui le compose, *i.e.*, 5, 17, et 36. On obtient, donc, trois modes de perte de qualité, représentant trois sous phénomène de *Y* à expliquer, notés *Y = {{5}, {17}, {36}}*.

Ainsi, cette étape a permis de transformer la partition initiale des plaques mesurées *M*, en une nouvelle partition $partition'_M$ composée de quatre groupes, les plaques *OK*, et trois groupes relatifs chacun à une plaque *OOC*. À partir de ce premier résultat, on peut en déduire que le phénomène de perte de qualité *Y* est potentiellement dû à trois causes différentes, qu'on cherchera à identifier, durant les étapes suivantes.

$$partition'_M = \{OK, \{5\}, \{17\}, \{36\}\}$$

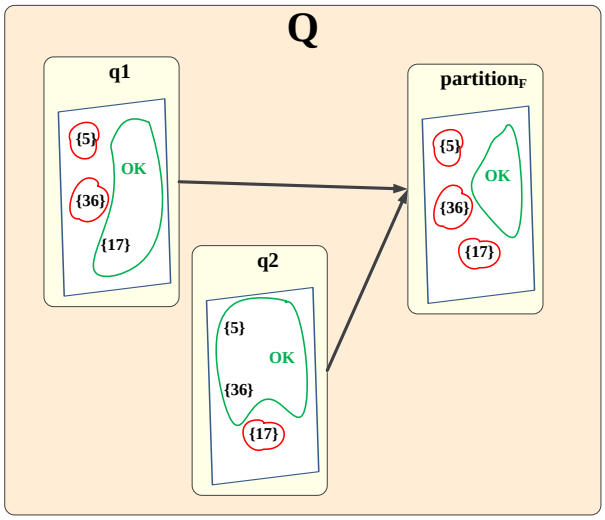


Figure 5.3: Schématisation du résultat de l''identification de modes descriptifs y de Y

5.1.2.2.2 Génération non supervisée des modes descriptifs candidats pour chaque étape de EC

Après l'identification des sous phénomènes composant le phénomène de perte de qualité *Y*, on s'intéresse, maintenant, à la génération de modes descriptifs de production candidats à expliquer les différents modes descriptifs $y \in Y$, et ceci en analysant les données relatives à chaque étape de production, $p \in P$. Pour identifier les différents modes de production, on applique l'algorithme *K-means*, en variant *k* de 2 à $\frac{n}{2}$, c.-à-d., 50, 86, 12, 148, respectivement les fichiers de données D^{p1t1}, D^{p1t2}, D^{p1t3}, et D^{p2t4}. Afin de générer des modes descriptifs de production qui décrivent au mieux les conditions de production sur chaque équipement, toutes les données relatives aux plaques traitées par cette machine, qu'elles soient ultérieurement mesurées aux étapes *Q* ou pas, sont utilisées pour l'analyse. Rappelons, que cette analyse a comme but d'identifier des groupes de plaques produites sous des conditions de production similaires.

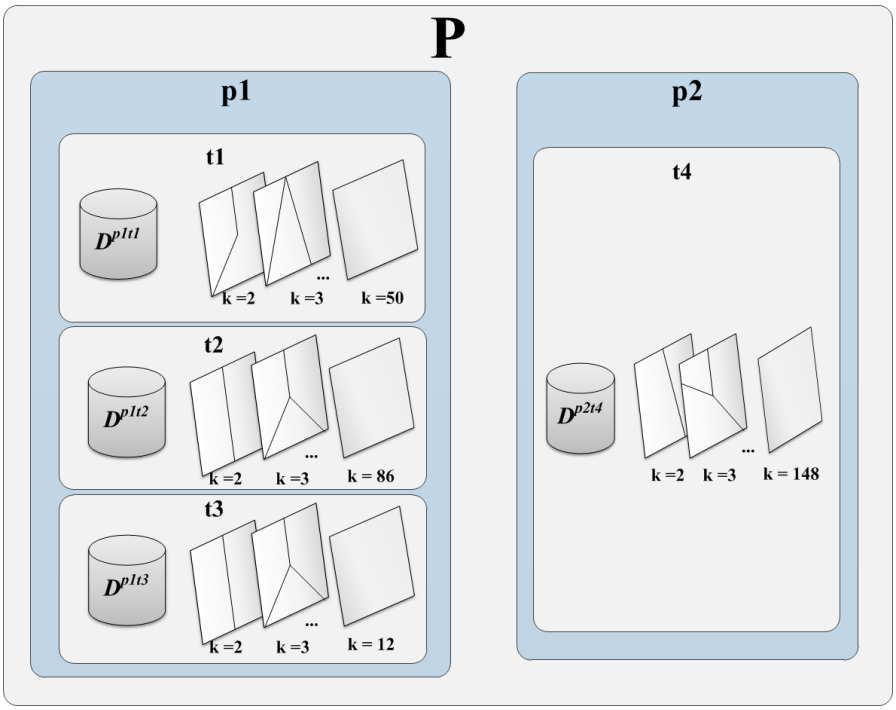


Figure 5.4: Schématisation du résultat d'identification de modes descriptifs de production

Une segmentation des différents modes de production identifiés pour les différentes partitions obtenues, en variant *k*, est donnée dans la *Figure 5.4*. Ainsi chaque plaque est affectée à plusieurs modes de production pour chaque équipement du chemin de production qu'elle a suivi. Dans la prochaine étape, ces différentes partitions sont confrontées avec la partition finale de qualité, $partition'_M$ (illustrée dans *Figure 5.3*) afin d'identifier quels modes de production expliquent les 3 modes de perte de qualité précédemment identifiés, dont le label est *5, 17* et *36*.

5.1.2.2.3 Identification des règles explicatives

La troisième étape de la méthode de fouille de données proposée concerne l'identification des causes explicatives des différents sous phénomènes identifiés *y* à expliquer. Comme décrit dans le précédent chapitre, la méthode proposée permet d'identifier deux types de règles, des règles simples et des règles de trajectoires.

Nous commençons par générer le fichier d'analyse *T*, tel que pour chaque plaque, ce fichier met en correspondance, d'un côté, le label du mode de production identifié pour chaque partition de *k=2* jusqu'à *k=n/2* des équipements de production, *t1, t2, t3* et *t4*, et d'un autre côté, le label du mode qualité, $partition'_M$, qui lui est associé. Rappelons que seules les plaques ayant été mesurées et traitée durant les étapes de production, *i.e. M,* sont considérées et ce afin de ne pas biaiser la qualité des règles identifiées.

L'analyse a permis l'identification de 52 règles simples, présentées dans les *Tableau 5.2*, Tableau 5.3 et

Tableau 5.4. Les règles identifiées ont des degrés de qualité différents. L'indicateur de *confiance* varie entre *0.03* et *1*, celui de *complexité* varie entre *2* et *104*. La contribution est toujours égale à *1*, puisque les modes de perte de qualité identifiés sont composés d'une seule plaque. Par ailleurs, 4, 10 et 192 règles de trajectoire pour expliquer, respectivement, les modes de perte de qualité *y= {5}, {17}, et {36}* ont aussi été identifiées.

Tableau 5.2: Liste des règles simples identifiées pour expliquer y = {5}

ID	Règle			Confiance	Complexité	Contribution
	Mode de production	→	***Mode de perte qualité***			
522	$<p_2, t_4, k_{39}m_{10}>$	→	y={5}	1	39	1
523	$<p_2, t_4, k_{19}m_2>$	→	y={5}	0.5	19	1
51	$<p_1, t_1, k_6m_5>$	→	y={5}	0.33	6	1
524	$<p_2, t_4, k_{22}m_{10}>$	→	y={5}	0.33	22	1
52	$<p_1, t_1, k_2m_1>$	→	y={5}	0.25	2	1
525	$<p_2, t_4, k_9m_9>$	→	y={5}	0.25	9	1
526	$<p_2, t_4, k_2m_2>$	→	y={5}	0.2	2	1

Tableau 5.3: Liste des règles simples identifiées pour expliquer y = {17}

ID	Règle			Confiance	Complexité	Contribution
	Mode de production	→	***Mode de perte qualité***			
173	$<p_1, t_1, k_{23}m_{16}>$	→	y={17}	1	23	1
1727	$<p_2, t_4, k_{70}m_{31}>$	→	y={17}	1	70	1
174	$<p_1, t_1, k_{16}m_9>$	→	y={17}	0.5	16	1
175	$<p_1, t_1, k_{31}m_{12}>$	→	y={17}	0.5	31	1
1728	$<p_2, t_4, k_{71}m_{31}>$	→	y={17}	0.5	71	1
1729	$<p_2, t_4, k_{11}m_9>$	→	y={17}	0.33	11	1
176	$<p_1, t_1, k_5m_4>$	→	y={17}	0.25	5	1
177	$<p_1, t_1, k_4m_3>$	→	y={17}	0.11	4	1
178	$<p_1, t_1, k_2m_2>$	→	y={17}	0.08	2	1
1730	$<p_2, t_4, k_7m_6>$	→	y={17}	0.08	7	1
1731	$<p_2, t_4, k_4m_1>$	→	y={17}	0.07	4	1
1732	$<p_2, t_4, k_9m_4>$	→	y={17}	0.07	9	1
1733	$<p_2, t_4, k_3m_2>$	→	y={17}	0.04	3	1
1734	$<p_2, t_4, k_2m_1>$	→	y={17}	0.03	2	1

Tableau 5.4: Liste des règles simples identifiées pour expliquer y = {36}

ID	Règle			Confiance	Complexité	Contribution
	Mode de production	→	***Mode de perte qualité***			
9	$<p_1, t_2, k_{21}m_7>$	→	y={36}	1	21	1
35	$<p_2, t_4, k_{68}m_{55}>$	→	y={36}	1	68	1
10	$<p_1, t_2, k_{70}m_{46}>$	→	y={36}	0,5	70	1
11	$<p_1, t_2, k_{34}m_{16}>$	→	y={36}	0,5	34	1
12	$<p_1, t_2, k_{33}m_9>$	→	y={36}	0,5	33	1
13	$<p_1, t_2, k_{55}m_{44}>$	→	y={36}	0,5	55	1
36	$<p_2, t_4, k_{86}m_{47}>$	→	y={36}	0,5	86	1
37	$<p_2, t_4, k_{104}m_{88}>$	→	y={36}	0,5	104	1
14	$<p_1, t_2, k_{27}m_5>$	→	y={36}	0,33	27	1
15	$<p_1, t_2, k_{37}m_{27}>$	→	y={36}	0,33	37	1
16	$<p_1, t_2, k_{50}m_{24}>$	→	y={36}	0,33	50	1
38	$<p_2, t_4, k_{19}m_4>$	→	y={36}	0,33	19	1
39	$<p_2, t_4, k_{23}m_{16}>$	→	y={36}	0,33	23	1
17	$<p_1, t_2, k_{10}m_{10}>$	→	y={36}	0,25	10	1
40	$<p_2, t_4, k_{20}m_2>$	→	y={36}	0,25	20	1
41	$<p_2, t_4, k_{21}m_{16}>$	→	y={36}	0,25	21	1
18	$<p_1, t_2, k_{24}m_6>$	→	y={36}	0,2	24	1
42	$<p_2, t_4, k_{16}m_1>$	→	y={36}	0,2	16	1
43	$<p_2, t_4, k_{15}m_{12}>$	→	y={36}	0,17	15	1
44	$<p_2, t_4, k_{12}m_5>$	→	y={36}	0,14	12	1
45	$<p_2, t_4, k_{13}m_3>$	→	y={36}	0,14	13	1
19	$<p_1, t_2, k_5m_3>$	→	y={36}	0,12	5	1
46	$<p_2, t_4, k_{14}m_9>$	→	y={36}	0,12	14	1
20	$<p_1, t_2, k_4m_4>$	→	y={36}	0,11	4	1
47	$<p_2, t_4, k_{11}m_6>$	→	y={36}	0,09	11	1
21	$<p_1, t_2, k_2m_1>$	→	y={36}	0,08	2	1
48	$<p_2, t_4, k_7m_6>$	→	y={36}	0,08	7	1
49	$<p_2, t_4, k_4m_1>$	→	y={36}	0,07	4	1
50	$<p_2, t_4, k_9m_4>$	→	y={36}	0,07	9	1
51	$<p_2, t_4, k_3m_2>$	→	y={36}	0,04	3	1
52	$<p_2, t_4, k_2m_1>$	→	y={36}	0,03	2	1

Une première conclusion, qu'on peut déduire de ces premières règles identifiées, est que les équipements potentiellement problématiques sont *t1*, *t2* et/ou *t4*, car aucune règle identifiée, n'associe un mode de production de l'équipement *t3* comme une cause potentiellement explicative d'un des trois modes de perte de qualité.

5.1.2.2.4 Sélection des règles les plus pertinentes

Pour chacun des modes de perte de qualité $y \in \{ \{5\}, \{17\}, \{36\}\}$, on commence par filtrer les règles correspondantes selon un front de Pareto. L'indicateur de *contribution* pour toutes les règles identifiées est à sa valeur maximale de 1. Le problème d'optimisation à trois critères (*confiance*, *contribution* et *complexité*) revient ainsi à un problème à deux critères:

$$\begin{cases} \max(\text{confiance}) \\ \max(-\text{complexité}) \end{cases}$$

Les règles sélectionnées sont représentées dans *Figure 5.5* par des points verts associés à leurs identifiants. Ainsi, on a réduit l'ensemble de règles de départ à *2, 4* et *7* règles, respectivement pour expliquer *y={5}, {17}, et {36}*, résumées dans le Tableau 5.5 et le Tableau 5.6.

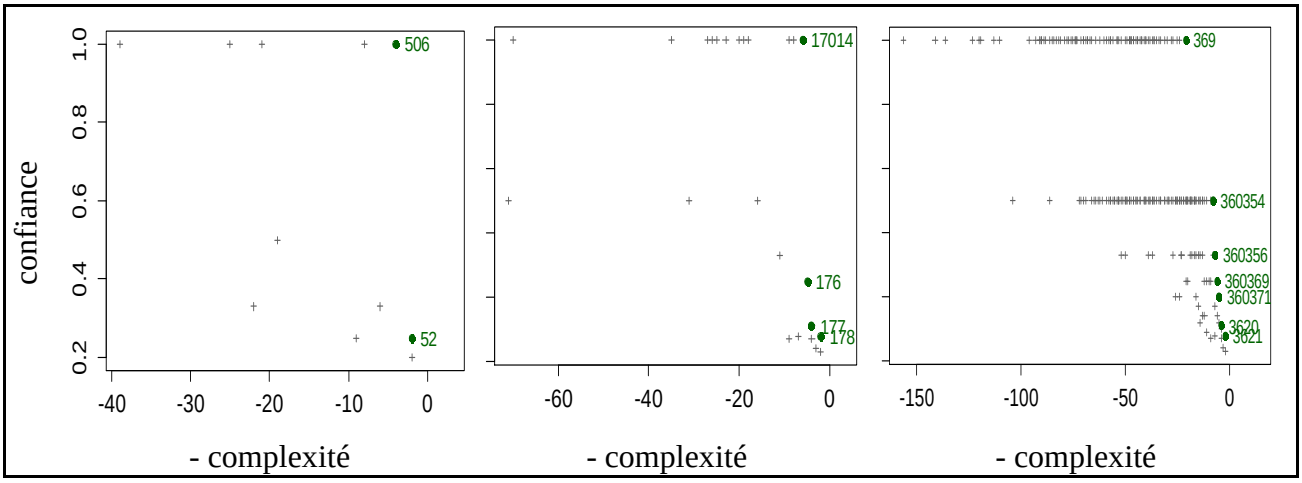


Figure 5.5: Sélection des règles pertinentes par front de Pareto (y= 5, 17 et 36)

Tableau 5.5: L'ensemble final de règles simples expliquant les modes de perte qualité y = {5}, {17} et {36}

ID	Règle			Confiance	Complexité	Contribution
	Mode de production	→	***Mode de perte qualité***			
52	$<p_1, t_1, k_2m_1>$	→	y={5}	0.25	2	1
176	$<p_1, t_1, k_5m_4>$	→	y={17}	0.25	5	1
177	$<p_1, t_1, k_4m_3>$	→	y={17}	0.11	4	1
178	$<p_1, t_1, k_2m_2>$	→	y={17}	0.8	2	1
369	$<p_1, t_2, k_{21}m_7>$	→	y={36}	1	21	1
3620	$<p_1, t_2, k_4m_4>$	→	y={36}	0.11	4	1
3621	$<p_1, t_2, k_2m_1>$	→	y={36}	0.08	2	1

Tableau 5.6: L'ensemble final de règles de trajectoire expliquant les modes de perte qualité y = {5}, {17} et {36}

ID	Règle					Confiance	Complexité	Contribution
	Mode de production sur p1	***&***	***Mode de production sur p2***	**→**	***Mode de perte qualité***			
506	$<p_1, t_1, k_2m_1>$	&	$<p_2, t_4, k_2m_2>$	→	y={5}	1	4	1
17014	$<p_1, t_1, k_2m_2>$	&	$<p_2, t_4, k_4m_1>$	→	y={17}	1	6	1
360354	$<p_1, t_2, k_4m_4>$	&	$<p_2, t_4, k_4m_1>$	→	y={36}	0.5	8	1
360356	$<p_1, t_2, k_4m_4>$	&	$<p, t_4, k_3m_2>$	→	y={36}	0.33	7	1
360369	$<p_1, t_2, k_2m_1>$	&	$<p_2, t_4, k_4m_1>$	→	y={36}	0.25	6	1
360371	$<p_1, t_2, k_2m_1>$	&	$<p_2, t_4, k_3m_2>$	→	y={36}	0.2	5	1

En plus de cette sélection par front de Pareto, les ingénieurs métiers ont défini un seuil minimal de confiance à *0.5*. Les règles respectant ce seuil sont résumées dans le Tableau 5.7**.**

Tableau 5.7: L'ensemble final de règles sélectionnées selon le front de Pareto et un seuil minimal de confiance à 0.5

ID	Règle					Confiance	Complexité	Contribution
	Modes de production problématiques			**→**	***Mode de perte qualité***			
506	$<p_1, t_1, k_2m_1>$	&	$<p_2, t_4, k_2m_2>$	→	y={5}	1	4	1
17014	$<p_1, t_1, k_2m_2>$	&	$<p_2, t_4, k_4m_1>$	→	y={17}	1	6	1
369	$<p_1, t_2, k_{21}m_7>$			→	y={36}	1	21	1
360354	$<p_1, t_2, k_4m_4>$	&	$<p_2, t_4, k_4m_1>$	→	y={36}	0.5	8	1

Comme représenté dans la *Figure 5.6*, les règles sélectionnées décrivent des trajectoires problématiques dans le segment *s*. D'un côté, on peut en conclure, que la méthode d'analyse proposée réussit à expliquer les pertes de qualité relatives à un mode de production spécifique à un équipement de production, comme définit par la règle 369. Cette règle propose une explication de la perte de qualité *y = {36}* à travers le mode de fonctionnement identifié par le *groupe 7* identifié à travers la partition avec *k = 21* sur les données de l'équipement *t2*, pour la première étape de production *p1*.

D'un autre côté, l'approche proposée réussit aussi à expliquer d'autres modes de perte de qualité relatifs à une combinaison de différents modes de production sur des équipements différents, comme identifiées par les règles *506* et *17014.*

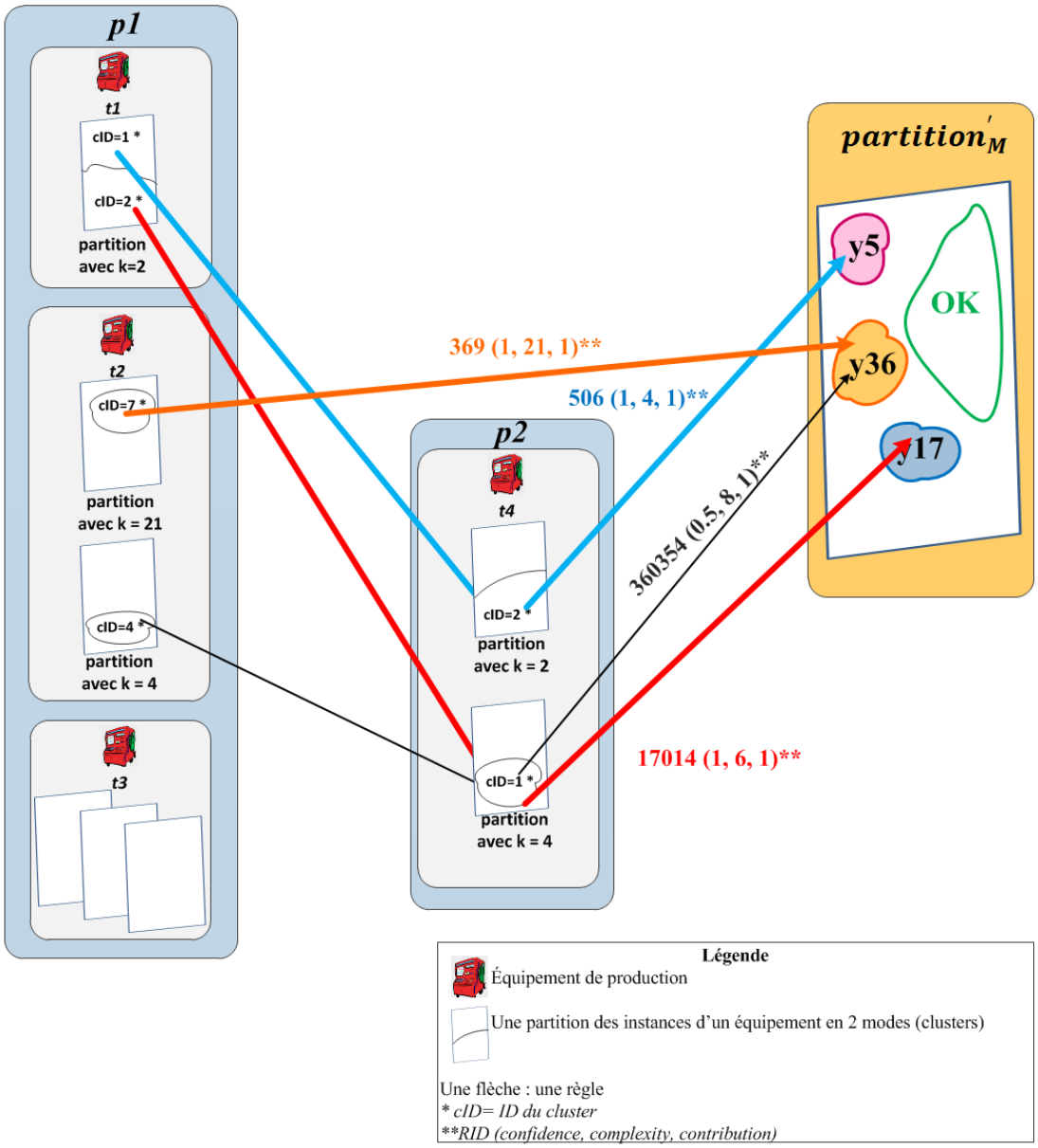


Figure 5.6: Une représentation graphique des règles sélectionnées

5.1.2.2.5 Transformation des règles pertinentes

Après les étapes d'identification et de sélection de règles, quatre règles sont retenues pour expliquer la perte de contrôle analysée. La dernière étape de fouille de données consiste à transformer ces "*règles discrètes*" en "*règles continues*", afin de donner une signification physique aux modes de production, identifiés comme problématiques.

Comme décrit dans la *section 4.6*, l'induction d'arbre de décision est utilisée pour traduire les identifiants de modes de production potentiellement problématiques par des intervalles de valeurs sur les paramètres de production. Ainsi, chaque mode de production

$x \in X$ est expliqué par les paramètres de production qui caractérisent le mieux les conditions de production des plaques appartenant à ce mode *x*.

À travers les 4 règles sélectionnées selon le front de Pareto et le seuil de confiance de 0.5, représenté dans la Figure 5.6 et le Tableau 5.7, six modes de production sont identifiés comme potentiellement problématiques :

- Pour l'étape *p1*, sur l'équipement de production *« t1 »* :
  - le *mode* de production *numéro 1* identifié par la partition à *k=2*
  - le *mode* de production *numéro 2* identifié par la partition à *k=2*
- Pour l'étape *p1*, sur l'équipement de production *« t2 »* :
  - le *mode* de production *numéro 7* identifié par la partition à *k=21*
  - le *mode* de production *numéro 4* identifié par la partition à *k=4*
- Pour l'étape *p2*, sur l'équipement de production *« t4 »* :
  - le *mode* de production *numéro 2* identifié par la partition à *k=2*
  - le *mode* de production *numéro 4* identifié par la partition à *k=1*

On s'intéresse, donc, à donner une explication physique à chacun de ces six modes de production. Comme décrit précédemment, on garde les trois premières interprétations, *x'1, x'2* et *x'3*, pour chaque mode problématique *x*, à travers l'induction d'arbre de décision avec l'algorithme *CART*.

Les interprétations des modes de production sur l'équipement « t1 » sont données dans les Tableau 5.8 et

Tableau 5.9. La première interprétation *x'1= Mean_Flow_MFC-AR-500_offset >= 0.75* du mode problématique *« <p1, t1, k2m1> »* représente un écart par rapport à la moyenne définie sur le flux du gaz *MFC-AR-500* supérieur ou égal à 0.75. Cette traduction est intéressante pour l'ingénieur et permet de caractériser un mode de production problématique, puisque, idéalement, cet écart doit être de 0.

Pour le mode de production problématique « *p1, t1, k2m2* », la première interprétation *x'1= Mean_Flow_MFC-AR-500_offset < 0.75* n'est pas intéressante puisqu'elle représente un écart par rapport à la moyenne définie sur le flux du gaz *MFC-AR-500* inférieur à 0.75. Selon l'expertise métier, l'intervalle de valeur de ce paramètre est normal. Par contre, la deuxième interprétation *x'2= Mean_Flow_MFC-SO2-200_offset < -2.58* est intéressante puisqu'elle décrit un écart éloigné du *0* ciblé, et ainsi, la cause de la perte de qualité qui en a découlé. On voit, ainsi, l'intérêt de donner les trois premières interprétations d'un mode de perte de qualité.

Tableau 5.8: Les trois interprétations obtenues pour expliquer le mode de production problématique "< p_1, t_1, k_2m_1>"

Mode de production problématique / Interprétation par induction d'arbre de décision	<p_1, t_1, k_2m_1>	Avec équilibrage des classes
x'_1	Mean_Flow_MFC-AR-500_offset >= 0.75	Non
x'_2	Mean_Flow_MFC-SO2-200_offset >= -2.58	Non
x'_3	TimeSincePM < 236.22	Non

Pour le deuxième mode de production problématique sur *« t1 »*, la première interprétation *x'1= Mean_Flow_MFC-AR-500_offset < 0.75* est moins intéressante puisqu'elle représente un écart par rapport à la moyenne définie sur le flux du gaz *MFC-AR-500* inférieur à 0.75, qui contient la zone de bon fonctionnement = 0. Par ailleurs, la deuxième interprétation *x'2= Mean_Flow_MFC-SO2-200_offset < -2.58* est intéressante puisqu'elle décrit un écart éloigné du *0* ciblé. On voit, ainsi, l'intérêt de donner les trois premières interprétations d'un mode de perte de qualité.

Tableau 5.9: Les trois interprétations obtenues pour expliquer le mode de production problématique "<p_1, t_1, k_2m_2> »

Mode de production problématique / Interprétation par induction d'arbre de décision	<p_1, t_1, k_2m_2>	Avec équilibrage des classes
x'_1	Mean_Flow_MFC-AR-500_offset < 0.75	Non
x'_2	Mean_Flow_MFC-SO2-200_offset < -2.58	Non
x'_3	TimeSincePM >= 236.22	Non

De même, les modes de production problématiques relatifs aux équipements *« t2 »* et *« t4 »* sont caractérisés par les chemins issus des trois premiers arbres de décision induits. À partir de ces traductions, les *règles continues* sont obtenues en remplaçant chaque mode de production problématique par ces trois interprétations obtenues. Toutes les combinaisons possibles entre les interprétations obtenues pour les modes problématiques constituant une règle, définissent l'ensemble des nouvelles règles continues.

Les indicateurs de qualité, à savoir la *confiance*, la *complexité* et la *contribution* sont calculés pour chaque nouvelle règle. Comme expliqué dans le précédent chapitre, cette transformation peut altérer les indicateurs de *confiance* et de *contribution*. Ainsi, seules les règles respectant le seuil minimal de confiance de 0.5, définit par les ingénieurs, et un indicateur de contribution = 1 sont gardées.

Tableau 5.10: Transformation de la règle 369 en règles à base d'arbre de décision

RID	Cause de production problématique	→ Mode de perte qualité	Confiance	Complexité	Contribution
369.1	t2 (Mean_Flow_HBR-100_offset >= -0.29 & TimeSincePM < 261.97)	→ y = {36}	1	2	1
369.2	t2 (Mean_Flow_SO2-200_offset >= -0.63)	→ y = {36}	0	1	0
369.3	t2 (Stage11_Mean_Position_C5cap >= 661.36 & Mean_Flow_O2-20_offset < -0.09 & Range_Temperature_TCP < 4.45)	→ y = {36}	0.5	3	1

Prenons l'exemple de la règle simple numéro *369*, sa transformation a donné trois nouvelles règles décrites dans *Tableau 5.10*. On remarque que la deuxième et troisième règle ont détérioré les indicateurs de *confiance* et de *contribution*. D'un côté, la nouvelle règle *369.2* n'est plus intéressante puisqu'elle ne respecte pas les critères de sélection requis (*confiance >=0.5* ; *contribution =1*). D'un autre côté, la règle *369.3* est retenue, avec la règle *369.1*, car même si elle détériore l'indicateur de *confiance*, elle respecte le seuil défini.

De même, les autres règles présentes dans le Tableau 5.5 et le Tableau 5.6, sont transformées à l'aide des traductions des modes afin de construire l'ensemble des règles expliquant la perte de qualité analysée, avec ses trois sous phénomènes *y={5}, y={17}* et *y={36}*.

5.1.2.3 Étape 3: Intégration des connaissances identifiées

L'ensemble des règles sélectionnées a été transformé et présenté aux ingénieurs afin de valider et explorer cette connaissance identifiée. Les résultats de l'entretien de validation avec les ingénieurs sont décrits comme suit :

Premièrement, à partir des règles expliquant la perte de qualité *y={36}*, les ingénieurs ont pu en déduire que cette perte de qualité est dû à une étape de production précédente au segment analysé, car les paramètres de production potentiellement problématiques caractérisent, soit à un mode de fonctionnement correct, soit un module de l'équipement qui s'adapte selon la qualité de la plaque en entrée. En effet, la règle *369.1* identifie le paramètre « Mean_Flow_HBR-100_offset » qui représente un écart par rapport la consigne assez faible (>= -0.29), qui ne peut, donc, pas caractériser un mode de fonctionnement problématique. La

règle *369.3* quant à elle, caractérise ce mode de fonctionnement potentiellement problématique par un fonctionnement de l'équipement *« t2 »* spécifique, relatif au module de régulation, qui dépend de la qualité de la plaque en entrée. Ainsi, ces deux règles, expliquant le sous phénomène *y=36,* ont permis au ingénieur, de les orienter à explorer les étapes de production précédentes pour expliquer cette perte de qualité détectée au niveau de ce segment, mais relative à une ou plusieurs étapes précédentes. Ainsi, on peut réappliquer l'approche proposée, en considérant cette fois, un nouvel ensemble d'étape de production *P*, incluant les étapes précédentes à celles étudiées.

Par ailleurs, les règles expliquant le sous phénomène *y={17}* sont potentiellement intéressantes. Par exemple, une règle identifiée explique cette perte de qualité par une maintenance éloignée de l'équipement *« t1 »*, associée à un mode de fonctionnement sur *« t4 »* caractérisé par une pression très variable durant le nettoyage des plaques (étape p2).

Finalement, les règles expliquant le sous phénomène *y={5}*, peuvent être exploitées afin de prévenir des pertes de qualité similaires. Par exemple, une des règles extraites identifie un écart par rapport à la consigne du flux du gaz AR-500 > 0.75 sur l'équipement *« t1 »* associé à une variance importante du flux du gaz N2-1000 (écart type >= 12.87) sur l'équipement *« t4 »*.

Pour conclure, pour ce cas d'étude, l'approche proposée a réussi à identifier des connaissances intéressantes, utiles et utilisables pour expliquer des cas de perte de qualité locale. En plus, d'identifier les modes de production problématiques, l'approche a permis d'orienter les ingénieurs à des étapes de production antérieures pour expliquer une perte de qualité identifiée localement au niveau du segment analysé *s*, mais qui est relative à une ou plusieurs étapes précédentes.

5.1.3 Application d'une approche classique d'induction d'arbre de décision

On s'intéresse à comparer les performances de l'approche proposée *CLARIF* avec une approche supervisée classique. Pour cela, nous commençons par poser la problématique d'identification de causes de perte de qualité d'un point d'un point de vue supervisé, dans la première sous-section. Par la suite, dans, nous donnons une justification du choix de l'algorithme d'induction d'arbre de décision pour cette tâche

5.1.3.1 Description de la problématique d'un point de vue « supervisé »

Pour expliquer une perte de qualité à travers une analyse supervisée, on formalise les données comme décrit dans le Tableau 5.11. Chaque instance des données peut être vue comme un couple (*observation, étiquette*), avec d'un côté, la description d'une situation à travers des variables explicatives, et d'un autre côté le label de la classe résultante qui est

fourni par l'expert. Dans notre cas, le label d'une plaque est soit (*OK*) si elle est de bonne qualité soit (*OOC*), si elle est de mauvaise qualité.

Tableau 5.11 : Description des données en entrée pour un algorithme d'apprentissage supervisé

Instance Id	***var1***	***var2***	**…**	**label**
1				
2				
…				

Classiquement, un algorithme de fouille de données pour l'apprentissage supervisé cherche à estimer le label de nouvelles observations à partir des données d'apprentissage, *i.e.* comme un outil de classification. Dans notre cas, nous allons utiliser un algorithme d'apprentissage supervisé dans un objectif, non pas de prédiction de tendances futures, mais afin d'extraire des motifs qui décrivent la classe des plaques problématiques, à savoir (*OOC*). Ainsi, comme un outil d'exploration des données.

Ainsi, en appliquant un algorithme d'apprentissage supervisé, dont le choix sera justifié dans la sous-section suivante, on cherche à identifier, à partir des variables explicatives (*var1, var2, ... du* Tableau 5.11) un ou plusieurs paramètres responsables du phénomène de perte de qualité.

5.1.3.2 Justification sur le choix de l'algorithme

Parmi les algorithmes d'apprentissage supervisé, notre étude de l'art (cf. chapitre 2) a montré que les algorithmes d'induction d'arbre de décision sont parmi les plus utilisés pour identifier les sources de perte de qualité. Ce succès est dû à leur facilité d'implémentation et d'interprétation.

Ainsi, nous avons choisi d'utiliser un algorithme d'induction d'arbre de décision.

Par ailleurs, les données disponibles pour l'analyse sont un mélange de variables catégoriques et continues. On choisit, donc, l'algorithme *CART* [79], à travers la méthode *rpart*, son implémentation sous *R*, puisqu'il permet l'analyse des deux types de variables.

5.1.3.3 Description des résultats obtenus

Dans un contexte d'apprentissage supervisé, toutes les plaques doivent avoir le label correspondant. Ainsi, dans notre cas d'étude, seules les données correspondantes aux 44 plaques mesurées seront analysées. Rappelons que, pour le cas analysé, ces plaques ont suivis trois chemins de production différents (cf. Figure 5.1). Un premier groupe de plaques ont été

traitées par l'équipement *t1*, à l'étape de production *p1*, puis par *t4* à l'étape *p2*. Un deuxième groupe de plaques ont été traitées par *t2*, puis *t4*. Finalement, le troisième groupe de plaques ont été traitées par *t3* puis *t4*. Ainsi, pour chaque chemin, on construit un fichier de données qui sera analysée plus tard séparément aux autres.

Ainsi, pour chaque chemin, on construit un fichier de données, pour obtenir trois fichiers D^{t1t4}, D^{t2t4} et D^{t3t4}. Chaque fichier contient (1) les données décrivant le fonctionnement de l'équipement correspondant durant la première étape *p1*, (2) suivies par les données décrivant le fonctionnement de l'équipement correspondant durant la deuxième étape *p2* et finalement (3) le label de la qualité de la plaque, et ce pour toutes les plaques qui ont suivi ce chemin de production. Par exemple, le fichier D^{t1t4} stocke les données des équipements *t1*, et celles de *t4*, en plus du label qualité, pour toutes les plaques W^{t1t4} qui ont suivi le chemin *t1_t4*.

Sur chaque fichier, on applique l'algorithme d'induction d'arbre *de deux façons* : *(1)* la première sur les données brutes sans pondération et *(2)* la deuxième en donnant une pondération aux plaques afin d'équilibrer les classes et éviter ainsi le problème des classes minoritaires. On garde les trois premiers arbres de décision, si possible. Quand un arbre de décision peut être induit pour expliquer le label *OOC*, nous calculons, comme c'était le cas avec *CLARIF*, les indicateurs de *confiance*, de *contribution* et de *complexité* des règles de décision qui en écoule. Les résultats sont présentés dans le *Tableau 5.12*.

À travers les résultats décrit dans le *Tableau 5.12*, on peut voir que si on applique la méthode sur les données, sans équilibrer les classes, on n'obtient aucun arbre de décision sur aucun chemin de production. Ceci peut être expliqué par le nombre réduit de plaques problématiques par rapport au nombre total de plaques analysées.

Pour remédier à ce problème, et comme nous l'avons proposé pour la méthode *CLARIF*, nous proposons de rééquilibrer les classes à travers une pondération des exemples. Les résultats de l'induction d'arbres de décision sur les données pondérées sont disponibles dans la deuxième colonne du *Tableau 5.12*. Malgré ce traitement, seule la cause potentielle d'une plaque des trois problématiques est identifiée à travers l'analyse des données du chemin de production, *t2*, *t4*. La cause identifiée n'est pas utilisable puisqu'elle présente un faible degré de confiance, à 14%, et ne peut être utilisée comme une connaissance.

Tableau 5.12: Identification des sources de perte de qualité à travers une induction d'arbres de décision

	Cause			confiance	contribution	complexité
	Données non pondérées	**Données pondérées**				
D^{t1t4}	Ø	Ø		-	-	-
D^{t2t4}	Ø	1ére interprétation	t2_Duration.AllSteps. < 335.21	0.14	0.33	1
		2éme interprétation	t2_L.Clamped.Mean.Pressure_HeBackside < 19.94	0.14	0.33	1
		3éme interprétation	t2_L.Clamped.Mean.Voltage_ESC_Clamp < 3616.25	0.14	0.33	1
D^{t3t4}	Ø	Ø		-	-	-

5.1.4 Conclusion

Cette étude comparative des résultats d'application de la méthode proposée *CLARIF* à ceux d'une méthode classique par induction d'arbre de décision permet de voir à quel point il est difficile, pour l'approche classique, de séparer les 3 plaques *OOC* des plaques bonnes. L'explication du phénomène de perte de qualité par induction d'arbre de décision n'a pas permis l'identification d'une cause utilisable, puisque les explications identifiées ont un faible degré de confiance, et n'explique qu'une seule plaque sur les trois à expliquer.

D'un autre côté, la méthode *CLARIF* proposée, a permis de proposer des explications aux plaques problématiques et ce avec des degrés de *confiance*, *complexité* et *contribution* intéressant. Permettant, ainsi de donner des explications aux ingénieurs afin de mieux comprendre ces nouveaux cas de perte de qualité, et d'intégrer cette connaissance au système de production.

5.2 ÉTUDE DU CAS *SECOM*

Le deuxième cas d'étude est issu du répertoire d'apprentissage automatique *UCI Machine Learning Repository* disponible ici [86], qui groupe 325 fichiers d'analyse. Nous avons choisi d'analyser le fichier nommé *SECOM* [87], pour deux raisons principales : (1) ce dataset correspond à données issues d'un système de production en industrie du semi-conducteur. (2) La tâche de fouille de données correspondante concerne la découverte de causalité.

L'objectif des expérimentations qui vont être décrites est de tester les performances de *CLARIF*, ainsi que de les comparer à celles d'une approche classique par induction d'arbres de décision sur un jeu de données indépendant.

Dans la section suivante, nous décrivons les deux expérimentations proposées pour ce cas d'étude. Par la suite, nous analysons ce cas avec la méthode *CLARIF* et une approche classique par induction d'arbre de décision.

5.2.1 Construction des différents fichiers d'analyse

Les données d'analyse relatives à ce cas d'étude sont stockées en deux fichiers séparés : le premier contient les 591 variables descriptives des 1567 instances, *i.e.* plaques. Le deuxième fichier contient l'heure de contrôle et la classe de qualité des plaques, *i.e. (-1)* si la plaque est de bonne qualité (*OK*) et *(1)* si celle-ci est de mauvaise qualité (*OOC*). Parmi les 1567 plaques disponibles, 104 plaques sont problématiques, et on cherche donc à remonter à leurs causes. Notons que puisqu'il s'agit de données issues d'un système de production réel, certaines valeurs sont manquantes et les valeurs sont bruitées.

Par ailleurs comme, contrairement au cas *STMicroelectronics*, nous ne disposons pas d'expert métier, nous devons proposer une autre méthode de validation des résultats obtenus. Pour cela, nous décomposons les données disponibles en deux parties : (1) une partie relative aux données d'apprentissage, notée D^A, et (2) une partie relative aux données de test, notée D^T. Cette décomposition est réalisée à travers un échantillonnage aléatoire sans remise en tirant séparément dans les instances problématiques, *OOC*, et de bonnes qualité, *OK*.

Nous proposons deux expériences en variant les données étudiées. Le tableau ci-dessous présente la répartition des instances problématiques *OOC* et de bonne qualité *OK* parmi les données d'apprentissage et les données de test. Pour la première expérience, nous avons sélectionné aléatoirement 75% des individus pour les données d'apprentissage et les 25% restant pour les données de test.

Pour la deuxième expérimentation, nous avons généré une base d'apprentissage composée à égalité de plaques *OOC* et de plaques *OK*, tout en réduisant la taille de la base d'apprentissage à 104 plaques (52 plaques *OOC* plus 52 plaques *OK*). De même, nous avons construit la base de test pour contenir à égalité des instances *OOC* et des instances *OK* et que sa taille représente 15% de la base d'apprentissage. Ainsi, 8 plaques ont été tirées aléatoirement dans les instances *OOC* restantes, de même pour les plaques *OK*.

Tableau 5.13 Description des deux expérimentations réalisées sur le cas SECOM

	Données d'Apprentissage			**Données de Test**		
	OOC	**OK**	**Total**	**OOC**	**OK**	**Total**
Expérimentation 1	75	1100	1175	29	363	392
Expérimentation 2	52	52	104	8	8	16

5.2.2 Description de la démarche d'analyse

5.2.2.1 La préparation des données

Comme énoncé précédemment, les données issues d'un système réel nécessitent un prétraitement de nettoyage avant d'appliquer la méthode de fouille de données. Pour cela, nous commençons par supprimer les variables pour lesquelles il y a des valeurs manquantes. Restent alors 52 variables parmi les 591 initiales à analyser plus la variable contenant le label qualité finale des plaques.

5.2.2.2 La fouille des données

Pour appliquer la méthode *CLARIF*, nous devons formaliser le problème sous la forme d'une problématique d'explication d'un phénomène *Y* détecté à des étapes effets *EE* à travers des conditions particulières sur des étapes causes *EC*. Pour ce cas d'étude, le phénomène à expliquer est celui des 104 plaques problématiques, *i.e.* la classe labélisée *(1)*. Nous avons besoin de définir la décomposition en étapes causes *EC*. Nous choisissons de grouper les variables explicatives en une seule étape cause. Ainsi, *EC* est composée d'une seule étape *ec* qui est décrite par les *52* variables qui restent à la suite de l'étape de prétraitement.

$EC = ec1$ dont les données D^{ec1} sont de taille *1175*52*

Par ailleurs, nous ne disposons pas des mesures du contrôle qualité qui ont conduit à la labélisation des plaques en bonne *(-1)* ou problématique *(1)*. Ainsi, nous ne pouvons pas définir les étapes *EE*. À défaut de disposer des données relatives aux étapes *EE*, *CLARIF* reste applicable, en ignorant sa première étape d'enrichissement de la partition de qualité basée sur l'expertise à travers les données des étapes *EE*. Ainsi, *CLARIF* est appliqué ici avec $Y = \{instances\ OOC\}$ et $\bar{Y} = l'ensemble\ complémentaire\ de\ Y$.

Pour l'approche classique par induction d'arbres de décision, et comme décrit pour le cas d'étude précédent (cf. section 5.1.3), nous appliquons la méthode *rpart* disponible sous *R*. Notons qu'indépendamment de la méthode de fouille de données utilisée, celle-ci sera appliquée, en premier lieu, sur les données d'apprentissage uniquement, D^A, pour l'identification des règles dans le cas de *CLARIF* et la construction du modèle pour l'induction d'arbre de décision.

5.2.2.3 La validation des résultats obtenus

Finalement, afin d'évaluer les résultats obtenus à travers les deux méthodes étudiées, nous construisons la table de contingence. Ainsi, nous allons tester les règles identifiées grâce à *CLARIF*, en premier lieu sur les données d'apprentissage D^A, puis sur les données de test D^T. De même, afin d'évaluer l'approche traditionnelle par induction d'arbre de décision sur ce cas d'étude, nous allons tester les règles de décision construites grâce à l'approche classique par induction d'arbre de décision sur les données d'apprentissage D^A et sur les données test D^T.

Pour finir, nous allons comparer les performances des deux méthodes sur les deux jeux de données.

5.2.3 Résultats expérimentaux

5.2.3.1 Expérimentation 1 :

L'étape de génération non supervisée des modes descriptifs candidats pour l'unique étape *ec1* a permis de transformer le fichier D^{ec1} (1175*52) en D'^{ec1} (1175 *586). À partir des modes générés, l'identification de règles explicatives sur le fichier *T* composé de 1175 lignes et (587 +1) colonnes, en fixant un seuil de contribution minimal à 1.3% (au moins une plaque problématique expliquée par les règles à identifier), et un seuil de confiance minimal à 0.75, a permis la génération de *14357 règles* avec des indicateurs de *confiance*, *contribution* et de *complexité* variant respectivement de 0.5 à 1, de 0.01 à 0.92 et de 14 à 587. Parmi ces 14357 règles, la sélection par front de Pareto, illustrée dans la Figure 5.7, a permis de trouver *une règle* Pareto-optimale, décrite ci-dessous.

Tableau 5.14 : Description de la règle Pareto-optimale

RID	rules	confidence	complexity	Contribution
15529	$< ec_1, k_{14}m_{11} > \rightarrow y= \{1\}$	1	14	0.92

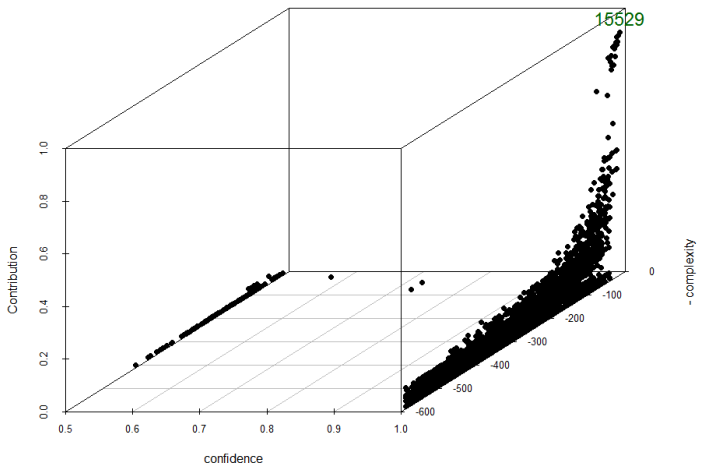
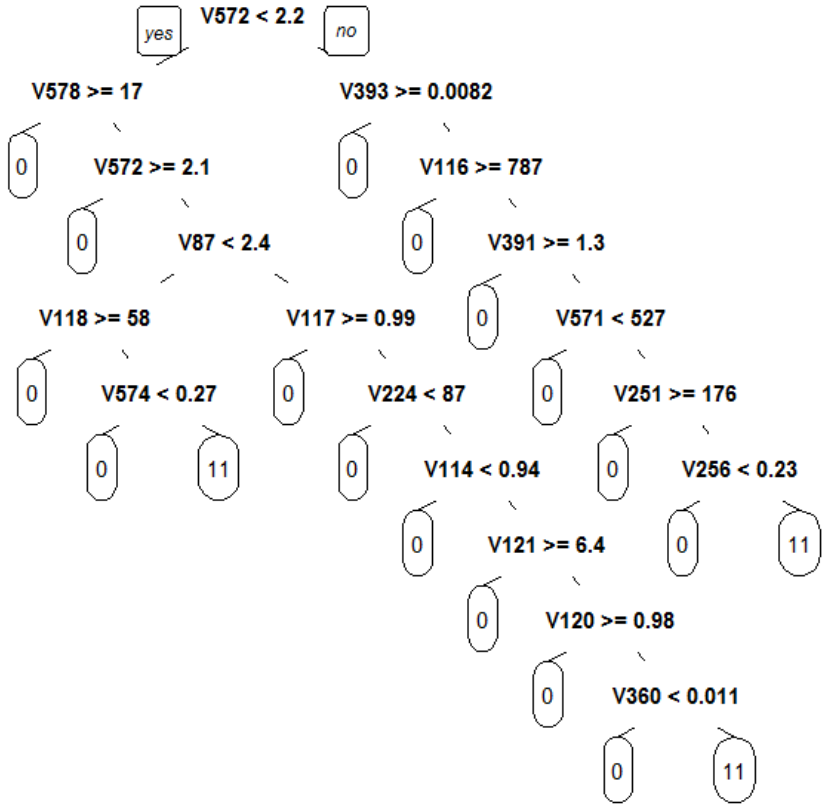


Figure 5.7: Représentation de la sélection de règles par front de Pareto pour l'expérimentation 1

La dernière étape de *CLARIF* consiste à décrire le mode cause identifié, $<ec_1, k_{14}m_{11}>$, par des intervalles de valeurs sur les paramètres de l'espace d'entrée du fichier D^{ec1}. Cette traduction est donnée dans le Tableau 5.15, ainsi que la table de contingence obtenue pour les données D^A et D^T. Pour finir, les mesures de performance, *i.e.* les valeurs pour les trois indicateurs de qualité sont données.

Tableau 5.15 : Synthèse des résultats de CLARIF pour l'expérimentation 1

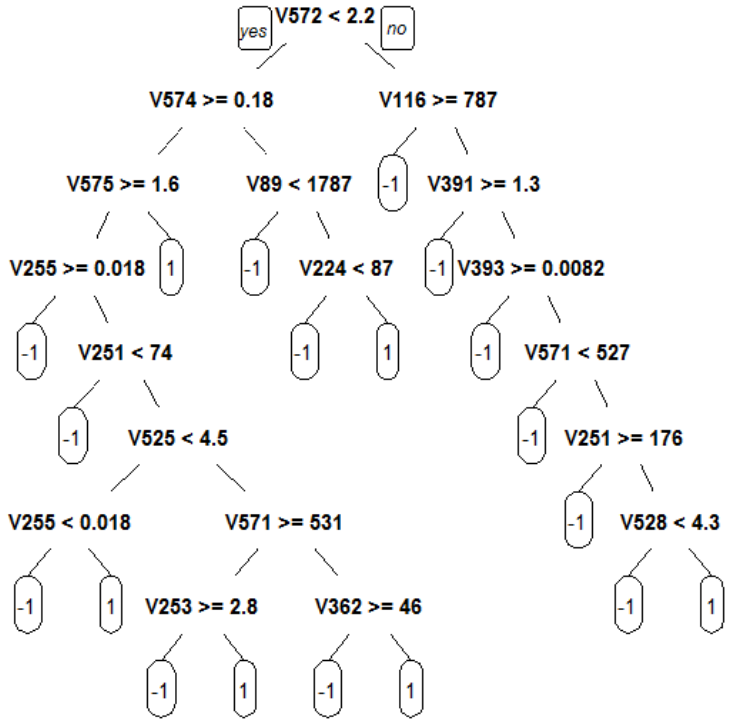
<table>
<tr><td>La transformation du mode problématique identifié
< ec_1, $k_{14}m_{11}$ >

NB : avec équilibrage de clusters</td><td colspan="2"> 
</td></tr>
<tr><td rowspan="2">Table de contingence</td><td>sur D^A</td><td><table>
<tr><th></th><th>0 (^-1)</th><th>11 (^1)</th></tr>
<tr><td>-1</td><td>841</td><td>259</td></tr>
<tr><td>1</td><td>6</td><td>69</td></tr>
</table></td></tr>
<tr><td>sur D^T</td><td><table>
<tr><th></th><th>0 (^-1)</th><th>11 (^1)</th></tr>
<tr><td>-1</td><td>247</td><td>116</td></tr>
<tr><td>1</td><td>19</td><td>10</td></tr>
</table></td></tr>
<tr><td>Mesures de performance pour expliquer les produits problématiques</td><td colspan="2"><table>
<tr><th></th><th>confiance</th><th>contribution</th><th>complexité</th></tr>
<tr><td>D^A</td><td>0.21</td><td>0.92</td><td rowspan="2">10</td></tr>
<tr><td>D^T</td><td>0.08</td><td>0.34</td></tr>
</table></td></tr>
</table>

À partir du Tableau 5.15, nous remarquons que la traduction du mode problématique identifié a fortement détériorée la qualité de la cause identifiée, puisque la confiance est passée de 100% à 21% sur les données d'apprentissage. Ceci implique que l'induction d'arbre de décision ne permet pas de décrire ce groupement de produits, et que ce groupement est difficile à expliquer.

En parallèle à l'application de *CLARIF*, nous avons testé l'approche classique par induction directe d'arbre de décision. Le résultat est donné dans le Tableau 5.16. La qualité de l'arbre de décision induit à partir des données est faible. Pour les données d'apprentissage, sur lesquelles on espère avoir un haut niveau de qualité, l'arbre de décision ne permet que 22% de confiance. Sur les données de test, la confiance est de 9% avec un taux de contribution à l'explication de 34%. Notons que la complexité est aussi importante, égale à 8, le nombre maximal de nœuds pour expliquer la classe problématique *(-1)*.

Tableau 5.16 : Synthèse des résultats de l'approche classique pour l'expérimentation 1

<table>
<tr><td colspan="2">L'arbre de décision induit</td><td> 
</td></tr>
<tr><td rowspan="2">Table de contingence</td><td>sur D^A</td><td><table>
<tr><th></th><th>^-1</th><th>^1</th></tr>
<tr><td>-1</td><td>850</td><td>250</td></tr>
<tr><td>1</td><td>4</td><td>71</td></tr>
</table></td></tr>
<tr><td>sur D^T</td><td><table>
<tr><th></th><th>^-1</th><th>^1</th></tr>
<tr><td>-1</td><td>263</td><td>100</td></tr>
<tr><td>1</td><td>19</td><td>10</td></tr>
</table></td></tr>
<tr><td colspan="2">Mesures de performance pour expliquer les produits problématiques</td><td><table>
<tr><th></th><th>confiance</th><th>contribution</th><th>complexité</th></tr>
<tr><td>D^A</td><td>0.22</td><td>0.94</td><td rowspan="2">8</td></tr>
<tr><td>D^T</td><td>0.09</td><td>0.34</td></tr>
</table></td></tr>
</table>

Pour cette première expérimentation, les performances des deux méthodes sont très proches, mais ne sont pas suffisantes. Ceci peut être expliqué par la présence de bruits dans les données sélectionnées pour construire les fichiers d'analyse. Lors de la deuxième

expérimentation, nous allons réduire la taille des échantillons sélectionnés afin de potentiellement réduire le risque de sélection de bruit.

5.2.3.2 Expérimentation 2 :

Nous appliquons maintenant *CLARIF* aux données de l'expérimentation 2. À partir des modes générés, l'identification de règles explicatives, en fixant un seuil de contribution minimal à 1.3% (au moins une plaque problématique), et un seuil de confiance minimal à 0.75, a permis la génération de *500 règles* avec des indicateurs de *confiance*, *contribution* et de *complexité* variant respectivement de 0.5 à 1, de 0.02 à 0.42 et de 2 à 52.

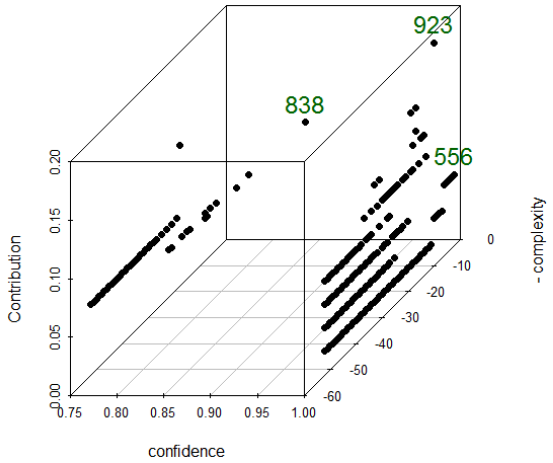


Figure 5.8 : Représentation de la sélection de règles par front de Pareto pour l'expérimentation 2

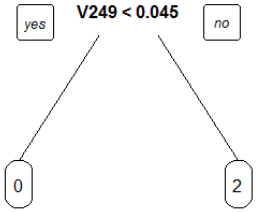
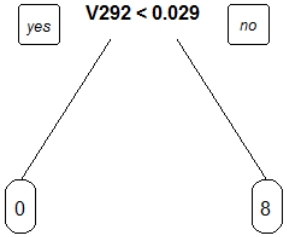
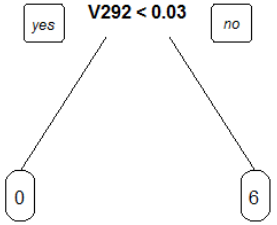
Parmi ces 500 règles, la sélection par front de Pareto, illustrée dans la Figure 5.8, a permis de distinguer trois règles Pareto-optimales. Elles sont décrites dans le Tableau 5.17.

Tableau 5.17 : Description des règles Pareto-optimales pour l'expérimentation 2

RID	rules	confidence	complexity	Contribution
556	$< e_1, k_2m_2 > \rightarrow y= \{1\}$	1	2	0.06
923	$< e_1, k_{10}m_8 > \rightarrow y= \{1\}$	1	10	0.19
838	$< e_1, k_9m_6 > \rightarrow y= \{1\}$	0.86	9	0.12

La transformation des différents modes problématiques identifiés par ces trois règles, ainsi que les tables de contingence et les mesures de performance résultantes, sont décrits dans le Tableau 5.18. Puisque ces trois règles ont été sélectionnées, les modes problématiques correspondants seront transformés et la (les) traduction(s) optimales selon les trois indicateurs sera (seront) considérée(s) comme cause(s) finale(s) des produits problématiques de cette expérimentation.

Tableau 5.18 : Synthèse des résultats de CLARIF pour l'expérimentation 2

<table>
<tr><th></th><th></th><th>R 556
< e_1, k_2m_2 >
(AVEC EQUILIBRAGE DE CLUSTERS)</th><th>R923
< e_1, $k_{10}m_8$ >
(SANS EQUILIBRAGE DE CLUSTERS)</th><th>R838
< e_1, k_9m_6 >
(SANS EQUILIBRAGE DE CLUSTERS)</th></tr>
<tr><td colspan="2">La transformation du mode problématique identifié par la règle correspondante</td><td> 
</td><td> 
</td><td> 
</td></tr>
<tr><td rowspan="2">Table de contingence</td><td>sur D^A</td><td><table><tr><th></th><th>0 (^-1)</th><th>2 (^1)</th></tr><tr><td>-1</td><td>49</td><td>3</td></tr><tr><td>1</td><td>48</td><td>4</td></tr></table></td><td><table><tr><th></th><th>0 (^-1)</th><th>8 (^1)</th></tr><tr><td>-1</td><td>52</td><td>0</td></tr><tr><td>1</td><td>41</td><td>11</td></tr></table></td><td><table><tr><th></th><th>0 (^-1)</th><th>6 (^1)</th></tr><tr><td>-1</td><td>52</td><td>0</td></tr><tr><td>1</td><td>44</td><td>8</td></tr></table></td></tr>
<tr><td>sur D^T</td><td><table><tr><th></th><th>0</th><th>2</th></tr><tr><td>-1</td><td>8</td><td>0</td></tr><tr><td>1</td><td>8</td><td>0</td></tr></table></td><td><table><tr><th></th><th>0</th><th>8</th></tr><tr><td>-1</td><td>8</td><td>0</td></tr><tr><td>1</td><td>6</td><td>2</td></tr></table></td><td><table><tr><th></th><th>0</th><th>6</th></tr><tr><td>-1</td><td>8</td><td>0</td></tr><tr><td>1</td><td>6</td><td>2</td></tr></table></td></tr>
<tr><td colspan="2">Mesures de performance pour expliquer les produits problématiques</td><td><table><tr><th></th><th>confiance</th><th>contribution</th><th>complexité</th></tr><tr><td>D^A</td><td>0.57</td><td>0.08</td><td rowspan="2">1</td></tr><tr><td>D^T</td><td>0</td><td>0</td></tr></table></td><td><table><tr><th></th><th>confiance</th><th>contribution</th><th>complexité</th></tr><tr><td>D^A</td><td>1</td><td>0.21</td><td rowspan="2">1</td></tr><tr><td>D^T</td><td>1</td><td>0.25</td></tr></table></td><td><table><tr><th></th><th>confiance</th><th>contribution</th><th>complexité</th></tr><tr><td>D^A</td><td>1</td><td>0.15</td><td rowspan="2">1</td></tr><tr><td>D^T</td><td>1</td><td>0.25</td></tr></table></td></tr>
</table>

On remarque que la transformation du mode problématique identifié par la règle n°556 a détérioré la qualité de l'explication donnée, puisque la confiance est passée de 100% à 57%. Pour la règle 923, la qualité n'a pas été impactée par cette transformation. Par contre, la transformation de la règle 838 a amélioré la qualité initiale, en passant d'une confiance de 86% à une confiance de 100%. Par ailleurs, selon les trois indicateurs de qualité, la transformation de la règle 923 est la meilleure cause, puisqu'elle maximise la confiance (1) et la contribution (0.21) et minime la complexité (1).

Par ailleurs, notons que les transformations des deux règles 923 et 838 ne dépendent que de la variable 292. Nous pouvons conclure que cette variable doit, donc, avoir un sens physique particulier, voir critique pour la production

Pour finir cette lecture des résultats de *CLARIF*, pour cette expérience, nous notons que la qualité de la cause ne s'est pas détériorée, comme c'était le cas pour la première expérience, puisque la confiance est stable à 1, et la contribution connaît une légère amélioration sur D^T (sauf pour la traduction de la règle 556).

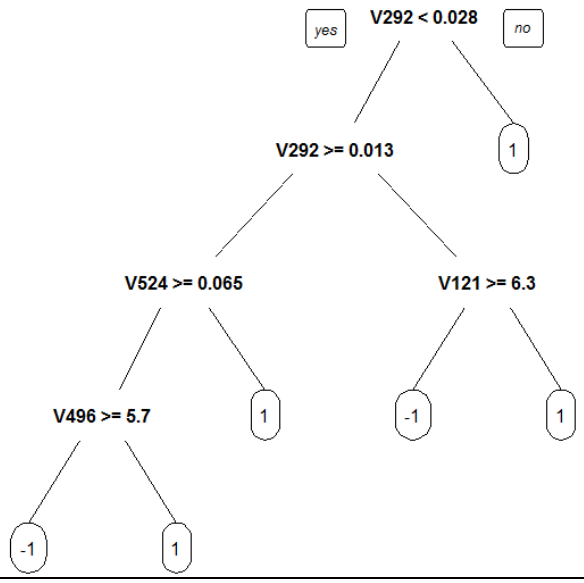
Nous nous intéressons, maintenant, à l'application de l'approche classique par induction directe d'arbre de décision sur les données de l'expérimentation 2. Les résultats sont décrits dans le Tableau 5.19. Sur les données d'apprentissage, l'arbre de décision induit permet d'expliquer avec une confiance de 80%, 80% des produits problématiques analysés. Cette confiance diminue à 63% pour les données D^T non apprises.

Cette dernière expérience illustre comment *CLARIF* peut permettre une meilleure qualité d'explication potentielle à la fois sur les données d'apprentissage et sur les données cachées des causes d'un défaut.

5.2.4 Conclusion

Les résultats obtenus sur ces deux expériences ont montré la sensibilité de *CLARIF* ainsi que de l'induction d'arbre de décision, face aux données de la base *SECOM*. Cette base, issue de mesures réelles, présente deux difficultés : les mesures sont bruitées et de nombreux relevés de mesures sont manquants. Le traitement des données manquantes par élimination des variables associées a réduit le nombre de variables et, par la même, accru l'influence du bruit sur l'apprentissage. Par ailleurs, la deuxième expérimentation a permis de démontrer l'efficacité de *CLARIF*, relativement à une induction directe d'arbre de décision, pour l'aide à l'explication d'un phénomène impactant une production. Dans les deux expériences, parmi le grand nombre de règles candidates générées (plusieurs milliers), on constate que peu (1 et 3, respectivement) sont dominantes au sens de Pareto suivant les trois critères que nous avons proposés.

Tableau 5.19: Synthèse des résultats de l'approche classique pour l'expérimentation 2

<table>
<tr><td>L'arbre de décision induit</td><td> 
</td></tr>
<tr><td rowspan="2">Table de contingence
sur D^A</td><td><table>
<tr><th></th><th>^-1</th><th>^1</th></tr>
<tr><td>-1</td><td>42</td><td>10</td></tr>
<tr><td>1</td><td>10</td><td>42</td></tr>
</table></td></tr>
<tr><td>sur D^T</td><td><table>
<tr><th></th><th>-1</th><th>1</th></tr>
<tr><td>-1</td><td>4</td><td>4</td></tr>
<tr><td>1</td><td>1</td><td>7</td></tr>
</table></td></tr>
<tr><td>Mesures de performance pour expliquer les produits problématiques</td><td><table>
<tr><th></th><th>confiance</th><th>contribution</th><th>complexité</th></tr>
<tr><td>D^A</td><td>0.8</td><td>0.8</td><td rowspan="2">4</td></tr>
<tr><td>D^T</td><td>0.63</td><td>0.87</td></tr>
</table></td></tr>
</table>

5.3 ÉTUDE CRITIQUE ET EXTENSIONS

5.3.1 Étude des limites de l'approche proposée

Pour expliquer un phénomène *Y*, la méthode proposée, *CLARIF*, se base sur la sélection des étapes causes potentielles de celui-ci. Par exemple, dans le cadre d'une explication de perte de qualité locale, lors de la première étape du processus *ECD* décrit dans le chapitre 3, pour expliquer un phénomène de perte de qualité *OOC* détecté à des étapes *Q*, on se base sur l'expertise métier pour définir l'ensemble des étapes de production *P* en relation avec les étapes de contrôle qualité.

Par ailleurs, quand une perte de qualité est détectée par des étapes de contrôle qualité avancées dans le processus de fabrication, on parle de perte de qualité globale. Une perte de

qualité globale est détectée à une ou plusieurs étapes de contrôle qualité plus globales. À la différence des étapes de contrôle qualité locale qui viennent valider les étapes de production les précédant directement, les étapes de contrôle qualité globale viennent valider un ensemble plus large d'étapes de production antérieures. Par exemple, le premier test paramétrique *PT*, qui vient à la fin de la création de la première couche de métallisation (début du processus *BEOL)*, en industrie du semi-conducteur, est un contrôle qualité global. Il permet de valider l'ensemble des étapes qui le précédent, à savoir toutes les étapes de création des composants du *CI*, *i.e.* toutes les étapes du processus *FEOL*, mais aussi les première étapes de métallisation du processus *BEOL*. Un autre exemple de contrôle qualité global est représenté par les tests de tri électriques *EWS* qui arrivent en fin du processus *FE* (*Front End*) et qui viennent valider la qualité des puces produites tout au long des étapes du processus *FE*.

Sachant que par exemple, le processus de fabrication en semi-conducteur présente environ 500 étapes de production, il est important de sélectionner un sous ensemble de celles-ci avant de pouvoir appliquer la méthode dans un objectif d'optimisation de temps d'analyse et d'effort.

Pour des étapes avancées dans le processus de fabrication. Il est ainsi, souvent difficile d'identifier les étapes potentiellement causes d'un phénomène *Y*. Pour cela, nous avons besoin de proposer une méthode de sélection des étapes potentiellement causes.

5.3.2 Proposition d'une analyse récursive par niveau

Pour cette analyse récursive par niveau, nous commençons par introduire une notion plus globale qu'un *segment*, qu'on nomme *« brique »*, noté *b*. Ainsi, une perte de qualité globale est relative à une "*b*", *i.e.* un ensemble de segments de production, $\mathbb{S}$, suivi par une ou plusieurs étapes de contrôle qualité globale, Q_G.

$$b=\{\mathbb{S}, Q_G\}$$

Autrement dit, une brique est définit comme la succession des étapes de production, ainsi que des étapes de contrôle qualité du premier segment, suivies par celles du deuxième segment, … Pour finir, un ensemble d'étapes de contrôle qualité globale, Q_G, permet de contrôler la qualité des plaques atteignant la fin de *b*.

$$b= \{P_{s1}, Q_{s1}, P_{s2}, Q_{s2}, \ldots, Q_G\}.$$

Notons $P_{\mathbb{S}}$ et $Q_{\mathbb{S}}$ les ensembles des étapes de production et de contrôle qualité de tous les segments relatifs à la brique analysée.

$$P_{\mathbb{S}} = \{P_s\} \ \forall s \in \mathbb{S}$$

$$Q_{\mathbb{S}} = \{Q_s\} \ \forall s \in \mathbb{S}$$

Les étapes de contrôle qualité globale Q_G servent à valider le bon déroulement de l'ensemble des étapes de production, $P_{\mathbb{S}}$, relatives aux différents segments $s \in \mathbb{S}$. L'intérêt de ce contrôle qualité global est de valider les relations entre les différentes étapes qui composent la brique à analyser, car des plaques contrôlées bonnes localement au niveau segment peuvent être finalement problématiques. Rappelons, aussi, que toutes les plaques ne sont pas mesurées durant les étapes de contrôle qualité intermédiaires, relatives aux segments $\mathbb{S}$, il est ainsi possible qu'une plaque, non mesurée aux étapes de contrôle intermédiaires, soit mesurée et détectée comme problématique à une étape de contrôle qualité globale.

Comme pour la perte de qualité locale, une perte de qualité globale est représentée par l'ensemble de toutes les plaques détectées problématiques à au moins une étape de contrôle qualité globale $q_G \in Q_G$. Les plaques bonnes sont celles réussissant tous les contrôles élémentaires des étapes de contrôle qualité globale.

$$OOC = \bigcup_{\forall q \in Q_G} OOC^q$$

$$OK = \bigcap_{q \forall \in Q_G} OK^q$$

Comme décrit dans la Figure 5.9, trois types de données sont disponibles :

- Premièrement, on a les données décrivant les différentes étapes de contrôle qualité globale Q_G, où la perte de qualité globale à expliquer a été détectée. Dans l'exemple illustré dans la Figure 5.9, le fichier D^{q_G} décrit les données collectées lors de la première étape de contrôle qualité globale q_G.
- Deuxièmement, on a des données décrivant les étapes de contrôle qualité intermédiaires $Q_{\mathbb{S}}$, relatives aux étapes de contrôle qualité Q_s des différents segments de production $s \in \mathbb{S}$. Le fichier D^{q1} décrit les données collectées lors de la première étape de contrôle qualité intermédiaire *q1* du segment *s*.
- Finalement, les données décrivant les étapes de production $P_{\mathbb{S}}$, relatives aux étapes de production P_s des différents segments de production $s \in \mathbb{S}$. Rappelons que ces données sont collectées pour chaque équipement de production. Ainsi, le fichier de données D^{p1t1} représente les données de production décrivant l'état de l'équipement *t1* lors du traitement des plaques à l'étape *p1* du segment *s*.

Ainsi, expliquer une perte de qualité détectée à une étape de contrôle qualité globale à partir des données disponibles au niveau d'une brique, revient à corréler les données décrivant les résultats des étapes de contrôle qualité globale Q_G, aux données des étapes de contrôle qualité intermédiaires $Q_{\mathbb{S}}$ et/ou à celles des étapes de production $P_{\mathbb{S}}$. Cela permettrait d'expliquer une perte de qualité globale par *des conditions sur les résultats de contrôles intermédiaires et/ou par des conditions de production*.

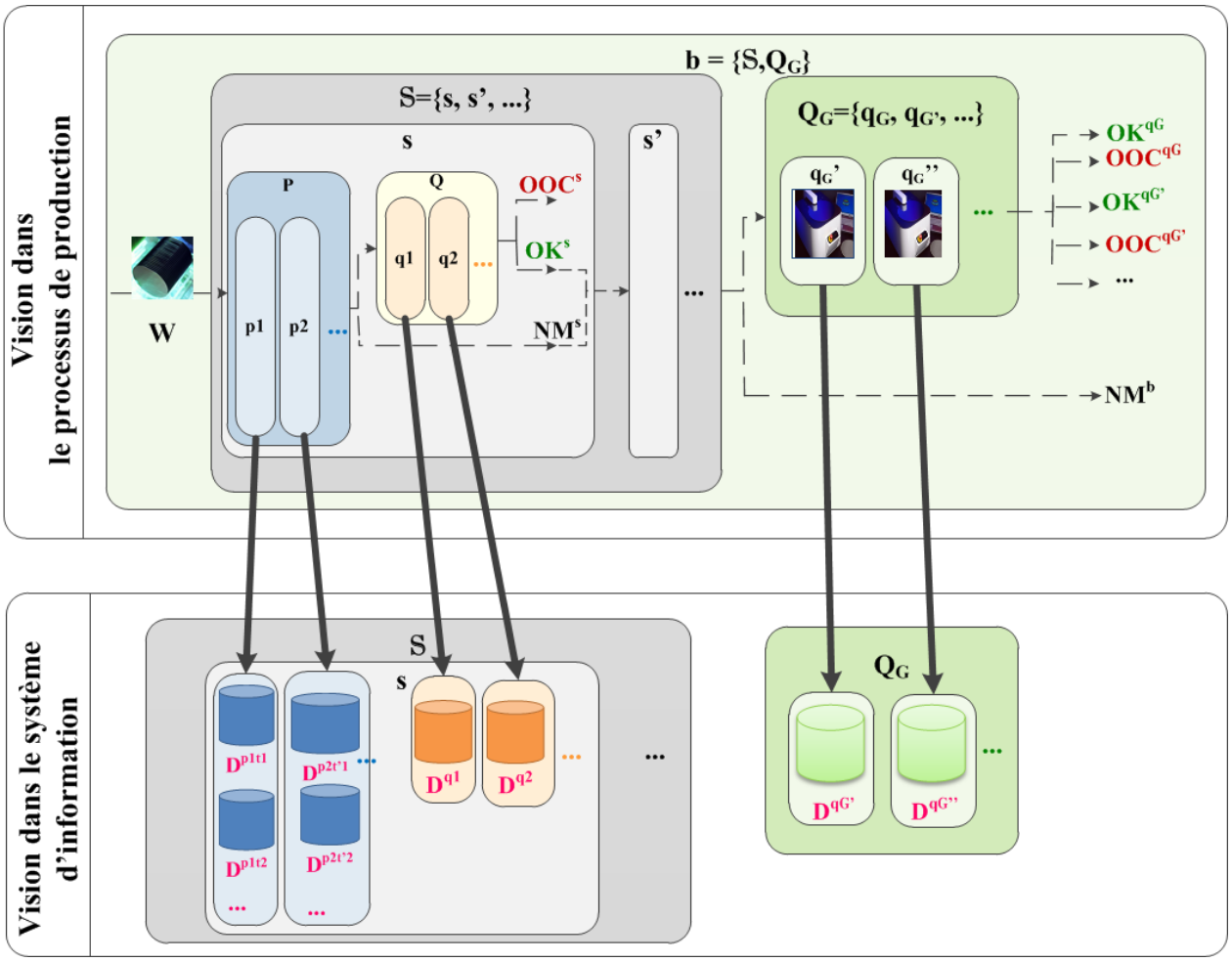


Figure 5.9 : Représentation des données collectées au niveau d'une brique.

À partir de cette décomposition en briques et segments du processus de fabrication, et telle que illustrée dans la Figure 5.10, nous proposons une approche d'*E*xtraction des *C*onnaissances à partir des *D*onnées *(ECD*) qui utilise les mêmes principes et méthodes que celle proposée pour l'analyse locale mono-segment, dans la section 3.4 du troisième chapitre.

Cette approche est proposée afin de permettre l'identification de connaissances globales, au niveau de plusieurs segments du processus de fabrication, décrivant un ensemble de conditions de contrôle qualité problématiques, potentiellement explicatives de cas de perte de qualité globales.

Ainsi, l'application de la méthode *CLARIF* à ce type de contexte peut être définie par un processus *ECD* en trois étapes, qui seront décrites dans les sections suivantes.

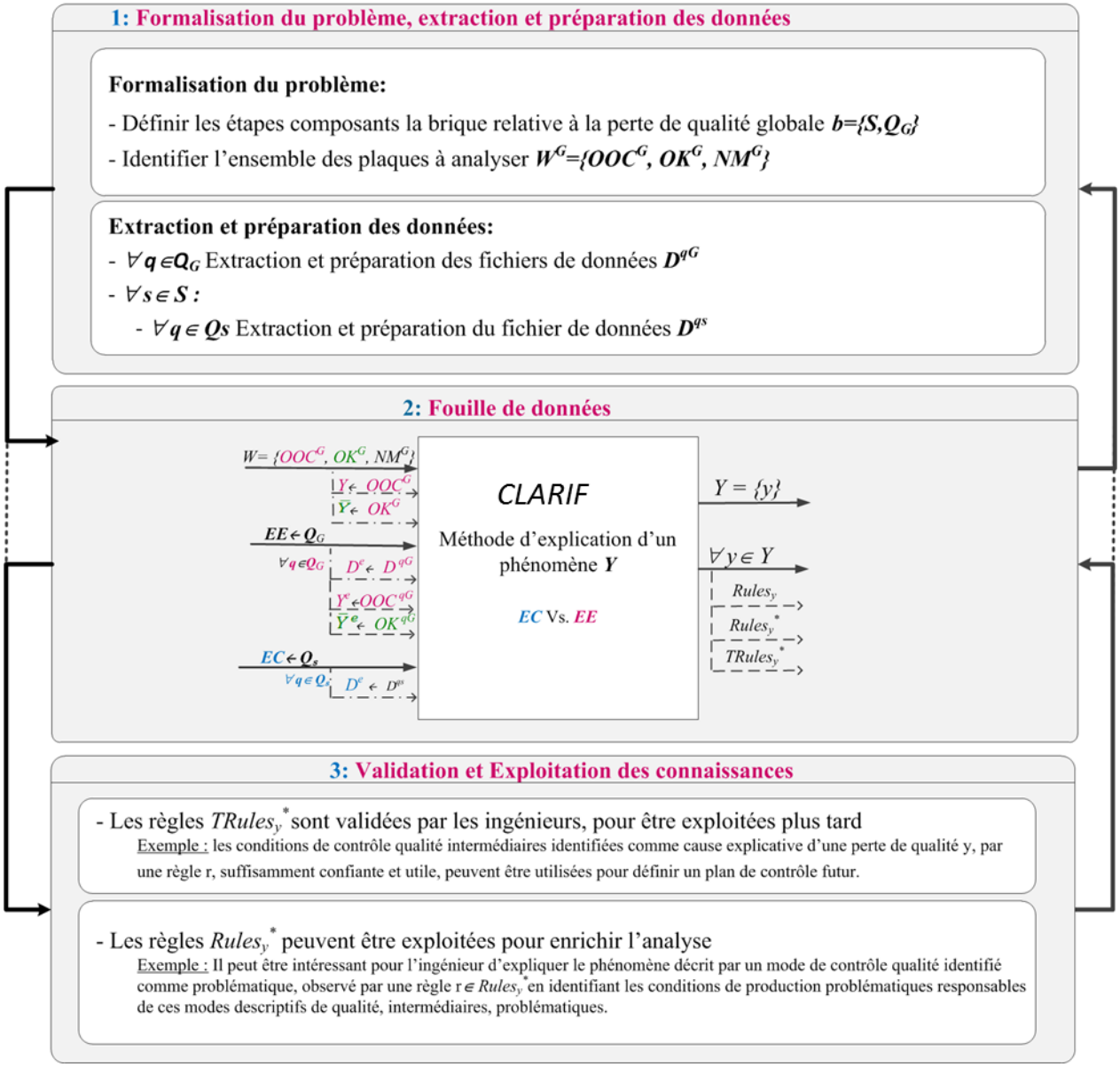


Figure 5.10: Une approche ECD pour l'explication des cas de perte de qualité globale

5.3.2.1 Étape 1: Formulation du problème, l'extraction et la préparation des données

5.3.2.1.1 Formulation du problème

Premièrement, la formulation du problème consiste à identifier l'ensemble des étapes de contrôle qualité globales, noté Q_G, qui ont détecté le phénomène à expliquer. Ce contrôle global peut être composé par une ou plusieurs étapes élémentaires q_G.

Deuxièmement, on s'intéresse à identifier l'ensemble des plaques à analyser, ainsi que la période temporelle. Comme pour le cas mono-segment, *ECD local*, trois types de plaques sont à analyser, les plaques problématiques, notées OOC^G, les plaques bonnes, notées OK^G et les plaques non mesurées NM^G. Les plaques OOC^G, sont des plaques détectées en dehors des limites de contrôle pour au moins une étape de contrôle qualité globale, $q \in Q_G$. Les plaques OK^G, quant à elles, respectent toutes les limites définies pour chacune des étapes de Q_G. Finalement, les plaques NM^G représentent l'ensemble des plaques non mesurées aux différentes étapes de contrôle qualité global Q_G. L'ensemble de ces plaques est noté

$$W^G = \{OK^G \cup OOC^G \cup NM^G\}.$$

On note par M^G les plaques mesurées aux étapes de contrôle qualité globale, *i.e.* les plaques bonnes OK^G et celles qui sont problématiques OOC^G.

Troisièmement, on s'intéresse à définir la structure de la brique à analyser. Pour cela, on identifie l'ensemble des segments qui la constitue, noté $\mathcal{S}$. Chaque segment, $s \in \mathcal{S}$, sera composé d'un ensemble d'étapes de production P_s et d'un ensemble d'étapes intermédiaires de contrôle qualité Q_s. Après avoir défini la brique d'analyse, on extrait les données décrivant l'historique des plaques W_G aux différentes étapes de la brique *b*.

5.3.2.1.2 Extraction et préparation des données d'analyse

D'un côté, on commence par extraire les fichiers de données relatifs aux étapes de contrôle qualité global Q_G. On obtient ainsi un fichier de donnée, noté D^{qG}, relatif à chaque étape de contrôle qualité globale $q_G \in Q_G$.

$$D^{qG} = \{q_G(w)\} \ \forall \ w \in W^{qG}$$

$q_G(w)$ est le vecteur de données représentant les mesures physiques ou électriques collectées à une étape de contrôle qualité globale $q_G \in Q_G$.

D'un autre côté, on extrait les données qui vont servir à la deuxième étape de l'approche. Cette étape servira pour la première itération d'analyse, où on s'intéresse à l'explication de la perte de qualité globale par des modes descriptifs de qualité intermédiaires. On se limite, ainsi, à l'extraction des données relatives aux étapes de contrôle qualité intermédiaires Q_s pour chaque segment $s \in \mathcal{S}$ de la brique analysée. On obtient ainsi un fichier de données, noté D^{qs} relatif à chaque étape de contrôle qualité $q \in Q_s$ de chaque segment $s \in \mathcal{S}$.

$$D^{qs} = \{q_s(w)\} \ \forall \ w \in W^{qs}$$

$q_s(w)$ est le vecteur de données représentant les mesures physiques ou électriques collectées à l'étape de contrôle qualité intermédiaire q_s relative au segment *s*. W^{qs} est l'ensemble des plaques mesurées durant cette étape.

5.3.2.2 Étape 2: Fouille de données

Une fois que les fichiers d'analyse sont extraits, on passe à l'approche de fouille de données, en appelant *CLARIF* la méthode proposée pour l'explication d'un phénomène *Y*. Dans notre cas, le phénomène à expliquer est représenté par les plaques OOC^G. On cherche à expliquer ces plaques problématiques à travers des conditions sur les étapes de contrôle qualité intermédiaires. Pour cela, on donne en entrée de méthode, les différents fichiers relatifs au contrôle de qualité final D^{q_G} ainsi que les fichiers relatifs aux étapes de contrôle intermédiairesD^{q_s}.

Cette méthode permet l'identification de différents modes descriptifs de perte de qualité *y* définissant le phénomène global de perte de qualité *Y*. Pour expliquer chacun de ces modes, les deux types de règles sont extraits (*simple* et *de trajectoire*). Deux représentations de ces règles sont disponibles, des règles discrètes $Rules_y^*$, et des règles continues, $TRules_y^*$, pour expliquer chacun des modes de perte de qualité *y*.

5.3.2.3 Étape 3: Intégration des connaissances identifiées

Pour l'analyse multi-segments, on propose d'explorer les deux ensembles de règles obtenus, à savoir, les règles continues $TRules_y$ et les règles discrètes $Rules_y^*$.

D'un côté, l'ensemble final des règles $TRules_y$ permet d'expliquer une perte de qualité globale par des conditions particulières sur une ou plusieurs étapes de contrôle qualité intermédiaires dans la brique, noté Q_s. Cela permet d'identifier des modes de qualité locale explicatifs de cette perte de qualité globale. Ces règles sont ainsi le résultat de la corrélation des données collectées aux étapes de contrôle intermédiaires Q_s à celles collectées pour les étapes de contrôles finales Q_G. Grâce aux indicateurs de qualité (*confiance*, *complexité* et *contribution*), les ingénieurs peuvent intégrer de différentes façons, ces nouvelles connaissances dans les futures processus de contrôle, afin d'éviter des pertes de qualité similaires. Une exploitation possible de ces connaissances est la définition de nouvelles limites de contrôle sur les étapes de contrôles qualité concernées telles que définies par les règles identifiées.

D'un autre côté, une deuxième itération d'analyse permettrait d'enrichir ces premiers résultats obtenus, à travers l'analyse des règles non transformées, $Rules_y^*$. Ces règles ont permis d'expliquer une perte de qualité globale *y* à travers des modes de contrôle qualité intermédiaires, *i.e.* des conditions particulières de contrôle qualité relatifs à un ou plusieurs segments de production.

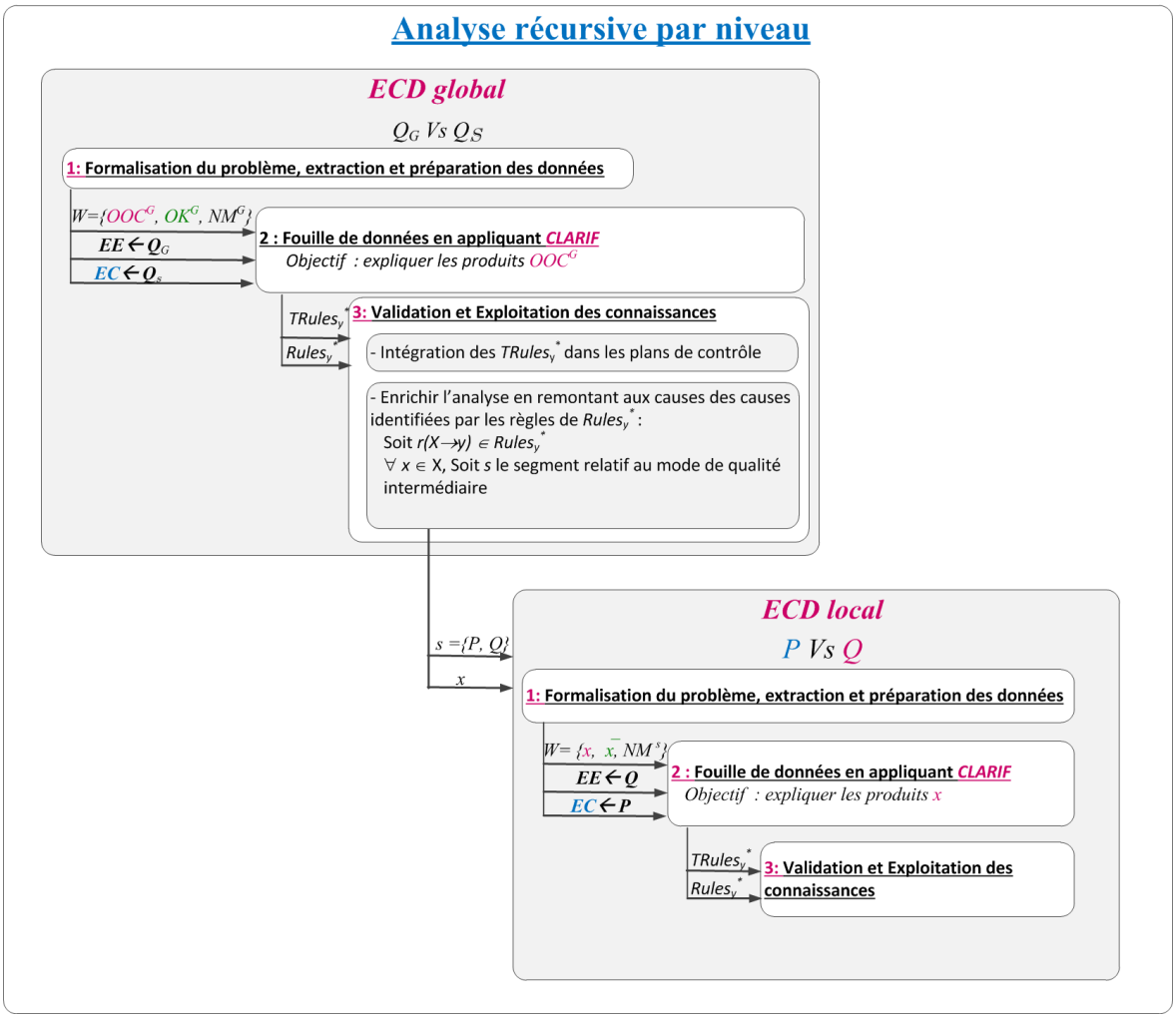


Figure 5.11 : Schématisation de l'approche d'analyse récursive par niveau

Comme illustrée dans la Figure 5.11, cette deuxième itération s'intéressera à expliquer chacun des modes problématiques, identifiés sur Q_S, à travers les étapes P_S correspondantes. Il s'agit d'expliquer itérativement, un des modes de contrôle qualité intermédiaires identifiés lors de l'analyse précédente comme une cause explicative potentielle de la perte de qualité globale. Ainsi, itérativement, on propose une analyse locale, mono-segment, en prenant, d'un côté, les étapes de contrôle qualité, Q, où le mode de contrôle qualité intermédiaire a été identifié à travers une règle $r \in Rules_y^*$, et d'un autre côté, les étapes de production qui leurs sont associées, P. On s'intéressera donc à corréler les données collectées aux étapes de production P, à celles collectées aux étapes Q. Cela permet d'identifier les équipements de production ainsi que leurs conditions particulières qui expliquent les modes de contrôle qualité intermédiaires, et par inférence la perte de qualité globale.

Sur la Figure 5.12, l'analyse globale a identifié une règle $r(x \rightarrow y1)$, qui identifie le mode de contrôle qualité intermédiaire x, sur l'étape de contrôle qualité intermédiaire $q1'$ relative au segment s', comme une cause explicative du sous-phénomène de perte de qualité

globale *y1*. Enrichir ce premier résultat reviendrait à appliquer la méthode d'analyse proposée en prenant comme contexte d'analyse le segment *s'*, et comme phénomène à expliquer le mode de qualité intermédiaire *x*. Cela implique de mettre en œuvre la première étape de l'approche *ECD* en local, en formalisant le problème dans le contexte d'analyse locale au segment *s'*, et en collectant les données des étapes de production $P_{s'}$ relatives au segment *s'*. Puis, de passer à la deuxième étape d'analyse, en considérant le phénomène à expliquer, *Y*, comme étant le mode de contrôle qualité intermédiaire *x*. Les règles discrètes permettront d'identifier les équipements de production ainsi que leurs conditions particulières qui expliquent la perte de qualité globale *y*.

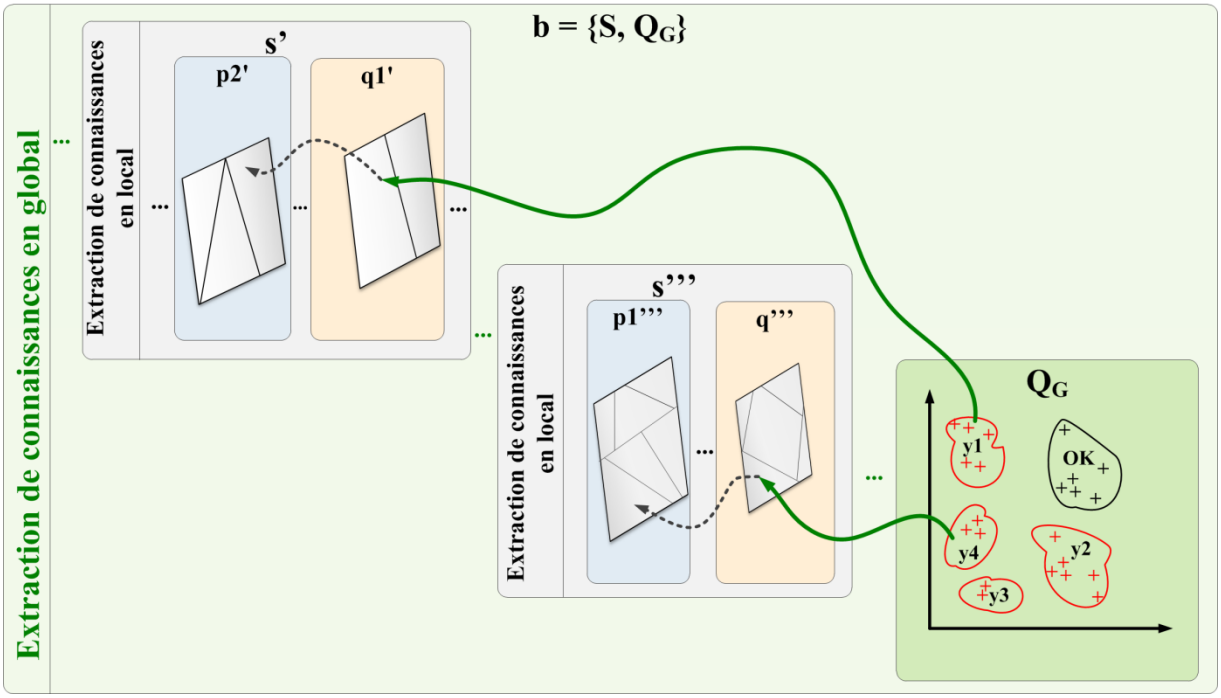


Figure 5.12: Intégration du processus *ECD* en local dans un processus *ECD* global

Ainsi, ces deux types d'analyse, illustrée dans la Figure 5.11 et la Figure 5.12, permettraient d'identifier des causes explicatives de différents types (qualité et production) et de différents niveaux (locale et globale) pour expliquer une perte de qualité globale.

5.4 RESUME ET CONCLUSION

Dans ce chapitre, nous avons testé la méthode proposée *CLARIF* dans différentes situations : pour commencer, nous avons appliqué l'approche *ECD* et *CLARIF*, sur un cas d'étude réel issu du site de fabrication *Crolles 300mm* de notre partenaire *STMicroelectronics*, afin d'expliquer des situations de perte de qualité locale. Les résultats ont démontré la capacité de cette approche à identifier les équipements de production, ainsi que leurs modes descriptifs de fonctionnement problématiques. Dans la deuxième section, nous avons testé *CLARIF* sur un cas indépendant disponible dans le répertoire d'apprentissage automatique

UCI Machine Learning Repository [86]. Pour les différentes expérimentations, une comparaison avec une approche classique d'induction d'arbre de décision a permis d'illustrer la valeur ajoutée de l'approche proposée. La dernière section de ce chapitre a été consacrée à une étude critique qui concerne le processus *ECD* proposé pour expliquer un phénomène local en proposant une extension à celui-ci, à travers un processus *ECD* qu'on a nommé *ECD global*, qui permet l'explication d'un phénomène plus globale. Ce processus est une proposition d'une analyse récursive par niveau pour expliquer des phénomènes globaux.

Dans le prochain chapitre, on clôture ce manuscrit de thèse avec une conclusion générale et une description des perspectives et travaux futurs.

Conclusion générale

Le secteur du semi-conducteur est caractérisé par un processus de fabrication long, complexe et couteux. Pour arriver à une puce électronique qui fonctionne, il y a de nombreuses étapes intermédiaires qui durent de 2 à 3 mois de fabrication. Le point de départ est une plaque de silicium sur laquelle sont appliquées diverses opérations, comme la lithographie, la gravure, l'implantation, la métallisation, le polissage …. Outre cette complexité intrinsèque, le système de production de l'industrie microélectronique est également caractérisé par des rythmes de changements et de transformations très élevés, classiquement illustrés par la loi de Moore.

Vue l'évolution rapide que connaît le secteur du semi-conducteur, il est crucial d'arriver à produire les puces avec une bonne qualité, le plus rapidement possible, c'est à dire d'atteindre un haut taux de rendement rapidement, notamment en phase d'implémentation de nouvelles technologies de production. Le rendement est un indicateur majeur sur la santé d'un site de fabrication : même un léger gain au niveau du rendement peut induire un important profit financier. Une parfaite maîtrise du processus de fabrication et l'identification rapide des causes de perte de rendement sont la clé de réussite d'un site de fabrication.

Ainsi, la maîtrise du rendement d'un site de fabrication et l'identification rapide des causes de perte de qualité restent un défi quotidien pour les industriels, qui font face à une concurrence continue et des processus de fabrication très variés. Dans ce cadre, cette thèse a eu pour ambition de proposer une démarche d'analyse permettant l'identification rapide de l'origine d'un défaut, à travers l'exploitation d'un maximum des données disponibles grâce aux outils de contrôle qualité, tel que la *FDC*, la métrologie, les tests paramétriques *PT*, et le tri électrique *EWS*.

L'état de l'art effectué a montré une exploitation limitée des types de données disponibles, ainsi qu'une utilisation limitée des nouvelles techniques de fouille de données, puisque les méthodes actuellement disponibles se basent principalement sur des outils d'apprentissage supervisé, comme outils exploratoires pour remonter à la cause du problème.

Grâce à une décomposition du processus de fabrication en un ensemble d'étapes, de segments et de briques, nous avons proposé une analyse par étape et par plaque, à travers une méthode de fouille hybride, qui combine différentes méthodes de fouille de données, nommée *CLARIF*.

Contrairement aux travaux traditionnels, la méthode proposée *CLARIF* propose d'aborder une approche non supervisée qui consiste à générer des modes causes candidats pour expliquer un phénomène indépendamment du résultat final. La sélection finale des

causes sera faite dans un second temps lors de la confrontation entre les modes générés de façon non supervisée et les résultats finaux, présence ou pas du phénomène à expliquer.

La problématique d'explication d'une perte de qualité est généralisée pour une problématique d'explication d'un phénomène *Y*. Ainsi, *CLARIF* est une méthode de fouille de données générique qui permet d'expliquer un phénomène *Y* détecté à une ou plusieurs étapes effet *EE*, par des conditions particulières à des étapes causes *EC*.

CLARIF est une méthode de fouille de données qui combine trois méthodes de fouille de données issues de différentes familles, à savoir :

(1) le *clustering* qui consiste à regrouper des individus par similarité. *CLARIF* utilise le clustering, entre autre, pour la génération non supervisée des modes candidats sur les étapes *EC*.
(2) la *fouille de règles d'association* intervient pour la confrontation des modes candidats générés avec le phénomène à expliquer, permettant ainsi de réduire les causes potentielles. Par ailleurs, afin de contourner la problématique principale des algorithmes de fouille de règle d'association qui consiste dans un nombre important de règles générées, nous proposons d'optimiser cette génération à travers la proposition de l'algorithme *ARCI*, qui est une adaptation du célèbre algorithme de fouille de règles d'association, *APRIORI*, afin de permettre d'intégrer les contraintes spécifiques à la problématique de *CLARIF*, et des indicateurs de qualité de filtrage des règles à identifier.
(3) Finalement, *l'induction d'arbre de décision* intervient pour caractériser le(s) mode(s) cause(s) identifiés.

Par ailleurs, les travaux effectués dans le cadre de ce doctorat ont débouché sur la proposition de deux démarches d'application, qui consiste en deux processus d'*E*xtraction des *C*onnaissances à partir des *D*onnées : Un premier *ECD* est proposé pour guider les ingénieurs dans l'application de la méthode *CLARIF* pour l'explication d'un phénomène de perte de qualité local, *i.e.* relatif à un segment du processus de production, en industrie du semi-conducteur. Un deuxième *ECD* est défini pour l'explication d'une perte de qualité plus globale, relative à une partie plus large, ou à l'ensemble du processus de production à travers l'application de *CLARIF*.

Dans l'environnement de fabrication de circuits intégrés de *STMicroelectronics Crolles 300mm*, nous avons pu valider le processus *ECD* local et la méthode *CLARIF*, pour expliquer un nouveau phénomène de perte de qualité. Une étude comparative avec une autre méthode classique a permis de valider l'utilité de la méthode *CLARIF* de fouille de données proposée.

D'autres expérimentations ont été conduites sur des données industrielles indépendantes de notre partenaire, afin de valider scientifiquement l'efficacité de *CLARIF*. Pour cela, en premier lieu, nous avons appliqué *CLARIF* sur des données d'apprentissage pour

identifier des explications potentielles pour les produits problématiques analysés, en étudiant leur qualité à travers les inducteurs proposés. En parallèle, nous avons appliqué une approche classique par induction d'arbres de décision. Et nous avons comparé les résultats obtenus par les deux méthodes. Dans un second temps, nous avons testé les performances des résultats obtenus par les deux méthodes à expliquer des pertes de qualité non analysées, en les testant sur des données cachées, i.e. sur des données de test qui ne nous ont pas servi pour la première étape d'apprentissage.

Les expérimentations sur le cas *SECOM* ont démontré la sensibilité de *CLARIF* face aux données bruitées et des valeurs manquantes. Ainsi, une voie d'amélioration de ce travail de thèse peut être d'intégrer un outil de traitement des données bruitées et de sélection de variables. Ceci permettrait à *CLARIF* de mieux s'appliquer sur les données réelles sans devoir passer par un prétraitement séparé.

Notons qu'un enjeu majeur pour l'industrialisation *CLARIF* concerne la première phase d'extraction des données utiles pour appliquer l'analyse. Ceci pointe une problématique commune pour l'application de toutes les méthodes d'analyse des données, qui est *« comment rendre les données déjà disponibles grâce aux différents systèmes de production et de contrôle, plus accessibles, i.e. stockées en un format directement exploitable pour l'application de méthode d'analyse* ». Pour cela, des projets sont en cours chez *STMicroelectronics* visant à mieux structurer les données utiles dans des entrepôts de données plus accessibles aux outils de fouille de données, qui pourront être exploités pour l'industrialisation de la méthode proposée *CLARIF*.

La généralisation de la problématique d'explication d'un défaut de qualité par des conditions de fonctionnement sur une machine permet à *CLARIF* d'être déployée dans toute situation industrielle, à conditions de pouvoir:

- Définir le phénomène *Y* à expliquer ;
- Définir les étapes effets *EE* qui ont détecté *Y ;*
- Définir les étapes causes *EC* sur lesquelles nous cherchons des causes potentielles.

Grâce à cette généralisation, *CLARIF* peut être appliquée en utilisant des données de différents types autres que ceux étudiés dans le cadre de ce manuscrit, à condition de pouvoir formaliser la problématique sous la forme précédemment citée. Notons, qu'une condition de réussite de *CLARIF* réside dans la bonne formulation du problème : la sélection des étapes causes *EC*, et des étapes effets *EE*, ainsi que des données décrivant chaque étape. Une première perspective serait d'appliquer *CLARIF* pour expliquer d'autres phénomènes relatifs à d'autres secteurs industriels afin de valider sa généricité.

Par ailleurs, nous savons que le processus de fabrication, en particulier en semi-conducteur n'est pas stationnaire puisque ce que nous identifions aujourd'hui sur un échantillon peut ne pas être applicable demain et ce pour différentes raisons : changement de recette, de produits, de réglages sur les machines ... Ainsi, une autre voie d'amélioration de ce

travail de thèse peut être de proposer une analyse dynamique qui permettrait de vérifier si les connaissances extraites auparavant sont encore valides ou sont-elles devenues obsolètes, et qu'il faudrait soit supprimer, soit archiver.

Dans la poursuite de la précédente perspective, une autre ouverture pour des travaux complémentaires à cette thèse peut être d'intégrer un outil de gestion des connaissances extraites grâce aux différentes analyses effectuées, afin d'être utilisées si possible ultérieurement. Ainsi, avant de commencer une nouvelle analyse pour expliquer un nouveau phénomène observé, nous pouvons commencer par comparer si ce nouveau phénomène est similaire à un phénomène déjà analysé et pour lequel nous disposons déjà d'une explication ou s'il s'agit d'un nouveau phénomène à expliquer.

Bibliographie

1] A. Chen, «Recipe-independent indicator for tool health diagnosis and predictive maintenance,» *Semiconductor Manufacturing,* pp. 522-535, 2009.

2] Y. Usami, N. Eguchi et S. Isogai, «Semiconductor Yield Enhancement Solutions for Next Generation,» pp. 118-124, 2002.

3] M. Lutz, X. Boucher et O. Roustant, «Information Technologies capacity planning in manufacturing systems : Proposition for a modelling process and application in the semiconductor industry,» *Computers in Industry,* vol. 63, pp. 659-668, 2012.

4] m. t. Jairo, «Modélisation conceptuelle d'une unité de fabrication Microélectronique,» *Revista EIA,* pp. 9-24, 2007.

5] G. E. Moore, «Cramming more components onto integrated circuits,» *Electronics,* 1965.

6] S. Bassetto, *Contribution à la qualification et amélioration des moyens de production - Application à une usine de recherche et production de semi-conducteurs,* 2005.

7] S. P. Cunningham, C. J. Spanos et K. Voros, «Semiconductor Yield Improvement : Results and Best Practices,» *IEEE Transactions on Semiconductor Manufacturing,* pp. 103-109, 1995.

8] S.-C. Hsu et C.-F. Chien, «Hybrid data mining approach for pattern extraction from wafer,» *Int. J. Production Economics,* vol. 107, n° %11, pp. 88-103, 2007.

9] M. Pillet, Appliquer la maîtrise statistique des procédés MSP/SPC, Ed. d'Organisation, 2003.

10] E. Duclos, L'ABC de la MSP en BD, Doussard, Savoie, FR, 2006.

11] D. C. Montgomery, Introduction to statistical quality control, New York: John Wiley & Sons, 2001.

12] K. Sanjoy, «Survey of various statistical process control methods,» *Symposium A Quarterly Journal In Modern Foreign Literatures,* pp. 387-390, 1991.

13] W. A. Shewhart, «Quality control charts,» *Bell System Technical Journal,* vol. 5, n° %14, p. 593–603, October 1926.

14] J. Scanlan et K. O'Leary, «Knowledge-based process control for fault detection and classification,» 2003.

C.-f. Chien, C.-y. Hsu et P.-n. Chen, «Semiconductor fault detection and
15] classification for yield enhancement and manufacturing intelligence,» *Flexible Services and Manufacturing Journal,* pp. 367-388, 2013.

C. Cortes et V. Vapnik, «Support-Vector Networks,» *Machine Learning,* vol. 20,
16] pp. 273-297, 1995.

K.-Y. Chen, L.-S. Chen, M.-C. Chen et C.-L. Lee, «Using SVM based method for
17] equipment fault detection in a thermal power plant,» *Computers in Industry,* vol. 62, n° %11, pp. 42-50, 2011.

S. Mahadevan et S. L. Shah, «Fault detection and diagnosis in process data using
18] one-class support vector machines,» *Journal of Process Control,* vol. 19, n° %110, pp. 1627-1639, 2009.

S. J. Hong, W. Y. Lim, T. Cheong et G. S. May, «Fault Detection and
19] Classification in Plasma Etch Equipment for Semiconductor Manufacturing e-Diagnostics,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 25, n° %11, pp. 83-93, 2012.

S.-F. Liu, H.-E. Chueh, K.-H. Liao et F.-L. Chen, «An Intelligent Approach To
20] Develop Fault Detection and Classification in Photolithography Process of Semiconductor Manufacturing,» *Information-an International Interdisciplinary Journal,* vol. 14, pp. 1043-1048, 2011.

Z. He, X. Xu et S. Deng, «Discovering cluster-based local outliers,» *Pattern*
21] *Recognition Letters,* vol. 24, pp. 1641-1650, 2003.

Q. He et J. Wang, «Fault detection using the k-nearest neighbor rule for
22] semiconductor manufacturing processes,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 20, n° %14, pp. 345-354, 2007.

W.-C. Chen, S.-S. Tseng et C.-Y. Wang, «A novel manufacturing defect detection
23] method using association rule mining techniques,» *Expert Systems with Applications,* vol. 29, n° %14, 2005.

P. Rastogi, N. Kozicki, F. Golshani et S. Member, «ExPro-An Expert System
24] Based Process Management System,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 6, n° %13, 1993.

U. Fayyad, G. Piatetsky-Shapiro and P. Smyth, "From Data Mining to Knowledge
25] Discovery in Databases.," *Artificial Intelligence Magazine,* pp. 37-54, 1996.

S. Weiss, R. Baseman, F. Tipu, C. Collins, W. Davies, R. Singh and J. Hopkins,
26] "Rule-based data mining for yield improvement in semiconductor manufacturing," *Applied Intelligence,* pp. 318-329, 2009.

W. Taam et M. Hamada, «Detecting spatial effects from factorial experiments: an
27] application from integrated-circuit manufacturing,» *Technometrics,* vol. 35, n° %12, pp. 149-160, 1993.

C. Stapper, «LSI yield modeling and process monitoring,» *IBM Journal of*

28] *Research and Development,* vol. 44, n° %12, pp. 112-118, 2000.

29] D. Friedman, M. Hansen, V. Nair et D. James, «Model-free estimation of defect clustering in integrated circuit fabrication,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 10, n° %13, pp. 344-359, 1997.

30] U. Kaempf, «The Binomial Test : A Simple Tool to Identify Process Problems,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 8, n° %12, 1995.

31] W. CHIH-HSUAN, K. WAY et H. BENSMAIL, «Detection and classification of defect patterns on semiconductor wafers,» *IIE Transactions,* vol. 38, pp. 1059-1068, 2006.

32] W. Zhang, X. Li, S. Saxena, A. Strojwas et R. Rutenbar, «Automatic clustering of wafer spatial signatures,» *Proceedings of the 50th Annual Design Automation Conference on - DAC '13,* 2013.

33] M. Gardner and J. Bieker, "Data Mining Solves Tough Semiconductor Manufacturing Problems," *KDD,* 2000.

34] H. W. Stoll, «Design for Manufacture: An Overview,» *Applied Mechanics Reviews,* vol. 39, n° %19, 1986.

35] Y. Hirano, W. Shindo, M. Ono, T. Yamaura, F. Satou et K. Imaoka, «Scrubber-induced substrate cracks found by data mining analysis,» *ISSM 2005, IEEE International Symposium on Semiconductor Manufacturing, 2005.,* pp. 257-259, 2005.

36] M. Anderson, «Design of Experiments,» *American Institute of Physics,* pp. 24-26, 1997.

37] H. Chou, M. Liao, J. Hsu, L. Yip et Y. Wu, «Design of Experiments to Achieve High Yield Manufacturing at 6-inch Foundry,» *GaAsMANTECH Conference,* vol. 18, n° %169, 2002.

38] R. M. Gardner, J. Bieker et S. Elwell, «Solving tough semiconductor manufacturing problems using data mining,» *IEEE/SEMI Advanced Semiconductor Manufacturing Conference.,* pp. 46-55, 2000.

39] A. Wong, «A statistical parametric and probe yield analysis methodology,» *Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems,* pp. 131-139, 1996.

40] M. Chakaroun, M. A. Djeziri, M. Ouladsine et J. Pinaton, «Set Theory for Root Cause Determination in Semiconductor Manufacturing System,» chez *10éme journées des Doctorants du LSIS*, Carquerianne, 2013.

41] D. Hand, H. Mannila et P. Smyth, Principles of Data Mining, Massachusetts Institute of Technology, 2001.

42] A. Cornuéjols, L. Miclet et J.-P. Haton, Apprentissage Artificiel: Concepts et algorithmes (2éme édition), 2010.

U. Fayyad, G. Piatetsky-Shapiro et P. Smyth, «The KDD process for extracting

43] useful knowledge from volumes of data,» *Communications of the ACM,* pp. 27-34, 1996.

44] F. Mieno et T. Sato, «Yield improvement using data mining system,» 1999.

45] C. J. Romanowski et R. Nagi, «Improving preventive maintenance scheduling using data mining techniques,» chez *Industrial Engineering Research Conference*, Phoenix AZ, 1999.

46] V. Raghavan, «Application of Decision Trees for Integrated Circuit Yield Improvement,» *IEEE/SEMI Advanced Semiconductor Manufacturing Conference,* pp. 262-265, 2002.

47] D. Braha et A. Shmilovici, «Data mining for improving a cleaning process in the semiconductor industry,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 15, n° %11, pp. 91-101, 2002.

48] D. Braha et A. Shmilovici, «On the use of decision tree induction for discovery of interactions in a photolithographic process,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 16, n° %14, pp. 644-652, 2003.

49] H. Tsuda, H. Shirai, O. Takagi et R. Take, «Yield analysis and improvement by reducing manufacturing fluctuation noise,» *The Ninth International Symposium on Semiconductor Manufacturing,* pp. 249-252, 2000.

50] a. K. Choudhary, J. a. Harding et M. K. Tiwari, «Data mining in manufacturing: a review based on the kind of knowledge,» *Journal of Intelligent Manufacturing,* pp. 501-521, 2008.

51] T. Kohonen, «Self-Organized Formation of Topologically Correct Feature Maps,» *Biological Cybernetics,* vol. 43, 1982.

52] J. Quinlan, C4.5 Programing f or machine Learning, San Mateo, California: Morgan Kaufmann, 1993.

53] R. Leivian, W. Peterson et M. Gardner, «CorDex : a Knowledge Discovery Tool,» *WSOM'97: Workshop on Self-organizing Maps,* 1997.

54] F. Bergeret et C. Le Gall, «Yield Improvement Using Statistical Analysis of Process Dates,» *IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING,* vol. 16, n° %13, pp. 535-542, 2003.

55] T.-S. Li, C.-L. Huang et Z.-Y. Wu, «Data Mining using Genetic Programming for Construction of a Semiconductor Manufacturing Yield Rate Prediction System,» *Journal of Intelligent Manufacturing,* vol. 17, n° %13, pp. 355-361, 2006.

56] J. R. Koza, «Genetic programming as a means for programming computers by natural selection,» *Statistics and Computing,* vol. 4, n° %12, pp. 87-112, 1994.

57] S. Sarawagi, «Sequence data mining,» chez *Advanced Methods for Knowledge Discovery from Complex Data*, Springer, 2005, pp. 153-187.

K. Kerdprasop et N. Kerdprasop, «Performance Analysis of Complex

58] Manufacturing Process with Sequence Data Mining Technique,» *International Journal of Control and Automation,* vol. 6, n° %13, pp. 301-312, 2013.

P.-N. Tan, M. Steinbach et V. Kumar, «Association Analysis: Basic Concepts and
59] Algorithms,» chez *Introduction to Data Mining*, 2006, pp. 327-414.

R. Agrawal et R. Srikant, «Fast algorithms for mining association rules,» *Proc.
60] 20th Int. Conf. Very Large Data Bases, VLDB,* 1994.

K. Chitra et B. Subashini, «Data Mining Techniques and its Applications in
61] Banking Sector,» *International Journal of Emerging Technology and Advanced Engineering,* vol. 3, n° %18, pp. 219-226, 2003.

M. Vijayalakshmi, S. Kumar et B. Kavyashree, «Investigating Interesting Rules
62] Using Association Mining for Educational Data,» *International Journal of Advanced Research in Computer Engineering & Technology,* vol. 3, n° %12, pp. 268-271, 2014.

S. Imberman, M. Kress et D. McCloskey, «Using Frequent Pattern Mining to
63] Identify Behaviors in a Naked Mole Rat Colony,» *International Florida Artificial Intelligence Research Society Conference,* pp. 394-399, 2012.

S. Garasia, D. Rana et R. Mehta, «HTTP botnet detection using frequent
64] patternset mining,» *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY,* pp. 619-624, 2012.

X. Zhong, «The Application Of Apriori Algorithm For Network Forensics
65] Analysis,» *Journal of Theoretical and Applied Information Technology,* pp. 430-434, 2013.

D. Bansal et L. Bhambhu, «Usage of Apriori Algorithm of Data Mining as an
66] Application to Grievous Crimes against Women,» *International Journal of Computer Trends and Technology,* vol. 4, n° %19, pp. 3194-3199, 2013.

A. L. Buczak et C. M. Gifford, «Fuzzy association rule mining for community
67] crime pattern discovery,» *ACM SIGKDD Workshop on Intelligence and Security Informatics - ISI-KDD '10,* 2010.

A. Casali et C. Ernst, «Discovering Correlated Parameters in Semiconductor
68] Manufacturing Processes: A Data Mining Approach,» *IEEE Transactions on Semiconductor Manufacturing,* pp. 118-127, 2012.

P. C. P. Chen, S. Wu, J. L. J. Lin, F. Ko, H. Lo, J. Wang, C. Yu et M. Liang,
69] «Virtual metrology: a solution for wafer to wafer advanced process control,» *IEEE International Symposium on Semiconductor Manufacturing,* 2005.

C. Yung-cheng et C. Fan-Tien, «Application development of virtual metrology in
70] semiconductor industry,» *31st Annual Conference of IEEE Industrial Electronics Society,* pp. 124-129, 2005.

P. C. P. Chen, «A virtual metrology system for semiconductor manufacturing,»
71] *Expert Systems with Applications,* vol. 36, n° %110, pp. 12554-12561, 2009.

A. Cornuéjols, L. Miclet et T. Mitchell, Apprentissage artificiel : concepts et

72] algorithmes, Eyrolles, 2002, ISBN : 2-212-11020-0.

Y. Liu, Z. Li, H. Xiong, X. Gao et J. Wu, «Understanding of internal clustering
73] validation measures,» *2010 IEEE International Conference on Data Mining,* pp. 911-916, 2010.

L. Zadeh, «Fuzzy Sets,» *Information and Control 8,* n° %18, p. 338–353, 1965.
74]

M. Ester, H. P. Kriegel, J. Sander et X. Xu, «A density-based algorithm for
75] discovering clusters in large spatial databases with noise,» *Second International Conference on Knowledge Discovery and Data Mining,* p. 226–231, 1996.

M. P. P. P. V. B. e. L. S. Lenca P., «Critères d'évaluation des mesures de qualité
76] des règles d'association,» *Revue des Nouvelles Technologies de l'Information, Entreposage et Fouille de données ,* pp. 123-134, 2003.

R. Fisher, «Iris Plants Database,» 1988. [En ligne]. Available:
77] https://archive.ics.uci.edu/ml/machine-learning-databases/iris/. [Accès le 01 Mai 2015].

J. Han et M. Kamber, Data Mining: Concepts and Techniques, The Morgan
78] Kaufmann Series in Data Management Systems, 2006.

l. Breiman, J. Friedman et C. Stone, Classification and Regression Trees,
79] Chapman and Hall/CRC, 1984.

D. MacKay, «An Example Inference Task : Clustering,» *Information Theory,*
80] *Inference and learning algorithms,* pp. 284-292, 2003.

A. K. Jain, «Data Clustering : 50 Years Beyond K-Means,» *Pattern Recognition*
81] *Letters,* 2009.

S. Bouker, R. Saidi, S. Ben Yahia et E. M. Nguifo, «Ranking and selecting
82] association rules based on dominance relationship,» *Tools with Artificial Intelligence (ICTAI), 2012 IEEE 24th International Conference on ,* pp. 658 - 665 , 2012.

K. Miettinen, Nonlinear Multiobjective Optimization, Boston, MA, USA, :
83] Kluwer, 1999.

S. Kotsiantis, D. Kanellopoulos et P. Pintelas, «Handling imbalanced datasets: A
84] review,» *GESTS International Transactions on Computer Science and Engineering,* vol. 30, 2006.

Y. Zhao, «R and Data Mining : Examples and Case Studies,» 26 April 2013. [En
85] ligne].

«UCI Machine Learning Repository,» [En ligne]. Available:
86] archive.ics.uci.edu/ml/. [Accès le 15 juin 2015].

«SECOM Data Set,» [En ligne]. Available:
87] https://archive.ics.uci.edu/ml/datasets/SECOM. [Accès le 15 juin 2015].

K. Mika, H. Mannila et H. Toivonen, «A Data Mining Methodology and its

88] application to Semi-Automatic Knowledge Acquisition,» *DEXA Workshop,* p. 670–677, 1997.

89] C. Chien, W. Wang and J.-C. Cheng, "Data mining for yield enhancement in semiconductor manufacturing and an empirical study," *Expert Systems with Applications,* 2007.

90] L. R. C et D. A. Hodges, «Benchmarking Semiconductor Manufacturing,» *IEEE Transactions on Semiconductor Manufacturing,* vol. 9, n° %12, pp. 158-169, 1996.

91] K. Ishikawa, Asian Productivity Organization , 1991.

92] W. H. Kruskal et W. A. Wallis, «Use of Ranks in One-Criterion Variance Analysis,» *Journal of the American Statistical Association,* vol. 47, n° %1260, pp. 583-621, 1952.

93] S. Lallich et O. Teytaud, «Évaluation et validation de l'intérêt des règles d'association,» *Revue des Nouvelles Technologies de l'Information,* Vols. %1 sur %2Mesures de Qualité pour la Fouille de Données, RNTI-E-1, pp. 183-208, 2004.

École Nationale Supérieure des Mines
de Saint-Étienne

NNT : *2015 EMSE 0795*

Hasna BARKIA Ep YAHYAOUI

A *post-hoc* Data Mining method for defect diagnosis - Application to the microelectronics sector

Speciality : Computer Sciences

Keywords : Semi-conductor manufacturing, quality loss causes identification, data mining, association rule mining, clustering, decision tree induction

Abstract :

Controlling the performance of a manufacturing site and the rapid identification of quality loss causes remain a daily challenge for manufacturers, who face continuing competition. In this context, this thesis aims to provide an analytical approach for the rapid identification of defect origins, by exploring data available thanks to different quality control systems, such *FDC*, *metrology*, *parametric tests PT* and the *Electrical Wafer Sorting EWS*. The proposed method, named *CLARIF*, combines three complementary data mining techniques namely *clustering*, *association rules* and decision trees induction. This method is based on unsupervised generation of a set of potentially problematic production modes, which are characterized by specific manufacturing conditions. Thus, we provide an analysis which descends to the level of equipment operating parameters. The originality of this method consists on (1) a pre-treatment step to identify spatial patterns from quality control data, (2) an unsupervised generation of manufacturing modes candidates to explain the quality loss case. We optimize the generation of association rules through the proposed *ARCI* algorithm, which is an adaptation of the famous association rules mining algorithm, *APRIORI* to integrate the constraints specific to our issue and filtering quality indicators, namely *confidence*, *contribution* and *complexity*, in order to identify the most interesting rules. Finally, we defined a Knowledge Discovery from Databases process, enabling to guide the user in applying *CLARIF* to explain both local and global quality loss problems.

École Nationale Supérieure des Mines
de Saint-Étienne

NNT : *2015 EMSE 0795*

Hasna BARKIA Ep YAHYAOUI

Méthode d'analyse de données pour le diagnostic *a posteriori* de défauts de production - Application au secteur de la microélectronique

Spécialité: Informatique

Mots clefs : industrie du semi-conducteur, identification des causes de pertes de qualité, fouille de données, fouille de règles d'association, clustering, induction d'arbres de décision.

Résumé :

La maîtrise du rendement d'un site de fabrication et l'identification rapide des causes de perte de qualité restent un défi quotidien pour les industriels, qui font face à une concurrence continue. Dans ce cadre, cette thèse a pour ambition de proposer une démarche d'analyse permettant l'identification rapide de l'origine d'un défaut, à travers l'exploitation d'un maximum des données disponibles grâce aux outils de contrôle qualité, tel que la *FDC*, la métrologie, les tests paramétriques *PT*, et le tri électriques *EWS*. Nous avons proposé une nouvelle méthode hybride de fouille de données, nommée *CLARIF*, qui combine trois méthodes de fouille de données à savoir, le *clustering*, les *règles d'association* et *l'induction d'arbres de décision*. Cette méthode se base sur la génération non supervisée d'un ensemble de modes de production potentiellement problématiques, qui sont caractérisés par des conditions particulières de production. Elle permet, donc, une analyse qui descend au niveau des paramètres de fonctionnement des équipements. L'originalité de la méthode consiste dans (1) une étape de prétraitement pour l'identification de motifs spatiaux à partir des données de contrôle, (2) la génération non supervisée de modes de production candidats pour expliquer le défaut. Nous optimisons la génération des règles d'association à travers la proposition de l'algorithme *ARCI*, qui est une adaptation du célèbre algorithme de fouille de règles d'association, *APRIORI*, afin de permettre d'intégrer les contraintes spécifiques à la problématique de *CLARIF*, et des indicateurs de qualité de filtrage des règles à identifier, à savoir *la confiance*, *la contribution* et *la complexité*. Finalement, nous avons défini un processus d'*E*xtraction de *C*onnaissances à partir des *D*onnées, *ECD* permettant de guider l'utilisateur dans l'application de *CLARIF* pour expliquer une perte de qualité locale ou globale.