HAL open science

Analyse macroscopique des grands systèmes : émergence épistémique et agrégation spatio-temporelle

Robin Lamarche-Perrin

To cite this version:

Robin Lamarche-Perrin. Analyse macroscopique des grands systèmes : émergence épistémique et agrégation spatio-temporelle. Autre [cs.OH]. Université de Grenoble, 2013. Français. NNT : 2013GRENM030 . tel-00933186

HAL Id: tel-00933186

https://theses.hal.science/tel-00933186v1

Submitted on 20 Jan 2014

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

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

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Informatique**

Arrêté ministériel : 7 août 2006

Présentée par

Robin LAMARCHE-PERRIN

Thèse dirigée par **Yves DEMAZEAU**
et codirigée par **Jean-Marc VINCENT**

préparée au sein du **Laboratoire d'Informatique de Grenoble**
et de l'**École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique**

Analyse macroscopique des grands systèmes

Émergence épistémique et agrégation spatio-temporelle

Thèse soutenue publiquement le **14 octobre 2013**,
devant le jury composé de :

Mme Brigitte PLATEAU
Professeure à Grenoble INP, Présidente du jury
M. Éric FLEURY
Professeur à l'École Normale Supérieure de Lyon, Rapporteur
M. Bernard MOULIN
Professeur à l'Université Laval, Québec, Rapporteur
Mme Salima HASSAS
Professeure à l'Université Claude Bernard, Lyon, Examinatrice
M. Yves DEMAZEAU
Directeur de recherche au CNRS, Directeur de thèse
M. Jean-Marc VINCENT
Maître de conférences à l'Université Joseph Fourier, Codirecteur de thèse

REMERCIEMENTS

Dans une semaine et un jour, si tout va bien,
Je serai donc Docteur – mais n'anticipons point.
N'étant, pour quelques jours, encore que doctorant,
Voici arrivée l'heure de mes remerciements.

Le dodécasyllabe, j'ai choisi pour format.
Mais n'étant pas poète, sachez (que je ne mente)
Que des « e » sont muets quand il ne faudrait pas,
Que les rimes, souvent pauvres, sont parfois suffisantes.

Je commence bien sûr par mes deux directeurs,
Yves et Jean-Marc, qui m'ont – tout à la fois – aidé,
Soutenu, initié au métier de chercheur,
Encouragé, instruit ! Vous n'avez pas idée.

Merci pour votre confiance, je vous dois bien la mienne,
Pour ces heures édifiantes que je compte par centaines.
Si c'était à refaire, je n'hésiterais pas.
Et à continuer ? Aucun doute, croyez-moi.

Merci également à toute l'équipe du CIST.
À Claude Grasland d'abord, pour son soutien – j'insiste !
Puisqu'il m'a savamment encadré en géo.
Merci pour son appui, à Marta Severo,

À Timothée Giraud et Nicolas Lambert,
Pour leurs avis carto et leurs conseils sous R,
À Marion Gentilhomme qui a – ce n'est pas rien
Relu ce manuscrit du début à la fin.

Lucas Schnorr a été un équipier hors-pair
Pour les expés, publis et discussions abstraites.
Et Damien Dosimont lui-aussi sait y faire.
Je leur dois pour beaucoup quant au chapitre 7.

Ma thèse ne serait pas, sans mon passé philo,
Il faut bien l'avouer, ce qu'elle est à présent.
Merci à ce propos, à Régis Catinaud,
Aux deux Denis de l'ARSH : à Perrin et Vernant.

Elle aurait mérité la participation
De l'écriture de Max, du parler de Simon,
Du raisonnement d'Alex, des couleurs de Tacha,
De Martin et Mélisse qui grandissent à grands pas.

Qu'il fut bon de bosser tous les jours à MAGMA,
Grâce à la bonne humeur et l'accent de Julie,
Aux coinches en compagnie de Carole et Wafa
Et aux contre-coinchées de Mika et Marty.

J'ai fait ces trois années d'innombrables trajets
Pour rejoindre tour à tour mes colocs préférés :
Du boulevard Édouard Rey (Mathieu, Julien, Régis)
Au 12e à Paris (Simon, Lucas, Alice).

Notez qu'Alice mérite un prix particulier,
Puisque, mieux que quiconque, elle me réconforta.
Salut aussi à Ju sur qui on peut compter
Pour un draft, un poker ou un Agricola.

Ces longs remerciements touchent bientôt à leur fin,
Mais je ne peux conclure sans le plus important :
Je veux parler bien sûr de Séverine et Jean.
Je leur dédie cette thèse et ces alexandrins.

Robin, le 6 octobre 2013

RÉSUMÉ

L'analyse des systèmes de grande taille est confrontée à des difficultés d'ordre *syntaxique* et *sémantique* : comment observer un million d'entités distribuées et asynchrones ? Comment interpréter le désordre résultant de l'observation microscopique de ces entités ? Comment produire et manipuler des abstractions pertinentes pour l'analyse macroscopique des systèmes ? Face à l'échec de l'approche analytique, le concept d'*émergence épistémique* – relatif à la nature de la connaissance – nous permet de définir une stratégie d'analyse alternative, motivée par le constat suivant : l'activité scientifique repose sur des *processus d'abstraction* fournissant des éléments de description macroscopique pour aborder la complexité des systèmes.

Cette thèse s'intéresse plus particulièrement à la production d'abstractions spatiales et temporelles par *agrégation de données*. Afin d'engendrer des représentations exploitables lors du passage à l'échelle, il apparaît nécessaire de contrôler deux aspects essentiels du processus d'abstraction. Premièrement, la complexité et le contenu informationnel des représentations macroscopiques doivent être conjointement optimisés afin de préserver les détails pertinents pour l'observateur, tout en minimisant le coût de l'analyse. Nous proposons des mesures de qualité (*critères internes*) permettant d'évaluer, de comparer et de sélectionner les représentations en fonction du contexte et des objectifs de l'analyse. Deuxièmement, afin de conserver leur pouvoir explicatif, les abstractions engendrées doivent être cohérentes avec les connaissances mobilisées par l'observateur lors de l'analyse. Nous proposons d'utiliser les propriétés organisationnelles, structurelles et topologiques du système (*critères externes*) pour contraindre le processus d'agrégation et pour engendrer des représentations viables sur les plans syntaxique et sémantique. Par conséquent, l'automatisation du processus d'agrégation nécessite

de résoudre un problème d'optimisation sous contraintes. Nous proposons dans cette thèse un algorithme de résolution *générique*, s'adaptant aux critères formulés par l'observateur. De plus, nous montrons que la complexité de ce problème d'optimisation dépend directement de ces critères.

L'approche macroscopique défendue dans cette thèse est évaluée sur deux classes de systèmes. Premièrement, le processus d'agrégation est appliqué à la visualisation d'applications parallèles de grande taille pour l'analyse de performance. Il permet de détecter les anomalies présentes à plusieurs niveaux de granularité dans les traces d'exécution et d'expliquer ces anomalies à partir des propriétés syntaxiques du système. Deuxièmement, le processus est appliqué à l'agrégation de données médiatiques pour l'analyse des relations internationales. L'agrégation géographique et temporelle de l'attention médiatique permet de définir des évènements macroscopiques pertinents sur le plan sémantique pour l'analyse du système international. Pour autant, nous pensons que l'approche et les outils présentés dans cette thèse peuvent être généralisés à de nombreux autres domaines d'application.

ABSTRACT

The analysis of large-scale systems faces *syntactic* and *semantic* difficulties: How to observe millions of distributed and asynchronous entities? How to interpret the disorder that results from the microscopic observation of such entities? How to produce and handle relevant abstractions for the systems' macroscopic analysis? Faced with the failure of the analytic approach, the concept of *epistemic emergence* – related to the nature of knowledge – allows us to define an alternative strategy. This strategy is motivated by the observation that scientific activity relies on *abstraction processes* that provide macroscopic descriptions to broach the systems' complexity.

This thesis is more specifically interested in the production of spatial and temporal abstractions through *data aggregation*. In order to generate scalable representations, the control of two essential aspects of the aggregation process is necessary. Firstly, the complexity and the information content of macroscopic representations should be jointly optimized in order to preserve the relevant details for the observer, while minimizing the cost of the analysis. We propose several measures of quality (*internal criteria*) to evaluate, compare and select the representations depending on the context and the objectives of the analysis. Secondly, in order to preserve their explanatory power, the generated abstractions should be consistent with the background knowledge exploited by the observer for the analysis. We propose to exploit the systems' organisational, structural and topological properties (*external criteria*) to constrain the aggregation process and to generate syntactically and semantically consistent representations. Consequently, the automation of the aggregation process requires solving a constrained optimization problem. We propose a generic algorithm that adapts to the criteria expressed by the observer. Furthermore, we show that the complexity of this optimization

problem directly depend on these criteria.

The macroscopic approach supported by this thesis is evaluated on two classes of systems. Firstly, the aggregation process is applied to the visualisation of large-scale distributed applications for performance analysis. It allows the detection of anomalies at several scales in the execution traces and the explanation of these anomalies according to the system syntactic properties. Secondly, the process is applied to the aggregation of news for the analysis of international relations. The geographical and temporal aggregation of media attention allows the definition of semantically consistent macroscopic events for the analysis of the international system. Furthermore, we believe that the approach and the tools presented in this thesis can be extended to a wider class of application domains.

TABLE DES MATIÈRES

Remerciements **iii**

Résumé **v**

Abstract **vii**

Table des matières **ix**

Table des figures **xiii**

Table des notations **xvii**

1 De la nécessité de l'approche macroscopique **1**
1.1 Difficultés et enjeux 1
1.2 Problématique et contributions 3
1.3 Organisation du manuscrit 3

I Représentation macroscopique des systèmes **7**

2 Enjeux méthodologiques liés à l'analyse des grands systèmes **9**
2.1 Difficultés syntaxiques et difficultés sémantiques 10
2.2 L'émergence épistémique comme modèle philosophique 12
2.2.1 Les enjeux philosophiques de l'émergentisme 13
2.2.2 Les conséquences méthodologiques de l'émergentisme . 16
2.3 Bilan de l'approche macroscopique 18

3 L'agrégation de données comme processus d'abstraction 21
3.1 État de l'art des techniques d'abstraction 22
3.1.1 Abstraction par définition de concepts 22
3.1.2 Abstraction par définition d'objets 23
3.1.3 Partitionnement, agrégation et interprétation 24
3.2 Formaliser le processus d'agrégation 27
3.2.1 Représentation microscopique 27
3.2.2 Partitionnement et agrégation de données 29
3.2.3 Interprétation des données agrégées 31
3.2.4 Structure algébrique de l'ensemble des partitions 33

II Le rôle de l'observateur 37

4 Évaluer et contrôler le processus d'agrégation 39
4.1 Passage à l'échelle et interprétation des données agrégées . . . 40
4.1.1 Réduire la complexité pour passer à l'échelle 40
4.1.2 Contrôler la perte d'information pour interpréter les données 42
4.1.3 Réaliser un compromis entre réduction de complexité et perte d'information 43
4.2 Définition générique des mesures de qualité 44
4.2.1 Problème des partitions optimales 44
4.2.2 Mesures de qualité monotones 45
4.2.3 Mesures de qualité décomposables 46
4.3 Définition de mesures de qualité spécifiques 47
4.3.1 La taille des représentations comme mesure de complexité 47
4.3.2 La divergence comme perte d'information 50
4.3.3 Réaliser un compromis entre réduction de complexité et perte d'information 53
4.4 Bilan et perspectives 57

5 Des abstractions cohérentes avec les connaissances externes 61
5.1 Prendre en compte les connaissances externes lors de l'agrégation 62
5.2 Organisation hiérarchique des systèmes 64
5.2.1 Exemples d'organisations hiérarchiques 65
5.2.2 Parties admissibles selon une hiérarchie 66
5.2.3 Partitions admissibles selon une hiérarchie 68
5.3 Organisation ordonnée des systèmes 69
5.3.1 Parties admissibles selon un ordre total 69
5.3.2 Partitions admissibles selon un ordre total 69

5.4 Agrégation multidimensionnelle 70
5.5 Bilan et perspectives 72

6 Calculer et choisir les meilleures représentations 77
6.1 Le problème des partitions admissibles optimales 78
6.2 Un algorithme de résolution efficace 81
6.2.1 Décomposition par la relation de couverture 81
6.2.2 Principe d'optimalité et calcul récursif 84
6.2.3 Algorithme des partitions admissibles optimales 86
6.2.4 Optimisation de l'algorithme 88
6.3 Implémentations spécialisées de l'algorithme 92
6.3.1 Algorithme des partitions hiérarchiques optimales . . . 93
6.3.2 Algorithme des partitions ordonnées optimales 95
6.4 Bilan et perspectives 97

III Agrégation et analyse de grands systèmes 99

7 Agrégation de traces pour la visualisation de performance 101
7.1 Enjeux de l'agrégation pour la visualisation de performance . . 102
7.2 Agrégation hiérarchique des traces d'exécution 105
7.2.1 Applications et traces analysées 105
7.2.2 Organisation hiérarchique de la plate-forme 106
7.2.3 Outils de visualisation et représentations treemap . . . 107
7.2.4 Mesures de qualité 109
7.3 Détection d'anomalies dans les traces d'exécution 110
7.3.1 Engorgement dans le réseau de communication 110
7.3.2 Détection d'anomalies multi-niveau 115
7.4 Visualiser un million de processus 116
7.5 Bilan et perspectives 119

8 Agrégation de données médiatiques pour l'analyse des relations internationales 125
8.1 Exploiter les flux d'information médiatique pour l'analyse du système international 126
8.1.1 Analyse médiatique des relations internationales 126
8.1.2 La notion d'évènement médiatique 128
8.2 Agrégation spatio-temporelle de l'attention médiatique 129
8.2.1 Représentation microscopique de l'attention médiatique 129
8.2.2 Détection d'évènements médiatiques 132
8.2.3 Hypothèse de répartition non-uniforme 138

8.2.4 Sémantique des abstractions engendrées 140
8.3 Détection d'évènements médiatiques par agrégation 141
8.3.1 Agrégation et abstractions géographiques 141
8.3.2 Agrégation ordonnée et abstractions temporelles 145
8.4 Bilan et perspectives 149

9 Bilan et perspectives 155
9.1 Contributions, limites et généralisation 155
9.2 Perspectives de recherche 158

Bibliographie 163

Annexes 177

A Agrégation de données et théorie de l'information 177

B Algorithmes et complexité 183
B.1 Algorithme des partitions hiérarchiques optimales 184
B.2 Algorithme des partitions ordonnées optimales 190

C Hiérarchie WUTS 197

TABLE DES FIGURES

3.1 Partitionnement, agrégation et interprétation d'une population de 6 individus selon une partition de 3 parties . . . 32
3.2 Diagramme de Hasse de l'ensemble des partitions possibles de la population $\{a, b, c, d\}$ ordonné selon la relation de couverture 35

4.1 Agrégation d'une population de 6 individus selon 3 partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$ 41
4.2 Évaluation des partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$: tableau des qualités normalisées 55
4.3 Comparaison des partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$: compromis de qualité linéaire CQL_α en fonction du coefficient α spécifié par l'observateur 56
4.4 Sélection des partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$: qualité normalisée des partitions optimales en fonction du coefficient α spécifié par l'observateur 56
4.5 Caractériser une distribution à partir du graphe de qualité associé à son agrégation 60

5.1 Hiérarchie à 3 niveaux : niveau des individus ($\{a\}$, $\{b\}$, $\{c\}$, $\{d\}$ et $\{e\}$), niveau des parties (A et B) et ensemble de tous les individus (Ω) 66
5.2 Arbre représentant une hiérarchie à 3 niveaux 67
5.3 Trois partitions hiérarchiques $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$ représentées par des « coupes » dans un arbre et par des « boîtes disjointes » . . . 67
5.4 Diagramme de Hasse de l'ensemble des partitions admissibles ordonné selon la relation de couverture 68

5.5 Deux partitions ordonnées $\mathcal{A}$ et $\mathcal{B}$ représentées par des « pyramides d'intervalles » 70
5.6 Agrégation d'une population bi-dimensionnelle composée d'une population hiérarchique $\{a, b, c, d, e\}$ et d'une population ordonnée $\{1, 2, 3, 4\}$ 72
5.7 Diagramme d'évènements (extrait de [Mat89]) 73
5.8 Graphe de voisinage défini sur une population de 13 individus 74
5.9 Graphe pondéré défini sur une population de 5 individus . . . 75

6.1 Décomposition de l'espace de recherche en fonction des partitions couvertes par la partition macroscopique 82
6.2 Trace d'exécution de l'algorithme des partitions admissibles optimales dans le cas d'une population ordonnée de taille 4 . . 87
6.3 Trace d'exécution de l'algorithme stockant les résultats intermédiaires pour éviter des appels récursifs 88
6.4 Trace d'exécution de l'algorithme stockant les partitions évaluées pour éviter les évaluations redondantes 90
6.5 Trace d'exécution de l'algorithme des partitions admissibles optimales dans le cas d'une population hiérarchique de taille 5 93

7.1 Graphe des ressources de la plate-forme GRID'5000 visualisée au niveau des machines (*hosts*), des clusters et des sites (extrait de [SLV13]) 102
7.2 Visualisation de la disponibilité d'un client BOINC au cours du temps (extrait de [SLV12]) 103
7.3 Représentation treemap d'une application exécutée sur la plate-forme GRID'5000 au niveau des machines (*hosts*), des clusters et des sites (extrait de [SHN12]) 108
7.4 Treemap microscopique (188 processus visualisés) 110
7.5 Treemap entièrement agrégée (1 valeur visualisée) 111
7.6 Treemap au niveau des sites (5 sites visualisés) 112
7.7 Trois interprétations possibles d'une valeur agrégée au niveau d'un site 112
7.8 Graphe de qualité des treemaps optimales en fonction du coefficient de compromis α 113
7.9 Treemap optimale préservant au moins 99% de l'information microscopique 113
7.10 Treemap microscopique (434 processus visualisés) 114
7.11 Treemap optimale préservant au moins 99% de l'information microscopique 115
7.12 Treemap au niveau des machines (10 000 machines visualisées) 116

7.13 Treemap optimale préservant au moins 95% de l'information microscopique 117
7.14 Treemaps donnant les valeurs de 4 attributs (extrait de [SHN12]) 121
7.15 Agrégation temporelle de la trace d'exécution d'un processus 122
7.16 Implémentation de l'algorithme des partitions ordonnées optimales au sein de l'outil OCELOTL 123

8.1 Proportion des articles publiés par un journal burkinabé et un journal canadien parlant des différents pays du monde (extrait de [G+11]) 127
8.2 Quantité d'articles parlant de l'Islande au cours du temps (base FACTIVA, extrait de [GGS11]) 128
8.3 Variation temporelle de l'attention géographique relative du *Guardian* concernant la Grèce (niveau hebdomadaire) 134
8.4 Variation géographique de l'attention temporelle relative du *Guardian* pendant le mois de juillet 2011 135
8.5 Zoom sur l'attention temporelle relative du *Guardian* concernant les pays africains 136
8.6 Zoom sur l'attention temporelle relative du *Guardian* concernant les pays européens 137
8.7 Graphe de qualité correspondant à l'agrégation de la carte de citations de la figure 8.4 141
8.8 Partition géographique optimale préservant au moins 50% de l'information microscopique 142
8.9 Partition géographique optimale préservant au moins 70% de l'information microscopique 144
8.10 Graphe de qualité correspondant à l'agrégation de la série de citations de la figure 8.11 145
8.11 Représentation microscopique de l'attention géographique relative du *Guardian* concernant la Grèce 146
8.12 Partition temporelle optimale préservant au moins 80% de l'information microscopique 147
8.13 Partition temporelle optimale préservant au moins 50% de l'information microscopique 148
8.14 Graphe de co-citations et graphe de domination issus de la matrice des co-citations (extrait de [GGLP+13]) 151
8.15 Variation temporelle de l'attention géographique relative de 4 flux concernant la Syrie (extrait de [GGLP+13]) 153

9.1 Agrégation d'une population de 6 individus selon un ensemble de parties non-disjointes et non-recouvrantes 159
9.2 Observation macroscopique des systèmes multi-agents (extrait de [LP10]) 160

B.1 Temps d'exécution de l'algorithme des partitions hiérarchiques optimales 189
B.2 Temps d'exécution de l'algorithme des partitions ordonnées optimales 195

C.1 Deuxième niveau de la hiérarchie WUTS composé de 36 micro-régions (extrait de [GD07]) 198
C.2 Troisième niveau de la hiérarchie WUTS composé de 17 meso-régions (extrait de [GD07]) 199
C.3 Quatrième niveau de la hiérarchie WUTS composé de 7 macro-régions (extrait de [GD07]) 200
C.4 Cinquième niveau de la hiérarchie WUTS composé de seulement 3 régions (extrait de [GD07]) 201

TABLE DES NOTATIONS

x,y,z	Individus (objets microscopiques)	28
Ω	Population (ensemble d'individus)	28
X,Y,Z	Parties de la population Ω (sous-ensembles d'individus)	29
$\mathcal{P}(\Omega)$	Ensemble des parties de la population Ω	29
$\mathcal{P}_a(\Omega)$	Ensemble des parties *admissibles* de la population Ω	63
$\mathcal{T}(\Omega)$	Hiérarchie de parties définie sur la population Ω	66
$\mathcal{X}$,$\mathcal{Y}$,$\mathcal{Z}$	Partitions de la population Ω	31
$\mathcal{P}_\perp(\Omega)$	Partition *microscopique* de la population Ω	29
$\mathcal{P}_\top(\Omega)$	Partition *macroscopique* de la population Ω	31
$\mathfrak{P}(\Omega)$	Ensemble des partitions de la population Ω	31
$\mathfrak{P}_a(\Omega)$	Ensemble des partitions *admissibles* de la population Ω	63
$\mathfrak{P}_\mathcal{T}(\Omega)$	Ensemble des partitions *admissibles* selon la hiérarchie $\mathcal{T}(\Omega)$	68
$\mathfrak{P}_<(\Omega)$	Ensemble des partitions *admissibles* selon l'ordre total $<$	69
$\mathfrak{P}^m(\Omega)$	Ensemble des partitions *optimales* selon la mesure de qualité m	45
$<$	Relation de raffinement entre deux partitions	33
$\prec$	Relation de couverture entre deux partitions	34

$\mathfrak{R}(\mathcal{X})$ Ensemble des partitions *raffinant* la partition $\mathcal{X}$ 33

$\mathfrak{C}(\mathcal{X})$ Ensemble des partitions *couvertes* par la partition $\mathcal{X}$ 34

$v(.)$ Attribut de dénombrement (quantité d'unités atomiques) 28

$v(x)$ Valeur de l'attribut $v(.)$ pour l'individu x 28

$v(\Omega)$ Valeur *agrégée* de l'attribut $v(.)$ pour la population Ω 30

$v(X)$ Valeur *agrégée* de l'attribut $v(.)$ pour la partie X 30

$v_{\mathcal{X}}(x)$ Valeur *redistribuée* de l'attribut $v(.)$ selon la partition $\mathcal{X}$ 32

$v(\mathcal{P}_{\perp})$ Représentation *microscopique* de l'attribut $v(.)$ 29

$v(\mathcal{X})$ Représentation *agrégée* de l'attribut $v(.)$ selon la partition $\mathcal{X}$ 31

$v_{\mathcal{X}}(\mathcal{P}_{\perp})$ Représentation *redistribuée* selon la partition $\mathcal{X}$ 32

$p(x)$ Probabilité d'apparition de l'individu x 28

$p(\mathcal{P}_{\perp})$ Distribution de probabilité *microscopique* 29

$p(\mathcal{X})$ Distribution de probabilité *agrégée* selon la partition $\mathcal{X}$ 31

m Mesure de qualité définie sur l'ensemble des partitions 44

C Mesure de complexité 47

T Taille des représentations 49

ΔC Réduction de complexité 48

ΔT Réduction de taille 49

L Perte d'information 50

D Divergence de Kullback-Leibler 52

CQ Compromis de qualité 53

CQL_{α} Compromis de qualité linéaire 54

α Coefficient de compromis linéaire 54

CHAPITRE 1

De la nécessité de l'approche macroscopique

> « Aujourd'hui, nous sommes confrontés à un autre infini : l'infiniment complexe. Mais cette fois, plus d'instrument. »
>
> Joël de Rosnay, *Le macroscope*

Expliquer le fonctionnement des sociétés humaines ; comprendre celui d'un organisme aux parties complexes, de par leur structure, de par leurs fonctions ou de par leurs interactions ; analyser le comportement de systèmes de calcul gigantesques, faisant intervenir des millions de processus distribués et exécutant chacun, de manière asynchrone, des millions d'instructions par seconde. Les sciences modernes se doivent de répondre à un problème épistémologique majeur : *comment aborder les systèmes qui nous entourent et qui nous constituent ?*

1.1 Difficultés et enjeux

Les systèmes sociaux, biologiques et physiques, qui font l'objet des sciences en général, sont extrêmement complexes sur le plan structurel ou sur le plan fonctionnel. À ces systèmes s'ajoutent ceux, construits par l'homme, destinés à produire, à manipuler et à transmettre de grandes quantités d'information. La notion de *système* désigne donc, de manière transversale, l'ensemble des objets que les sciences modernes abordent aujourd'hui comme des assemblages d'entités organisées et interagissantes, et qui présentent dès lors

une grande complexité. L'*analyse* de tels systèmes désigne – au sens large – l'activité scientifique visant à leur compréhension : il s'agit d'identifier leur structure et leur organisation, de caractériser le rôle et la fonction de chacune des entités qui les composent, de décrire leurs comportements, leurs interactions avec l'environnement et d'*expliquer* les phénomènes observés en rendant explicites leurs causes et leurs effets. La connaissance des chaînes causales œuvrant au sein des systèmes débouche, enfin, sur des perspectives pragmatiques. Pour les systèmes artificiels, il s'agit de contrôler l'activité et d'améliorer les performances. Pour les systèmes naturels, la compréhension et l'explication des phénomènes visent à améliorer notre condition biologique et sociale.

Force est de constater que, dans le cas de systèmes de *grande taille*, comprenant plusieurs milliards d'entités et dont les « chaînes causales » font intervenir autant d'évènements, la méthode *analytique*, consistant – au sens strict – à étudier un système à partir de ses constituants, échoue à satisfaire ces objectifs. Tout d'abord, afin d'aborder chacune de ces parties, l'approche analytique nécessite des outils d'observation égaux en taille aux systèmes observés. De tels instruments doivent en effet décrire le comportement de chacune de ces entités, potentiellement distribuées dans l'espace, et de chacune de leurs interactions, constituant la temporalité du système. De plus, une analyse des systèmes reposant sur le résultat d'une telle observation microscopique est elle-même extrêmement complexe. Le scientifique est rapidement dépassé par la quantité de paramètres à prendre en compte et la quantité des phénomènes à expliquer. Dans le cas de simulations informatiques ou de procédures d'analyse automatisées, la représentation microscopique est coûteuse à manipuler et compromet ainsi le « passage à l'échelle ».

L'analyse des dynamiques microscopiques ne permet donc pas, en pratique, la compréhension des grands systèmes. Il est dès lors nécessaire de trouver une alternative à l'approche analytique. Le « macroscope » est un outil symbolique imaginé par Joël de Rosnay pour appréhender la complexité des systèmes [dR75]. Il participe notamment à l'approche *systémique*, consistant à étudier un système « comme un tout ». La première mission du « macroscope » est de distinguer ce qui est important de ce qui est de l'ordre du détail. Plus généralement, cette approche vise à fournir aux experts (en sciences sociales, en sciences du vivant et en sciences informatiques) un point de vue *macroscopique* sur les systèmes qu'ils étudient. C'est ce que nous appelons l'*analyse macroscopique des grands systèmes*.

1.2 Problématique et contributions

Cette thèse propose des outils conceptuels et méthodologiques pour l'édification d'un tel point de vue macroscopique. Elle vise ainsi à favoriser le passage à l'échelle des techniques d'analyse. La méthode proposée repose sur l'hypothèse suivante : nous disposons de moyens d'observation microscopique, c'est-à-dire d'instruments d'observation capables de représenter le comportement et les interactions des entités constituant le système. Le problème est alors le suivant : *comment mettre en évidence les phénomènes macroscopiques à partir des résultats de l'observation microscopique ?* Nous nous intéressons donc à la conception d'un « macroscope » à partir de « microscopes ». Le relâchement de cette hypothèse sera abordé en perspective de cette thèse.

Vis-à-vis de la problématique exposée ci-dessus, cette thèse défend que :

1. Les phénomènes macroscopiques sont le résultat d'un *processus d'abstraction*. Ils correspondent ainsi à des descriptions macroscopiques du système, réductibles *en principe* au niveau de description microscopique, mais nécessaires *en pratique* à l'analyse des grands systèmes. En particulier, les abstractions spatio-temporelles peuvent être engendrées par des techniques d'*agrégation*.
2. Les processus d'abstraction reposent sur des mécaniques épistémiques complexes, au sein desquelles l'observateur a une place prépondérante. Lors de l'édification du point de vue macroscopique, il est notamment essentiel de prendre en compte le contexte de l'analyse et les connaissances mobilisées par les experts pour comprendre, interpréter et expliquer les phénomènes.
3. Les processus d'abstraction sont alors capables d'engendrer, en pratique, des phénomènes pertinents pour l'analyse macroscopique d'une large classe de systèmes. Ils permettent ainsi de relever – au moins en partie – les défis du « macroscope ».

1.3 Organisation du manuscrit

Les trois parties de cette thèse sont chacune consacrée à l'un des trois points exposés ci-dessus. La **première partie** présente la notion d'abstraction et le processus d'agrégation.

Le **chapitre 2** expose plus en détail les difficultés, d'ordre syntaxique et sémantique, liées à l'analyse des grands systèmes, à travers deux exemples : l'analyse des systèmes de calculs, développés en informatique, et l'analyse

des systèmes sociaux. Afin de dépasser ces difficultés, il apparaît nécessaire de préciser la notion de *phénomène macroscopique*. Le concept d'*émergence épistémique*, issu de la philosophie de la connaissance, permet de définir un cadre méthodologique adéquat : les phénomènes macroscopiques sont le résultat d'un processus d'abstraction *subjectif* (ils sont « dans l'œil de l'observateur ») et *pragmatique* (ils dépendent du contexte et des objectifs de l'analyse) visant à réduire la complexité des systèmes observés.

Parmi les techniques d'abstraction développées en informatique, le **chapitre 3** identifie et formalise une technique cohérente avec la position émergentiste. Cette technique repose sur l'*agrégation* des objets issus de l'observation microscopique dans le but d'engendrer une *représentation macroscopique* du système. Le processus d'abstraction proposé consiste en trois étapes : (1) *partitionnement* des objets microscopiques, (2) *agrégation* des objets, visant à l'édification d'une sémantique macroscopique, et (3) *interprétation* des résultats par l'observateur.

Ces trois étapes de l'agrégation, au sein desquels l'observateur à une place prépondérante, induisent des problématiques de recherche particulières. La **deuxième partie** est consacrée à ces problématiques et constitue la contribution théorique de la thèse.

Le **chapitre 4** s'intéresse à l'exploitation des représentations macroscopiques pour l'analyse des grands systèmes. Il apparaît nécessaire, lors de l'agrégation, de contrôler la *complexité* des représentations, afin de réduire le coût de l'analyse lors du passage à l'échelle, et leur *contenu informationnel*, afin de s'assurer que l'observateur interprète correctement les abstractions engendrées et que celles-ci ne suppriment pas d'informations significatives pour l'analyse. Nous proposons des *mesures de qualité* permettant d'évaluer, de comparer et de sélectionner les représentations macroscopiques en fonction de ces critères et du contexte de l'analyse (approche pragmatiste).

Le **chapitre 5** se concentre sur la notion de *sémantique macroscopique* : les abstractions doivent être dotées d'une signification pour l'observateur et, notamment, être cohérentes avec les connaissances et les modèles experts mobilisés lors de l'analyse (approche subjectiviste). Nous proposons de formaliser ces critères *externes* – reposant sur des connaissances préliminaires à l'observation – en contraignant l'ensemble des représentations *admissibles* par l'observateur. Nous donnons deux exemples de contraintes, liées à l'organisation *topologique* des systèmes observés : les *hiérarchies constitutives* et les *relations d'ordre*, respectivement utilisées pour engendrer des abstractions *spatiales* et *temporelles* cohérentes avec les attentes de l'observateur.

Le **chapitre 6** résout un problème d'optimisation pour automatiser le processus d'agrégation : étant donnés une mesure de qualité (critère interne)

et un ensemble de représentations admissibles (critère externe), *quelles sont les « meilleures » représentations pour l'observateur ?* Nous proposons un algorithme de résolution efficace (en temps de calcul et en espace mémoire) reposant sur deux hypothèses concernant les propriétés algébriques des critères à optimiser : (1) les mesures à optimiser sont *additivement décomposables* et (2) les contraintes d'admissibilité « simplifient » l'espace de recherche. L'algorithme proposé a alors une complexité temporelle *linéaire*, dans le cas de contraintes hiérarchiques, et *quadratique*, dans le cas d'une relation d'ordre.

La **troisième partie** propose une évaluation expérimentale du processus d'agrégation. Elle montre qu'il permet d'engendrer *en pratique* des représentations macroscopiques pertinentes pour l'analyse des grands systèmes.

Le **chapitre 7** propose d'appliquer le processus d'abstraction à la visualisation d'applications distribuées de grande taille pour l'analyse de performance. Nous constatons que les techniques de visualisation utilisées dans ce domaine passent difficilement à l'échelle. L'algorithme d'agrégation est utilisé pour réduire la complexité des représentations spatiales de l'exécution, tout en conservant le maximum d'information sur le niveau microscopique (niveau des processus). Les abstractions engendrées permettent ainsi de détecter – à moindre coût – des anomalies dans les traces d'exécution et de les expliquer à partir des propriétés topologiques de la plate-forme. En outre, nous montrons que le processus d'abstraction passe à l'échelle en agrégeant un million de processus.

Le **chapitre 8** s'intéresse à un deuxième domaine d'application : l'analyse des relations internationales. Les marqueurs géographiques contenus dans les flux d'information médiatique (radio, presse, télévision) permettent de donner une représentation microscopique des relations économiques, culturelles et politiques entre les pays du monde à travers la notion d'*évènement médiatique*. Pour les géographes, ces évènements peuvent être abordés à plusieurs granularités spatiales et temporelles. L'agrégation permet alors d'engendrer des abstractions cohérentes avec les connaissances mobilisées par les experts, afin d'analyser le système international à différents niveaux de représentation.

Le **chapitre 9** conclut ce manuscrit par un bilan du travail accompli et quelques perspectives de recherche. En particulier, nous y abordons la question de l'*observation macroscopique*. Elle consiste à fournir une représentation macroscopique du système sans passer par l'observation complète et détaillée des entités qui le constituent. Cette perspective vise à dépasser le second problème de l'approche analytique en intégrant les techniques d'agrégation au sein même du système. Ainsi, le processus d'abstraction devient lui-même un phénomène émergent.

Première partie

Représentation macroscopique des systèmes

« [Le macroscope sert] à observer ce qui est *à la fois* trop grand, trop lent et trop complexe pour nos yeux. »

Joël de Rosnay, *Le macroscope*

CHAPITRE 2

Enjeux méthodologiques liés à l'analyse des grands systèmes

Cette thèse participe à l'approche systémique pour au moins deux raisons. La première est qu'elle vise à l'édification d'une « théorie générale des systèmes » [vB69], c'est-à-dire une approche adaptée à la compréhension de *tous types de systèmes*, indépendamment du domaine scientifique de référence. Ainsi, elle embrasse à la fois – et parmi d'autres – les systèmes physiques (*e.g.*, systèmes gazeux, systèmes planétaires), les systèmes biologiques (*e.g.*, cellules, organes, organismes) et les systèmes sociaux (*e.g.*, intelligence collective, interactions sociales et sociétés humaines). La notion de système sert donc d'abstraction pour identifier des problématiques transversales et pour développer des méthodes de résolutions génériques. La section 2.1 présente de telles problématiques selon deux axes, liés à la *syntaxe* et à la *sémantique* des systèmes observés.

Le second point de rencontre entre cette thèse et l'approche systémique concerne le rejet de la démarche analytique (*cf.* chapitre précédent). Nous pensons en effet que la compréhension d'un système ne peut être le résultat d'un examen séparé des parties qui le constituent. S'oppose à cette démarche réductionniste une approche globale, parfois nommée *holisme*, qui consiste à aborder les systèmes comme des ensembles indivisibles, comme des « touts » cohérents dont une analyse parcellaire ne peut expliquer les dynamiques. Les sections 2.2 et 2.3 précisent cette approche que nous empruntons pour pallier les limites de la démarche analytique et pour répondre aux difficultés mises en évidence dans la section 2.1. Cette approche est fondée sur la notion d'*émergence épistémique*.

2.1 Difficultés syntaxiques et difficultés sémantiques

Parmi les difficultés relatives à l'analyse des grands systèmes, nous distinguons deux catégories : les difficultés liées à la *syntaxe* du système et celles liées à sa *sémantique*. Nous utilisons les termes « syntaxe » et « sémantique » sur la base d'une analogie avec la distinction linguistique classique entre la « forme » et le « fond ». La syntaxe est la branche de la linguistique chargée de l'étude des relations formelles entre les mots constituant un énoncé. La sémantique s'intéresse à la signification de l'énoncé, c'est-à-dire à la manière dont il est interprété.

Définition 2.1. Dans le cas de l'analyse des systèmes, les difficultés *syntaxiques* sont liées à la structure physique du système observé : agencement des entités, relations qu'elles entretiennent et structure globale de l'édifice.

Les difficultés *sémantiques* sont liées à la signification de ces entités et de ces relations pour l'observateur, c'est-à-dire à leur interprétation au sein d'un cadre d'expertise particulier.

Un système est *complexe sur le plan syntaxique* lorsque la compréhension de sa structure, et des relations entre les entités, nécessite un effort – plus ou moins important – de la part de l'observateur. Un système est *complexe sur le plan sémantique* lorsque les entités et les relations ont une signification non-triviale vis-à-vis d'un contexte épistémique donné. Pour reprendre l'analogie linguistique, la complexité syntaxique d'un énoncé dépend de l'agencement des mots et sa complexité sémantique du sens qu'ils véhiculent, l'une et l'autre pouvant être éventuellement corrélées. Quoi qu'il en soit, ces difficultés sont abordables pour des systèmes de petite taille. Un énoncé de vingt-six mots – prenez celui-ci – ne résistera jamais longtemps à l'analyse linguistique, quelque complexes que soient sa syntaxe ou sa sémantique. En revanche, il est extrêmement difficile d'analyser un énoncé de mille mots lorsque leur agencement ou leur signification n'est pas triviale. L'objectif de ce chapitre consiste à définir une approche adaptée à l'analyse des systèmes à la fois complexes et de grande taille. Le critère de réussite est donc la réduction des difficultés syntaxiques et sémantiques de l'analyse.

La plupart des systèmes sont à la fois complexes sur le plan syntaxique et sur le plan sémantique. Pour un organisme vivant, par exemple, le jeu des interactions chimiques entre les organes et leur interprétation en termes

fonctionnels par la biologie sont extrêmement complexes. Cependant, afin d'illustrer notre propos, nous nous intéressons à deux classes de systèmes dont les difficultés relèvent plutôt de l'une ou de l'autre de ces deux catégories : les *systèmes de calcul* et les *systèmes sociaux*. Ils participeront, dans la troisième partie, à l'évaluation de notre approche en montrant qu'elle parvient à résoudre certaines des difficultés syntaxiques et sémantiques liées à l'analyse des grands systèmes. Nous nous intéresserons en particulier à la détection des irrégularités au sein de ces systèmes et à leur explication à partir de leurs propriétés syntaxiques et sémantiques. Nous reviendrons en conclusion (chapitre 9) sur la possibilité de généraliser cette approche à une classe plus large de systèmes.

Difficultés syntaxiques liées à l'analyse des systèmes de calcul. Les systèmes informatiques développés ces dernières années mettent en présence une quantité inégalée de ressources de calcul. Le supercalculateur en tête de la liste *Top500* de juin 2013 coordonne par exemple plus de trois millions de cœurs [1]. Le projet de calcul distribué FOLDING@HOME a exploité 5 780 000 processeurs différents depuis sa création, répartis sur les machines de plusieurs millions de participants [2]. Le protocole de transfert de données pair-à-pair BITTORRENT met en réseau plus de 150 millions d'utilisateurs partageant chaque jour d'énormes quantités de données [3].

La complexité de tels systèmes est moins liée à l'interprétation de l'exécution – relativement bien formalisée par le domaine – qu'à leur structure distribuée. Premièrement, la décentralisation des processus de calcul rend difficile tout aperçu de l'état global du système [CL85, Mat89]. De plus, l'asynchronisme des comportements rend la représentation du temps extrêmement délicate [CMV01]. En pratique, la représentation du système se limite donc à une description des exécutions locales, pour chaque ressource, c'est-à-dire une représentation *microscopique* du système. Le grand nombre de ressources, d'évènements et d'interactions fait qu'une telle représentation microscopique est extrêmement coûteuse à visualiser et *a fortiori* à analyser. Afin de passer à l'échelle, les techniques d'analyse doivent donc résoudre les difficultés syntaxiques suivantes : comment *observer* l'activité de plusieurs millions d'entités distribuées et asynchrones ? Comment *représenter* les données issues de l'observation de manière à visualiser l'exécution tout

[1] Supercalculateur TIANHE-2 MILKYWAY construit par la *National University of Defense Technology* de Changsha, en Chine, voir http://www.top500.org/lists/2013/06/.

[2] Voir http://fah-web.stanford.edu/cgi-bin/main.py?qtype=osstats2 (consulté le 26 juin 2013).

[3] Selon la société *BitTorrent Inc.* elle-même, le 9 juin 2012. Voir http://www.bittorrent.com/intl/fr/company/about/ces_2012_150m_users.

en représentant la structure globale du système ? Comment procéder, enfin, à l'*analyse* détaillée d'une telle exécution ?

Difficultés sémantiques liées à l'analyse des systèmes sociaux. Nous entendons par « système social » tout ensemble d'individus interdépendants (humains ou non) qui interagissent et s'organisent à partir de normes ou de rôles [BB04]. Une *ville*, par exemple, est un système social constitué d'habitants, d'institutions, d'entreprises, de services publics, *etc.* [PRT06] Une colonie de fourmis [DF94], un réseau social sur Internet, une famille, le marché de l'éducation [BB04], en sont d'autres exemples. Même s'ils ne sont pas toujours formalisés de manière explicite, ni même définis sans ambiguïté, les termes introduits ici (individus, communautés, normes, institutions, *etc.*) constituent la base conceptuelle des sciences sociales [BB04]. L'analyse des systèmes sociaux consiste donc à exploiter ce vocabulaire pour décrire et expliquer les phénomènes qui y sont observés.

Une difficulté sémantique majeure concernant l'utilisation de ce vocabulaire réside dans le fait que les entités, les comportements et les relations ainsi désignés sont de natures différentes et, dès lors, s'inscrivent chacun dans un domaine spécialisé de la sociologie. Par exemple, le comportement d'un individu au sein d'une famille, d'une entreprise au sein d'une ville ou d'un État au sein du système international, sera respectivement expliqué par la sociologie de la famille, par la sociologie urbaine ou par le domaine des Relations internationales (de même qu'une conversation entre un père et une fille, un contrat de vente entre deux entreprises et des négociations diplomatiques entre deux États). L'analyse macroscopique doit donc aborder les difficultés sémantiques suivantes : comment observer ces entités de nature et de granularité différentes ? Comment rendre compte des relations sociologiques entre les entités de deux domaines spécialisés ? Comment expliquer l'organisation globale du système social à partir de ses parties hétérogènes ?

2.2 L'émergence épistémique comme modèle philosophique

Cette section vise à définir un cadre d'analyse pertinent pour pallier les difficultés syntaxiques et sémantiques présentées ci-dessus. Pour ce faire, elle montre que le concept d'*émergence épistémique* est adéquat. Celui-ci consiste à décrire les phénomènes macroscopiques comme des *abstractions*, réductibles *en principe* aux phénomènes microscopiques, mais néanmoins nécessaires *en pratique* pour aborder la complexité des grands systèmes.

L'argumentation consiste en deux temps. Nous exposons d'abord les origines *philosophiques* de la position émergentiste (2.2.1). Dans un second temps, nous en tirons des conséquences *méthodologiques* (2.2.2). Ainsi, nous ne prétendons pas argumenter en faveur ou en défaveur des différentes positions philosophiques abordées dans cette section. Notre objectif est de tirer de ce débat *philosophique* des principes *méthodologiques* pour l'analyse des grands systèmes[4]. Il ne s'agit pas de formuler des objections catégoriques à l'encontre de méthodes d'analyse, mais de donner des directions envisageables pour résoudre les problèmes qui nous intéressent. Les critères de validation du cadre d'analyse proposé sont donc d'ordre strictement méthodologique.

2.2.1 Les enjeux philosophiques de l'émergentisme

Il existe de nombreux travaux recensant les différentes acceptions de la notion d'émergence, aussi bien en philosophie [Ste99, OW06, Kis07] qu'en Intelligence Artificielle [BPMG05, DMD06, DFP08]. Nous nous intéressons ici à celle qui a été développée par la philosophie britannique, au tournant du XIXe siècle, dans le but de donner un cadre épistémique adéquat à l'étude des phénomènes macroscopiques. Dans ce contexte, la notion d'émergence n'est pas interprétée comme une propriété du *système* (*cf.* notions d'« adaptation » et d'« auto-organisation » [Pic04]), mais comme une propriété de l'*observateur*. La discussion repose donc sur la distinction suivante :

Définition 2.2. L'*ontologie* d'un système désigne tout ce qui est relatif au système lui-même, indépendamment de toute connaissance empirique.

L'*épistémie*[5] d'un système désigne tout ce qui est relatif à la connaissance que nous en avons. Elle dépend notamment d'un procédé d'observation.

Historiquement, un débat oppose deux positions philosophiques concernant l'explication des phénomènes biologiques [Kim99, OW06]. La science doit en effet rendre compte d'objets très différents sur le plan empirique : la matière inanimée et les êtres vivants. La question est de savoir si la vie

[4] Cette démarche participe ainsi à un travail plus général concernant les apports possibles entre la philosophie et l'Intelligence Artificielle [LP12].

[5] Dans cette thèse, le terme « épistémologie » désigne la branche de la philosophie chargée de l'étude des modes de connaissance du réel. Cette acception est à ce titre plus proche du terme anglais « *epistemology* » que du terme français désignant également, par extension, la philosophie des sciences. Nous utilisons donc le terme « épistémie » pour désigner ce qui est en rapport à la *connaissance* des systèmes, et non nécessairement à leur étude *scientifique*.

est une propriété propre aux êtres vivants (position *vitaliste*) ou si elle est simplement le résultat d'un agencement particulier de la matière inanimée (position *mécaniste*). En d'autres termes, il s'agit de savoir si cette propriété macroscopique a une existence en soi ou si elle est réductible à des propriétés microscopiques de la matière. La question de la *réduction* des phénomènes correspond à une discussion bien plus générale en philosophie [Kis07]. Elle oppose, de manière très schématique, deux positions concernant la structure de la réalité et la nature de la connaissance : le *dualisme* et le *monisme*.

Le coût ontologique du dualisme. La position *dualiste* fait l'hypothèse de plusieurs principes indépendants pour expliquer la diversité des phénomènes observés. Dans le cas particulier du vitalisme, les êtres vivants répondent à un principe de « force vitale » qui ne peut pas être entièrement expliqué par les principes gouvernant la matière inanimée. En d'autres termes, l'analyse des phénomènes macroscopiques (lois du vivant) ne peut être *réduite* à l'analyse de phénomènes microscopiques (lois physico-chimiques).

Dès lors, la réalité est constituées de plusieurs « couches » qui nécessitent le concours de *sciences spéciales* indépendantes (physique, chimie, biologie, sociologie, *etc.*). De ce fait, le dualisme met en cause l'unité de la science chère aux philosophes [Kis07]. On dit alors que le dualisme est *coûteux sur le plan ontologique* : il donne une explication *ad hoc* des phénomènes macroscopiques en multipliant les niveaux de réalité. Le principe de parcimonie, aussi connu sous le nom de « rasoir d'Ockham », invite au contraire à choisir une théorie explicative unifiée lorsque cela est possible.

La limite épistémique du monisme. La position *moniste* affirme que l'ensemble des phénomènes peut être expliqué à partir d'un seul niveau de réalité. Dans le cas particulier du mécanisme, par exemple, les phénomènes du vivant sont simplement le résultat des principes physiques sous-jacents. Dès lors, les sciences spéciales sont toutes *réductibles* à une science fondamentale (par exemple, la physique des particules).

Un reproche fréquemment adressé à l'encontre du monisme est qu'il conduit irrémédiablement à l'*élimination* des sciences spéciales [vdV97, Kis07]. En effet, dès lors que les lois de la biologie sont réductibles à celles de la physique, la biologie est démise de ses responsabilités épistémiques. À terme, tous les phénomènes doivent être expliqués par la science fondamentale. On dit alors que le monisme est *limité sur le plan épistémique* : il rend compte de tous les phénomènes selon un unique cadre épistémique qui ne parvient pas à mettre en évidence les différences fondamentales observées en pratique. Au contraire, de nombreux auteurs [Bed97, vdV97, Cha06] s'opposent au monisme *élimi-*

nativiste et considèrent que les phénomènes macroscopiques ont besoin d'un statut explicatif propre, notamment lorsqu'« une certaine batterie *explicative* s'est montrée sans efficacité véritable. » [vdV97]

Le recours au dualisme n'est pas une approche satisfaisante sur le plan explicatif dans la mesure où il ne rend pas compte des fondements microscopiques de ces phénomènes. Il est donc nécessaire de défendre une position à la fois économe sur le plan ontologique et efficace sur le plan épistémique. En d'autres termes, il est nécessaire de fonder un *monisme non-éliminativiste*[6].

La voie intermédiaire de l'émergentisme. Historiquement, la position *émergentiste* a été défendue pour trouver un compromis entre vitalisme et mécanisme [OW06] et, plus généralement, entre dualisme et monisme. L'émergentisme consiste à dissocier clairement la nature d'un phénomène (ontologie) et le cadre scientifique qui en fait l'analyse (épistémologie). Ainsi, les sciences spéciales fournissent des abstractions utiles à l'analyse des phénomènes macroscopiques, mais elles ne constituent en aucun cas un engagement quant à la structure de la réalité. En pratique, lorsque le scientifique est confronté à des phénomènes trop complexes pour en expliciter les causes microscopiques, il a recours à ces abstractions pour décrire et expliquer ces phénomènes à un niveau épistémique adéquat. Ainsi, sur le plan ontologique, les êtres vivants sont uniquement constitués de matière inanimée (étudiée par la physique), mais, sur le plan épistémique, ils peuvent être étudiés à un niveau de description plus adéquat (par la biologie). Dès lors, la distinction entre matière inanimée et êtres vivants n'est pas une différence objective. Elle est « dans l'œil du scientifique. »

On parle alors d'émergence *épistémique*. Il existe de nombreuses autres acceptions de l'émergence en philosophie, notamment l'émergence *ontologique*, défendue par certains philosophes britanniques (*e.g.*, Mill et Broad) et selon laquelle les phénomènes émergents sont irréductibles *en principe* [OW06]. Nous avons une approche plus contemporaine de l'émergence, notamment issue du philosophe Alexander, selon qui ces phénomènes sont seulement irréductible *en pratique* (voir [JC97, OW06, Kis07, Bed08] pour plus de détails concernant la distinction entre émergence épistémique et ontologique).

[6] Voir également les caractéristiques paradoxales de l'émergence énoncées par Bedau [Bed97, Bed08], selon qui les phénomènes émergents sont à la fois « *engendrés* par les processus sous-jacents » et « *autonomes* vis-à-vis de ces processus ». Dans notre cas, l'autonomie des phénomènes émergents est interprétée comme une autonomie *épistémique*, c'est-à-dire une autonomie de la batterie explicative employée pour leur analyse.

2.2.2 Conséquences méthodologiques de l'émergentisme

Afin de tirer des conséquences méthodologiques relatives à la modélisation des phénomènes macroscopiques en informatique, les termes du débat philosophique présenté ci-dessus sont adaptés sur la base d'analogies.

Application du dualisme et du monisme aux systèmes de calcul.

Définition 2.3. L'*ontologie d'un système de calcul* désigne tout ce qui est relatif à sa *conception*, c'est-à-dire en amont de son exécution : *e.g.*, spécifications, code, implémentation, plate-forme d'exécution.

L'*épistémie d'un système de calcul* désigne tout ce qui est relatif à son *analyse* : données recueillies au cours de l'exécution, traitement, visualisation et analyse de ces données[7].

Par analogie, une approche dualiste suppose que les systèmes de calcul comprennent au moins deux niveaux de conception. Ceci consiste par exemple à supposer l'existence de ressources macroscopiques chargées de centraliser, de synchroniser et d'agréger les informations produites localement par les processus de calcul[8]. L'objectif de l'approche dualiste est de donner ainsi un support physique aux phénomènes macroscopiques et de les exploiter pour réduire la complexité syntaxique du système. Cependant, la décentralisation et l'asynchronisme des ressources microscopiques (*cf.* section précédente) est incompatible avec l'intégration de telles ressources macroscopiques. La méthode dualiste ne peut donc être appliquée aux systèmes qui nous intéressent.

Du fait de la décentralisation et de l'asynchronisme, l'exécution a lieu uniquement au niveau microscopique. Cette constatation du moniste méthodologique ne doit pas, cependant, inciter aux mêmes travers que son analogue philosophique. Ainsi, l'*analyse* des phénomènes ne doit pas se limiter au niveau microscopique[9]. En effet, du fait de la taille et de la complexité syntaxique des systèmes, l'analyse complète et détaillée des ressources n'est pas envisageable en pratique. Il est donc nécessaire d'avoir recours à une

[7] Les raisons de cette analogie sont données plus en détail dans [LP12]. Elle repose essentiellement sur la distinction entre la notion d'algorithme vue comme un *objet mathématique* (ontologie) ou vue comme un *processus physique* (épistémie).

[8] Les « systèmes à tableaux noirs » font par exemple intervenir des espaces de mémoire partagée [Saw01]. Les modèles « multi-niveaux », dans le domaine de la simulation à base d'agents, maintiennent et synchronisent plusieurs niveaux d'exécution afin de modéliser le système à différentes échelles [GQLH12].

[9] C'est par exemple le cas des approches défendues dans [Dar94, Bed08], définissant les phénomènes émergents comme des propriétés globales du système, calculées à partir de la description complète des dynamiques microscopiques. Voir [LP12] pour plus de détails.

« science spéciale », c'est-à-dire un niveau de description plus abstrait pour expliquer les comportements. Selon l'approche émergentiste, les phénomènes macroscopiques apparaissent donc lors de l'étape d'analyse.

Application du dualisme et du monisme aux systèmes sociaux.

Définition 2.4. L'*ontologie d'un système social* désigne tout ce qui compose l'activité sociale (les entités, leurs comportements, leurs interactions) indépendamment de toute description ou analyse sociologique.

L'*épistémie d'un système social* désigne le résultat d'enquêtes sociologiques, les données démographiques et les modèles utilisés par les sociologues pour décrire et expliquer les phénomènes sociaux.

Une approche dualiste consiste à affirmer que les systèmes sociaux sont constitués d'entités de natures différentes, répondant à des lois hétérogènes : individus, entreprises, institutions, États, gouvernements, *etc.*[10] Dès lors, les phénomènes macroscopiques (*e.g.*, la crise économique mondiale de 2008) sont le fait d'entités de haut-niveau (marchés financiers, associations d'États, institutions internationales) dont le comportement ne peut pas être réduit au comportement des entités de plus bas niveau (les nations elles-mêmes, leurs institutions ou leurs populations). La complexité sémantique est directement expliquée par cette composition hétérogène des systèmes sociaux. Outre les problèmes ontologiques qu'elle soulève, l'hypothèse dualiste induit des problèmes d'ordre méthodologique. Il est notamment nécessaire de disposer d'outils d'observation et de méthodes d'analyse *ad hoc* pour chaque catégorie d'entités. Plus grave encore, les liens de causalité entre les phénomènes observés à différents niveaux du système sont difficilement explicités dans la mesure où chaque phénomène est régit par ses propres lois causales.

Afin de rendre compte du soubassement microscopique des phénomènes macroscopiques, il est nécessaire de définir un niveau de référence à partir duquel expliquer causalement les phénomènes observés[11]. Par exemple, dans

[10] C'est par exemple le cas de l'*holisme méthodologique*, largement exploité par la sociologie. « Dans cette perspective macroscopique, [...] chacun de ces touts est censé dépasser la seule somme de ses parties constituantes : au niveau d'organisation supérieur qui est le sien, il présente en effet des propriétés nouvelles irréductibles à celles de ces dernières et plus riches qu'elles en performances. » [Lau94], pages 12 et 13.

[11] Pour l'*individualisme méthodologique*, antagoniste de l'holisme, les phénomènes sociaux doivent être décrits et expliqués à partir des actions et des interactions des individus. Ce paradigme des sciences sociales est assez bien résumé par les principes énoncés par John Stuart Mill selon lesquels « les hommes ne se changent pas, quand ils sont rassemblés, en une autre espèce de substance douée de propriétés différentes. [...] Dans les phénomènes so-

le chapitre 8, nous choisirons le niveau des États comme niveau de référence pour l'analyse des phénomènes internationaux[12]. Pour autant, ce niveau de référence – préconisé par le monisme méthodologique – n'est pas suffisant pour rendre compte de la complexité sémantique des systèmes observés. Le recours aux abstractions – préconisé par l'émergentisme – est alors nécessaire, notamment dans le cas de grands systèmes. Il est cependant primordial de se souvenir que les entités et les relations macroscopiques ainsi définies n'en sont pas moins causalement liées au niveau de référence.

2.3 Bilan de l'approche macroscopique

En conclusion des discussions philosophiques et méthodologiques présentées dans ce chapitre, nous formulons deux principes résumant l'approche émergentiste pour l'analyse macroscopique des systèmes.

Monisme. Les systèmes doivent être abordés à partir d'un niveau de référence, composé d'entités de même nature et causalement responsables de l'ensemble des phénomènes que l'on souhaite analyser.

Non-éliminativisme. Dans le cas de grands systèmes, l'analyse ne doit pas se limiter au niveau de référence, mais proposer des abstractions adaptées à la complexité syntaxique et sémantique du niveau de référence.

Ces deux principes méthodologiques permettent de caractériser la notion de *phénomène macroscopique* dont voici les propriétés les plus importantes.

Épiphénoménisme. La notion d'émergence est parfois caractérisée par l'existence de *causalités descendantes* au sein des systèmes. Dans ce cas, les entités macroscopiques ont un pouvoir causal sur les entités microscopiques [Saw01, DMD06]. Une telle conception de l'émergence n'est pas compatible avec le monisme méthodologique que nous préconisons. De ce point de vue, les phénomènes macroscopiques sont des *épiphénomènes* : ils sont engendrés par les dynamiques microscopiques du système, mais n'ont en retour

ciaux, la composition des causes est la loi universelle. » (*Système de logique*, 1843) Citation extraite de [Lau94], page 29.

[12] Cette approche, donnant au concept d'« État » la primauté méthodologique pour l'étude des relations internationales, peut être qualifiée de *réaliste*, par opposition aux approches *idéalistes*, qui proposent de gérer et d'expliquer les relations internationales à partir de principes plus généraux (valeurs morales universelles, droit international, droits de l'homme, *etc.*) [Bat09].

aucun pouvoir causal sur le niveau microscopique. En ce sens, les phénomènes macroscopiques sont bien des abstractions, c'est-à-dire des *aspects* particuliers du niveau microscopique et non des *entités* réelles.

Approche ascendante. La « clôture causale » du niveau microscopique invite à une véritable *approche ascendante*. Sur le plan méthodologique, l'analyse doit reposer sur des données relatives au niveau de référence, afin de rendre compte des phénomènes dans leur intégralité, c'est-à-dire depuis leurs fondements microscopiques. Dans le chapitre suivant, nous présentons la notion d'*agrégation* permettant d'engendrer des abstractions à partir de la *représentation microscopique* du système, constituant le niveau de référence.

Subjectivisme. Les phénomènes macroscopiques dépendent d'un procédé d'observation et d'un cadre épistémique de référence (*cf.* notion de science spéciale). Par conséquent, le processus d'abstraction n'est pas neutre. Il est « dans l'œil de l'observateur. » En particulier, la sémantique des phénomènes macroscopiques doit être en adéquation avec les connaissances mobilisées par l'observateur pour analyser et expliquer ces phénomènes. Cet aspect *subjectif* des abstractions sera abordé plus en détail dans le chapitre 5.

Pragmatisme. L'activité du scientifique ne repose pas sur l'observation de phénomènes macroscopiques préexistants, mais sur un processus créatif d'abstraction. Le scientifique *fait-émerger* [VTR92] les phénomènes utiles à la compréhension macroscopique du système. Ainsi, il n'y a pas de « bonne » abstraction *per se*. Les phénomènes macroscopiques doivent donc être sélectionnés en fonction du contexte et des objectifs de l'analyse. Cette approche *pragmatiste* des abstractions sera abordée dans le chapitre 4.

Passage à l'échelle. L'objectif du processus d'abstraction est de donner une vue synthétique du système malgré sa complexité. Sur ce point, nous rejoignons l'acception de Bonabeau et Dessalles, identifiant l'émergence à une réduction de complexité réalisée par l'intermédiaire d'outils d'observation ou de description adéquats [BD97][13]. La notion de complexité est donc ici liée à la *représentation du système*, et non au *système lui-même*. Les phénomènes macroscopiques visent donc à réduire la complexité de la représentation microscopique dans le but de « passer à l'échelle », c'est-à-dire de pouvoir analyser de grands systèmes à partir de représentations simples, mais adéquates. Cette notion sera formalisée dans le chapitre 4.

[13] Voir [LPDV11a] pour plus de détails concernant les liens entre l'émergence épistémique et la définition de Bonabeau et Dessales.

CHAPITRE 3

L'agrégation de données comme processus d'abstraction

Selon la position émergentiste, les phénomènes macroscopiques sont des abstractions. En ce sens, il s'agit de représentations réductibles au niveau de description microscopique, mais néanmoins nécessaires pour aborder la complexité syntaxique et sémantique des grands systèmes. L'objectif de cette thèse consiste à mettre en place des processus d'abstraction automatisés répondant aux enjeux énoncés dans le chapitre précédent : *engendrer un point de vue macroscopique à partir de données microscopiques.* Ce chapitre présente et discute les différentes techniques d'abstraction développées en informatique. Parmi elles, l'*agrégation* satisfait pleinement les critères énoncés.

La section 3.1 définit l'agrégation comme un processus d'abstraction visant à la constitution d'*objets macroscopiques* en trois temps : (1) une étape préliminaire de partitionnement, (2) une étape d'agrégation et (3) une étape d'interprétation des résultats. Du fait des points (2) et (3), visant à une réelle sémantique macroscopique, l'agrégation consiste notamment à enrichir les techniques de partitionnement classiques. Elle induit donc des critères de partitionnement (internes et externes) particuliers. La section 3.2 formalise le processus d'agrégation : celui-ci consiste à modéliser une distribution de probabilité, définie sur un ensemble d'individus, à partir (1) d'une partition de ces individus, (2) d'un opérateur d'agrégation et (3) d'une hypothèse de redistribution permettant d'interpréter les résultats. Le processus d'agrégation consiste donc à trouver dans l'ensemble des partitions possibles celles qui optimisent les critères de partitionnement exposés dans les chapitres suivants.

3.1 État de l'art des techniques d'abstraction

Dans cette section, nous utilisons une décomposition classique des techniques d'abstraction [SS77] : celles qui visent à la définition de *concepts macroscopiques*, regroupant les objets microscopiques par catégories sémantiques en fonction des propriétés qu'ils partagent (3.1.1), et celles qui visent à la définition d'*objets macroscopiques*, par composition d'objets microscopiques (3.1.2). Selon [SS77], la première catégorie, s'intéressant à la notion de *généralisation*, est amplement développée en Intelligence Artificielle, et la seconde catégorie, s'intéressant à la notion d'*agrégation*, appartient plus au domaine des bases de données. Bien qu'il puisse également exister des techniques d'abstraction hybrides, mêlant objets et concepts, cette thèse s'intéresse principalement à la création d'objets, particulièrement adaptée à l'analyse spatiale et temporelle des grands systèmes.

Nous affirmons cependant que les techniques d'agrégation visent, tout comme les techniques de généralisation, à l'édification d'une *sémantique macroscopique*. Elles constituent par conséquent un sujet de recherche légitime pour l'Intelligence Artificielle. Cette sémantique, relative aux objets engendrés et non aux propriétés partagées, repose sur un processus en trois temps : *partitionnement* des objets microscopiques, *agrégation* de ces objets et *interprétation* des résultats par l'observateur (3.1.3).

3.1.1 Abstraction par définition de concepts

En apprentissage automatique, les techniques de *généralisation*, de *classification* ou de *catégorisation* consistent à organiser les connaissances relatives à un domaine particulier, sous la forme par exemple de « hiérarchies de concepts » [SS77, GW99, Wil05]. Les objets du domaine sont alors regroupés en classes (ou en catégories) en fonction des propriétés qu'ils partagent.

Exemple. Les taxonomies développées en biologie constituent un exemple canonique de généralisation : les CHIENS et les CHATS appartiennent à la classe des MAMMIFÈRES, qui peut être définie par un ensemble de propriétés telles que « avoir une température corporelle constante », « posséder des poils », « allaiter ses petits », *etc.* Les MAMMIFÈRES forment à leur tour, avec les REPTILES et les OISEAUX, le clade des AMNIOTES. En programmation orientée objet, il s'agit de la notion d'*héritage* : les CHIENS « héritent » des propriétés communes à tous les MAMMIFÈRES et, également, de celles partagées par les AMNIOTES.

Ces techniques d'abstraction consistent à définir des concepts à partir des objets observés. En analyse formelle de concepts [GW99, DP02, Wil05], la

définition *intensive* d'un concept (ensemble des propriétés partagées par les objets d'une classe) est donnée à partir de sa définition *extensive* (ensemble des objets appartenant à cette classe). Ici, l'objectif n'est pas de travailler sur la granularité des objets, mais sur la généralité de leurs propriétés : le terme « mammifère » ne désigne pas un objet macroscopique, formé de tous les MAMMIFÈRES, mais un ensemble de propriétés générales, partagées par les MAMMIFÈRES.

Cette thèse poursuit un objectif relativement différent. Comme il a été discuté dans le chapitre précédent, l'analyse des grands systèmes nécessite d'avoir recours à des objets de granularités spatiales et temporelles variées, afin de représenter le système et d'aborder sa complexité syntaxique et sémantique. Nous nous concentrons donc sur la notion d'*objet macroscopique*. Nous reviendrons sur la possibilité d'exploiter la technique d'abstraction proposée dans cette thèse à la définition de *concepts macroscopiques* en perspective de ces travaux (*cf.* section 9.2).

3.1.2 Abstraction par définition d'objets

Les techniques d'*agrégation* visent à construire des objets composites par le regroupement des objets qui les constituent [SS77][1].

Exemple. En programmation orientée objet, cela consiste à définir un objet à partir de ses composants. Par exemple, un ensemble de CHIENS composent une MEUTE. Dans ce contexte, on ne dira pas qu'un CHIEN est un type de MEUTES, ou qu'il hérite des propriétés de cette MEUTE, mais qu'il en est, plus simplement, un élément constitutif.

Les techniques d'agrégation se distinguent donc des techniques de généralisation par le fait que les abstractions définies sont des objets, et non des concepts. Cela n'empêche pas de s'intéresser aux propriétés de l'objet macroscopique ainsi défini (une MEUTE a un comportement, une taille, une localisation, *etc.*). Cependant, ces propriétés sont relatives à l'*ensemble* des objets agrégés, et non des propriétés *partagées* par ces objets.

Les techniques de partitionnement (*clustering*) [HBV01, LMO04, Moc09] sont des techniques d'apprentissage non-supervisé visant à « partitionner un ensemble fini de points d'un espace multidimensionnel en classes telles que (1) les points appartenant à la même classe sont *similaires* et (2) les points appartenant à des classes différentes sont *dissimilaires*. » [LMO04]

[1] L'*agrégation* de données doit être distinguée de la *fusion* de données consistant à intégrer plusieurs sources d'information concernant un objet donné pour procéder à son analyse. L'agrégation peut, au contraire, être appliquée à une seule source d'information.

Le partitionnement est donc une étape préliminaire aux processus d'abstraction, aussi bien dans le cas de concepts que d'objets macroscopiques. En effet, une fois les classes constituées, il est possible de déterminer les propriétés partagées par les membres de la classe (*généralisation*) ou de définir un nouvel objet à partir de ses membres (*agrégation*). Le partitionnement a donc un objectif général de *réduction* : en termes de données, il vise à « la compression de l'information contenue [par celles-ci] » [HBV01], page 109. Le partitionnement dépend néanmoins d'un objectif d'abstraction plus large (*généralisation* ou *agrégation* des objets partitionnés) et doit, par conséquent, être défini en fonction de cet objectif.

De plus, l'agrégation ne peut être dissociée d'une étape conclusive visant à l'*interprétation* des objets agrégés. En effet, dans la mesure où ils permettent de décrire le système, les agrégats « sont porteurs d'un sens pour l'observateur » [EF10]. Premièrement, ils sont utilisés pour décrire et expliquer les phénomènes de manière macroscopique. Ils doivent, par conséquent, être cohérents avec les batteries explicatives utilisées par les experts du domaine [DB07]. Deuxièmement, les agrégats peuvent être traduits en termes microscopiques afin d'obtenir une représentation approximative des objets qu'ils contiennent. Il s'agit notamment de savoir si les abstractions engendrées représentent adéquatement les objets sous-jacents [Cui07] et sont, à ce titre, correctement interprétées par l'observateur. Cette étape d'interprétation, responsable de la sémantique macroscopique et de la justesse des abstractions, doit donc être prise en compte lors du processus d'agrégation.

3.1.3 Partitionnement, agrégation et interprétation

L'agrégation utilise la notion de partitionnement comme point de départ du processus d'abstraction. Cependant, puisqu'elle vise la constitution d'objets macroscopiques, l'agrégation consiste à enrichir le partitionnement par l'introduction d'une sémantique macroscopique. Elle doit donc être distinguée des techniques de partitionnement classiques sur les points suivants.

Critères de partitionnement. Les algorithmes de partitionnement sont chargés de résoudre un problème d'optimisation : quelles partitions des objets microscopiques optimisent un critère donné ? Dans le cas de l'agrégation, le critère retenu doit prendre en compte l'objectif final du processus d'abstraction, c'est-à-dire la constitution d'un objet macroscopique correctement interprété par l'observateur. Nous souhaitons donc mesurer la *qualité* de l'objet engendré vis-à-vis du niveau de référence à partir duquel il a été construit. Dans ce contexte, les critères de partitionnement reposant uniquement sur l'analyse des propriétés microscopiques objets ou des classes

ne sont pas satisfaisants : *e.g.*, densité, connectivité, distance intra-classe et extra-classe [HBV01].

Nous nous orientons donc vers les techniques de partitionnement reposant sur la comparaison entre les objets microscopiques et les structures macroscopiques induites par leur agrégation. Le partitionnement par barycentres (*centroid-based clustering* [JMF99]) consiste par exemple à définir des objets centraux pour représenter les classes. Leur qualité dépend donc de la centralité de ces objets. Cependant, ils sont toujours interprétés comme des objets *microscopiques* et ne parviennent donc pas à rendre compte de la visée macroscopique de l'agrégation. En revanche, le partitionnement par distributions (*distribution-based clustering* [JMF99, Bis06]) permet de donner une véritable sémantique macroscopique aux classes engendrées. Celles-ci sont équivalentes à des distributions de probabilité, donnant un modèle de répartition des objets sous-jacents et permettant d'interpréter les abstractions au niveau microscopique[2]. La qualité des classes dépend alors de l'ajustement (*goodness-of-fit*) entre le modèle et les données, souvent optimisé par des techniques d'estimation du *maximum de vraisemblance* [Aka73, FJ02, Bis06]. De manière générale, les critères de partitionnement que nous considérons correspondent donc à des mesures de similarité entre les distributions, et non entre les objets.

Cette notion de « modèle de distribution », particulièrement adaptée à la définition d'objets macroscopiques, est formalisée dans la section suivante. Le chapitre 4 présente des mesures issues de la théorie de l'information pour évaluer la qualité de ces modèles et ainsi constituer des critères de partitionnement adaptés à la juste interprétation des objets macroscopiques.

Partitionnement contraint. L'étape cruciale du processus d'agrégation consiste à donner un sens aux abstractions engendrées. En complément des techniques de partitionnement utilisées en amont, l'agrégation dote les objets d'une *sémantique macroscopique*. Celle-ci doit notamment reposer sur des connaissances *externes*, c'est-à-dire indépendantes des données partitionnées et issues du domaine d'expertise chargé de l'analyse du système (*cf.* section 2.3). En termes de partitionnement, il s'agit d'introduire un *biais externe* [TB99], c'est-à-dire un critère de sélection des partitions pertinentes indépendant du critère de partitionnement *interne* (*cf.* paragraphe précédent). L'objectif est d'accroître ainsi la *compréhensibilité* des abstractions, c'est-à-dire leur adéquation avec « les connaissances *a priori* des experts » [TB99], page 213. L'agrégation nécessite donc un partitionnement *supervisé*, réalisé

[2] On parle plus généralement de *modèles de mélange* [JMF99, FJ02] et [Bis06], chapitre 9 « *Mixture Models and EM* », pages 423-459.

par exemple à partir de contraintes logiques sur les objets à partitionner. Ces contraintes permettent de formaliser les connaissances externes nécessaires à la compréhension des abstractions [TB99, WC00, DB07].

De ce point de vue, les classes engendrées par le partitionnement sont en partie *expliquées* par les connaissances externes : on fait l'hypothèse d'une correspondance entre le critère de partitionnement interne et les critères externes formulés par les experts. Ainsi, ce n'est pas seulement la similarité intra-classe qui est évaluée, mais la similarité *vis-à-vis d'un modèle externe du système*. Si les classes engendrées ne correspondent pas aux contraintes formulées, deux conclusions sont possibles : (1) les données sous-jacentes révèlent un comportement inattendu du système constituant potentiellement un point crucial de l'analyse (détection d'anomalies) ; (2) le modèle externe n'est pas adapté à l'analyse des données. La *validation du biais* consiste ainsi à valider ou à invalider le modèle utilisé par les experts [TB99] et participe, plus largement, à la *validation externe* du partitionnement, au cours duquel les classes engendrées sont confrontées aux abstractions du domaine [HBV01].

Le chapitre 5 présente une méthode pour la formalisation de ces modèles externes à partir de contraintes sur l'espace de recherche (ensemble des partitions *admissibles*).

Partitionnement optimal. L'objectif du partitionnement consiste à trouver les classes qui optimisent une mesure de qualité donnée (critère interne). Nous supposons donc qu'il existe un ordre total sur l'ensemble des partitions permettant de les comparer et de sélectionner les meilleures. Dans la suite de cette thèse, nous appellerons ce problème d'optimisation le *problème des partitions optimales* (formalisé dans la sous-section 4.2.1). Dans son acception la plus générale, il se distingue des problèmes de partitionnement classiques (*clustering problem* [JMF99, HBV01]) dans la mesure où aucune métrique n'est *a priori* définie sur l'ensemble des objets à partitionner. Dans le cas général, le nombre des partitions possibles d'un ensemble à n éléments est donné par le n^{e} nombre de Bell [Rot64]. Cette série a une *croissance exponentielle*. Elle ne permet donc pas en pratique un examen exhaustif de l'espace de recherche.

Les algorithmes de partitionnement sont donc des heuristiques utilisant un *biais interne* (*i.e.*, relatif aux données elles-mêmes) pour calculer des optima *locaux* (voir [HBV01] pour un état de l'art concernant les techniques de partitionnement classiques et leur complexité). La *validation interne* des algorithmes consiste alors à comparer les résultats engendrés par différentes techniques pour déterminer celles qui calculent les meilleures partitions vis-à-vis du critère interne [HBV01]. Cependant, l'introduction d'un *biais ex-*

terne permet d'orienter le processus d'optimisation [TB99]. Les partitions compréhensibles par les experts constituent un sous-ensemble des partitions possibles. Dans ce contexte, nous sommes alors intéressés par le calcul des optima *globaux* sur les espaces de recherche ainsi réduits. Le chapitre 6 présente un algorithme calculant ces optima globaux en un temps *polynomial* (pour des contraintes de type ordonnées ou hiérarchiques).

3.2 Formaliser le processus d'agrégation

Cette section donne un cadre formel à la notion d'agrégation. Elle définit en premier lieu le niveau de représentation microscopique du système comme une *population* (ensemble d'*individus*) et un *attribut* donnant la *distribution microscopique* des *unités atomiques* observées (3.2.1). L'agrégation utilise une *partition* de la population pour résumer l'attribut à l'aide d'un *opérateur d'agrégation* et donner ainsi la *distribution macroscopique* des unités atomiques (3.2.2). L'interprétation consiste à donner un *modèle* de la distribution microscopique à partir de la distribution macroscopique et d'une *hypothèse de redistribution* (3.2.3). L'*ensemble de toutes les partitions* est l'objet mathématique au centre du processus d'agrégation. Nous explicitons certaines de ses propriétés algébriques (3.2.4).

Notations. Cette section fait intervenir plusieurs types d'objets mathématiques. Nous utilisons un système de casses cohérent pour bien les distinguer :

- les *individus* sont notés par des lettres minuscules : x, y, z ;
- les *parties* (ensembles d'individus) par des majuscules : X, Y, Z ;
- les *partitions* (ensembles de parties) par des calligraphiques : $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$;
- les ensembles de partitions par des gothiques. Nous utilisons notamment le « C », le « P » et le « R » gothiques : $\mathfrak{C}$, $\mathfrak{P}$, $\mathfrak{R}$.

De manière générale, la table en page xvii donne un aperçu des notations utilisées dans cette thèse.

3.2.1 Représentation microscopique

L'agrégation a pour point de départ une représentation microscopique du système, discrétisant une dimension de l'analyse (espace, temps, entités, *etc.*) en objets microscopiques (portions d'espace, périodes de temps, membres du système, *etc.*). Nous parlons, plus généralement, d'*individus* et de *populations* pour désigner ces objets et ces dimensions.

Définition 3.1. Une *population* Ω est un ensemble d'individus $\{x_1, \ldots, x_n\}$ discrétisant une dimension du système. On note $|\Omega|$ la *taille* de la population (nombre d'individus).

Exemple. Prenons Alice, Bob, Carole, Denise, Élodie et Fernand, six voisins habitant dans le même immeuble et composant la population $\Omega = \{a, b, c, d, e, f\}$ de taille $|\Omega| = 6$.

Définition 3.2. Un *attribut* v est une application de Ω dans un ensemble de valeurs V qui associe à chaque individu $x \in \Omega$ une valeur $v(x) \in V$. Ces valeurs constituent les *données* de l'analyse. Elles peuvent représenter divers aspects des individus (*e.g.*, états, qualités, quantités) supposés utiles à la compréhension du système.

Exemple. Le nom, l'âge, le sexe et la taille des six voisins sont autant d'attributs qui peuvent servir à l'analyse du système qu'ils composent.

Dans cette thèse, les attributs que nous considérons sont interprétés comme des *distributions d'unités atomiques* (observations, évènements, ressources) réparties selon la dimension choisie. Nous nous intéressons donc aux attributs exprimant un *dénombrement*. En ce sens, la représentation microscopique est le résultat d'une agrégation préliminaire, réalisée par l'instrument d'observation et donnant le niveau de discrétisation de référence pour l'analyse. Les valeurs observées correspondent donc à des *quantités d'unités* et l'agrégation des individus implique toujours l'agrégation des unités atomiques.

Définition 3.3. Un *attribut de dénombrement* est un attribut à valeurs entières positives ($V = \mathbb{N}^+$) interprétées comme les *quantités d'unités atomiques* associées aux individus par l'instrument d'observation. Un tel attribut exprime également la *probabilité d'apparition* d'un individu lorsqu'on choisit une unité au hasard, de manière uniforme, parmi toutes les unités observées : $\forall x \in \Omega, \quad p(x) = \frac{v(x)}{v(\Omega)}$.

Exemple. Supposons que les six voisins sont tous amateurs de musique. Nous souhaitons analyser leurs collections de disques vinyle en mesurant, pour chaque individu x, le nombre $v(x)$ de disques qu'il possède dans ses étagères. Dans cet exemple, les unités observées sont donc des *disques vinyle*, répartis au niveau microscopique selon la dimension des *locataires de l'immeuble*, et l'attribut analysé selon cette dimension correspond à des

quantités de disque : $v(a) = 40$ disques, $v(b) = 41$ disques, $v(c) = 27$ disques, $v(d) = 25$ disques, $v(e) = 2$ disques et $v(f) = 45$ disques.

L'immeuble contient donc $v(\Omega) = 180$ disques (ensemble des unités observées). Imaginons que nous en choisissions un au hasard, de manière uniforme : $p(x) = \frac{v(x)}{v(\Omega)}$ est la probabilité qu'il appartienne à l'individu x.

Définition 3.4. On note $\mathcal{P}_\perp(\Omega) = \{\{x_1\}, \ldots, \{x_n\}\}$ la *partition microscopique* de Ω ou plus simplement $\mathcal{P}_\perp$, quand cela n'est pas ambigu. La *représentation microscopique* de l'attribut v, notée $v(\mathcal{P}_\perp)$, est le n-uplet des valeurs prises par tous les individus :

$$v(\mathcal{P}_\perp) = (v(x_1), \ldots, v(x_n))$$

De même, la *distribution de probabilité microscopique* des unités est :

$$p(\mathcal{P}_\perp) = \frac{v(\mathcal{P}_\perp)}{v(\Omega)} = (p(x_1), \ldots, p(x_n))$$

Exemple. Dans notre cas, nous avons $\mathcal{P}_\perp(\Omega) = \{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}\}$ et la représentation microscopique de la collection de disques des locataires est donnée par le 6-uplet $v(\mathcal{P}_\perp) = (40, 41, 27, 25, 2, 45)$ (*cf.* figure 3.1a). La distribution de probabilité associée est $p(\mathcal{P}_\perp) = \left(\frac{40}{180}, \frac{41}{180}, \frac{27}{180}, \frac{25}{180}, \frac{2}{180}, \frac{45}{180}\right)$.

3.2.2 Partitionnement et agrégation de données

Un *agrégat* est un objet macroscopique qui représente les objets qu'il contient. Il donne à ce titre un point de vue synthétique de l'attribut selon la dimension représentée. Formellement, un agrégat est une *partie* de la population sur laquelle les valeurs microscopiques sont agrégées en fonction d'un *opérateur d'agrégation*. Cet opérateur est responsable de la réduction des données et il donne la manière dont est construite la sémantique de l'attribut associé à l'objet macroscopique. On parle alors de *valeur agrégée*.

Définition 3.5. Étant donnée une population Ω, on note :

- $\mathcal{P}(\Omega)$ l'ensemble des parties de Ω, correspondant à l'ensemble des agrégats possibles ;
- $X \in \mathcal{P}(\Omega)$ une partie quelconque de la population et $|X|$ la *taille* de cette partie (nombre d'individus).

Exemple. Supposons que l'immeuble où vivent nos protagonistes soit agencé en trois appartements. On note $AB = \{a, b\}$, $CDE = \{c, d, e\}$ et $F = \{f\}$ les trois parties de Ω représentant la répartition des individus au sein de ces appartements. Leurs tailles sont donc $|AB| = 2$ locataires, $|CDE| = 3$ locataires et $|F| = 1$ locataire.

Définition 3.6. Un attribut v défini sur la population Ω peut être étendu à l'ensemble $\mathcal{P}(\Omega)$ des parties de la population en fonction d'un opérateur *-aire dans V, nommé *opérateur d'agrégation* :

$$\forall X \in \mathcal{P}(\Omega), \quad v(X) = \underset{x \in X}{\text{op}}\, v(x)$$

La *valeur agrégée* $v(X)$ résume ainsi les valeurs prises par les individus de la partie X.

Dans la mesure où nous nous intéressons à des attributs de dénombrement, la *somme* des valeurs mesure la quantité d'unités associées aux agrégats. Nous prenons donc $op = \sum$. Cependant, en perspective de cette thèse, nous pourrons généraliser l'agrégation à d'autres opérateurs permettant de résumer les valeurs microscopiques (produit, extrema, quantiles, *etc.*).

Exemple. Dans le cas du dénombrement de disques vinyle, la somme est un opérateur d'agrégation adéquat. Étant donné un appartement X, la valeur agrégée $v(X)$ représente ainsi la quantité de disques possédés par *tous* les locataires de l'appartement. En particulier : $v(AB) = 81$ disques, $v(CDE) = 54$ disques, $v(D) = 45$ disques et $v(\Omega) = 180$ disques (quantité de disques dans tout l'immeuble).

L'agrégation consiste à combiner *plusieurs* agrégats pour engendrer une représentation macroscopique de l'attribut. Dans cette thèse, nous nous limitons aux agrégats *disjoints* (un individu n'appartient jamais à deux agrégats différents) et *recouvrants* (tous les individus appartiennent au moins à un agrégat). Une telle agrégation correspond à une *partition* de la population représentée. Comme nous le verrons dans les chapitres suivants, les partitions d'une population permettent déjà d'engendrer de nombreuses représentations pertinentes pour l'analyse macroscopique. L'utilisation d'agrégats *non-disjoints* ou *non-recouvrants* sera abordée en perspective de cette thèse (section 9.2).

Définition 3.7. Étant donnée une population Ω, on note :

- $\mathfrak{P}(\Omega)$ l'ensemble des partitions de Ω, correspondant à l'ensemble des représentations macroscopiques possibles ;
- $\mathcal{X} \in \mathfrak{P}(\Omega)$ une partition quelconque de la population et $|\mathcal{X}|$ la *taille* de cette partition (nombre de parties) ;
- $\mathcal{P}_\perp(\Omega) = \{\{x_1\}, \ldots, \{x_n\}\}$ la *partition microscopique* de Ω ;
- $\mathcal{P}_\top(\Omega) = \{\{x_1, \ldots, x_n\}\} = \{\Omega\}$ la *partition macroscopique*[3] de Ω.

Définition 3.8. Étant donnée une partition $\mathcal{X} = \{X_1, \ldots, X_m\} \in \mathfrak{P}(\Omega)$ de taille m, la *représentation agrégée* de l'attribut v selon la partition $\mathcal{X}$, notée $v(\mathcal{X})$, est le m-uplet des valeurs agrégées sur les parties appartenant à $\mathcal{X}$:

$$v(\mathcal{X}) = (v(X_1), \ldots, v(X_m))$$

De même, la *distribution de probabilité agrégée* des unités est :

$$p(\mathcal{X}) = (p(X_1), \ldots, p(X_m))$$

Exemple. Nous décidons de représenter les collections de disques au niveau des appartements. L'agrégation repose donc sur la partition $\mathcal{X} = \{AB, CDE, F\} = \{\{a, b\}, \{c, d, e\}, \{f\}\}$ de taille $|\mathcal{X}| = 3$ appartements. La représentation résultante est le triplet $v(\mathcal{X}) = (81, 54, 45)$ correspondant aux quantités de disques dans les appartements AB, CDE et F (*cf.* figure 3.1). La distribution de probabilité des unités au niveau des appartements est donc $p(\mathcal{X}) = \left(\frac{81}{180}, \frac{54}{180}, \frac{45}{180}\right)$.

3.2.3 Interprétation des données agrégées

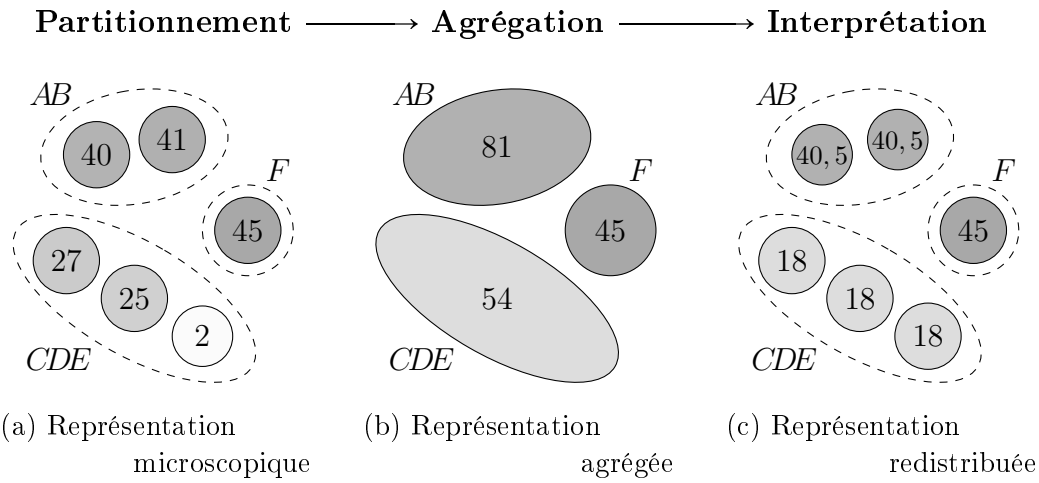
Les représentations agrégées offrent une vue synthétique du niveau microscopique. Cependant, en vue de donner une explication microscopique des phénomènes macroscopiques, il est nécessaire d'interpréter les valeurs agrégées vis-à-vis des individus sous-jacents. Un observateur ne disposant que des données agrégées doit nécessairement, pour ce faire, formuler une hypothèse concernant la *distribution effective des unités atomiques* au sein des agrégats qu'il observe. En ce sens, il s'agit pour lui de redistribuer les unités au niveau

[3] Les notations $\mathcal{P}_\perp$ et $\mathcal{P}_\top$ correspondent aux notions d'*élément minimum* $\perp$ et d'*élément maximum* $\top$ d'un ensemble partiellement ordonné [DP02]. La sous-section 3.2.4 montre en effet que l'ensemble des partitions $\mathfrak{P}(\Omega)$ est un ensemble partiellement ordonné dont les partitions microscopique et macroscopique sont les extrema.

microscopique. Cette *hypothèse de redistribution* exprime la manière dont les données agrégées sont interprétées par l'observateur (*cf.* figure 3.1).

Définition 3.9. Une représentation agrégée $v(\mathcal{X})$ peut être interprétée comme une représentation microscopique $v_{\mathcal{X}}(\mathcal{P}_\perp)$ à partir d'une *hypothèse de redistribution*. Il s'agit d'une application $v_{\mathcal{X}}$ qui associe à chaque individu $x \in \Omega$, une *valeur redistribuée* $v_{\mathcal{X}}(x) \in V$.
Par exemple, l'*hypothèse de redistribution uniforme* donne $v_{\mathcal{X}}(x) = \frac{v(X)}{|X|}$ lorsque $x \in X$. La représentation $v_{\mathcal{X}}(\mathcal{P}_\perp)$ est alors la *représentation redistribuée* au niveau microscopique. Elle exprime la manière dont les valeurs agrégées sont interprétées par l'observateur.

Exemple. Supposons que nous ne disposions que de la représentation agrégée au niveau des appartements, mais que nous aimerions avoir une idée des collections de disques de chaque locataire. En première approximation, nous pouvons supposer que la répartition des disques vinyles entre les locataires d'un même appartement est uniforme. Il s'agit là d'une interprétation possible – parmi d'autres – des valeurs agrégées. Nous avons alors : $v_{\mathcal{X}}(a) = v_{\mathcal{X}}(b) = 40,5$ disques, $v_{\mathcal{X}}(c) = v_{\mathcal{X}}(d) = v_{\mathcal{X}}(e) = 18$ disques et $v_{\mathcal{X}}(f) = 45$ disques. Le 6-uplet $v_{\mathcal{X}}(\mathcal{P}_\perp) = (40.5, 40.5, 18, 18, 18, 45)$ donne la représentation uniformément redistribuée des collections de disques (*cf.* figure 3.1c). Elle indique la manière dont l'observateur interprète la représentation agrégée (figure 3.1b) à partir de la distribution uniforme.



(a) Représentation microscopique (b) Représentation agrégée (c) Représentation redistribuée

FIGURE 3.1 – Partitionnement, agrégation et interprétation d'une population de 6 individus (valeurs sommées et uniformément redistribuées)

Définition 3.10. Dans le cas général, supposons que l'on dispose d'une *distribution de probabilité* p', définie au niveau des individus et utilisée pour interpréter les données agrégées (c'est-à-dire pour les redistribuer). L'hypothèse de redistribution correspondante, pour $x \in X$, est :

$$v_{\mathcal{X}}(x) = p'(x|X)\, v(X) = \frac{p'(x)}{p'(X)}\, v(X)$$

Exemple. Supposons que nous disposions d'informations externes sur les locataires pouvant être corrélées à la possession de disques (salaire, nombre d'achats mensuels, place réservée dans les étagères de l'appartement), de telles informations peuvent être utilisées pour définir une distribution de probabilité servant à interpréter les données agrégées de manière plus fine.

3.2.4 Structure algébrique de l'ensemble des partitions

L'agrégation consiste à chercher, à évaluer et à comparer des partitions pour sélectionner les « meilleures » représentations possibles. L'objet mathématique au centre du processus d'abstraction est donc l'*ensemble des partitions* $\mathfrak{P}(\Omega)$. Cette sous-section caractérise cet objet et en explicite certaines propriétés algébriques.

L'ensemble des partitions est notamment un *treillis*, c'est-à-dire un ensemble *partiellement ordonné* dont chaque couple d'éléments admet une *borne inférieure* et une *borne supérieure*. Intuitivement, la relation d'ordre est définie de la manière suivante : une partition $\mathcal{X}$ est « plus petite » qu'une partition $\mathcal{Y}$ si et seulement si la représentation $v(\mathcal{X})$ contient « plus de détails » que la représentation $v(\mathcal{Y})$. On dit alors que $\mathcal{X}$ « raffine » $\mathcal{Y}$.

Définition 3.11. Une partition $\mathcal{X} \in \mathfrak{P}(\Omega)$ *raffine* une partition $\mathcal{Y} \in \mathfrak{P}(\Omega)$ si et seulement si chaque partie appartenant à $\mathcal{X}$ est incluse dans une partie appartenant à $\mathcal{Y}$. On note alors $X < Y$.
Formellement : $X < Y \quad \Leftrightarrow \quad (\forall X \in \mathcal{X}, \quad \exists Y \in \mathcal{Y}, \quad X \subseteq Y)$

Étant donnée une partition $\mathcal{X} \in \mathfrak{P}(\Omega)$, on note $\mathfrak{R}(\mathcal{X})$ l'ensemble des partitions raffinant $\mathcal{X}$. En particulier, la partition microscopique raffine toutes les partitions et toutes les partitions raffinent la partition macroscopique : $\quad \forall \mathcal{X} \in \mathfrak{P}(\Omega), \quad \mathcal{P}_{\perp} \in \mathfrak{R}(\mathcal{X}) \quad$ et $\quad \mathcal{X} \in \mathfrak{R}(\mathcal{P}_{\top})$

Définition 3.12. Une partition $\mathcal{X} \in \mathfrak{P}(\Omega)$ est *couverte* par une partition $\mathcal{Y} \in \mathfrak{P}(\Omega)$ si et seulement si $\mathcal{X}$ raffine $\mathcal{Y}$ et il n'existe pas de « raffinement intermédiaire » entre $\mathcal{X}$ et $\mathcal{Y}$. On note alors $\mathcal{X} \prec\!\!\!- \mathcal{Y}$.
Formellement : $\mathcal{X} \prec\!\!\!- \mathcal{Y} \quad \Leftrightarrow \quad (\mathcal{X} < \mathcal{Y} \quad \text{et} \quad \nexists \mathcal{Z}, \quad \mathcal{X} < \mathcal{Z} < \mathcal{Y})$

Étant donnée une partition $\mathcal{X} \in \mathfrak{P}(\Omega)$, on note $\mathfrak{C}(\mathcal{X})$ l'ensemble des partitions couvertes par $\mathcal{X}$. Notons qu'une partition couverte est toujours raffinante : $\forall \mathcal{X} \in \mathfrak{P}(\Omega), \quad \mathfrak{C}(\mathcal{X}) \subset \mathfrak{R}(\mathcal{X})$

La relation de *raffinement* représente les partitions que l'on peut atteindre en agrégeant ou en désagrégeant une partition donnée. La relation de *couverture* représente les *désagrégations atomiques* que l'on peut appliquer à une partition donnée, c'est-à-dire les différentes façons de scinder une partition « de manière minimale »[4].

Exemple. La figure 3.2 donne le diagramme de Hasse – utilisé pour représenter un ensemble partiellement ordonné [DP02] – de l'ensemble des partitions $\mathfrak{P}(\{a, b, c, d\})$. Les 15 partitions possibles sont représentées. Les flèches représentent la relation de couverture (« $\mathcal{X}$ est couverte par $\mathcal{Y}$ » est représenté par une flèche allant de $\mathcal{Y}$ vers $\mathcal{X}$). La relation de raffinement est représentée par des séquences de flèches (« $\mathcal{X}$ raffine $\mathcal{Y}$ » est représenté par une séquence de flèches allant de $\mathcal{Y}$ à $\mathcal{X}$). On remarque en particulier que la partition microscopique $\{\{a\}, \{b\}, \{c\}, \{d\}\}$, située en bas du diagramme, peut être atteinte depuis toutes les partitions et que la partition macroscopique $\{\{a, b, c, d\}\}$, située en haut du diagramme, permet d'atteindre toutes les partitions.

La relation de couverture peut également être définie de manière *constructive* : pour tout couple de partition $\mathcal{X}$ et $\mathcal{Y}$ dans $\mathfrak{P}(\Omega)$, nous avons $\mathcal{X} \in \mathfrak{C}(\mathcal{Y})$ si et seulement si $\mathcal{X} \in \mathfrak{R}(\mathcal{Y})$ et $|\mathcal{X}| = |\mathcal{Y}| + 1$. La construction des partitions couvertes repose donc sur la notion de cardinalité. Nous verrons cependant dans le chapitre 6 que, lorsqu'on s'intéresse à un sous-ensemble de $\mathfrak{P}(\Omega)$, les partitions couvertes sont construites de manière différente.

[4] Notons que la relation de *raffinement* détermine – et est déterminée par – la relation de *couverture* [DP02] : $\mathcal{X}$ raffine $\mathcal{Y}$ si et seulement si « il existe une séquence de désagrégations atomiques de la partition $\mathcal{Y}$ qui aboutit à la partition $\mathcal{X}$. »

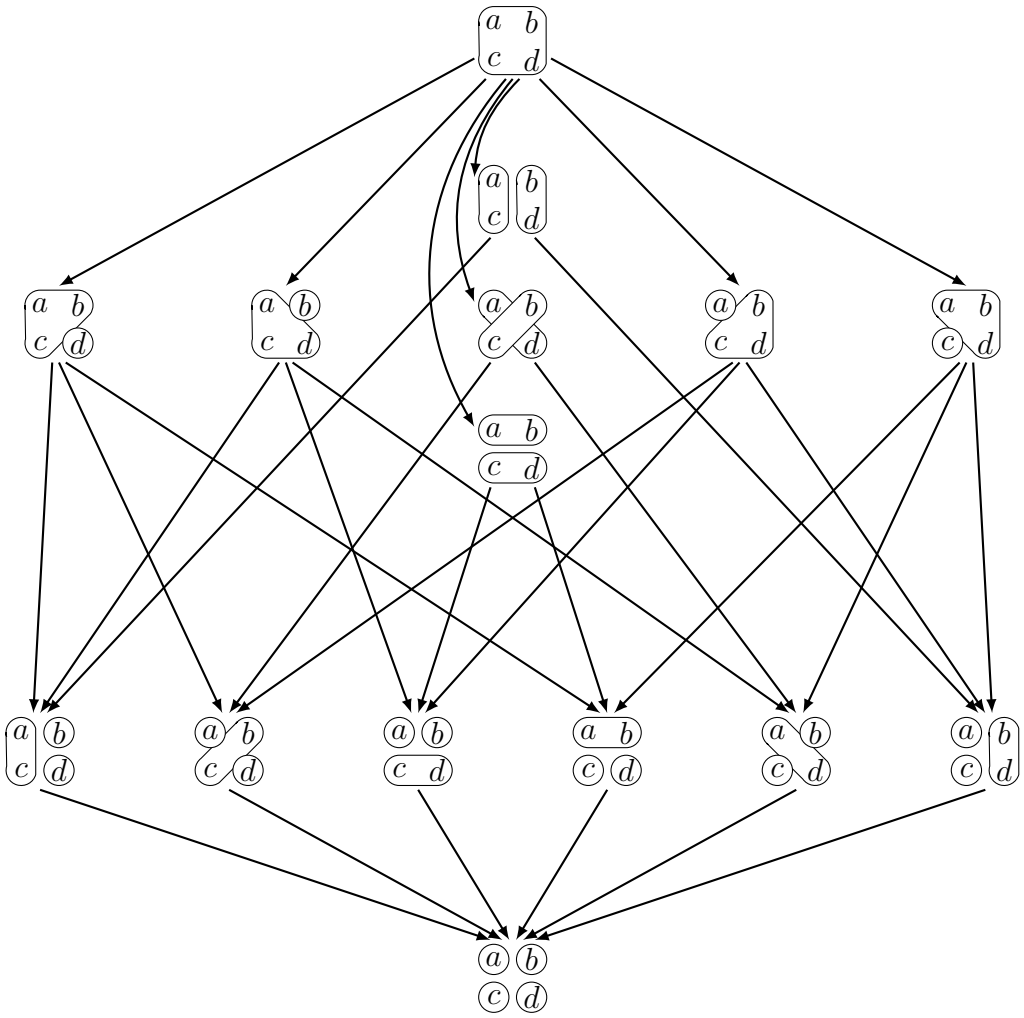


FIGURE 3.2 – Diagramme de Hasse de l'ensemble des partitions possibles de la population $\{a, b, c, d\}$ ordonné selon la relation de couverture

Deuxième partie

Le rôle de l'observateur

« Le macroscope filtre les détails, amplifie ce qui relie, fait ressortir ce qui rapproche. »

Joël de Rosnay, *Le macroscope*

CHAPITRE 4

Évaluer et contrôler le processus d'agrégation

Dans la mesure où ils sous-tendent et orientent la compréhension des systèmes, les processus d'abstraction ne sont pas neutres vis-à-vis de l'analyse. Dès lors, la caractérisation, l'évaluation et le contrôle du processus sont des conditions cruciales pour la mise en place d'un outil scientifique performant. En particulier, parmi l'ensemble des représentations macroscopiques possibles d'un même système, il est nécessaire de déterminer celles qui sont les plus pertinentes pour l'observateur. Celui-ci doit donc disposer de méthodes pour estimer la qualité des représentations qu'il utilise [Cui07]. Selon l'approche pragmatiste (*cf.* section 2.3), la mesure et la sélection des représentations dépend du contexte d'analyse (objectifs à réaliser et ressources disponibles). Ainsi, l'évaluation du processus d'agrégation doit être ancrée dans ce contexte.

La section 4.1 présente les enjeux essentiels du processus d'agrégation vis-à-vis de l'analyse. Il s'agit de produire des représentations passant à l'échelle tout en garantissant une juste interprétation des données macroscopiques. Nous explicitons donc des critères d'évaluation selon deux axes : *réduire la complexité* des représentations et *contrôler leur contenu informationnel*. Dans la section 4.2, nous proposons de formaliser ces critères à partir de *mesures de qualité* définies sur les partitions de la population à agréger. La section 4.3 propose des mesures pour formaliser les critères propres au processus d'agrégation : la *taille* des représentations comme mesure de complexité, la *divergence de Kullback-Liebler* comme perte d'information et un *compromis de qualité* pour exprimer les enjeux contradictoires de l'agrégation.

4.1 Passage à l'échelle et interprétation des données agrégées

Pour [Cui07], le processus d'abstraction consiste à « cacher certains détails tout en préservant les caractéristiques essentielles des données. »[1] L'objectif de l'agrégation est donc double. Premièrement, la suppression des détails permet de garantir l'exploitabilité des représentations lors du passage à l'échelle. Deuxièmement, la mise en évidence des caractéristiques essentielles permet de procéder néanmoins à une analyse informée du système. L'agrégation vise donc à résoudre deux problèmes contradictoires (passage à l'échelle et interprétation des données) afin d'engendrer des représentations exploitables et pertinentes pour l'analyse des grands systèmes.

Cette section discute plus en détail ces objectifs et définit, à partir de ceux-là, les critères d'évaluation du processus d'agrégation. Nous proposons de les articuler selon deux principes fondamentaux : la *réduction de complexité* et la *perte d'information* causées par l'agrégation des données. Le reste du chapitre formalise ces critères.

Exemple. L'ensemble du chapitre sera illustré par l'exemple introduit dans le chapitre précédent. Six amateurs de musique vivent dans le même immeuble et possèdent chacun une collection plus ou moins importante de disques vinyle. La figure 4.1 présente les résultats de l'agrégation, réalisée selon trois partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$, engendrant chacune une représentation particulière de l'attribut $v(.)$ (quantités de disques). L'objectif de ce chapitre est de déterminer laquelle de ces représentations est la plus adaptée à l'analyse des collections de disques des différents locataires.

4.1.1 Réduire la complexité pour passer à l'échelle

Comme il a été discuté dans la section 2.3, l'objectif premier du processus d'abstraction consiste à réduire la *complexité* des systèmes afin de procéder à leur analyse. Nous empruntons la définition de Bonabeau et Dessalles [BD97] selon laquelle la complexité dépend à la fois de la tâche à accomplir (analyse du système) et des outils de description disponibles pour réaliser cette tâche (dans notre cas, la représentation utilisée pour l'analyse). Dans ce contexte, la complexité dépend des abstractions dont dispose un observateur pour aborder le système : plus une représentation est abstraite, plus l'analyse est « facile ». Nous précisons cette définition à partir de la notion de « coût de l'analyse ».

[1] « *We define data abstraction as the process of hiding details of data while maintaining the essential characteristics of data.* » [Cui07], page 1.

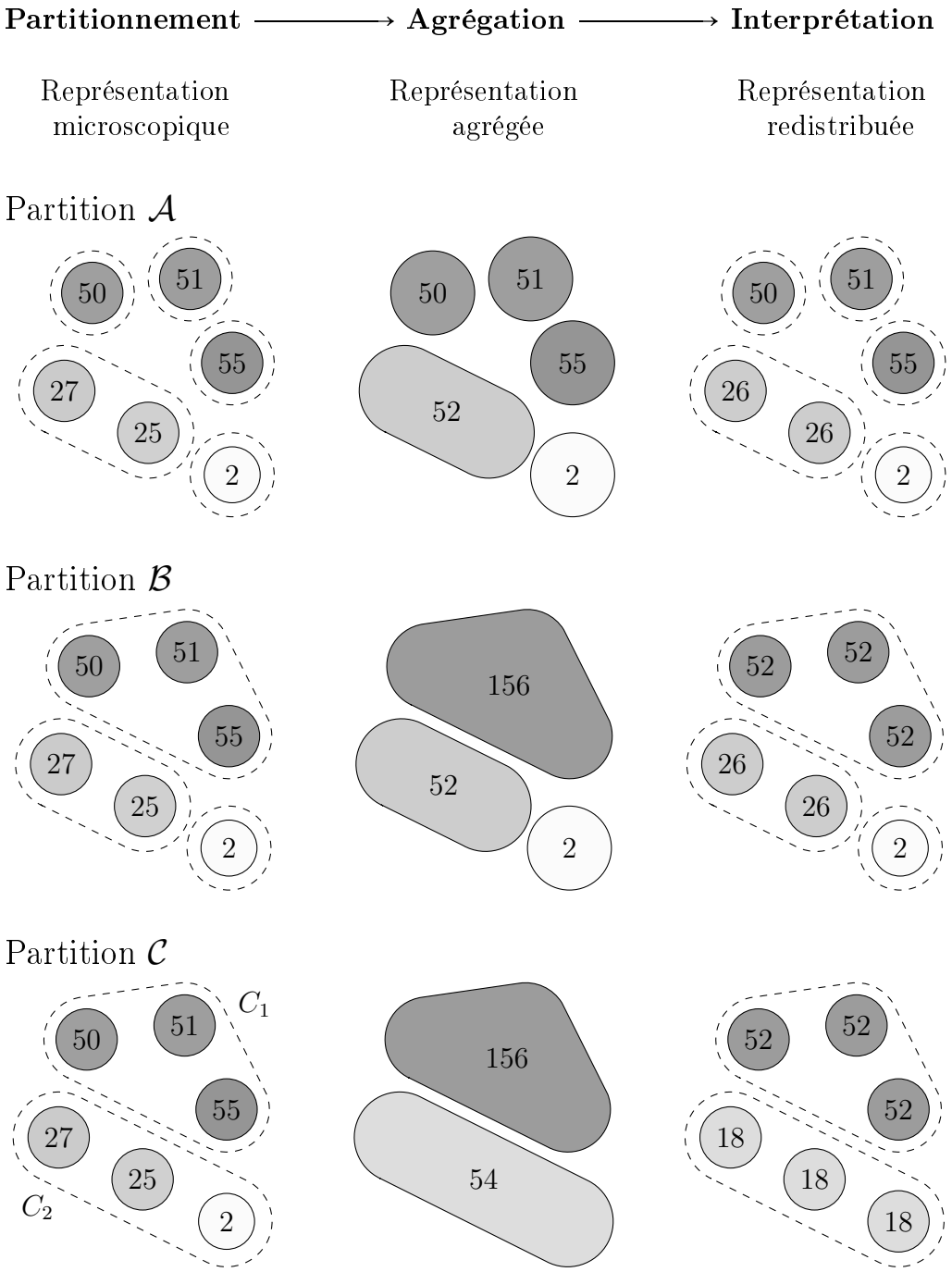


FIGURE 4.1 – Agrégation d'une population de 6 individus selon 3 partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$ (valeurs sommées et uniformément redistribuées)

Définition 4.1. La *complexité* d'une représentation désigne la difficulté qu'a un observateur à analyser le système à partir de cette représentation. La complexité est alors caractérisée par le « coût » d'une telle analyse, c'est-à-dire *la quantité de ressources nécessaires*[2] *à l'exploitation de la représentation.*

Dans le cas de grands systèmes, une analyse reposant sur la représentation microscopique est extrêmement coûteuse, voire irréalisable en pratique. Pour passer à l'échelle, il est nécessaire de *simplifier* la représentation microscopique, c'est-à-dire de travailler à partir d'une représentation qui demande moins de ressources. L'agrégation vise donc à *réduire la complexité* de la représentation microscopique afin de diminuer le coût de l'analyse reposant sur cette représentation et, ainsi, de passer à l'échelle. La sous-section 4.3.1 présente une mesure de qualité permettant d'évaluer les représentations en fonction de ce premier critère.

Exemple. L'agrégation selon la partition $\mathcal{A}$ conserve la plupart des détails microscopiques (*cf.* figure 4.1). Dans le cas de grands systèmes, une telle représentation est extrêmement coûteuse à manipuler, à visualiser et à interpréter[3]. À ce titre, les partitions $\mathcal{B}$ et $\mathcal{C}$ lui sont préférables, en termes de complexité, dans la mesure où elles offrent un véritable point de vue macroscopique sur le système.

4.1.2 Contrôler la perte d'information pour interpréter les données

Parce qu'elles contiennent moins de détails que les données microscopiques, les données agrégées fournissent nécessairement moins d'information concernant l'état du système. Cependant, la notion d'information doit clairement être distinguée de la notion de données. Telles que définies dans le chapitre précédent, les *données* désignent les valeurs contenues dans une représentation, indépendamment de toute interprétation. C'est notamment à partir de la taille des données que peut être définie la complexité d'une représentation

[2] Il s'agit par exemple (1) des ressources informatiques nécessaires à l'encodage de la représentation (espace mémoire) ou à la production d'indicateurs statistiques (temps de calcul), (2) des ressources graphiques (taille d'écran, nombre de pixels) nécessaires à sa visualisation ou encore (3) des « ressources cognitives » nécessaires à sa compréhension.

[3] Lorsqu'il s'agit de représenter 6 individus, bien évidemment, l'analyse de la représentation microscopique ne demande jamais beaucoup de ressources. Ainsi, la réduction de complexité n'a d'intérêt que lors du passage à l'échelle.

(*cf.* sous-section 4.3.1). L'*information* est, au contraire, le résultat d'une interprétation des données en vue de l'analyse du système. Une information est donc un fait significatif servant à décrire et à expliquer l'état ou la dynamique du système (*e.g.*, un évènement, une perturbation, une transition). Lorsque beaucoup de données fournissent peu d'information, on parle de *données redondantes*. L'agrégation vise alors à supprimer ces données afin de réduire la complexité de la représentation. Lorsqu'au contraire l'agrégation supprime des données non-redondantes, on parle de *perte d'information*. Il en résulte la suppression d'un fait important pour l'analyse du système.

La perte d'information est un point critique du processus d'abstraction dans la mesure où elle peut introduire un biais dans l'analyse (déformation de la représentation microscopique, perte d'éléments significatifs, homogénéisation des données, *etc.*). Il est donc primordial de mesurer et de contrôler cette perte afin de garantir que les données représentées sont correctement interprétée par l'observateur. Ainsi, l'agrégation, en plus de réduire la complexité des représentations, vise à en extraire le maximum d'information. La sous-section 4.3.2 présente une mesure de qualité permettant d'évaluer les représentations en fonction de ce second critère.

Exemple. Dans la figure 4.1, les partitions $\mathcal{B}$ et $\mathcal{C}$ donnent moins d'indications que la partition $\mathcal{A}$ sur la distribution microscopique des disques vinyle. Pour ces deux partitions, lors de l'étape d'interprétation, les données sont homogénéisées et les variations microscopiques disparaissent. Dans le cas de la partition $\mathcal{C}$, on distingue l'homogénéisation de la partie C_1, où l'agrégation de données redondantes ne cause pas une grande perte d'information, et l'homogénéisation de la partie C_2, provoquant la suppression d'un fait important : un des trois individus de l'agrégat possède beaucoup moins de disques que les deux autres (2 contre 25 et 27). La représentation agrégée ne donne aucune information sur cet état particulier de l'attribut, ce qui peut nuire à l'analyse des collections de disques.

4.1.3 Réaliser un compromis entre réduction de complexité et perte d'information

Le coût de l'analyse ne peut être réduit sans perdre un peu d'information. En d'autres termes, on ne peut passer à l'échelle sans supprimer un certain nombre de faits, plus ou moins significatifs pour l'analyse. Lorsqu'elle engendre une représentation macroscopique, l'agrégation réalise donc un compromis entre la complexité de cette représentation et la quantité d'information qu'elle contient. La qualité d'une représentation dépend du contexte d'analyse : une bonne représentation *pour l'analyse détaillée du système*

contient beaucoup d'information, mais elle est alors très coûteuse ; une bonne représentation *pour passer à l'échelle* est peu complexe, mais elle ne permet pas d'analyser en détail l'ensemble des variations microscopiques. Dans la plupart des cas, il s'agit de trouver *un équilibre entre réduction de complexité et perte d'information* en fonction de la quantité de détails attendus par l'observateur et des ressources dont il dispose pour procéder à l'analyse du système. Cette description du processus d'agrégation est cohérente avec l'acception pragmatiste de l'émergence (*cf.* section 2.3) selon laquelle une abstraction doit être évaluée en fonction du contexte épistémique au sein duquel elle est élaborée. La sous-section 4.3.3 présente une mesure de qualité combinant les mesures présentées dans les sous-sections 4.3.1 et 4.3.2 afin d'évaluer les représentations en fonction des deux critères énoncés.

Exemple. Dans la figure 4.1, la partition $\mathcal{B}$ est meilleure que la partition $\mathcal{A}$ en termes de complexité et elle est meilleure que la partition $\mathcal{C}$ en termes d'information. Ainsi, la représentation agrégée $v(\mathcal{B})$ est correctement interprétée par l'observateur et reste exploitable à moindre coût. On dira qu'elle réalise un « bon compromis » pour l'analyse du système.

4.2 Définition générique des mesures de qualité

Une « bonne » représentation est une représentation satisfaisant un certain nombre de critères attestant de sa *qualité* pour l'analyse. Dans la section précédente, nous avons mis en évidence deux critères importants : (1) les représentations doivent être *le moins complexe possible*, afin de pouvoir être exploitées lors du passage à l'échelle et (2) elles doivent contenir *le maximum d'information*, afin d'être correctement interprétées. Cette section donne un cadre générique pour la mesure et l'optimisation de tels critères.

4.2.1 Problème des partitions optimales

La première étape consiste à associer aux critères d'évaluation des *mesures* permettant de quantifier et de comparer la qualité des représentations. Nous parlons donc, de manière générique, de *mesures de qualité*.

Définition 4.2. Une *mesure de qualité* m est une application qui associe à toute partition $\mathcal{X} \in \mathfrak{P}(\Omega)$ une valeur dans $\mathbb{R}$ exprimant sa qualité vis-à-vis d'un critère particulier. On parlera indifféremment de la *qualité* de la partition $\mathcal{X}$ ou de la *qualité* de la représentation agrégée $v(\mathcal{X})$.

Définition 4.3. Une mesure de qualité *positive* est une mesure que l'on cherche à maximiser (*e.g.*, la réduction de complexité). Une partition $\mathcal{X}$ est alors *meilleure* qu'une partition $\mathcal{Y}$ si et seulement si elle a une qualité supérieure : $m(\mathcal{X}) > m(\mathcal{Y})$. Réciproquement, une *mesure de qualité négative* est une mesure que l'on cherche à minimiser (*e.g.*, la perte d'information). $\mathcal{X}$ est *meilleure* que $\mathcal{Y}$ si et seulement si $m(\mathcal{X}) < m(\mathcal{Y})$.

Le *problème des partitions optimales* est un problème d'optimisation consistant à trouver les partitions qui maximisent ou qui minimisent une mesure de qualité donnée (selon qu'il s'agisse d'une mesure positive ou négative). Résoudre ce problème revient à trouver les *meilleures représentations* pour l'analyse du système. Nous parlons donc des partitions *optimales*[4].

Définition 4.4. Étant donnée une mesure de qualité positive m, l'*ensemble des partitions optimales selon* m, noté $\mathfrak{P}^m(\Omega)$, est le sous-ensemble de $\mathfrak{P}(\Omega)$ contenant les partitions qui maximisent m :

$$\mathfrak{P}^m(\Omega) = \arg\max_{\mathcal{X}\in\mathfrak{P}(\Omega)} m(\mathcal{X})$$

Réciproquement, pour une mesure de qualité négative, nous avons :

$$\mathfrak{P}^m(\Omega) = \arg\min_{\mathcal{X}\in\mathfrak{P}(\Omega)} m(\mathcal{X})$$

Trouver $\mathfrak{P}^m(\Omega)$ consiste à résoudre le *problème des partitions optimales*.

La complexité du *problème des partitions optimales* est, dans le cas général, *exponentielle* (*cf.* section 3.1.3). Le chapitre 6 discute plus en détail cette complexité et donne un algorithme de résolution efficace.

4.2.2 Mesures de qualité monotones

La monotonie vise à assurer que les mesures de qualité utilisées sont cohérentes, en termes de variation, avec le processus d'agrégation. Intuitivement, pour une mesure de qualité *positive et croissante* ou *négative et décroissante*, plus une représentation est agrégée, meilleure elle est. Pour une mesure *positive et décroissante* ou *négative et croissante*, c'est l'inverse. Ainsi, lorsqu'on

[4] Notons que, bien que cela soit rare en pratique, *plusieurs* partitions peuvent être simultanément optimales selon une mesure de qualité donnée. Nous parlons donc bien du problème *des* partitions optimales.

parcourt le treillis des partitions (*cf.* sous-section 3.2.4), depuis la partition microscopique $\mathcal{P}_\perp$ jusqu'à la partition macroscopique $\mathcal{P}_\top$, la qualité des représentations ne fait qu'augmenter (ou elle ne fait que diminuer).

Définition 4.5. Une mesure de qualité m est *croissante* lorsque, pour tout couple de partitions $\mathcal{X}$ et $\mathcal{Y}$, nous avons $\mathcal{X} < \mathcal{Y} \rightarrow m(\mathcal{X}) < m(\mathcal{Y})$. La qualité minimale est alors atteinte par la partition microscopique et la qualité maximale par la partition macroscopique :

$$\min_{\mathcal{X} \in \mathfrak{P}(\Omega)} m(\mathcal{X}) = m(\mathcal{P}_\perp) \quad \text{et} \quad \max_{\mathcal{X} \in \mathfrak{P}(\Omega)} m(\mathcal{X}) = m(\mathcal{P}_\top)$$

Respectivement, une mesure m est *décroissante* lorsque, pour tout couple de partitions $\mathcal{X}$ et $\mathcal{Y}$, nous avons $\mathcal{X} < \mathcal{Y} \rightarrow m(\mathcal{X}) > m(\mathcal{Y})$.

La sous-section 4.3.1 présente une mesure de qualité *positive et croissante* pour quantifier la réduction de complexité et la sous-section 4.3.2 une mesure de qualité *négative et croissante* pour quantifier la perte d'information. Lorsqu'on cherche à optimiser ces mesures monotones, le problème des partitions optimales n'admet que des solutions triviales :

	Mesure positive	**Mesure négative**
Mesure croissante	$\mathfrak{P}^m(\Omega) = \{\mathcal{P}_\top\}$	$\mathfrak{P}^m(\Omega) = \{\mathcal{P}_\perp\}$
Mesure décroissante	$\mathfrak{P}^m(\Omega) = \{\mathcal{P}_\perp\}$	$\mathfrak{P}^m(\Omega) = \{\mathcal{P}_\top\}$

Cependant, la sous-section 4.3.3 présente une mesure de qualité *positive et non-monotone*, combinant les deux mesures introduites précédemment. Dès lors, le problème des partitions optimales n'admet pas de solution simple. Un algorithme de résolution est proposé dans le chapitre 6.

4.2.3 Mesures de qualité décomposables

Intuitivement, une mesure de qualité est *décomposable* si et seulement si la qualité d'une partition est entièrement déterminée par la qualité de ses parties. La décomposabilité suppose donc (1) que la notion de qualité peut être étendue aux *parties* de la population et (2) que la qualité d'une partie *ne dépend pas* du reste de la partition. Il est ainsi possible d'évaluer, de comparer et de sélectionner les agrégats, indépendamment des représentations auxquelles ils appartiennent. De plus, le problème des partitions optimales est alors décomposable et peut être efficacement résolu selon une stratégie *diviser pour régner* (*cf.* section 6.2).

Définition 4.6. Une mesure de qualité m, définie sur l'ensemble des partitions $\mathfrak{P}(\Omega)$, est *décomposable* si et seulement si elle est l'extension d'une mesure de qualité définie sur l'ensemble des parties $\mathcal{P}(\Omega)$. Il existe alors un opérateur *-aire op tel que :

$$\forall \mathcal{X} \in \mathfrak{P}(\Omega), \quad m(\mathcal{X}) = \operatorname*{op}_{X \in \mathcal{X}} m(X)$$

En particulier, m est *additivement décomposable* [Csi08], ou plus simplement *additive* [JSB+05], si et seulement si :

$$\forall \mathcal{X} \in \mathfrak{P}(\Omega), \quad m(\mathcal{X}) = \sum_{X \in \mathcal{X}} m(X)$$

4.3 Définition de mesures de qualité spécifiques

Cette section présente des mesures de qualité adaptées aux critères d'évaluation internes présentés dans la section 4.1.

4.3.1 La taille des représentations comme mesure de complexité

Dans la section 4.1, nous avons défini la complexité d'une représentation à partir de la quantité de ressources nécessaires à son exploitation (encodage, traitement, visualisation, *etc.*). L'agrégation vise donc à réduire le coût de l'analyse afin de passer à l'échelle. Cependant, de telles ressources peuvent être définies et quantifiées de nombreuses manières. Avant de préciser celle qui satisfait au mieux à notre contexte d'analyse, nous donnons une formalisation générique de la notion de complexité.

Définition 4.7. Une mesure de complexité C est une mesure de qualité estimant le coût de l'analyse en termes de ressources. Elle est :

- *négative* (une partition est « bonne » lorsque l'exploitation de la représentation associée nécessite peu de ressources) ;
- *décroissante* (plus une représentation est agrégée, moins son exploitation est coûteuse).

Le processus d'agrégation induit une *réduction de complexité*. Il s'agit de la quantité de ressources « économisées » par l'exploitation d'une représentation agrégées au lieu de la représentation microscopique. La réduction de complexité est donc une *différence*[5] entre deux complexités.

Définition 4.8. Étant donnée une partition $\mathcal{X} \in \mathfrak{P}(\Omega)$, la *réduction de complexité* ΔC est la différence entre la complexité de $\mathcal{X}$ et celle de la partition microscopique : $\Delta C(\mathcal{X}) = C(\mathcal{P}_\perp) - C(\mathcal{X})$

ΔC est une mesure de qualité :

- *positive* (une partition est « bonne » lorsqu'elle réduit la quantité de ressources nécessaires à l'exploitation de la représentation associée) ;
- *croissante* (plus une partition est agrégée, plus elle réduit le coût de l'analyse).

Dans cette thèse, nous utilisons une mesure de complexité relativement simple : la *taille* de la représentation[6], c'est-à-dire le nombre de valeurs représentées. En termes de ressources, cette mesure est cohérente avec plusieurs contextes d'analyse :

1. La taille d'une représentation est proportionnelle à *la quantité d'espace mémoire nécessaire à son encodage* (en supposant que chaque valeur nécessite le même espace mémoire). L'agrégation est alors interprétée comme un processus de compression de données visant à l'encodage efficace des représentations lors du passage à l'échelle. Dans ce contexte, d'autres mesures, telle que la complexité de Kolmogorov [Kol65] (théorie algorithmique de l'information) ou l'entropie de Shannon [Sha48] (théorie probabiliste de l'information), pourraient être utilisées pour estimer l'espace mémoire nécessaire à l'encodage. Ces mesures consti-

[5] Le *rapport* de deux complexités n'indique pas la quantité de ressources effectivement économisées : « 10 fois moins de ressources » ne désigne pas la même chose lorsqu'on parle d'*un million* de ressources ou de *cent* ressources. Afin de comparer entre eux les processus d'agrégation et d'exprimer les échelles auxquelles ils appartiennent, nous mesurons donc la *différence* entre les deux complexités : dans ce cas, 900 000 ou 90 ressources.

[6] La *taille* figure dans la liste des mesures de complexité dressée par Edmonds (*cf.* [Edm99], page 157). Bien qu'il précise que « la taille ne semble pas être une condition suffisante pour la complexité, » nous nous plaçons dans un tout autre contexte : nous ne cherchons pas à mesurer la complexité d'un *système*, mais la complexité d'une *représentation*. Or, un système simple peut être représenté de manière complexe, ce qui explique pourquoi la complexité d'une représentation n'implique pas la complexité du système.

tuent d'intéressantes perspectives de recherche. Cette thèse se concentre néanmoins sur la taille des représentations, plus simple à interpréter.

2. La taille d'une représentation est proportionnelle au *temps de calcul nécessaire pour parcourir les données qu'elle contient*. Plus généralement, elle est proportionnelle au temps de calcul nécessaire pour réaliser tout autre processus de traitement à complexité algorithmique *linéaire* (en particulier, la plupart des techniques de visualisation classiques). Si l'analyse fait intervenir des traitements à complexité algorithmique *polynomiale*, la mesure de complexité doit être adaptée en fonction[7].

3. La taille d'une représentation correspond au *nombre de paramètres du modèle statistique associé* [Aka74]. On considère alors que la représentation agrégée modélise la distribution des unités atomiques au sein de la population. Or, lors de la sélection de modèles statistiques, le nombre de paramètres est une mesure classique que l'on cherche à minimiser pour favoriser la sélection de modèles simples (*cf.* par exemple le critère AIC [Aka73, Aka74]).

Définition 4.9. La *taille* $T(\mathcal{X})$ d'une représentation $v(\mathcal{X})$ est le nombre de valeurs qu'elle contient. Elle correspond à la taille de la partition $\mathcal{X}$:

$$T(\mathcal{X}) = |\mathcal{X}|$$

Définition 4.10. La *réduction de taille* $\Delta T(\mathcal{X})$ induite par l'agrégation d'une partition $\mathcal{X}$ est :

$$\Delta T(\mathcal{X}) = |\Omega| - |\mathcal{X}|$$

Elle caractérise (entre autre) le nombre de paramètres « économisés » lors de l'analyse de la représentation agrégée $v(\mathcal{X})$ au lieu de l'analyse de la représentation microscopique $v(\mathcal{P}_\perp)$.

Notons que ΔT est additivement décomposable (*cf.* sous-section 4.2.3) :

$$\Delta T(\mathcal{X}) = \sum_{X \in \mathcal{X}} (|X| - 1)$$

[7] Notons cependant que les polynômes de la taille des représentations $|\mathcal{X}|^k$ ne sont pas *décomposables* (*cf.* section 4.2). De telles mesures de complexité ne peuvent donc pas être optimisées par l'algorithme présenté dans le chapitre 6.

Exemple. Dans la figure 4.1, les représentations agrégées modélisent la distribution microscopique des unités atomiques. La partition $\mathcal{A}$ permet de passer de $T(\mathcal{P}_\perp) = 6$ paramètres à $T(\mathcal{A}) = 5$ paramètres. L'agrégation induit donc la suppression de $\Delta T(\mathcal{A}) = 1$ paramètre, signifiant qu'il y a une valeur de moins à encoder, à visualiser et à traiter lors de l'analyse. Dans le cas de la partition $\mathcal{B}$, nous avons $\Delta T(\mathcal{B}) = 4$ paramètres supprimés. La partition $\mathcal{B}$ est donc meilleure que la partition $\mathcal{A}$ pour réduire le coût de l'analyse : elle passe mieux à l'échelle dans le cas de grands systèmes.

De manière générale, la taille de la représentation définit assez simplement sa *granularité*. Dans les chapitres applicatifs constituant la troisième partie de cette thèse, nous montrons que cette mesure de complexité est notamment cohérente avec les techniques de visualisation développées dans le domaine des systèmes de calcul (représentations *treemap*, *cf.* chapitre 7) et en sciences sociales (représentations cartographiques, *cf.* chapitre 8). Cependant, en perspective, l'approche pourra être généralisée à d'autres mesures de complexité selon le contexte d'analyse.

4.3.2 La divergence comme perte d'information

La réduction de complexité engendre, en contrepartie, une perte d'information (*cf.* section 4.1). Lorsque les données agrégées sont interprétées par l'observateur, la représentation microscopique $v(\mathcal{P}_\perp)$ est approchée par la représentation redistribuée $v_{\mathcal{X}}(\mathcal{P}_\perp)$. Elles constituent donc respectivement la *source* du processus d'agrégation et le *modèle* qui en résulte. La notion d'*information* désigne les indications fournies par le modèle à propos de la source. La *perte d'information* est la quantité d'information qui n'est plus disponible par l'observateur suite au processus d'agrégation.

Définition 4.11. Une *mesure de perte d'information* L est une mesure de qualité exprimant la quantité d'information perdue lors de l'agrégation des données. Elle permet d'évaluer le contenu informationnel d'une représentation redistribuée $v_{\mathcal{X}}(\mathcal{P}_\perp)$ (*modèle*) par rapport à la représentation microscopique $v(\mathcal{P}_\perp)$ (*source*). Une telle mesure est :

- *négative* (une partition est « bonne » lorsqu'il y a peu de pertes entre la source et le modèle) ;
- *croissante* (plus une représentation est agrégée, plus il y a de pertes).

La divergence de Kullback-Leibler est une mesure classique utilisée par la théorie de l'information pour comparer des distributions de probabilité [KL51]. Elle a plusieurs interprétations cohérentes avec le processus d'agrégation, justifiant son utilisation pour quantifier la perte d'information :

1. La divergence est tout d'abord une mesure de *dissimilarité*[8] asymétrique entre deux distributions de probabilité (la source et le modèle). Elle mesure donc la ressemblance des représentations et permet de s'assurer que les données interprétées sont proches des données initiales. D'autres mesures de dissimilarité communément utilisées sont définies par la somme de distances entre les valeurs des distributions (*e.g.*, distance de Manhattan [WF94b, Cui07], distance euclidienne [Cui07], autres distances de Minkowski, distance du χ^2 [WF94b] et autres adaptations de tests statistiques). Cependant, il est difficile de leur donner une sémantique cohérente vis-à-vis du processus d'agrégation dans la mesure où elles ne traitent pas les valeurs comme des dénombrements d'unités atomiques, mais comme des points dans un espace euclidien quelconque. La divergence a l'avantage d'exprimer la dissimilarité en termes d'*information* relative au dénombrement des unités.

2. Plus précisément, la divergence de Kullback-Leibler mesure la quantité d'information « gaspillée » lorsqu'on utilise un modèle de distribution pour déterminer la valeur d'une variable aléatoire [KL51]. Dans notre cas, il s'agit du nombre de bits supplémentaires nécessaires en moyenne pour trouver l'individu associé à une unité (choisie de manière uniforme) lorsqu'on utilise la représentation redistribuée au lieu de la représentation microscopique. La divergence est à ce titre interprétée comme une perte d'information. Il existe cependant d'autres manières de quantifier l'information manipulée par le processus d'agrégation : l'*entropie conditionnelle* [ASZA11] mesure la quantité d'information microscopique qui *n'est pas transmise* lors de l'agrégation ; l'*information de désagrégation* [LPVD12] mesure la quantité d'information nécessaire pour *renverser* le processus ; dans ce contexte, la *divergence* mesure alors l'*irréversibilité* du processus, c'est-à-dire le supplément d'information nécessaire pour renverser le processus lorsqu'on dispose du modèle et de l'information non-transmise (pour plus de détails concernant ces mesures et leurs interprétations, *cf.* annexe A).

[8] Notons que la notion de dissimilarité est ici différente de celle communément utilisée pour le partitionnement de données dans la mesure où nous sommes intéressés par la dissimilarité entre l'objet macroscopique (modèle) et les objets microscopiques (source), et non par la dissimilarité des objets microscopiques entre eux (*cf.* sous-section 3.1.3).

3. La divergence peut également être interprétée comme une diminution de la vraisemblance (*likelihood*) des représentations [LPVD12]. L'agrégation réduit la probabilité de retrouver la représentation microscopique en associant aléatoirement les unités aux individus, en fonction de la distribution agrégée et de l'hypothèse de redistribution : plus les données sont agrégées, plus on choisit les individus de manière uniforme, moins on a de chance de retrouver les détails du niveau microscopique. Dans l'annexe A, nous montrons que la divergence de Kullback-Leibler mesure cette diminution de probabilité. Ce principe est également exploité pour la sélection de modèles statistiques par les techniques de *maximum de vraisemblance* [Aka73, Aka74, FJ02, Bis06].

Définition 4.12. Étant données une partition $\mathcal{X} \in \mathfrak{P}(\Omega)$ et une hypothèse de redistribution, la *divergence* (en bits/unité) engendrée par $\mathcal{X}$ est donnée par la formule de Kullback-Leibler [KL51] :

$$D(\mathcal{X}) = \sum_{X \in \mathcal{X}} \sum_{x \in X} p(x) \log_2 \left(\frac{p(x)}{p_{\mathcal{X}}(x)} \right)$$

Notons que D est additivement décomposable.

Définition 4.13. Dans le cas de l'hypothèse de redistribution uniforme, pour tout $x \in \Omega$, nous avons $p_{\mathcal{X}}(x) = \frac{p(X)}{|X|}$ où X est la partie à laquelle appartient l'individu x dans la partition $\mathcal{X}$. La divergence est donc donnée par la formule suivante :

$$D(\mathcal{X}) = \sum_{X \in \mathcal{X}} \sum_{x \in X} p(x) \log_2 \left(\frac{p(x)\ |X|}{p(X)} \right)$$

Dans le cas général, supposons que l'on dispose d'une distribution de probabilité p' utilisée pour interpréter les données – c'est-à-dire pour les redistribuer, nous avons $p_{\mathcal{X}}(x) = \frac{p'(x)\, p(X)}{p'(X)}$ (*cf.* section 3.2.2). La divergence est donc donnée par la formule suivante :

$$D(\mathcal{X}) = \sum_{X \in \mathcal{X}} \sum_{x \in X} p(x) \log_2 \left(\frac{p(x)\ p'(X)}{p(X)\ p'(x)} \right)$$

Exemple. Dans les trois exemples de la figure 4.1, les données agrégées sont interprétées en fonction de l'hypothèse de redistribution uniforme. Nous avons alors $D(\mathcal{A}) = 2,6 \times 10^{-4}$ bits/unité, $D(\mathcal{B}) = 1,2 \times 10^{-3}$ bits/unité et $D(\mathcal{C}) = 1,0 \times 10^{-1}$ bit/unité. Ainsi, l'agrégation des partitions $\mathcal{A}$ et $\mathcal{B}$ induit une perte d'information bien inférieure à celle provoquée par l'agrégation de la partition $\mathcal{C}$ (respectivement 390 fois et 87 fois moins de perte). Pour donner un ordre de grandeur, la perte d'information maximale est de $D(\mathcal{P}_\top) = 2,8 \times 10^{-1}$ bits/unité. Ces valeurs permettent à l'observateur de s'assurer que son interprétation des données est cohérente avec le niveau microscopique. Il est notamment averti que la partition $\mathcal{C}$, responsable de 37% de la perte d'information maximale, supprime des irrégularités microscopiques significatives. Plus de 99% de cette perte d'information est liée à l'agrégation de la partie C_2, au sein de laquelle les individus sont hétérogènes. L'abstraction correspondante est de piètre qualité informationnelle dans la mesure où son contenu peut être mal interprété par l'observateur.

4.3.3 Réaliser un compromis entre réduction de complexité et perte d'information

Dans la section 4.1, le processus d'agrégation a été décrit comme réalisant un compromis entre le coût de l'analyse et le niveau de détails. L'objectif de cette section est de formaliser cette notion à partir des mesures de qualité définies précédemment et d'exploiter ce compromis pour sélectionner les partitions optimales en fonction des attentes de l'observateur.

Définition 4.14. Étant données une mesure de réduction de complexité ΔC et une mesure de perte d'information L, un *compromis de qualité CQ* est une mesure de qualité combinant ces deux mesures :

$$CQ(\mathcal{X}) = f(\Delta C(\mathcal{X}), L(\mathcal{X}))$$

La correspondance linéaire constitue une première façon, simple et cohérente, de mettre en relation les deux mesures. Dans cette thèse, nous proposons donc d'exprimer le compromis par une *combinaison linéaire* de la réduction de complexité et de la perte d'information. Pour généraliser la méthode, d'autres correspondances pourront néanmoins être envisagées en perspective de recherche (rapport des deux mesures, polynôme, fonctions plus complexes).

Afin de pouvoir combiner les mesures de manière linéaire, il est d'abord nécessaire de les normaliser à partir des valeurs maximales $\Delta C(\mathcal{P}_\top)$ et $L(\mathcal{P}_\top)$:

- Le rapport $\frac{\Delta C(\mathcal{X})}{\Delta C(\mathcal{P}_\top)}$ exprime la proportion des ressources économisées par $\mathcal{X}$ parmi toutes celles qui peuvent l'être. On dira par exemple que la partition $\mathcal{X}$ « réalise 10% de la réduction de complexité maximale ».
- Le rapport $\frac{L(\mathcal{X})}{L(\mathcal{P}_\top)}$ exprime la proportion d'information effectivement perdue par rapport à la perte maximale. On dira par exemple que la partition $\mathcal{X}$ « réalise 10% de la perte d'information maximale ».

Définition 4.15. Le *compromis de qualité linéaire* CQL_α est donné par la formule suivante :

$$CQL_\alpha(\mathcal{X}) = \alpha \, \frac{\Delta C(\mathcal{X})}{\Delta C(\mathcal{P}_\top)} - (1-\alpha) \, \frac{L(\mathcal{X})}{L(\mathcal{P}_\top)}$$

où $\alpha \in [0,1]$ est le *coefficient de compromis* permettant d'ajuster la mesure de qualité en pondérant les termes de la combinaison linéaire.

Notons que si ΔC et L sont additivement décomposables, alors CQL_α l'est également.

Comme ΔC est une mesure de qualité positive et L une mesure de qualité négative, CQL_α est une mesure de qualité positive : maximiser le compromis revient à (1) maximiser la réduction de complexité et (2) minimiser la perte d'information. Le coefficient de compromis α permet de donner plus ou moins de poids à l'un ou l'autre de ces deux objectifs. Il exprime donc les priorités de l'observateur concernant l'évaluation des représentations. L'observateur peut ainsi adapter le processus d'abstraction en fonction des ressources dont il dispose et du niveau de détail qu'il souhaite pour l'analyse.

- Pour $\alpha = 0$, nous avons $CQL_0 = -\frac{L(\mathcal{X})}{L(\mathcal{P}_\top)}$: maximiser le compromis revient à minimiser la perte d'information. Dans ce cas, l'observateur veut disposer du plus de détails possibles. La partition optimale est donc la partition microscopique : $\mathfrak{P}^{CQL_0}(\Omega) = \{\mathcal{P}_\bot\}$
- Pour $\alpha = 1$, nous avons $CQL_1 = \frac{\Delta C(\mathcal{X})}{\Delta C(\mathcal{P}_\top)}$: maximiser le compromis revient à maximiser la réduction de complexité. Dans ce cas, l'observateur veut manipuler la représentation la plus simple. La partition optimale est donc la partition macroscopique : $\mathfrak{P}^{CQL_1}(\Omega) = \{\mathcal{P}_\top\}$
- Pour $\alpha \in]0,1[$, le compromis de qualité linéaire est maximisé par une ou plusieurs partitions situées entre ces deux extrema.

Exemple.

Parmi les trois partitions de la figure 4.1, laquelle est la meilleure ?

1. **Évaluation des partitions.** La figure 4.2 donne, pour chacune des trois partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$, ainsi que pour la partition microscopique $\mathcal{P}_\perp$ et pour la partition macroscopique $\mathcal{P}_\top$, la réduction de complexité normalisée (*taille* T) et la perte d'information normalisée (*divergence* D) induites par l'agrégation. Dans la mesure où $\mathcal{P}_\perp < \mathcal{A} < \mathcal{B} < \mathcal{C} < \mathcal{P}_\top$, les qualités croissent dans cet ordre.

2. **Comparaison des partitions.** La figure 4.3 donne le compromis de qualité linéaire CQL_α pour chacune de ces partitions en fonction du coefficient de compromis α choisi par l'observateur. Nous remarquons que les partitions optimisant le compromis dépendent de ce coefficient. Les lignes verticales en pointillés indiquent les valeurs de α pour lesquelles deux partitions sont simultanément optimales (elles réalisent alors le même compromis).

3. **Sélection des partitions.** La figure 4.4 donne les mesures de qualité normalisées des partitions optimisant le compromis CQL_α en fonction du coefficient α. Voici ce que nous pouvons en déduire sur le processus d'abstraction :

   - Pour $\alpha = 0$, la partition microscopique $\mathcal{P}_\perp$ est optimale. Elle n'induit ni réduction de complexité, ni perte d'information.
   - La partition $\mathcal{B}$ permet de réduire considérablement la complexité de la représentation microscopique (60%) sans perdre beaucoup d'information (0,4%). Ainsi, pour $\alpha \in [0.008, 0.644]$, la partition $\mathcal{B}$ constitue un bon compromis représentant les données selon trois agrégats relativement homogènes, mettant en évidence les variations significatives de l'attribut (*cf.* figure 4.1).

$\mathcal{X}$	$\frac{\Delta T(\mathcal{X})}{\Delta T(\mathcal{P}_\top)}$	$\frac{D(\mathcal{X})}{D(\mathcal{P}_\top)}$
$\mathcal{P}_\perp$	**0%**	**0%**
$\mathcal{A}$	**20%**	**0,1%**
$\mathcal{B}$	**60%**	**0,4%**
$\mathcal{C}$	**80%**	**37%**
$\mathcal{P}_\top$	**100%**	**100%**

FIGURE 4.2 – Évaluation des partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$: tableau des qualités normalisées

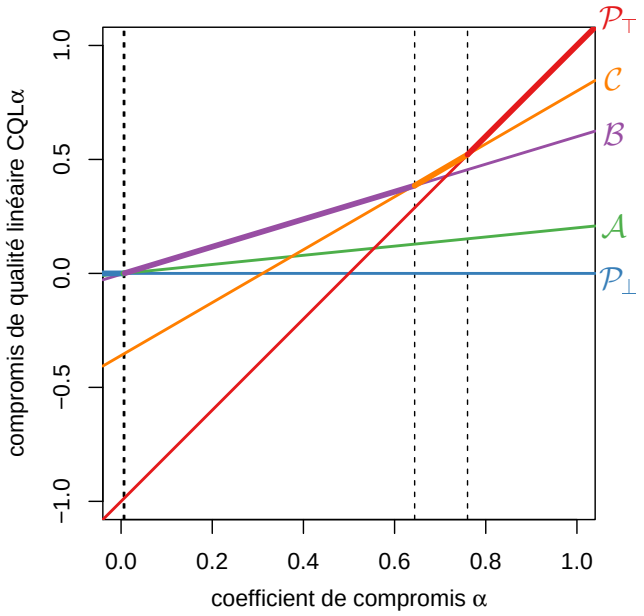


FIGURE 4.3 – Comparaison des partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$: compromis de qualité linéaire CQL_α en fonction du coefficient α spécifié par l'observateur

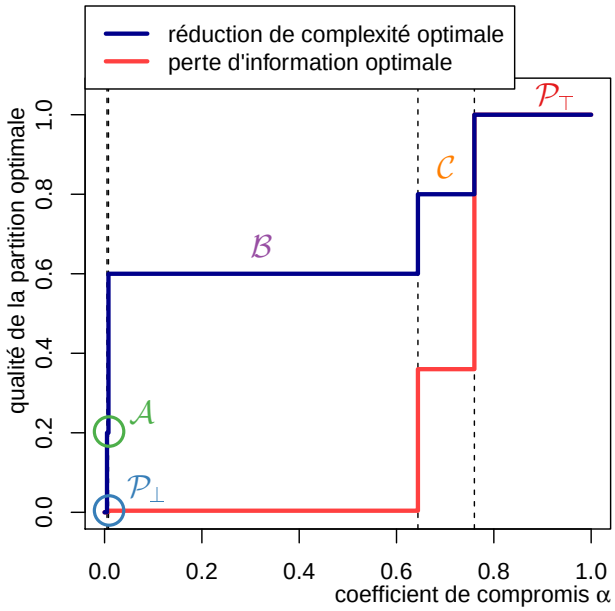


FIGURE 4.4 – Sélection des partitions $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$: qualité normalisée des partitions optimales en fonction du coefficient α spécifié par l'observateur

- Si l'on souhaite réduire un peu plus la taille de la représentation (grâce à la partition $\mathcal{C}$), il faut être prêt à perdre 36% de l'information microscopique. Malgré cette perte importante, ce compromis révèle tout de même une caractéristique intéressante des données : la partition $\mathcal{C}$ révèle une variation significative entre les deux agrégats constitués, de granularité supérieure aux variations mises en évidence par la partition $\mathcal{B}$. L'observateur travaille alors avec un niveau d'abstraction plus élevé.
- Pour $\alpha > 0.760$, la partition macroscopique $\mathcal{P}_{\top}$ est optimale. Elle donne seulement la valeur de l'attribut pour l'ensemble du système.
- Notons que la partition $\mathcal{A}$ n'est optimale que sur une plage très réduite du coefficient α (entre 0.005 et 0.008, à peine visible dans la figure 4.4). En effet, puisque cette partition n'induit pas beaucoup plus de perte d'information que la partition $\mathcal{B}$, mais qu'elle réduit nettement moins la complexité de la représentation, on préfèrera $\mathcal{B}$ à $\mathcal{A}$ dans la majorité des cas.

Pour conclure, l'observateur peut, à l'aide de ces différents outils, choisir la représentation qui convient le mieux aux objectifs de l'analyse et contrôler ainsi le processus d'abstraction. Notons que, dans cet exemple, seules cinq partitions sont évaluées et comparées. Pour une population de 6 individus, il existe en vérité 203 partitions possibles. L'expérience pourrait être reproduite pour sélectionner, parmi elles, les partitions optimales.

4.4 Bilan et perspectives

Les processus d'abstraction ne sont pas des processus triviaux. Au cœur de l'activité scientifique, à la fois nécessaires à l'analyse des grands systèmes et critiques en ce qui concerne interprétation des données, il est crucial de bien comprendre leur fonctionnement et de disposer d'outils de contrôle adaptés. L'objectif de ce chapitre est de fournir une méthode pour (1) évaluer les abstractions engendrées par le processus d'agrégation, (2) comparer ces abstractions et (3) choisir celles qui satisfont au mieux les attentes de l'observateur. Pour ce faire, il est nécessaire de définir des critères – internes au processus de partitionnement – exprimant la *qualité* des représentations macroscopiques pour l'analyse des systèmes.

Dans ce chapitre, nous avons formalisé la qualité des représentations selon deux axes, traduisant chacun un objectif propre à l'agrégation.

1. La *complexité* d'une représentation mesure le coût de l'analyse (*cf.* section 4.1). Afin d'appréhender les grands systèmes et de mettre en place des techniques d'analyse passant à l'échelle, l'agrégation vise à réduire la complexité des représentations. Dans les applications présentées dans la troisième partie de cette thèse, nous utilisons simplement la taille des représentations comme mesure de complexité (*i.e.*, le nombre d'agrégats représentés). En d'autres termes, *une partition est d'autant plus utile qu'elle contient peu de données.*

2. Le *contenu informationnel* d'une représentation mesure la quantité de détails qu'elle contient vis-à-vis du niveau microscopique. En particulier, la *perte* d'information peut nuire à l'analyse dans la mesure où l'agrégation homogénéise les comportements et supprime les variations significatives de l'attribut. Dans cette thèse, nous mesurons la perte d'information par la divergence de Kullback-Leibler [KL51] entre la représentation microscopique, d'une part, et la manière dont sont interprétées les données agrégées, d'autre part. En d'autres termes, *une partition est d'autant plus utile qu'elle contient beaucoup d'information.*

Cependant, nous prétendons développer ici une méthode d'évaluation générique. Premièrement, les notions de complexité et d'information ne se limitent pas aux processus d'agrégation. Elles peuvent également servir aux méthodes d'abstractions en général, telle que l'analyse formelle de concepts [DP02] et d'autres techniques d'apprentissage. Deuxièmement, la méthode d'évaluation que nous proposons peut être adaptée à l'ensemble des critères de qualité mesurables (et décomposables, *cf.* section 4.2). La démarche générale défendue dans ce chapitre consiste à exprimer un compromis entre plusieurs axes d'évaluation, afin de donner des indications complémentaires à l'observateur pour qu'il sélectionne les représentations optimales en fonction du contexte d'analyse. Ce compromis dépend notamment de *paramètres d'ajustement*, permettant de spécifier les pondérations relatives aux différents critères d'évaluation en présence. Cette démarche est ainsi cohérente avec l'acception pragmatiste des phénomènes macroscopiques, selon laquelle la qualité d'une abstraction dépend du contexte de l'analyse et des attentes de l'observateur (*cf.* section 2.3).

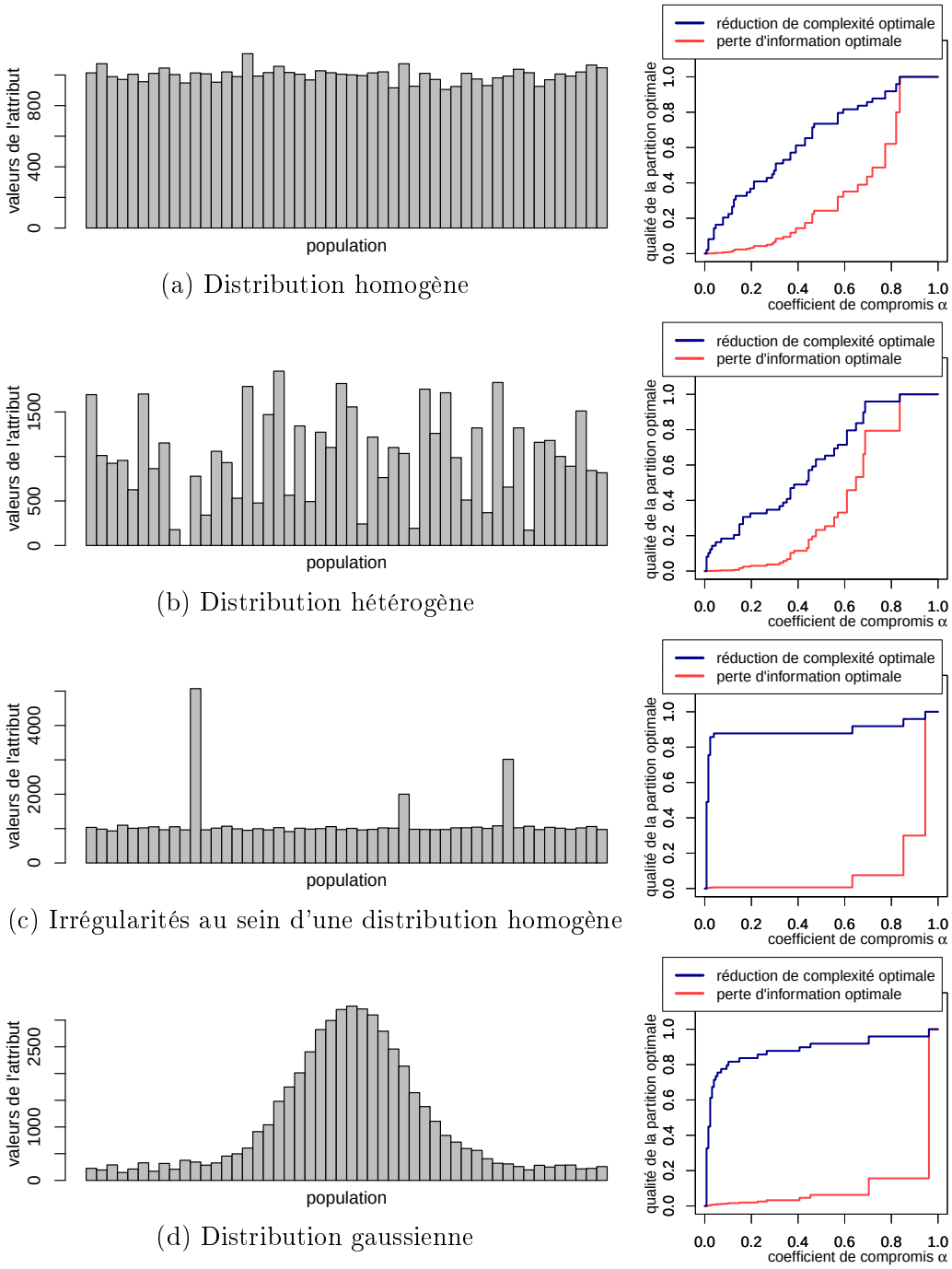
Lister les mesures de qualité et leur contexte d'analyse. Les mesures de qualité définies dans ce chapitre traduisent des objectifs d'analyse particuliers (détection de comportements hétérogènes et passage à l'échelle des méthodes d'analyse, mis en pratique dans les chapitres 7 et 8). Cependant, d'autres mesures peuvent être exploitées en fonction des tâches à réaliser. L'annexe A donne à ce titre des pistes de recherche pour la mise en place d'une batterie de mesures permettant de quantifier l'information manipulée durant les différentes étapes du processus d'agrégation. En perspective de ce travail, il apparaît donc nécessaire de dresser une liste exhaustive des mesures de complexité et d'information cohérentes avec le processus d'agrégation et de préciser les contextes d'analyse au sein desquels elles s'inscrivent.

Caractériser le comportement du système vis-à-vis du processus d'abstraction. Une autre piste de recherche concerne la caractérisation des jeux de données en fonction de la manière dont ceux-ci sont abstraits. Par exemple, le *graphe de qualité* de la figure 4.4 donne le profil du processus d'agrégation appliqué à l'exemple illustrant ce chapitre (*cf.* figure 4.1). Dans ce graphe, les qualités des représentations optimales sont représentées en fonction du coefficient de compromis spécifié par l'observateur. L'étude de tels graphes permet d'identifier plusieurs classes de systèmes :

- ceux dont on peut facilement réduire la complexité sans perdre beaucoup d'information (*e.g.*, distribution homogène, *cf.* figure 4.5a) ;
- ceux dont la complexité ne peut être réduite sans perdre beaucoup d'information (*e.g.*, distribution hétérogène, *cf.* figure 4.5b) ;
- ceux qui « résistent par paliers » à l'agrégation, témoignant d'irrégularités *discrètes* dans le jeu de données (*e.g.*, pics au sein de la distribution, *cf.* figure 4.5c) ;
- ceux qui « résistent progressivement » à l'agrégation, témoignant d'irrégularités *continues* dans le jeu de données (*e.g.*, distribution gaussienne, *cf.* figure 4.5d).

L'objet résultant de l'agrégation n'est pas une représentation unique du système, mais un *ensemble* de représentations ordonnées et distanciées en fonction du coefficient de compromis. Nous pensons que l'étude d'un tel objet permet de donner des indications extrêmement utiles sur le système observé. Il ne s'agit pas seulement d'engendrer une représentation macroscopique pertinente du système, mais de comprendre comment celui-ci se comporte vis-à-vis du processus d'abstraction et d'adapter la démarche scientifique en fonction.

FIGURE 4.5 – Caractériser une distribution à partir du graphe de qualité associé à son agrégation (population de 50 individus ordonnés, *cf.* section 5.3)



(a) Distribution homogène

(b) Distribution hétérogène

(c) Irrégularités au sein d'une distribution homogène

(d) Distribution gaussienne

CHAPITRE 5

Des abstractions cohérentes avec les connaissances externes

Le contrôle du contenu informationnel des représentations macroscopiques ne garantit pas à lui seul leur exploitation pour l'analyse. En effet, la compréhension des systèmes est un processus épistémique complexe qui repose également sur un large éventail de connaissances préliminaires à l'analyse. Les experts ont notamment recours aux modèles, aux connaissances et aux représentations classiques de leur domaine pour interpréter et expliquer les données propres à un cas d'étude particulier. Ainsi, pour qu'il puisse être exploité par les experts, le résultat du processus d'agrégation doit être en adéquation avec l'ensemble de ces outils épistémiques. Il est donc nécessaire, en plus de conserver l'information contenue dans un jeu de données particulier, de conserver ses propriétés *syntaxiques* et *sémantiques*. Ce chapitre discute donc d'un second point crucial concernant la création d'abstractions pertinentes pour l'analyse des grands systèmes, à savoir leur adéquation avec le domaine d'expertise (*cf.* approche *subjectiviste*, section 2.3).

La section 5.1 présente les enjeux liés à l'incorporation de connaissances externes dans le processus d'agrégation. L'approche proposée consiste à définir un ensemble des partitions *admissibles* par l'observateur et servant à *expliquer* les phénomènes observés. Les sections 5.2, 5.3 et 5.4 présentent des ensembles de partitions admissibles exprimant deux propriétés topologiques des populations agrégées : *organisations hiérarchiques*, notamment pour l'agrégation spatiale, et *relations d'ordre*, pour l'agrégation temporelle. Ces contraintes seront appliquées à l'agrégation de systèmes de calcul et du système international dans la troisième partie de cette thèse.

5.1 Prendre en compte les connaissances externes lors de l'agrégation

Les connaissances *externes* désignent l'ensemble des connaissances mobilisées par un expert pour interpréter un jeu de données, mais qui ne sont pas explicitement contenues dans ces données. Elles peuvent être relatives à la *syntaxe* du système, par exemple dans le cas d'un système informatique organisé selon un réseau de communication, ou à la *sémantique* associée par le domaine d'expertise aux entités observées. Par exemple, un géographe étudiant certaines variables démographiques (*e.g.*, population, taux de mortalité, âge médian) procède à partir d'un espace géographique bien connu (*e.g.*, entités territoriales, représentations cartographiques). Cet espace offre un cadre de référence pour importer les connaissances géographiques, politiques, économiques et culturelles nécessaires à la description et l'explication des variables analysées. Une représentation des données ne s'inscrivant pas dans un tel cadre est souvent inexploitable : incohérences avec les schémas explicatifs classiques, difficulté de lier les variables analysées aux connaissances externes, visualisation incompatible avec les modes de représentation classiques. Ainsi, lorsque les données sont partitionnées uniquement en fonction de critères internes, les abstractions engendrées, bien qu'optimales en termes de contenu informationnel, sont inutilisables. Il est donc nécessaire d'introduire un *biais externe* dans le processus de partitionnement (*cf.* sous-section 3.1.3).

Contraindre l'ensemble des partitions admissibles. Le partitionnement contraint (*constrained clustering*) est une technique d'apprentissage semi-supervisé permettant la mise en place de biais externes. Elle consiste à interdire ou à favoriser certains regroupements en fonction de règles logiques formalisées par les experts et exprimant leur connaissance *a priori* du système [DB07]. Cependant, la plupart des travaux du domaine définissent les contraintes au niveau des individus (*instance-level constrained clustering*) : à partir de contraintes de type *must-link* et *cannot-link* [WC00, DB07] ou de règles propositionnelles [TB99] obligeant ou interdisant le regroupement de plusieurs individus. Ces techniques centrées individu ne permettent d'exprimer que des propriétés relativement simples au niveau des agrégats : *e.g.*, diamètre minimal des agrégats, distance minimale entre agrégats [DB07].

Afin d'exprimer des connaissances complexes, nous proposons de travailler directement au niveau des parties $\mathcal{P}(\Omega)$ ou des partitions $\mathfrak{P}(\Omega)$. Les contraintes que nous définissons s'appliquent donc à ces ensembles et en extraient des sous-ensembles de parties ou de partitions *admissibles*.

Définition 5.1. Un *ensemble de partitions admissibles* $\mathfrak{P}_a(\Omega)$ est un sous-ensemble de $\mathfrak{P}(\Omega)$ contenant les partitions en adéquation avec la syntaxe et la sémantique de la population Ω. Ces partitions sont donc organisées en fonction de la connaissance des experts. Les partitions *non-admissibles* sont, au contraire, incompatibles avec le cadre d'analyse et, de ce fait, engendrent des représentations inexploitables.

Définition 5.2. Dans certains cas, $\mathfrak{P}_a(\Omega)$ est engendré à partir d'un *ensemble de parties admissibles* $\mathcal{P}_a(\Omega)$, c'est-à-dire d'un sous-ensemble de $\mathcal{P}(\Omega)$ contenant les parties en adéquation avec la syntaxe et la sémantique de Ω. Ces parties admissibles constituent les « briques » à partir desquelles sont construites les partitions admissibles.
Formellement : $\forall \mathcal{X} \in \mathfrak{P}(\Omega), \quad \mathcal{X} \in \mathfrak{P}_a(\Omega) \quad \Leftrightarrow \quad \forall X \in \mathcal{X}, X \in \mathcal{P}_a(\Omega)$

Expliquer les données à partir des connaissances externes. Les experts mobilisent en permanence des connaissances externes pour interpréter et analyser les données. Ces connaissances constituent une *batterie explicative* permettant de mettre en évidence les causes des phénomènes observés. Par exemple, on peut supposer en géographie que les relations de voisinage entre les pays, leurs échanges économiques et leurs échanges démographiques, peuvent *expliquer* les relations politiques, sociales et culturelles de ces pays. Dans ce cas, les propriétés géographiques du système sont utilisées pour rendre compte de phénomènes sociaux.

Cette stratégie épistémique fait l'hypothèse que les partitions admissibles – définies par des critères *externes* – sont également pertinentes sur le plan informationnel – *cf.* critères *internes* présentés dans le chapitre précédent. Cela signifie notamment que les partitions admissibles ne suppriment pas de détails microscopiques importants : on suppose que les données sont homogènes au sein des partitions admissibles (*cf.* section 4.1). Par exemple, l'*homogénéité* d'une variable démographique (fait interne) est expliquée par la proximité géographique des pays concernés (fait externe). L'*hétérogénéité* est au contraire le symptôme d'un comportement imprévu, d'une anomalie vis-à-vis des connaissances mobilisées. Elle met en évidence les points cruciaux de l'analyse, lorsque les connaissances externes ne suffisent pas à expliquer les données. De plus, lorsque ces connaissances n'engendrent aucune représentation pertinente sur le plan informationnel, elles doivent être considérées comme inadéquates pour l'analyse. La *validation du biais* consiste ainsi à évaluer les connaissances en fonction des données analysées [TB99].

Exploiter les connaissances externes pour optimiser les critères internes. Comme il a été discuté dans la sous-section 3.1.3, l'introduction d'un *biais externe* permet de guider le calcul des partitions optimales [TB99, DB07]. En effet, l'espace de recherche étant réduit à l'ensemble des partitions *admissibles*, les algorithmes d'optimisation procèdent à moins d'évaluations. Comme dans la sous-section précédente, on fait alors l'hypothèse que les critères externes sont en adéquation avec les critères internes. En d'autres termes, on suppose que les partitions pertinentes sur le plan informationnel appartiennent bien à l'ensemble des partitions admissibles définies par les experts. Dans ce cas, le partitionnement contraint permet de calculer rapidement des représentations pertinentes en utilisant les connaissances externes. Le chapitre suivant montre plus en détail comment ces contraintes affectent la complexité algorithmique du problème des partitions optimales.

La suite de ce chapitre présente deux ensembles de partitions admissibles définis à partir de propriétés *topologiques* des populations agrégées, c'est-à-dire de propriétés spatiales essentielles. La section 5.2 présente des contraintes liées à l'organisation hiérarchique des individus. Elles sont notamment utilisées pour l'agrégation d'individus spatialisés. La section 5.3 présente des contraintes liées à la conservation d'un ordre entre les individus. Elles sont notamment utilisées pour l'agrégation temporelle. La section 5.4, enfin, présente l'agrégation simultanée de plusieurs populations par le produit cartésien des partitions admissibles.

5.2 Organisation hiérarchique des systèmes

La notion de hiérarchie est couramment utilisée pour formaliser l'organisation multi-niveau des systèmes. Comme nous l'avons vu dans la sous-section 3.1.1, les *hiérarchies de concepts* servent notamment à l'organisation des connaissances (taxinomies biologiques, hiérarchies de classes en programmation orientée objet et systèmes de classification bibliographique). Selon ces démarches, les entités des niveaux supérieurs sont des *concepts* macroscopiques généralisant les propriétés des niveaux inférieurs. En vue d'un processus d'abstraction par définition d'*objets* (*cf.* sous-section 3.1.2), nous nous intéressons plutôt aux *hiérarchies constitutives* [GQLH12], c'est-à-dire aux hiérarchies dont les entités des niveaux supérieurs sont des objets macroscopiques composés d'objets appartenant aux niveaux inférieurs[1].

[1] Contrairement aux *hiérarchies exclusives*, exprimant des relations de catégories (*e.g.*, hiérarchie de concepts), et aux *hiérarchies inclusives*, où les niveaux supérieurs sont occupés par des objets microscopiques (*e.g.*, hiérarchie d'une entreprise) [GQLH12].

5.2.1 Exemples d'organisations hiérarchiques

En géographie, l'espace est parfois hiérarchisé pour construire des statistiques multi-niveaux à propos des différentes parties du globe : régions, pays, groupes de pays, continents, *etc.* C'est par exemple l'objectif des hiérarchies WUTS [GD07] (*cf.* annexe C) et UNEP [Uni07], qui partitionnent l'espace géographique en régions imbriquées, du niveau national au niveau mondial. Ces hiérarchies sont construites dans le respect de propriétés topologiques élémentaires. Elles conservent notamment la relation de voisinage entre les unités territoriales : par exemple, dans le cas de WUTS et de UNEP, les agrégats sont composés de *pays connexes*. Cependant, ces hiérarchies expriment également des relations non-spatiales, propres aux sciences sociales, pour engendrer une organisation territoriale cohérente avec les connaissances externes du domaine : relations politiques, économiques, culturelles et historiques existant entre les unités territoriales. Ainsi, les agrégats sont des groupes territoriaux cohérents sur le plan *sémantique* et respectant les propriétés *syntaxiques* élémentaires du système (organisation spatiale des unités territoriales).

La structure physique et logicielle de certains systèmes informatiques est également organisée de manière hiérarchique. C'est le cas par exemple de la grille de calcul GRID'5000 [BCC+06] : le réseau de communication reliant les ressources de calcul (processus) est alors constitué de machines, regroupées en clusters, eux-mêmes regroupés en sites de calcul. Cette organisation hiérarchique peut également être induite par l'application parallèle exécutée sur de telles plates-formes : distribution hiérarchique des tâches de calcul, réseau d'utilisateurs dans les systèmes pair-à-pair, *etc.*

Lorsque de telles hiérarchies sont utilisées pour interpréter et expliquer les données en fonction de propriétés syntaxiques et sémantiques du système observé, on suppose que le comportement des individus est régi par une telle organisation. En particulier, on suppose que les individus appartenant aux mêmes branches de la hiérarchie ont de fortes chances d'être similaires. Les hiérarchies définissent donc *des ensembles imbriqués d'individus homogènes.* La hiérarchie WUTS a par exemple été construite dans cette optique : les pays sont regroupés en fonction de leur proximité géographique et des outils sémantiques du domaine (*e.g.*, politique, économie, histoire, *cf.* annexe C). L'objectif est alors de définir des agrégats cohérents avec les connaissances externes des experts [GD07].

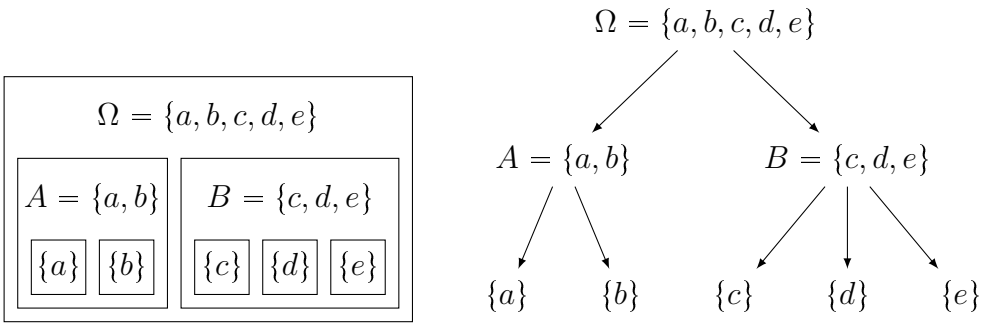
5.2.2 Parties admissibles selon une hiérarchie

Formellement, les *hiérarchies constitutives* [GQLH12] définissent un ensemble de parties imbriquées représentant les agrégats cohérents vis-à-vis de la syntaxe et de la sémantique de la population représentée.

Définition 5.3. Une *hiérarchie* $\mathcal{T}(\Omega)$ organisant une population Ω est un ensemble de parties disjointes ou incluses les unes dans les autres. Ces parties définissent les « briques » à partir desquelles sont construites les partitions admissibles. $\mathcal{T}(\Omega)$ est donc l'*ensemble des parties admissibles*. Formellement :

$$\forall (X_1, X_2) \in \mathcal{T}(\Omega)^2, \quad X_1 \cap X_2 = \varnothing \ \text{ ou } \ X_1 \subset X_2 \ \text{ ou } \ X_1 \supset X_2$$

Une hiérarchie peut également être modélisée par un *arbre*[2]. La figure 5.1 présente donc deux représentations équivalentes (sous forme de « boîtes imbriquées » et sous forme d'arbre) d'une hiérarchie à 3 niveaux organisant une population de 5 individus.

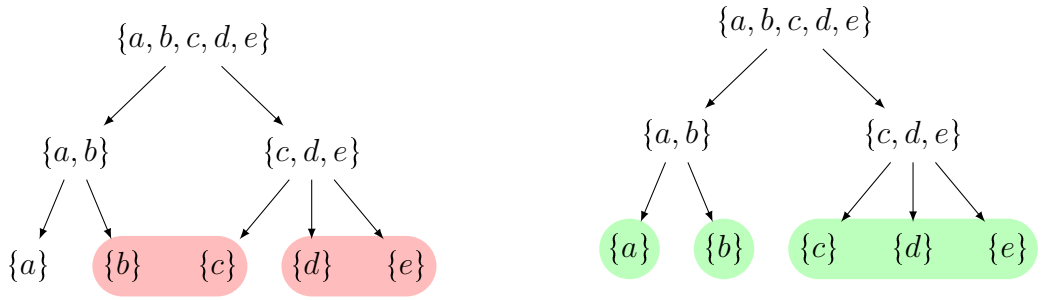


(a) Boîtes imbriquées (b) Arbre

FIGURE 5.1 – Hiérarchie à 3 niveaux : niveau des individus ($\{a\}$, $\{b\}$, $\{c\}$, $\{d\}$ et $\{e\}$), niveau des parties (A et B) et ensemble de tous les individus (Ω)

L'organisation hiérarchique oblige à agréger la population par branches entières. Il est notamment interdit d'agréger deux individus sans agréger également tous les individus appartenant à la plus petite partie commune dans $\mathcal{T}(\Omega)$. Par exemple, dans la figure 5.1, si on agrège les individus b et c, il faut également agréger tous les individus des parties A et B. La figure 5.2 donne d'autres exemples de parties admissibles et non-admissibles.

[2] Il faut pour cela que la hiérarchie ait une « racine », c'est-à-dire que l'union de toutes les parties admissibles soit elle-même une partie admissible. Si les parties admissibles n'ont pas de racine commune, la hiérarchie est modélisée par une *forêt*.



(a) Parties non-admissibles

(b) Parties admissibles

FIGURE 5.2 – Arbre représentant une hiérarchie à 3 niveaux : les parties superposées sur deux branches et celles qui ne s'étendent pas complètement sur une branche ne sont pas admissibles (zones rouges à gauche)

$\mathcal{A} = \{\{a\}, \{b\}, \{c, d, e\}\}$ $\mathcal{B} = \{\{a, b\}, \{c, d, e\}\}$ $\mathcal{C} = \{\{a, b, c, d, e\}\}$

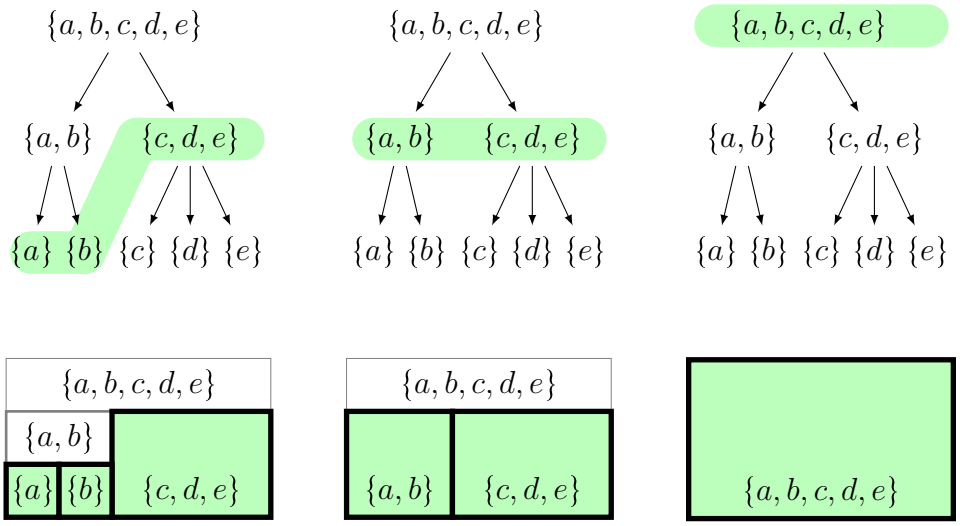


FIGURE 5.3 – Trois partitions hiérarchiques $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$ représentées par des « coupes » dans un arbre et par des « boîtes disjointes »

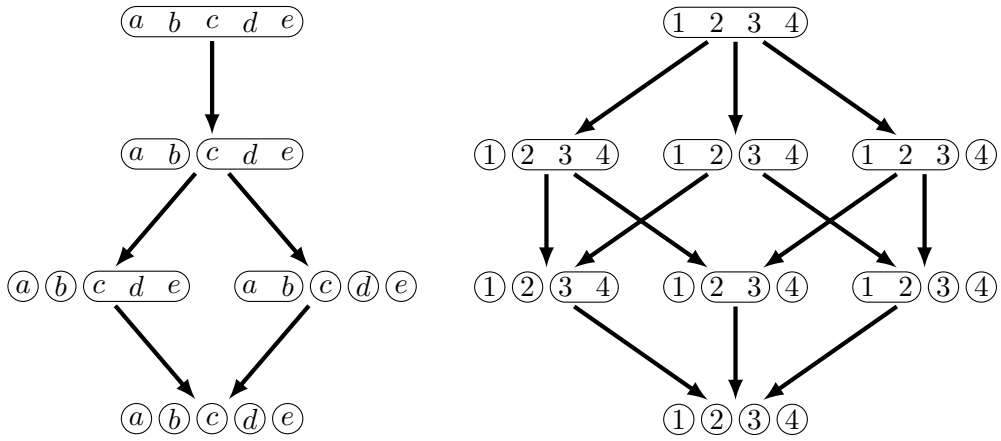
5.2.3 Partitions admissibles selon une hiérarchie

Formellement, une partition est admissible lorsqu'elle est uniquement constituée de parties admissibles.

Définition 5.4. Étant donnée une hiérarchie $\mathcal{T}(\Omega)$, l'*ensemble des partitions admissibles* $\mathfrak{P}_{\mathcal{T}}(\Omega)$ est le sous-ensemble de $\mathfrak{P}(\Omega)$ contenant toutes les partitions construites à partir des parties appartenant à $\mathcal{T}(\Omega)$.
Formellement : $\forall \mathcal{X} \in \mathfrak{P}(\Omega), \quad \mathcal{X} \in \mathfrak{P}_{\mathcal{T}}(\Omega) \quad \Leftrightarrow \quad (\forall X \in \mathcal{X},\ X \in \mathcal{T}(\Omega))$

Une partition admissible correspond à une *coupe* dans l'arbre modélisant la hiérarchie, c'est-à-dire un ensemble de nœuds tels que chaque feuille de l'arbre descend d'un et d'un seul de ces nœuds (*cf.* figure 5.3). Remarquons qu'une telle coupe peut traverser l'arbre à différents niveaux, engendrant ainsi une partition multi-niveaux. Il est ainsi possible d'utiliser une organisation hiérarchique pour construire des représentations multirésolution, détaillées sur certaines branches importantes pour l'analyse et agrégées sur les branches nécessitant moins de détail.

La figure 5.4a donne la structure algébrique de l'ensemble des partitions hiérarchiques induite par la relation de couverture (*cf.* sous-section 3.2.4). Ce diagramme montre que, dans le cas d'une organisation hiérarchique, une « désagrégation atomique » consiste à descendre *un* nœud de la partition d'*un* niveau dans la hiérarchie.



(a) Population hiérarchique (b) Population ordonnée

FIGURE 5.4 – Diagramme de Hasse de l'ensemble des partitions admissibles ordonné selon la relation de couverture

5.3 Organisation ordonnée des systèmes

Les populations modélisant les dimensions temporelles ou causales du système sont naturellement ordonnées. La notion de « proximité temporelle » constitue alors un facteur explicatif reposant sur l'hypothèse que des individus proches dans le temps ont de grandes chances d'être similaires L'homogénéité est alors la marque de périodes stables du système et l'hétérogénéité témoigne de *ruptures* potentiellement importantes pour l'analyse.

5.3.1 Parties admissibles selon un ordre total

Lorsqu'on agrège des dates, des évènements ou des individus temporellement localisés, la relation d'ordre doit être conservée. Cela consiste à partitionner la population en *intervalles* d'individus.

Définition 5.5. Étant donné un ordre total $<$ sur la population Ω, l'*ensemble des parties admissibles* $\mathcal{P}_<(\Omega)$ contient tous les intervalles d'individus $[x, y]$ définis dans Ω à partir de la relation d'ordre. Ces parties définissent les « briques » à partir desquelles sont construites les partitions admissibles. Formellement :

$$\forall X \in \mathcal{P}(\Omega),\ X \in \mathcal{P}_<(\Omega) \ \Leftrightarrow\ (\forall (x,y) \in X^2,\ \forall z \in \Omega,\ x < z < y \ \rightarrow\ z \in X)$$

5.3.2 Partitions admissibles selon un ordre total

Définition 5.6. L'*ensemble des partitions admissibles* $\mathfrak{P}_<(\Omega)$ selon la relation d'ordre $<$ est le sous-ensemble de $\mathfrak{P}(\Omega)$ contenant toutes les partitions construites à partir des intervalles de Ω. Formellement :

$$\forall \mathcal{X} \in \mathfrak{P}(\Omega), \quad (\forall X \in \mathcal{X}, \quad X \in \mathcal{P}_<(\Omega) \quad \rightarrow \quad \mathcal{X} \in \mathfrak{P}_<(\Omega))$$

La figure 5.5 représente l'ensemble des parties admissibles d'une population ordonnée de 5 individus sous la forme d'une « pyramide d'intervalles ». Chaque niveau de la pyramide donne l'ensemble des intervalles d'une taille donnée. Leurs descendants, c'est-à-dire les intervalles agrégés dans les niveaux inférieurs, sont indiqués par des flèches descendant la pyramide. Une partition admissible est un ensemble de « nœuds » de cette pyramide tel que chaque « feuille » descend d'un et d'un seul de ces nœuds. Deux partitions $\mathcal{A}$ et $\mathcal{B}$, engendrant des représentations multi-niveaux, sont présentées dans cette figure.

La figure 5.4b donne la structure algébrique de l'ensemble des partitions ordonnées induite par la relation de couverture. Dans ce cas, une « désagrégation atomique » consiste à couper *une* partie de la partition en *un* endroit.

$\mathcal{A} = \{\{1,2\},\{3\},\{4,5\}\}$ $\mathcal{B} = \{\{1,2\},\{3,4,5\}\}$

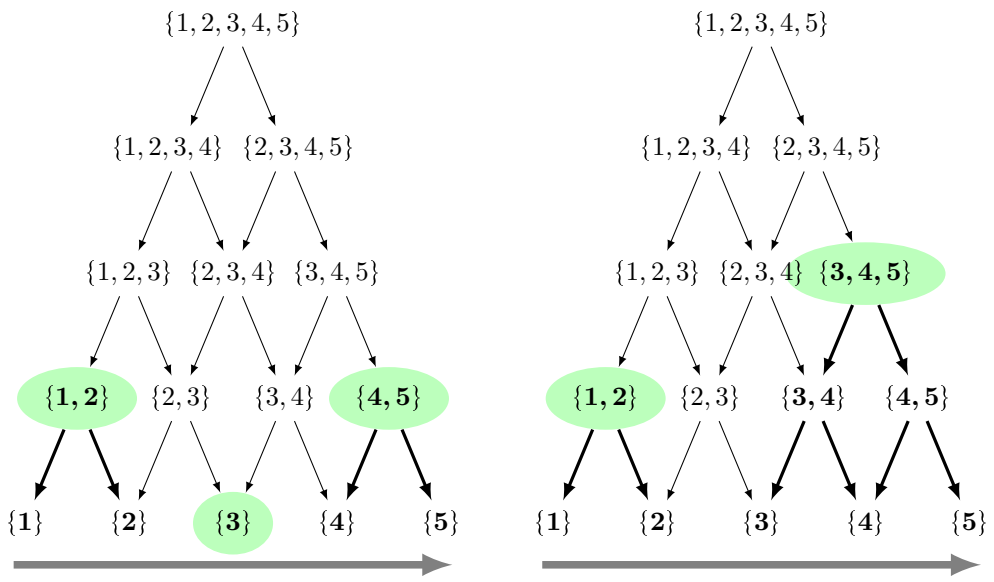


FIGURE 5.5 – Deux partitions ordonnées $\mathcal{A}$ et $\mathcal{B}$ représentées par des « pyramides d'intervalles »

5.4 Agrégation multidimensionnelle

La notion d'agrégation, définie jusqu'à présent sur une seule population, peut être étendue à des représentations multidimensionnelles. Par exemple, les interactions peuvent être représentées par des couples d'individus (émetteur et récepteur) ; l'espace physique, par le croisement de trois dimensions spatiales ; les évènements, par des points de l'espace-temps (une date et un lieu). Pour agréger de tels objets, il est possible de se ramener au cas connu en définissant une population *multidimensionnelle* à partir du produit cartésien des populations concernées.

Définition 5.7. Étant données k populations $\Omega_1, \ldots, \Omega_k$, servant à représenter le système selon plusieurs dimensions, la *population multidimensionnelle résultante* est donnée par le produit cartésien :

$$\Omega = \Omega_1 \times \ldots \times \Omega_k$$

Un *individu multidimensionnel* $x \in \Omega$ est donc un k-uplet d'individus $(x_1, \ldots, x_k) \in \Omega_1 \times \ldots \times \Omega_k$

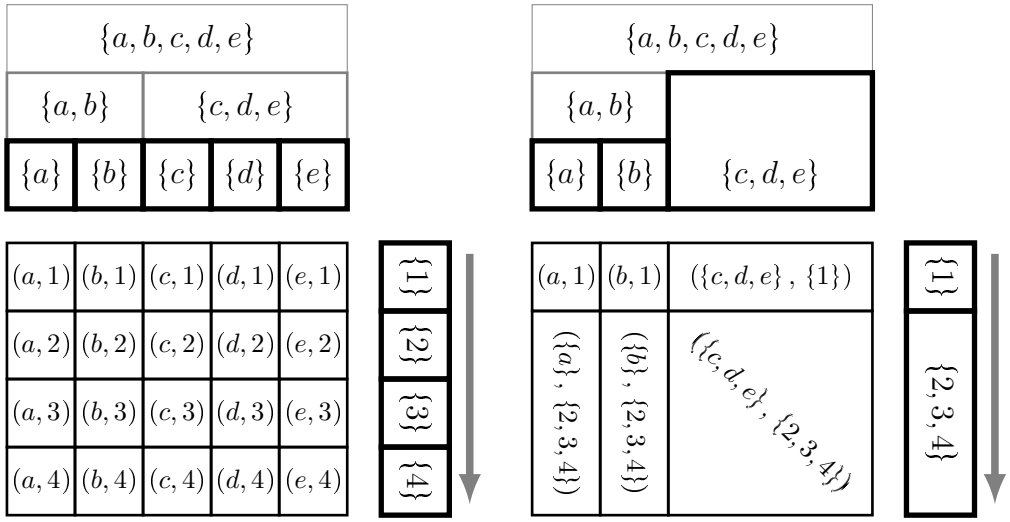
L'ensemble des parties (resp. des partitions) de cette population multidimensionnelle est également défini par le produit cartésien des ensembles des parties (resp. des partitions) des k populations. Une *partie multidimensionnelle* est donc un k-uplet de parties et une *partition multidimensionnelle* un k-uplet de partitions :

$$\begin{aligned} X \in \mathcal{P}(\Omega) &= (X_1, \ldots, X_k) \in \mathcal{P}(\Omega_1) \times \ldots \times \mathcal{P}(\Omega_k) \\ \mathcal{X} \in \mathfrak{P}(\Omega) &= (\mathcal{X}_1, \ldots, \mathcal{X}_k) \in \mathfrak{P}(\Omega_1) \times \ldots \times \mathfrak{P}(\Omega_k) \end{aligned}$$

Dans le cas d'une agrégation bi-dimensionnelle, la population Ω est une matrice ; les parties $\mathcal{P}(\Omega)$ sont des sous-matrices ; les partitions $\mathfrak{P}(\Omega)$ sont des ensembles de sous-matrices disjointes dont l'union est la matrice complète. De plus, chaque dimension peut avoir des propriétés syntaxiques et sémantiques différentes. On dispose alors de k ensembles de partitions admissibles $\mathfrak{P}_a(\Omega_1), \ldots, \mathfrak{P}_a(\Omega_k)$ dont les *partitions multidimensionnelles admissibles* sont des k-uplets :

$$\mathcal{X} \in \mathfrak{P}_a(\Omega) = (\mathcal{X}_1, \ldots, \mathcal{X}_k) \in \mathfrak{P}_a(\Omega_1) \times \ldots \times \mathfrak{P}_a(\Omega_k)$$

La figure 5.6 donne un exemple d'agrégation bi-dimensionnelle couplant une population hiérarchique $\Omega_1 = \{a, b, c, d, e\}$ et une population ordonnée $\Omega_2 = \{1, 2, 3, 4\}$. La population des individus multidimensionnels, donnée par le produit cartésien $\Omega = \{a, b, c, d, e\} \times \{1, 2, 3, 4\}$, est représentée par une matrice. La figure 5.6a donne la représentation microscopique de cette matrice : chaque couple appartenant à Ω est détaillé. La figure 5.6b donne une représentation agrégée, admissible selon la hiérarchie définie sur Ω_1 et selon la relation d'ordre définie sur Ω_2. Il en résulte un partitionnement de la matrice en sous-matrices disjointes de différentes tailles.



(a) Représentation microscopique (b) Représentation agrégée

FIGURE 5.6 – Agrégation d'une population bi-dimensionnelle composée d'une population hiérarchique $\{a, b, c, d, e\}$ et d'une population ordonnée $\{1, 2, 3, 4\}$

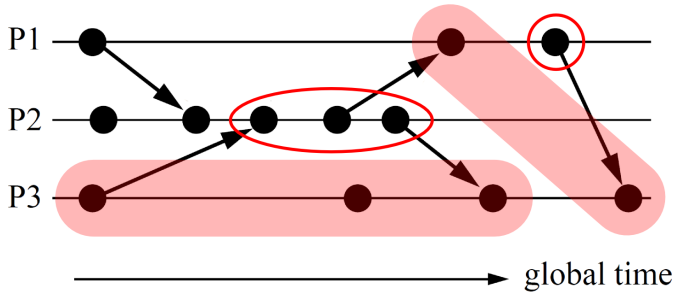
5.5 Bilan et perspectives

Afin d'engendrer des abstractions exploitables par les experts, le processus d'agrégation doit être en adéquation avec le domaine scientifique au sein duquel il s'insère. En particulier, les propriétés syntaxiques (structure du système) et sémantiques (interprétation du système) définies par le domaine servent à expliquer les phénomènes observés. Ces propriétés doivent donc être prises en compte lors de l'agrégation, sous peine de produire des abstractions dénuées de tout pouvoir explicatif. Ce chapitre propose de formaliser ces connaissances externes à partir de contraintes sur l'ensemble des partitions *admissibles*. Une partition est admissible lorsqu'elle peut être utilisée pour l'analyse. Elle constitue dès lors un candidat pour la représentation macroscopique du système.

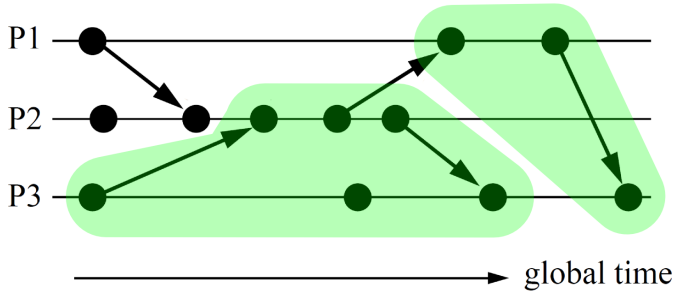
Ce chapitre se concentre sur les propriétés *topologiques* élémentaires des dimensions de l'analyse. Deux classes de partitions admissibles sont présentées, formalisant les connaissances associées à deux types d'organisation des populations : *organisation hiérarchique* (de par la structure physique du système ou de par les connaissances externes associées aux individus) et *relation d'ordre* (dans le cas de l'analyse temporelle). Les propriétés topologiques de ces organisations, telles que la relation de voisinage, servent à contraindre le processus d'agrégation et à engendrer ainsi des représentations cohérentes avec l'organisation syntaxique et sémantique du système.

Bien que l'agrégation de populations hiérarchiques ou ordonnés est exploitable pour l'analyse spatiale et temporelle de nombreux systèmes[3], l'approche présentée dans ce chapitre vise à la généricité. Le reste de cette section présente d'autres ensembles de partitions admissibles permettant d'exprimer des connaissances plus fines sur la topologie des dimensions agrégées.

Partitions admissibles selon un ordre partiel. Les *diagrammes d'évènements*, notamment utilisés dans le domaine du calcul distribué [Mat89], servent à représenter la dynamique des processus, leurs changements d'état et leurs communications (*cf.* figure 5.7). Ils mettent en évidence des relations de causalité complexes entre les évènements du système : transition entre deux états, émissions ou réceptions de messages, chaînes causales. Lorsqu'on agrège ces évènements, il est important de conserver l'*ordre partiel* induit



(a) Parties non-admissibles



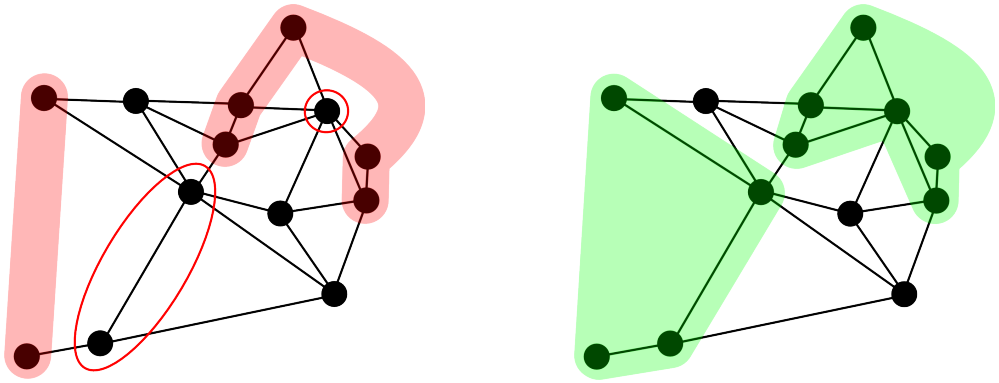
(b) Parties admissibles

FIGURE 5.7 – Diagramme d'évènements (extrait de [Mat89]) : les parties qui excluent des évènements causalement liés ne sont pas admissibles (zones rouges du premier diagramme) ; ces évènements y sont ajoutés pour les rendre admissibles (zones vertes du second diagramme)

[3] Elle sera notamment appliquée à l'agrégation de systèmes de calcul et de systèmes géographiques dans la troisième partie de cette thèse.

par la relation de causalité [LPDV11a]. Les parties admissibles sont, comme dans le cas d'un ordre total, des intervalles d'évènements. Cependant, plusieurs intervalles peuvent être définis de manière parallèle, s'ils ne sont pas liés causalement (*cf.* figure 5.7). Les contraintes correspondantes permettent ainsi de formaliser une *dimension temporelle non-linéaire*.

Partitions admissibles selon un graphe. Plus généralement, les populations peuvent être organisées selon un graphe, représentant une relation quelconque entre les individus : relation de causalité entre deux évènements (ordre partiel), relation de voisinage entre deux individus, réseau de communication, *etc.* Le fait que le graphe soit orienté ou non n'influence pas la nature des parties admissibles. Celles-ci doivent former des ensembles de nœuds *connexes* (*cf.* figure 5.8). L'agrégation selon un ordre total est alors un cas particulier d'agrégation selon un graphe filiforme (« chaîne de nœuds »).

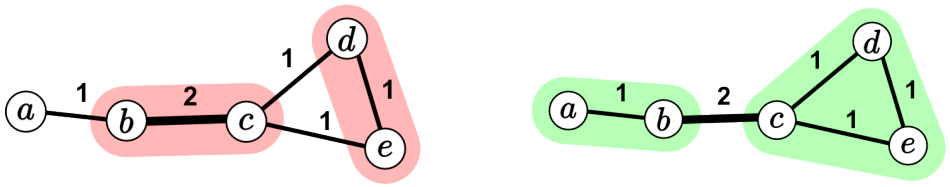


(a) Parties non-admissibles (b) Parties admissibles

FIGURE 5.8 – Graphe de voisinage défini sur une population de 13 individus : les parties non-connexes ne sont pas admissibles (zones rouges du graphe de gauche) ; des nœuds y sont ajoutés pour les rendre admissibles (zones vertes du graphe de droite)

Partitions admissibles selon un graphe pondéré. Une formalisation plus large permet de généraliser la notion de graphe et de hiérarchie. Il s'agit d'agréger les nœuds d'un graphe dont les arêtes sont pondérées, en faisant en sorte que les poids des arêtes *sortant* d'un agrégat soient tous supérieurs aux poids des arêtes *à l'intérieur* de l'agrégat. Les relations d'ordre total et d'ordre partiel sont des cas particuliers pour lesquels les arêtes ont toutes le même poids. Cependant, il est également possible de définir l'ensemble des partitions admissibles selon une hiérarchie à partir d'un tel graphe pondéré.

La figure 5.9 donne le graphe équivalent de la hiérarchie présentée dans la figure 5.1. De nombreuses autres propriétés peuvent être ainsi exprimées, élargissant le champ des connaissances topologiques que l'on souhaite conserver lors de l'agrégation.



(a) Parties non-admissibles (b) Parties admissibles

FIGURE 5.9 – Graphe pondéré défini sur une population de 5 individus (équivalent à la hiérarchie de la figure 5.1) : les parties dont un des poids vers l'extérieur est inférieur ou égal à un poids intérieur ne sont pas admissibles (zones rouges du graphe de gauche)

CHAPITRE 6

Calculer et choisir les meilleures représentations

Les deux chapitres précédents proposent des critères internes et externes afin de garantir que les représentations engendrées lors de l'agrégation sont (1) correctement interprétées par l'observateur, (2) exploitables lors du passage à l'échelle et (3) cohérentes avec les connaissances mobilisées par les experts. Dans ce chapitre, nous abordons la question de la *calculabilité* de telles représentations : étant donnés une mesure de qualité à optimiser (critère interne) et un ensemble de partitions admissibles (critère externe), *comment calculer en un temps raisonnable les partitions admissibles et optimales ?*

Nous nommons ce problème le *problème des partitions admissibles optimales*. La section 6.1 montre que, dans le cas général, la complexité algorithmique de ce problème est **exponentielle** relativement à la taille des populations analysées. La section 6.2 propose néanmoins un algorithme de résolution efficace reposant sur deux hypothèses : (1) les mesures de qualité à optimiser satisfont le *principe d'optimalité*, permettant de décomposer le calcul des partitions optimales, et (2) la *relation de couverture* définie sur l'ensemble des partitions admissibles permet de simplifier le calcul. La section 6.3 montre que la complexité de l'algorithme dépend directement de la structure induite par la relation de couverture : plus l'ensemble des partitions admissibles est contraint, plus la résolution du problème est « facile » sur le plan algorithmique. La complexité de l'algorithme devient par exemple **linéaire** (en temps et en espace) dans le cas de populations hiérarchiques et **quadratique** dans le cas de populations ordonnées.

6.1 Le problème des partitions admissibles optimales

Dans le chapitre 4, nous avons abordé un problème de partitionnement classique, lié à la production d'abstractions *exploitables* pour l'analyse : *quelles partitions* optimisent *une mesure de qualité donnée (critère* interne*) ?* Dans le chapitre 5, nous avons introduit un second problème, lié à la production d'abstractions *compréhensibles* par les experts : *quelles partitions sont* admissibles *pour l'analyse (critère* externe*) ?* L'objectif de ce chapitre consiste à résoudre ces deux problèmes de manière simultanée : *quelles partitions sont à la fois* admissibles *et* optimales *?*

Définition 6.1. Étant donnée une mesure de qualité m et un ensemble de partitions admissibles $\mathfrak{P}_a(\Omega)$, l'*ensemble des partitions admissibles optimales* $\mathfrak{P}_a^m(\Omega)$ est le sous-ensemble de $\mathfrak{P}_a(\Omega)$ contenant les partitions qui optimisent m. Dans le cas où m est une mesure de qualité *positive*,

$$\mathfrak{P}_a^m(\Omega) = \underset{\mathcal{X} \in \mathfrak{P}_a(\Omega)}{\arg\max}\, m(\mathcal{X})$$

Trouver $\mathfrak{P}_a^m(\Omega)$ consiste à résoudre le *problème des*[1] *partitions admissibles optimales.*

La résolution du problème des partitions admissibles optimales nécessite d'évaluer et de comparer entre elles les partitions admissibles. La *complexité algorithmique* de ce problème dépend donc de la structure de l'ensemble des partitions admissibles (critère externe) et des propriétés algébriques de la mesure de qualité à optimiser (critère interne).

Cette section s'intéresse à la complexité algorithmique du problème dans le cas général, c'est-à-dire sans faire de supposition concernant les propriétés algébriques de la mesure de qualité à optimiser. Chaque partition admissible doit alors être évaluée indépendamment pour résoudre le problème. Dans la section suivante, nous verrons que, dans le cas de mesures de qualité *additivement décomposables* (*cf.* sous-section 4.2.3), il est cependant possible de résoudre le problème de manière plus efficace sur le plan algorithmique.

[1] Notons que, dans la plupart des cas, une seule partition appartenant à $\mathfrak{P}_a(\Omega)$ optimise m. Dans ce cas, $\mathfrak{P}_a^m(\Omega)$ est un singleton. Cependant, en théorie, plusieurs optima peuvent exister. On parle donc bien du problème *des* partitions admissibles optimales.

Définition 6.2. Étant donnés une population Ω et un critère externe définissant la notion d'admissibilité, le nombre de partitions admissibles $|\mathfrak{P}_a(n)|$ dépend de la taille n de la population.

De plus, nous notons $E^m(n)$ la complexité de l'évaluation par la mesure m d'une partition quelconque[2]. La complexité algorithmique du problème des partitions admissibles optimales est donc en $\Theta(|\mathfrak{P}_a(n)|\ E^m(n))$.

Dans la suite de cette section, nous calculons $|\mathfrak{P}_a(n)|$ pour les différents ensembles de partitions admissibles présentés dans le chapitre précédent, afin de donner une idée de la complexité algorithmique du problème dans chacun de ces cas.

Nombre de partitions dans le cas non-contraint. Étant donnée une population de taille n pour laquelle toute partition est admissible ($\mathfrak{P}_a(\Omega) = \mathfrak{P}(\Omega)$, aucune syntaxe ou sémantique particulière n'est conservée lors de l'agrégation), le nombre de partitions à évaluer est donné par le n^{e} nombre de Bell [Rot64] :

$$|\mathfrak{P}(n)| = B_n = \sum_{k=0}^{n-1} \binom{n-1}{k} B_k$$

Il a été montré que la suite $(B_n)_{n\in\mathbb{N}}$ est asymptotiquement bornée par la suite exponentielle suivante [BT10] :

$$|\mathfrak{P}(n)| = \Theta\left(\left(\frac{0,792\,n}{\ln(n+1)}\right)^n\right) = \Theta\left(e^{n\,\log n}\right)$$

Le nombre total de partitions d'une population dépend **plus qu'exponentiellement** de la taille de cette population.

Nombre de partitions admissibles selon une hiérarchie. Dans le cas d'une hiérarchie $\mathcal{T}(\Omega)$ (*cf.* section 5.2), le nombre de partitions admissibles $|\mathfrak{P}_{\mathcal{T}}(n)|$ dépend du nombre de niveaux et du nombre de branches dans la hiérarchie. Celui-ci peut être calculé de manière récursive : dans l'arbre représentant la hiérarchie $\mathcal{T}(\Omega)$, les partitions d'une partie $X \in \mathcal{T}(\Omega)$ – représentée par un nœud de l'arbre – sont (1) la partition macroscopique $\{X\}$ ou

[2] Dans le cas de la *divergence de Kullback-Leibler D* (*cf.* sous-section 4.3.2), l'évaluation dépend *linéairement* de la taille de la population. Dans le cas de la *réduction de taille* ΔT (*cf.* sous-section 4.3.1), l'évaluation est réalisée en temps *constant*.

(2) l'union de partitions des sous-parties de X – représentées par les fils du nœud. Le nombre de partitions admissibles est donc donné par la formule récursive suivante :

$$|\mathfrak{P}_{\mathcal{T}}(X)| = 1 + \prod_{Y \in \text{fils}(X)} |\mathfrak{P}_{\mathcal{T}}(Y)|$$

où fils(X) est l'*ensemble des fils de* X dans l'arbre représentant la hiérarchie.

Le nombre maximal de partitions est atteint lorsque l'arbre représentant la hiérarchie est un *arbre binaire complet*. À chaque nœud, le nombre de partitions est multiplié par deux. Supposons que $n = 2^k$, où k est la hauteur de l'arbre, nous avons alors :

$$|\mathfrak{P}_{\mathcal{T}}(2^k)| = U_k = 1 + (U_{k-1})^2 \qquad \text{avec } U_0 = 1$$

La suite $(U_k)_{k \in \mathbb{N}}$ est dominée par[3] :

$$|\mathfrak{P}_{\mathcal{T}}(n)| = O(c^n) \quad \text{avec } c \approx 1,226$$

Notons qu'on trouve des résultats similaires pour des arbres ternaires complets[4] ($c \approx 1,084$), pour des arbres quaternaires complets, *etc.* Ainsi, pour une taille bornée des agrégats constituant la hiérarchie, la classe de complexité reste la même que dans le cas d'un arbre binaire.

Même si le nombre de partitions admissibles dans le cas d'une population hiérarchique est inférieur au nombre total de partitions, il dépend néanmoins **exponentiellement** de la taille de la population.

Nombre de partitions admissibles selon un ordre total. Dans le cas d'une population ordonnée de n individus (*cf.* section 5.3), réaliser une partition de taille k consiste à couper la population en $k-1$ endroits et de former ainsi k intervalles. Le nombre de partitions de taille k est donc $\binom{n-1}{k-1}$ et le nombre total de partitions admissibles selon l'ordre total est :

$$|\mathfrak{P}_{<}(n)| = \sum_{k=0}^{n-1} \binom{n-1}{k} = 2^{n-1}$$

Dans le cas d'une population ordonnée également, le nombre de partitions admissibles dépend **exponentiellement** de la taille de la population.

[3] Voir la formule présentée sur le site *Integer Sequences* : http://oeis.org/A003095 (formule de Benoit Cloitre, 27 novembre 2002).

[4] Voir : http://oeis.org/A135361.

Passage à l'échelle de l'algorithme. Un simple parcours de l'ensemble $\mathfrak{P}_a(\Omega)$ n'est pas exploitable en pratique dans le cas de grands systèmes. En effet, du fait de cette complexité **exponentielle**, un algorithme évaluant une-à-une les partitions admissibles, et renvoyant celles qui optimisent la mesure de qualité, ne passe pas à l'échelle. Pour résoudre le problème des partitions admissibles optimales en un temps raisonnable, il est nécessaire d'établir une stratégie de résolution non-triviale. La section suivante propose un algorithme reposant sur la décomposabilité des mesures pour calculer les partitions optimales avec une complexité **linéaire**, dans le cas des populations hiérarchiques, ou **quadratique**, dans le cas des populations ordonnées.

6.2 Un algorithme de résolution efficace

L'algorithme décrit dans cette section utilise une stratégie de type *diviser pour régner* afin de résoudre le problème des partitions admissibles optimales. Cette stratégie consiste (1) à décomposer le problème en sous-problèmes plus simples et (2) à exécuter récursivement l'algorithme sur ces sous-problèmes, jusqu'à l'obtention de problèmes triviaux (calculs des partitions optimales d'une population de taille 1). Dans l'algorithme que nous proposons, l'étape (1) utilise la *relation de couverture* pour décomposer le problème (*cf.* sous-section 3.2.4) et l'étape (2) la *décomposabilité* des mesures de qualité présentées dans le chapitre 4 pour appliquer l'algorithme aux sous-problèmes (*cf.* sous-section 4.2.3).

6.2.1 Décomposition par la relation de couverture

Définition 6.3. Une décomposition de l'espace de recherche $\mathfrak{P}_a(\Omega)$ consiste à définir des sous-ensembles $\mathfrak{P}_1, \ldots, \mathfrak{P}_k$ tels que :

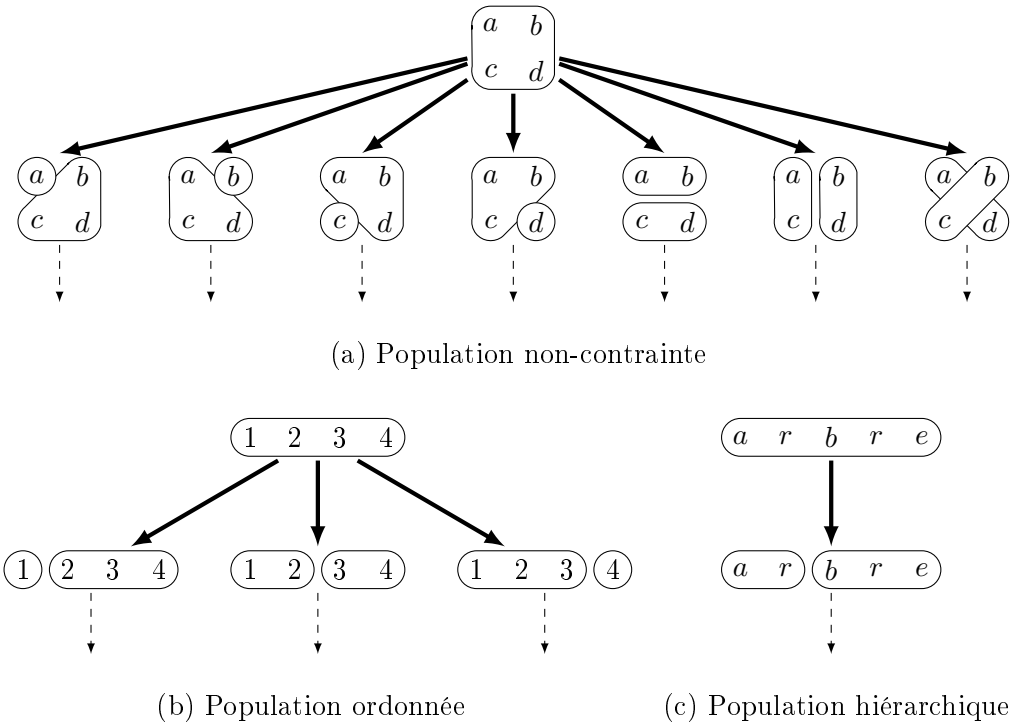
$$\mathfrak{P}_a(\Omega) = \mathfrak{P}_1 \cup \ldots \cup \mathfrak{P}_k$$

Une fois calculées les partitions optimales $\mathfrak{P}_1^m, \ldots, \mathfrak{P}_k^m$ au sein de ces sous-ensembles, nous avons :

$$\mathfrak{P}_a^m(\Omega) = \underset{\mathcal{X} \in \mathfrak{P}_1^m \cup \ldots \cup \mathfrak{P}_k^m}{\arg\max} m(\mathcal{X}) \tag{6.1}$$

La *relation de couverture* introduite dans la sous-section 3.2.4 représente les *désagrégations atomiques* que l'on peut appliquer à une partition donnée, c'est-à-dire les différentes façons de scinder une partition « de manière minimale ». La figure 6.1 donne des exemples de désagrégations atomiques de la partition macroscopique $\mathcal{P}_{\top}(\Omega)$ en fonction de l'ensemble des partitions admissibles définies sur la population Ω :

- dans le cas non-contraint, il s'agit de partitionner la population en *deux* parties quelconques (figure 6.1a) ;
- dans le cas d'une population ordonnée, il s'agit de scinder la partition en *un* endroit pour former deux intervalles (figure 6.1b) ;
- dans le cas d'une population hiérarchique, il s'agit de descendre d'*un* niveau dans la hiérarchie (figure 6.1c).



(a) Population non-contrainte

(b) Population ordonnée

(c) Population hiérarchique

FIGURE 6.1 – Décomposition de l'espace de recherche en fonction des partitions couvertes par la partition macroscopique (les flèches épaisses représentent la relation de couverture)

De cette manière, la structure induite par la relation de couverture peut être exploitée pour décomposer l'espace de recherche. Notons tout d'abord que toute partition appartenant à $\mathfrak{P}_a(\Omega)$ raffine la partition macroscopique $\mathcal{P}_\top$. Nous avons donc :

$$\mathfrak{P}_a(\Omega) = \mathfrak{R}_a(\mathcal{P}_\top) \tag{6.2}$$

où $\mathfrak{R}_a(\mathcal{P}_\top)$ est l'ensemble des partitions admissibles raffinant $\mathcal{P}_\top$. La notion de couverture peut ensuite être utilisée pour décomposer cet ensemble : étant donnée une partition $\mathcal{X}$, une partition raffinant $\mathcal{X}$ est soit *la partition $\mathcal{X}$ elle-même*, soit *une partition raffinant une partition couverte par $\mathcal{X}$*. Formellement :

$$\mathfrak{R}(\mathcal{X}) = \{\mathcal{X}\} \cup \left(\bigcup_{\mathcal{Y} \in \mathfrak{C}(\mathcal{X})} \mathfrak{R}(\mathcal{Y}) \right) \tag{6.3}$$

L'espace de recherche peut donc être décomposé de la manière suivante[5] (*cf.* équations 6.2 et 6.3 ci-dessus) :

$$\mathfrak{P}_a(\Omega) = \{\mathcal{P}_\top\} \cup \left(\bigcup_{\mathcal{Y} \in \mathfrak{C}_a(\mathcal{P}_\top)} \mathfrak{R}_a(\mathcal{Y}) \right) \tag{6.4}$$

Supposons que nous disposions d'une méthode pour calculer $\mathfrak{R}_a^m(\mathcal{Y})$, l'ensemble des partitions admissibles optimales *parmi celles raffinant la partition* $\mathcal{Y}$. Nous avons alors (*cf.* équations 6.1 et 6.4) :

$$\mathfrak{P}_a^m(\Omega) = \underset{\mathcal{X} \in \{\mathcal{P}_\top\} \cup \left(\bigcup_{\mathcal{Y} \in \mathfrak{C}(\mathcal{P}_\top)} \mathfrak{R}_a^m(\mathcal{Y}) \right)}{\arg\max} m(\mathcal{X}) \tag{6.5}$$

Le calcul des partitions admissibles optimales se ramène donc au calcul *des partitions admissibles optimales raffinant les partitions couvertes par la partition macroscopique*. La section suivante présente une méthode pour calculer ces ensembles $(\mathfrak{R}_a^m(\mathcal{Y}))_{\mathcal{Y} \in \mathfrak{C}(\mathcal{P}_\top)}$ de manière récursive.

[5] On suppose ici que la partition macroscopique est admissible : $\mathcal{P}_\top \in \mathfrak{P}_a(\Omega)$. Dans le cas contraire, il suffit d'enlever le terme $\{\mathcal{P}_\top\}$ de la décomposition.

6.2.2 Principe d'optimalité et calcul récursif

Le *principe d'optimalité* [JSB+05] est une propriété algébrique de la mesure à optimiser. Intuitivement, une mesure a cette propriété lorsque la notion d'optimalité est « récursive » : *une partition optimale $\mathcal{X}$ raffinant une partition $\mathcal{Y} = \{Y_1, \ldots, Y_k\}$ est l'union de partitions optimales $\mathcal{X}_1, \ldots, \mathcal{X}_k$ définies sur les parties $Y_1, \ldots, Y_k$, et réciproquement.* Notons que le principe d'optimalité énoncé dans [JSB+05] ne comprend que la première partie de cette proposition, et non sa réciproque. Ainsi, il s'énonce plus facilement : *si une partition $\mathcal{X}$ est optimale sur Ω, alors les partitions $\mathcal{X}_1, \ldots, \mathcal{X}_k$ sont optimales sur $Y_1, \ldots, Y_k$.* Cependant, cette définition ne permet pas de mettre en place une méthode de résolution récursive, dans la mesure où, selon cette seule condition suffisante, *l'union de partitions optimales sur $Y_1, \ldots, Y_k$ n'est pas nécessairement optimale sur Ω.* C'est pourquoi nous adjoignons au principe d'optimalité la condition nécessaire suivante : *si les partitions $\mathcal{X}_1, \ldots, \mathcal{X}_k$ sont optimales sur $Y_1, \ldots, Y_k$, alors leur union $\mathcal{X}$ est optimale parmi les partitions raffinant $\mathcal{Y} = \{Y_1, \ldots, Y_k\}$.* Dans l'encadré suivant, nous donnons une formalisation plus simple – mais équivalente – de cette propriété, pour $k = 2$.

Définition 6.4. Une mesure de qualité m satisfait *le principe d'optimalité* si et seulement si, pour tout couple de parties Y_1 et Y_2 partitionnant Ω (formellement : $Y_1 \cap Y_2 = \varnothing$ et $Y_1 \cup Y_2 = \Omega$) et pour tout couple de partitions $\mathcal{X}_1$ et $\mathcal{X}_2$ respectivement définies Y_1 et Y_2 ($\mathcal{X}_1 \in \mathfrak{P}(Y_1)$ et $\mathcal{X}_2 \in \mathfrak{P}(Y_2)$), l'union $\mathcal{X}_1 \cup \mathcal{X}_2$ est optimale parmi les partitions raffinant $\{Y_1, Y_2\}$ si et seulement si $\mathcal{X}_1$ et $\mathcal{X}_2$ sont optimales sur Y_1 et Y_2.

Formellement, m satisfait le principe d'optimalité si et seulement si, pour tout Y_1, Y_2, $\mathcal{X}_1$ et $\mathcal{X}_2$ définis tel que précédemment, nous avons :

$$\mathcal{X}_1 \cup \mathcal{X}_2 \in \mathfrak{R}^m(\{Y_1, Y_2\}) \quad \Leftrightarrow \quad \mathcal{X}_1 \in \mathfrak{P}^m(Y_1) \quad \text{et} \quad \mathcal{X}_2 \in \mathfrak{P}^m(Y_2) \tag{6.6}$$

Si le principe d'optimalité est vérifié pour toute partition de Ω en deux parties Y_1 et Y_2, il est facile de montrer qu'il est également vérifié pour toute partition $\mathcal{Y} = \{Y_1, \ldots, Y_k\} \in \mathfrak{P}(\Omega)$. Pour tout k-uplet de partitions $\mathcal{X}_1, \ldots, \mathcal{X}_k$ définies sur $Y_1, \ldots, Y_k$, nous avons alors :

$$\mathcal{X}_1 \cup \ldots \cup \mathcal{X}_k \in \mathfrak{R}^m(\{Y_1, \ldots, Y_k\}) \quad \Leftrightarrow \quad \forall i \in \{1, \ldots, k\}, \quad \mathcal{X}_i \in \mathfrak{P}^m(Y_i)$$

Le calcul des partitions optimales peut ainsi être réalisé de manière récursive : si nous connaissons les partitions optimales sur des parties de la population, nous pouvons facilement en déduire les partitions optimales raffinant la population. Il suffit pour cela de prendre le *produit cartésien* des ensembles de partitions.

Étant données une mesure de qualité m qui satisfait le principe d'optimalité et une partition $\mathcal{Y} = \{Y_1, \ldots, Y_k\}$ de la population Ω, l'ensemble des partitions optimales raffinant $\mathcal{Y}$ est donné par le produit cartésien des ensembles de partitions optimales sur les parties $Y_1, \ldots, Y_k$:

$$\mathfrak{R}^m(\mathcal{Y}) \;=\; \mathfrak{P}^m(Y_1) \bigtimes \ldots \bigtimes \mathfrak{P}^m(Y_k) \tag{6.7}$$

Théorème. Une mesure de qualité *additivement décomposable* (*cf.* sous-section 4.2.3) satisfait le principe d'optimalité.

Preuve. Soient m une mesure de qualité additivement décomposable, $\{Y_1, Y_2\}$ une partition de Ω et $\mathcal{X}_1$ et $\mathcal{X}_2$ des partitions de Y_1 et Y_2. Montrons que la proposition 6.6 est vérifiée. La preuve repose sur le fait que :

$$m(\mathcal{X}_1 \cup \mathcal{X}_2) = m(\mathcal{X}_1) + m(\mathcal{X}_2)$$

Condition suffisante. Supposons que $\mathcal{X}_1 \cup \mathcal{X}_2 \in \mathfrak{R}^m(\{Y_1, Y_2\})$. Pour toute partition $\mathcal{X}'_1 \cup \mathcal{X}'_2 \in \mathfrak{R}(\{Y_1, Y_2\})$, nous avons donc $m(\mathcal{X}_1 \cup \mathcal{X}_2) \geqslant m(\mathcal{X}'_1 \cup \mathcal{X}'_2)$. Raisonnons par l'absurde et supposons que $\mathcal{X}_1 \notin \mathfrak{P}^m(Y_1)$. Il existe donc une partition $\mathcal{X}'_1 \in \mathfrak{P}(Y_1)$ telle que $m(\mathcal{X}'_1) > m(\mathcal{X}_1)$. Nous avons donc $m(\mathcal{X}'_1 \cup \mathcal{X}_2) = m(\mathcal{X}'_1) + m(\mathcal{X}_2) > m(\mathcal{X}_1) + m(\mathcal{X}_2) = m(\mathcal{X}_1 \cup \mathcal{X}_2)$. Or, $\mathcal{X}'_1 \cup \mathcal{X}_2 \in \mathfrak{R}(\{Y_1, Y_2\})$. Contradiction. De même si on suppose que $\mathcal{X}_2 \notin \mathfrak{P}^m(Y_2)$. Nous avons donc $\mathcal{X}_1 \in \mathfrak{P}^m(Y_1)$ et $\mathcal{X}_2 \in \mathfrak{P}^m(Y_2)$. □

Condition nécessaire. Supposons que $\mathcal{X}_1 \in \mathfrak{P}^m(Y_1)$ et $\mathcal{X}_2 \in \mathfrak{P}^m(Y_1)$. Pour toutes partitions $\mathcal{X}'_1 \in \mathfrak{P}(Y_1)$ et $\mathcal{X}'_2 \in \mathfrak{P}(Y_2)$, nous avons donc $m(\mathcal{X}_1) \geqslant m(\mathcal{X}'_1)$ et $m(\mathcal{X}_2) \geqslant m(\mathcal{X}'_2)$. Soit une partition $\mathcal{X}' \in \mathfrak{R}(\{Y_1, Y_2\})$. Notons $\mathcal{X}'_1$ et $\mathcal{X}'_2$ les partitions de Y_1 et Y_2 telles que $\mathcal{X}'_1 \cup \mathcal{X}'_2 = \mathcal{X}'$. Nous avons $m(\mathcal{X}_1 \cup \mathcal{X}_2) = m(\mathcal{X}_1) + m(\mathcal{X}_2) \geqslant m(\mathcal{X}'_1) + m(\mathcal{X}'_2) = m(\mathcal{X}')$. Donc $\mathcal{X}_1 \cup \mathcal{X}_2 \in \mathfrak{R}^m(\{Y_1, Y_2\})$. □

Notons que le même raisonnement est valable pour toute mesure dont l'opérateur de décomposition op (*cf.* sous-section 4.2.3) est *croissant* ($x > x' \;\rightarrow\; \mathrm{op}(x, y) > \mathrm{op}(x', y)$). Ainsi, une mesure *décomposable par un opérateur croissant* satisfait le principe d'optimalité. C'est notamment le cas des mesures présentées dans la section 4.3 : la *taille des partitions* et la *divergence de Kullback-Leibler*.

6.2.3 Algorithme des partitions admissibles optimales

La méthode de décomposition (équation 6.4) et la méthode de récursion (équation 6.7) permettent de définir un algorithme de type *diviser pour régner*. Celui-ci repose sur la formule récursive suivante :

$$\mathfrak{P}_a^m(\Omega) = \underset{\mathcal{X} \in \{\mathcal{P}_\top\} \cup \left(\bigcup_{\{Y_1,\ldots,Y_k\} \in \mathfrak{C}_a(\mathcal{P}_\top)} \mathfrak{P}_a^m(Y_1) \times \ldots \times \mathfrak{P}_a^m(Y_k) \right)}{\arg\max} m(\mathcal{X})$$

Ainsi, pour calculer les partitions admissibles optimales $\mathfrak{P}_a^m(\Omega)$, il faut :

- calculer les partitions admissibles couvertes $\mathfrak{C}_a(\mathcal{P}_\top)$;
- calculer les ensembles de partitions admissibles optimales $\mathfrak{P}_a^m(Y_i)$ sur les parties Y_i des partitions $\mathcal{Y} \in \mathfrak{C}_a(\mathcal{P}_\top)$;
- calculer le produit cartésien de ces ensembles de partitions ;
- évaluer les partitions résultantes et la partition macroscopique $\mathcal{P}_\top$;
- garder celles qui optimisent la mesure de qualité m.

Algorithme des partitions admissibles optimales

Entrée : une population Ω, un ensemble de partition admissibles $\mathfrak{P}_a(\Omega)$ et une mesure de qualité positive m satisfaisant le principe d'optimalité

Sortie : l'ensemble des partitions admissibles optimales $\mathfrak{P}_a^m(\Omega)$

1. Initialisation : $\mathfrak{P}_a^m(\Omega) \leftarrow \{\mathcal{P}_\top\}$, avec $\mathcal{P}_\top = \{\Omega\}$
2. Calculer l'ensemble $\mathfrak{C}_a(\mathcal{P}_\top)$ des partitions couvertes par $\mathcal{P}_\top$
3. Pour chaque partition couverte $\mathcal{Y} \in \mathfrak{C}_a(\mathcal{P}_\top)$:
   3.1. Pour chaque partie $Y \in \mathcal{Y}$:
      3.1.1. Appliquer l'algorithme à Y pour calculer $\mathfrak{P}_a^m(Y)$
   3.2. En déduire l'ensemble $\mathfrak{R}_a^m(\mathcal{Y})$ des partitions admissibles optimales raffinant $\mathcal{Y}$ à l'aide du principe d'optimalité (*cf.* équation 6.6) :
   $$\mathfrak{R}_a^m(\mathcal{Y}) = \bigtimes_{Y \in \mathcal{Y}} \mathfrak{P}_a^m(Y)$$
   3.3. Si les partitions appartenant à $\mathfrak{R}_a^m(\mathcal{Y})$ ont une qualité supérieure aux partitions appartenant à $\mathfrak{P}_a^m(\Omega)$, alors $\mathfrak{P}_a^m(\Omega) \leftarrow \mathfrak{R}_a^m(\mathcal{Y})$
4. Renvoyer $\mathfrak{P}_a^m(\Omega)$

Exemple. Afin d'illustrer cet algorithme, la figure 6.2 donne un exemple d'exécution dans le cas d'une population ordonnée de taille 4. La figure se lit de haut en bas et de gauche à droite. Le point de départ est la partition macroscopique (1 2 3 4). Les flèches épaisses (numérotées) indiquent l'ordre d'exécution de l'algorithme, correspondant à la décomposition selon la relation de couverture (*cf.* étape 3 de l'algorithme). Chacune de ces flèches correspond donc à un certain nombre d'appels récursifs (3.1), à un produit cartésien (3.2) et à une comparaison (3.3). Chaque flèche en pointillées (non-numérotée) correspond à un appel récursif (3.1.1). Dans la sous-section suivante, nous montrons que la complexité temporelle de l'algorithme peut être réduite en évitant certains de ces appels.

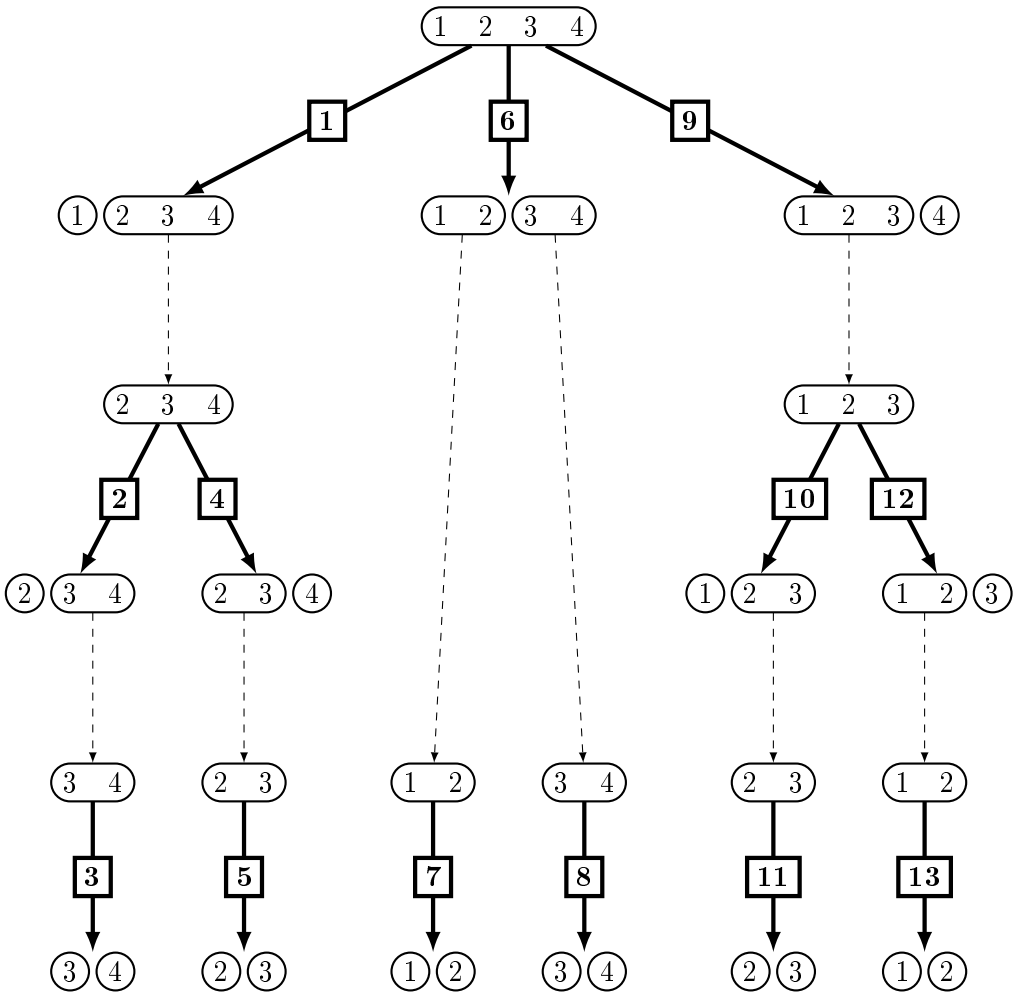


FIGURE 6.2 – Trace d'exécution de l'algorithme des partitions admissibles optimales dans le cas d'une population ordonnée de taille 4

6.2.4 Optimisation de l'algorithme

L'algorithme proposé dans la sous-section précédente résout le problème des partitions admissible optimales. Cependant, il n'est pas optimal en termes de temps de calcul. Dans cette sous-section, nous mettons en évidence les redondances de l'algorithme et nous proposons deux optimisations pour réduire sa complexité temporelle : *stocker les résultats intermédiaires* et *éviter les évaluations redondantes*. Notons que ces optimisations nécessitent un espace mémoire plus important. Ainsi, elles réduisent la complexité *temporelle* de l'algorithme, mais accroissent sa complexité *spatiale*. La section 6.3 quantifie et discute ces deux aspects.

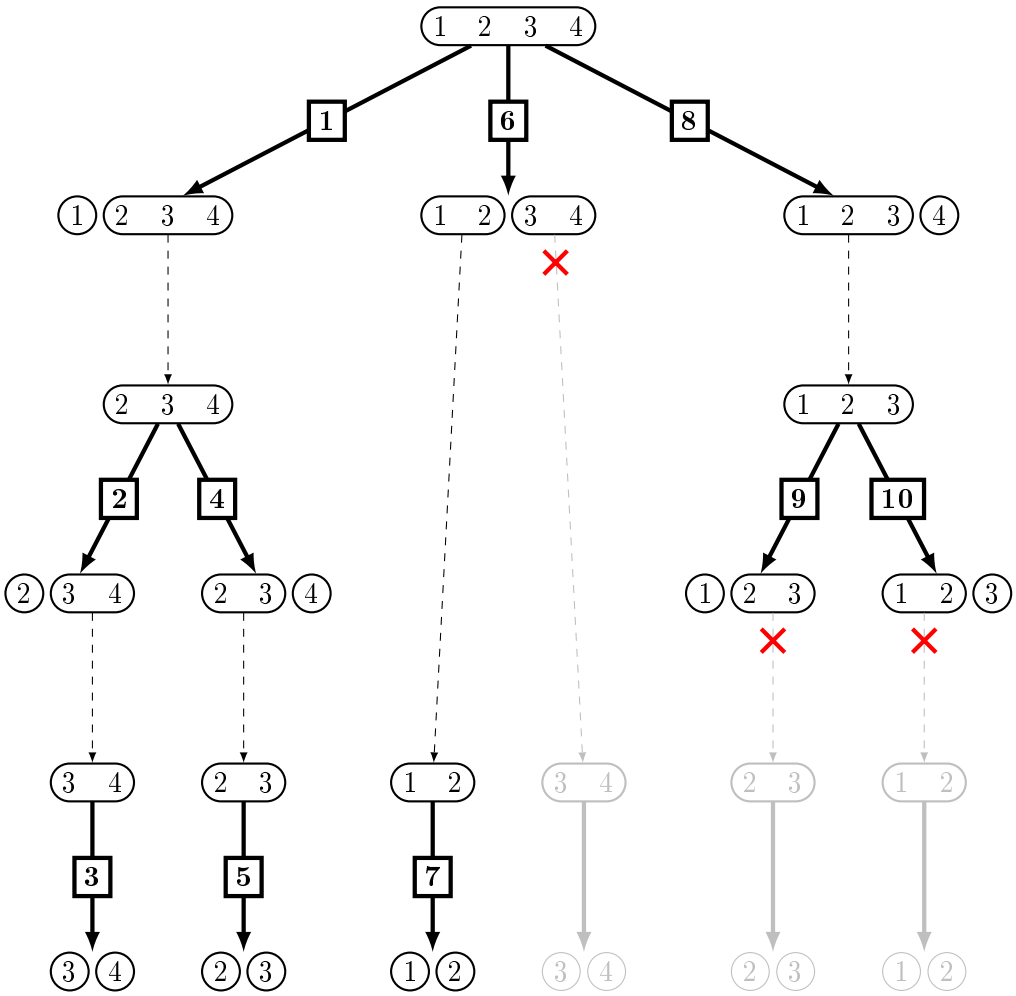


FIGURE 6.3 – Trace d'exécution de l'algorithme stockant les résultats intermédiaires pour éviter des appels récursifs (croix en rouge)

Stocker les résultats intermédiaires. Dans le cas d'une mesure de qualité décomposable, la qualité d'une partie ne dépend pas de la partition au sein de laquelle elle s'insère (*cf.* sous-section 4.2.3). Il est donc possible de garder en mémoire l'ensemble des partitions admissibles optimales $\mathfrak{P}_a^m(Y)$ pour chaque partie $Y \in \mathcal{P}_a(\Omega)$ et de réutiliser ces résultats. Ainsi, l'algorithme n'est exécuté qu'une seule fois par partie admissible. Les appels récursifs ne sont réalisés que si le résultat intermédiaire n'a pas déjà été calculé.

Exemple. Dans la figure 6.2, l'algorithme est appliqué deux fois aux parties $\textcircled{1\ \ 2}$, $\textcircled{2\ \ 3}$ et $\textcircled{3\ \ 4}$. En stockant le résultat lors du premier appel, on peut économiser les appels suivants. La figure 6.3 représente l'exécution résultant de cette optimisation. Les croix en rouge indiquent les appels ainsi évités.

Éviter les évaluations redondantes. Selon l'ensemble des partitions admissibles considéré, la décomposition de l'espace de recherche peut être *redondante* : $\mathfrak{P}_a(\Omega) = \mathfrak{P}_1 \cup \ldots \cup \mathfrak{P}_k$ et il existe i et j tels que $\mathfrak{P}_i \cap \mathfrak{P}_j \neq \varnothing$. Dans le cas de la décomposition que nous proposons (équation 6.4), il existe par exemple deux partitions $\mathcal{Y}_1$ et $\mathcal{Y}_2$ couvertes par la partition macroscopique ($\mathcal{Y}_1 \in \mathfrak{C}_a(\mathcal{P}_\top)$ et $\mathcal{Y}_2 \in \mathfrak{C}_a(\mathcal{P}_\top)$) telles que $\mathfrak{R}_a(\mathcal{Y}_1) \cap \mathfrak{R}_a(\mathcal{Y}_2) \neq \varnothing$. Dans ce cas de figure, les partitions appartenant à l'ensemble $\mathfrak{R}_a(\mathcal{Y}_1) \cap \mathfrak{R}_a(\mathcal{Y}_2)$ sont évaluées plusieurs fois lors de l'exécution de l'algorithme : une fois en tant que raffinement de $\mathcal{Y}_1$ et une fois en tant que raffinement de $\mathcal{Y}_2$.

Exemple. Dans la figure 6.3, la flèche $\boxed{3}$ correspond à l'évaluation et à la comparaison des partitions $\textcircled{3\ \ 4}$ et $\textcircled{3}\textcircled{4}$. Les flèches $\boxed{2}+\boxed{3}$ correspondent donc à l'évaluation des partitions $\textcircled{2\ \ 3\ \ 4}$, $\textcircled{2}\textcircled{3\ \ 4}$ et $\textcircled{2}\textcircled{3}\textcircled{4}$. Ensuite, les flèches $\boxed{4}+\boxed{5}$ correspondent à l'évaluation des partitions $\textcircled{2\ \ 3}\textcircled{4}$ et $\textcircled{2}\textcircled{3}\textcircled{4}$. Ainsi, la partition $\textcircled{2}\textcircled{3}\textcircled{4}$ est évaluée deux fois : une fois avec $\boxed{2}+\boxed{3}$ et une fois avec $\boxed{4}+\boxed{5}$. Par conséquent, la comparaison $\boxed{5}$ n'est pas nécessaire, $\boxed{2}+\boxed{3}+\boxed{4}$ suffisent à évaluer et à comparer les quatre partitions $\textcircled{2\ \ 3\ \ 4}$, $\textcircled{2}\textcircled{3\ \ 4}$, $\textcircled{2\ \ 3}\textcircled{4}$ et $\textcircled{2}\textcircled{3}\textcircled{4}$.

Ces évaluations redondantes peuvent être évitées afin d'améliorer les performances de l'algorithme en temps de calcul. Il suffit pour cela de transmettre, lors des appels récursifs, l'ensemble des partitions couvertes $\mathcal{Y}_1, \ldots, \mathcal{Y}_k$ qui ont été évaluées et comparées lors des appels « supérieurs » (étapes 3.1, 3.2 et 3.3). Ainsi, au sein des appels « inférieurs », lorsqu'une partition couverte $\mathcal{Y}'$ est évaluée (étape 3), on vérifie d'abord que celle-ci ne raffine aucune des partitions $\mathcal{Y}_1, \ldots, \mathcal{Y}_k$ précédemment évaluées. S'il existe une partition $\mathcal{Y}_i$

telle que $\mathcal{Y}' \in \mathfrak{R}_a(\mathcal{Y}_i)$, alors on passe directement à la partition suivante (sans exécuter les étapes 3.1, 3.2 et 3.3). Ainsi, ne sont évaluées que les partitions couvertes qui n'ont pas déjà été évaluées[6]. La figure 6.4 représente l'exécution résultant de cette optimisation. Les croix en bleu indiquent les évaluations évitées.

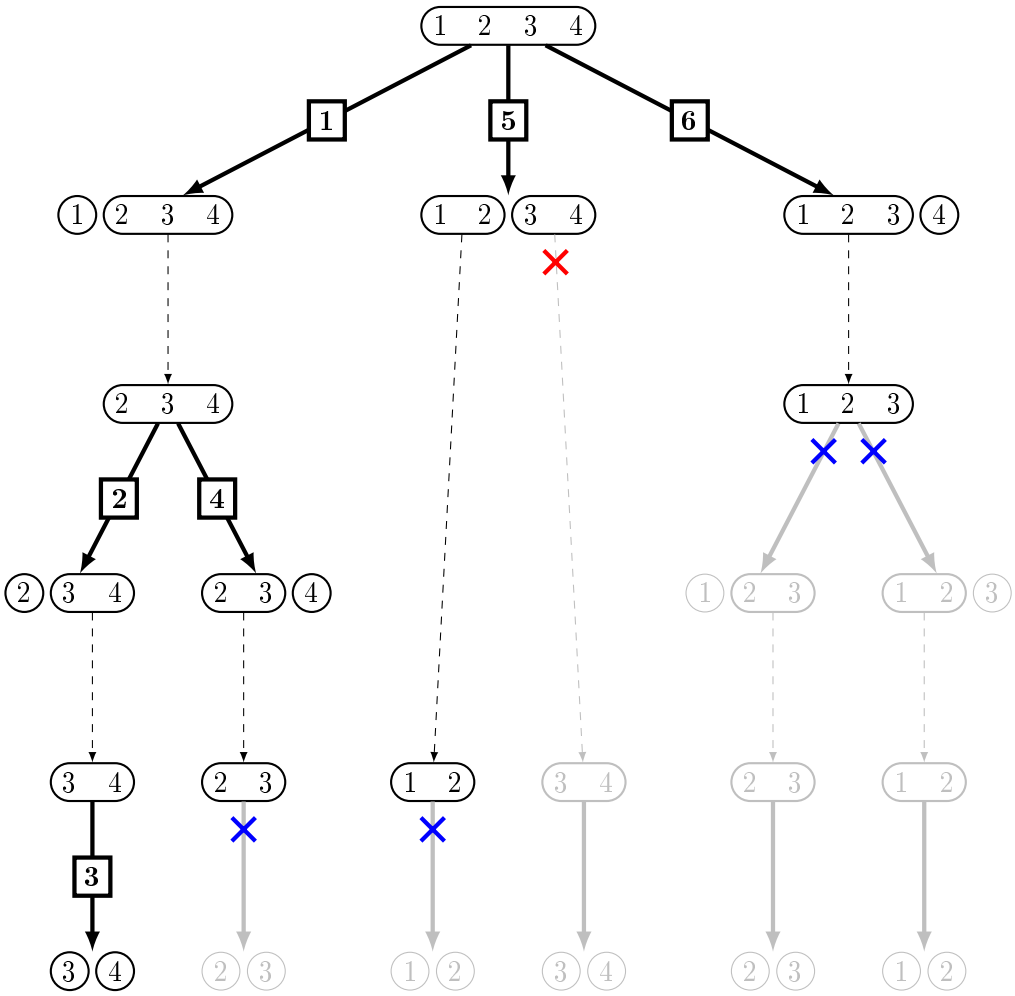


FIGURE 6.4 – Trace d'exécution de l'algorithme stockant les partitions évaluées pour éviter les évaluations redondantes (croix en bleu)

[6] Cette méthode doit être implémentée avec le plus grand soin, sous peine d'être extrêmement coûteuse en temps de calcul. Dans les implémentations spécialisées présentées en section 6.3, le critère d'admissibilité est fixé. On connaît alors la structure induite par la relation de couverture et la méthode peut être implémentée de manière implicite en ne calculant, lors de l'étape 3, que les partitions couvertes qu'il est nécessaire d'évaluer. La méthode de suppression des évaluations redondantes a alors un coût linéaire.

Voici la version optimisée de l'algorithme des partitions admissibles optimales. Les ajouts sont mis en évidence : en **rouge**, pour les ajouts concernant le stockage des résultats intermédiaire, et en **bleu**, pour ceux concernant la surpression des évaluations redondantes.

Dans la section suivante, nous précisons l'implémentation de cet algorithme dans le cas de populations hiérarchiques et de populations ordonnées. Nous donnons également la complexité algorithmique (spatiale et temporelle) de ces implémentations.

Algorithme des partitions admissibles optimales (version optimisée)

Entrée : une population Ω, un ensemble de partitions admissibles $\mathfrak{P}_a(\Omega)$, une mesure de qualité positive m satisfaisant le principe d'optimalité et **l'ensemble des partitions $\mathfrak{E}$ dont les raffinements ont été évalués**

Sortie : l'ensemble des partitions admissibles optimales $\mathfrak{P}_a^m(\Omega)$

Variable globale : l'ensemble des partitions admissibles optimales $\mathfrak{P}_a^m(Y)$ pour chaque partie $Y \in \mathcal{P}_a(\Omega)$ déjà évaluée

0. **Si $\mathfrak{P}_a^m(\Omega)$ a déjà été calculé, renvoyer $\mathfrak{P}_a^m(\Omega)$**
1. Initialisation : $\mathfrak{P}_a^m(\Omega) \leftarrow \{\mathcal{P}_\top\}$, avec $\mathcal{P}_\top = \{\Omega\}$, et **$\mathfrak{E}' \leftarrow \mathfrak{E}$**
2. Calculer l'ensemble $\mathfrak{C}_a(\mathcal{P}_\top)$ des partitions couvertes par $\mathcal{P}_\top$
3. Pour chaque partition couverte $\mathcal{Y} \in \mathfrak{C}_a(\mathcal{P}_\top)$,

   si $\mathcal{Y}$ ne raffine aucune partition de $\mathfrak{E}$:

   3.1. Pour chaque partie $Y \in \mathcal{Y}$:

   3.1.1. Appliquer l'algorithme à Y **avec $\mathfrak{E}'$** pour calculer $\mathfrak{P}_a^m(Y)$

   3.2. En déduire l'ensemble $\mathfrak{R}_a^m(\mathcal{Y})$ des partitions admissibles optimales raffinant $\mathcal{Y}$, à l'aide du principe d'optimalité (*cf.* équation 6.6) :

$$\mathfrak{R}_a^m(\mathcal{Y}) = \bigtimes_{Y \in \mathcal{Y}} \mathfrak{P}_a^m(Y)$$

   3.3. Si les partitions appartenant à $\mathfrak{R}_a^m(\mathcal{Y})$ ont une qualité supérieure aux partitions appartenant à $\mathfrak{P}_a^m(\Omega)$, alors $\mathfrak{P}_a^m(\Omega) \leftarrow \mathfrak{R}_a^m(\mathcal{Y})$

   3.4. **Ajouter $\mathcal{Y}$ à $\mathfrak{E}'$**
4. Renvoyer $\mathfrak{P}_a^m(\Omega)$ **et enregistrer le résultat**

6.3 Implémentations spécialisées de l'algorithme

L'algorithme proposé dans la section précédente est un algorithme générique au sens où il peut être appliqué à tout ensemble de partitions admissibles $\mathfrak{P}_a(\Omega) \subset \mathfrak{P}(\Omega)$. Cependant, en termes d'implémentation, il est préférable de *spécialiser* l'algorithme pour les différentes structures qui nous intéressent. En effet, la manière de représenter les individus, les parties admissibles, les partitions admissibles, la manière de calculer les partitions couvertes, d'évaluer la qualité des partitions et d'éviter les évaluations redondantes, dépendent de l'espace de recherche parcouru. Les structures de données utilisées dans les implémentations spécialisées de l'algorithme sont donc adaptées aux structures algébriques de l'ensemble des partitions admissibles. Bien qu'il soit possible de donner une implémentation générique de l'algorithme, celle-ci ne serait pas aussi performante, en temps de calcul et en espace mémoire, qu'une implémentation spécialisée.

Exemple. Une technique d'encodage *générique* des parties admissibles consiste par exemple à lister les identifiants des individus qu'elles contiennent. Dans ce cas, l'espace mémoire nécessaire à l'encodage d'une partie dépend **linéairement** de sa taille (en supposant que chaque identifiant nécessite un espace mémoire constant). Dans le cas d'une population hiérarchique, une partie admissible peut simplement être représentée par l'identifiant du nœud correspondant dans la hiérarchie et, dans le cas d'une population ordonnée, par les identifiants des premiers et derniers individus de l'intervalle correspondant. Dans ces deux cas, les techniques d'encodage *spécialisées* nécessitent un espace mémoire **constant** et sont donc plus performantes.

L'algorithme des partitions optimales doit donc être considéré comme *un point de départ* pour l'implémentation d'algorithmes spécialisés. Cela consiste à analyser l'exécution de l'algorithme, dans le cas général, pour en déduire un algorithme performant, dans un cas particulier.

Cette section explicite ce processus de spécialisation dans le cas de populations hiérarchiques (sous-section 6.3.1) et de populations ordonnées (sous-section 6.3.2). Le pseudo-code des deux algorithmes spécialisés résultants est fourni en annexe B. Cette section donne également la complexité *spatiale* et *temporelle* de ces deux algorithmes, c'est-à-dire le comportement asymptotique de l'espace mémoire et du nombre d'instructions nécessaires à leur exécution.

6.3.1 Algorithme des partitions hiérarchiques optimales

Exécution de l'algorithme générique. Étant données une population Ω de taille n et une hiérarchie $\mathcal{T}(\Omega)$, pour chaque partie admissible $Y \in \mathcal{T}(\Omega)$, la partition macroscopique correspondante $\{Y\}$ ne couvre qu'*une seule partition* : il s'agit de l'ensemble des fils $\{Y_1, \ldots, Y_k\}$ du nœud Y dans l'arbre représentant la hiérarchie. Ainsi, dans le cas d'une population hiérarchique, la décomposition de l'espace de recherche n'est pas redondante (*cf.* équation 6.4) :

$$\mathfrak{P}_<(Y) = \{Y\} \cup \mathfrak{R}_<(\{Y_1, \ldots, Y_k\})$$

où $\{Y_1, \ldots, Y_k\}$ est la partition couverte par $\{Y\}$.

Dans ce cas, l'algorithme des partitions admissibles optimales consiste simplement à s'exécuter récursivement sur les parties $Y_1, \ldots, Y_k$ (étape 3.1), à constituer les partitions résultantes $\mathfrak{P}_<(Y_1) \times \ldots \times \mathfrak{P}_<(Y_k)$ (étape 3.2) et à comparer leur qualité à celle de la partition macroscopique $\{Y\}$ (étape 3.3). La figure 6.5 donne un exemple d'exécution d'un tel algorithme.

Implémentation spécialisée de l'algorithme. La section B.1 de l'annexe B donne le pseudo-code de l'*algorithme des partitions hiérarchiques optimales* issue de cette spécialisation. Cet algorithme consiste à réaliser différents parcours en profondeur de l'arbre représentant la hiérarchie : pour mesurer la qualité des partitions admissibles (procédure COMPUTEQUALITY),

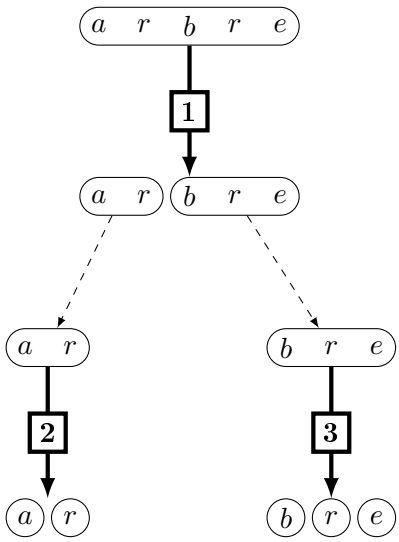


FIGURE 6.5 – Trace d'exécution de l'algorithme des partitions admissibles optimales dans le cas d'une population hiérarchique de taille 5

pour normaliser ces qualités (procédure NORMALIZEQUALITY) et pour trouver une partition optimale (procédure COMPUTEOPTIMALPARTITION). Cet algorithme a été implémenté en C++ au sein du logiciel de visualisation VIVA[7] pour l'agrégation de traces d'exécution de systèmes distribués (*cf.* chapitre 7), également en C++ au sein du module OCELOTL[8] de la plate-forme FRAMESOC, pour l'agrégation de traces d'exécution de systèmes embarqués (*cf.* perspectives du chapitre 7), et sous la forme d'un script PHP, dans le cadre du projet ANR CORPUS GEOMEDIA, pour l'agrégation de systèmes géographiques (*cf.* chapitre 8).

Notons que l'implémentation proposée renvoie *une seule* partition optimale. Dans le cas où plusieurs partitions sont optimales (elles ont la même qualité), l'algorithme renvoie toujours la moins agrégée. Dans la majorité des cas rencontrés en pratique, il n'y a qu'une seule partition optimale ($\mathfrak{P}_a^m$ est un singleton) et cela ne pose pas de problème. Il faudrait cependant, en perspective de cette implémentation, donner une preuve empirique de ce résultat.

Complexité spatiale de l'algorithme. La complexité spatiale de l'algorithme des partitions hiérarchiques optimales est déterminée par la taille de l'arbre représentant la hiérarchie. En effet, pour chaque nœud, différentes étiquettes sont créées et manipulées par l'algorithme afin de représenter : la valeur de l'attribut, la taille de l'agrégat, la réduction de complexité et la perte d'information associées à l'agrégat, ainsi que la coupe de l'arbre afin de représenter la partition optimale. Toutes ces étiquettes nécessitent un espace mémoire constant. De plus, l'arbre contient au maximum $2n-1$ nœuds (dans le cas d'un arbre binaire complet), où n est la taille de la population.

Complexité temporelle de l'algorithme. L'ensemble des opérations exécutées pour créer et manipuler les étiquettes (affectations, sommes, produits, comparaisons, *etc.*) sont réalisées en temps constant. La complexité temporelle de l'algorithme est donc celle d'un simple parcours en profondeur de l'arbre représentant la hiérarchie (*cf.* annexe B pour une évaluation empirique du temps d'exécution).

L'algorithme des partitions hiérarchiques optimales a donc une complexité **spatiale et temporelle linéaire** par rapport à la taille de la population.

[7] Implémentation réalisée par Lucas M. Schnorr, concepteur du logiciel VIVA. *Cf.* fonctions `computeGainDivergence` et `findBestAggregation` dans https://github.com/schnorr/pajeng/blob/entropy/src/libpaje/PajeContainer.cc et dans https://github.com/schnorr/pajeng/blob/entropy/src/libpaje/PajeEntropy.cc.

[8] Implémentation réalisée par Damien Dosimont, concepteur de OCELOTL. *Cf.* https://github.com/dosimont/lpaggreg.

6.3.2 Algorithme des partitions ordonnées optimales

Exécution de l'algorithme générique. Étant donnée une population Ω constituée de n individus $\{x_1, \ldots, x_n\}$ ordonnés selon la relation suivante : $x_i < x_j$ si et seulement si $i < j$, l'ensemble des parties admissibles $\mathcal{P}_<(\Omega)$ est l'ensemble des intervalles $Y_{i,j} = \{x_i, \ldots, x_j\}$ avec $1 \leqslant i \leqslant j \leqslant n$. Pour chacun de ces intervalles, les partitions admissibles couvertes par la partition macroscopique $\{Y_{i,j}\}$ sont des couples de sous-intervalles $\mathcal{Y}_{i,k,j} = \{Y_{i,k}, Y_{k+1,j}\} = \{\{x_i, \ldots, x_k\}, \{x_{k+1}, \ldots, x_j\}\}$ avec $i \leqslant k < j$. Nous avons donc la décomposition suivante : $\mathfrak{C}_<(\{Y_{i,j}\}) = \{\mathcal{Y}_{i,i,j}, \ldots, \mathcal{Y}_{i,j-1,j}\}$.

Supposons que l'algorithme des partitions admissibles optimales est appliqué à la population $\Omega = Y_{1,n}$ et que les partitions couvertes sont évaluées (étape 3) dans l'ordre suivant : $\mathcal{Y}_{1,1,n}, \ldots, \mathcal{Y}_{1,n-1,n}$ (*cf.* figure 6.4 pour $n = 4$).

1. **Première évaluation :** la partition $\mathcal{Y}_{1,1,n} = \{\{x_1\}, \{x_2, \ldots, x_n\}\}$ est évaluée en premier. L'algorithme est alors appelé récursivement (étape 3.1.1) sur la partie $Y_{2,n} = \{x_2, \ldots, x_n\}$, puis sur la partie $Y_{3,n} = \{x_3, \ldots, x_n\}$, *etc.* Après cette première évaluation, l'algorithme a donc été appliqué à l'ensemble des parties $Y_{2,n}, \ldots, Y_{n,n}$ et les résultats intermédiaires ont été stockés en mémoire (étape 4).
2. **Évaluations suivantes :** supposons que l'algorithme en soit à la k^{e} évaluation, avec $1 < k \leqslant n$. Les partitions $\mathcal{Y}_{1,1,n}, \ldots, \mathcal{Y}_{1,k-1,n}$ ont déjà été évaluées, ainsi que l'ensemble des partitions qu'elles raffinent (étape 3.4), et l'algorithme est maintenant chargé d'évaluer la partition $\mathcal{Y}_{1,k,n} = \{Y_{1,k}, Y_{k+1,n}\}$. Il est donc appliqué récursivement (étape 3.1.1) aux deux parties $Y_{1,k}$ et $Y_{k+1,n}$:
   - La partie $Y_{1,k}$ est décomposée (étape 2) de la manière suivante : $\mathfrak{C}_<(\{Y_{1,k}\}) = \{\mathcal{Y}_{1,1,k}, \ldots, \mathcal{Y}_{1,k-1,k}\}$. Or, pour tout $i \in [\![1, k-1]\!]$, la partition $\mathcal{Y}_{1,i,k}$ raffine la partition $\mathcal{Y}_{1,i,n}$. Cette dernière ayant déjà été évaluée lors des étapes précédentes, elle n'a pas besoin de l'être à nouveau. Par conséquent, aucune des partitions couvertes par $\{Y_{1,k}\}$ n'est évaluée (suppression des étapes 3.1, 3.2, 3.3 et 3.4). La partition retenue est donc la partition macroscopique $\{Y_{1,k}\}$ (étape 1).
   - L'algorithme a déjà été appliqué à la partie $Y_{k+1,n} = \{x_{k+1}, \ldots, x_n\}$ lors de la première évaluation. On récupère donc le résultat stocké en mémoire (étape 0).

Pour résumer, pour calculer $\mathfrak{P}^m_<(Y_{1,n})$, l'algorithme générique :

- calcule récursivement les partitions optimales $\mathfrak{P}^m_<(Y_{2,n}), \ldots, \mathfrak{P}^m_<(Y_{n,n})$ pour les $n-1$ parties $Y_{2,n}, \ldots, Y_{n,n}$;
- utilise les résultats pour comparer les partitions couvertes par $\{Y_{1,n}\}$, à savoir (*cf.* équation 6.7) : $\{Y_{1,1}\} \times \mathfrak{P}^m_<(Y_{2,n}), \ldots, \{Y_{1,n-1}\} \times \mathfrak{P}^m_<(Y_{n,n})$.

Implémentation spécialisée de l'algorithme. La section B.2 de l'annexe B donne le pseudo-code de l'*algorithme des partitions ordonnées optimales*. Il s'agit d'une version non-récursive de l'exécution décrite précédemment : les différentes évaluations sont réalisées de manière itérative. Une première itération permet de mesurer la qualité des parties admissibles (procédure COMPUTEQUALITY), une autre de les normaliser (procédure NORMALIZEQUALITY). Une troisième itération (procédure COMPUTEOPTIMALPARTITION) est chargée de calculer les partitions optimales de la partie $Y_{n-1,n}$, puis d'utiliser le résultat pour calculer les partitions optimales de la partie $Y_{n-2,n}$, et ainsi de suite, jusqu'à calculer les partitions optimales de la population $\Omega = Y_{1,n}$.

Cet algorithme a également été implémenté en C++ au sein du module OCELOTL (*cf.* perspectives du chapitre 7) et sous la forme d'un script PHP pour le projet ANR CORPUS GEOMEDIA (*cf.* chapitre 8).

Notons que cet algorithme est similaire à celui proposé dans [JSB+05]. À cet égard, notre contribution principale est de montrer que l'algorithme présenté dans l'article de Jackson *et al.* est un *cas particulier* d'algorithme des partitions optimales. Ce chapitre a bien une visée générique.

Complexité spatiale de l'algorithme. L'algorithme des partitions ordonnées optimales nécessite de stocker, pour chaque partie $Y_{1,n}, \ldots, Y_{n,n}$, la partition optimale associée. Dans l'implémentation proposée en annexe B, nous utilisons le principe d'optimalité pour encoder les partitions optimales de manière efficace. Pour chaque partie $Y_{i,n} = \{x_i, \ldots, x_n\}$, avec $1 \leqslant i \leqslant n$, nous ne stockons que l'indice k, avec $i < k$, de la seconde partie de la partition optimale, signifiant que $Y_{i,k-1} = \{x_i, \ldots, x_{k-1}\}$ est agrégé et que le reste de la partition est déterminé par la partition optimale de $Y_{k,n} = \{x_k, \ldots, x_n\}$ ($k = n + 1$ indique que l'intégralité de la partie $Y_{i,n} = \{x_i, \ldots, x_n\}$ est agrégée). Pour connaître la partition optimale de $Y_{i,n}$, il nous faut regarder celle de $Y_{k,n}$, et ainsi de suite. Ainsi, un simple vecteur d'entiers permet de stocker tous les résultats intermédiaires de l'algorithme.

La complexité spatiale de l'algorithme pourrait donc être **linéaire**. Cependant, dans l'implémentation proposée, les qualités des parties admissibles $Y_{i,j}$, avec $1 \leqslant i \leqslant j \leqslant n$, sont calculées de manière itérative, grâce à la décomposabilité des mesures de qualité (*cf.* sous-section 4.2.3), en construisant une matrice de taille $n \times n$ (procédure COMPUTEQUALITY). Dans ce cas la complexité spatiale est donc **quadratique**. Dans le cas d'un calcul des qualités « à la volée », c'est-à-dire sans allouer d'espace mémoire pour leur

calcul. Dans ce cas, l'évaluation d'une partie dépend linéairement de la taille de celle-ci. Or, l'algorithme nécessite de mesurer la qualité de chaque partie. Dans ce cas, la complexité temporelle serait alors **cubique**. Nous préférons éviter cela en allouant de l'espace mémoire pour la mesure des qualités des parties admissibles.

Complexité temporelle de l'algorithme. Pour chaque partie $Y_{i,n}$, avec $1 \leqslant i \leqslant n$, l'algorithme doit effectuer $i-1$ comparaisons pour identifier les partitions optimales parmi $\{Y_{i,i}\} \times \mathfrak{P}^m_<(Y_{i+1,n}), \ldots, \{Y_{i,n-1}\} \times \mathfrak{P}^m_<(Y_{n,n})$. En tout, $\frac{n(n-1)}{2}$ comparaisons sont effectuées par l'algorithme (*cf.* annexe B pour une évaluation empirique du temps d'exécution).

L'algorithme des partitions ordonnées optimales a donc :

- une complexité **spatiale linéaire** et **temporelle cubique**
- *ou* une **complexité spatiale et temporelle quadratique**.

6.4 Bilan et perspectives

Le problème des partitions admissibles optimales est difficile sur le plan algorithmique dans la mesure où, dans les cas présentés dans cette thèse (cas non-contraint, populations hiérarchiques et populations ordonnées), le nombre de partitions admissibles dépend **exponentiellement** de la taille de la population analysée. Ce chapitre propose un algorithme de résolution efficace reposant sur deux hypothèses :

1. La mesure de qualité à optimiser satisfait le *principe d'optimalité* (ce qui est le cas des mesures *additivement décomposables* présentées dans le chapitre 4).
2. La structure induite par la *relation de couverture* permet de décomposer l'espace de recherche afin de limiter le nombre de partitions à évaluer.

Un résultat important réside donc dans le fait que la complexité du problème des partitions admissibles optimales dépend de la structure de l'ensemble des partitions admissibles : plus les partitions sont contraintes, moins de désagrégations atomiques sont autorisées, plus la résolution du problème est facile sur le plan algorithmique. Ainsi, nous parvenons à résoudre le problème en un temps **polynomial** (linéaire dans le cas des populations hiérarchiques et quadratique dans le cas des populations ordonnées).

Ce chapitre propose un algorithme *générique*, applicable à tout critère externe définissant un ensemble de partitions admissibles. Il permet ainsi de

mettre en place des algorithmes spécialisés en fonction du contexte d'analyse. En perspective de ce chapitre, il apparaît donc important d'appliquer cet algorithme à d'autres critères externes, tels que ceux présentés en perspective du chapitre précédent (partitions admissibles selon un ordre partiel, selon un graphe et selon un graphe pondéré). Pour chacun de ces critères, il est nécessaire de donner une implémentation particulière de l'algorithme et de préciser la complexité spatiale et temporelle associée à cette implémentation. Dans le cas des partitions admissibles selon un graphe, cette complexité dépend notamment de la densité de celui-ci : **exponentielle** dans le cas d'un graphe complet et **quadratique** dans le cas d'un graphe filiforme (équivalent à une relation d'ordre). Une étude empirique permettrait donc de déterminer l'efficacité de l'algorithme en fonction de paramètres tels que la centralité, la connectivité et la densité du graphe.

Troisième partie

Agrégation et analyse de grands systèmes

« Le macroscope [...] est un instrument symbolique, fait d'un ensemble de méthodes et de techniques empruntées à des disciplines très différentes. »

Joël de Rosnay, *Le macroscope*

CHAPITRE 7

Agrégation de traces pour la visualisation de performance

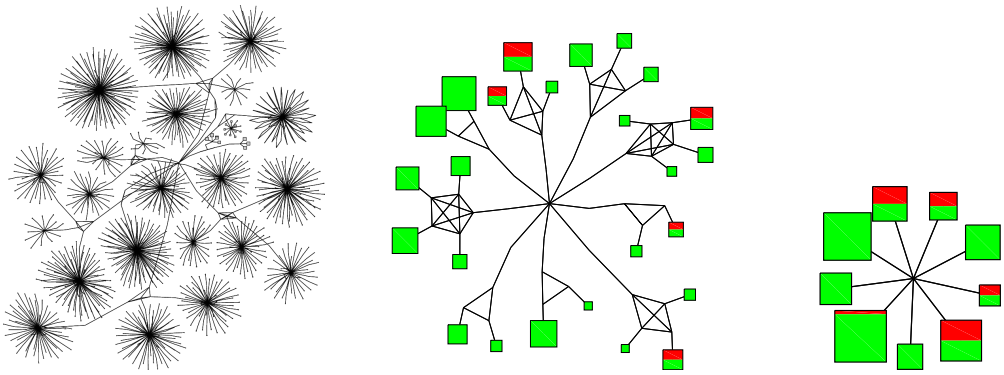
Les applications haute performance (*High Performance Computing*) s'exécutent sur des systèmes possédant aujourd'hui plusieurs milliers, voire plusieurs millions de cœurs. Les machines développées par l'exa-informatique (*exascale computing*), visant à atteindre 10^{18} opérations par seconde, pourraient atteindre l'échelle du milliard de cœurs au cours des prochaines années. Les applications exécutées sur ces plates-formes font intervenir autant de processus, engendrant une concurrence extrême entre les tâches de calcul. Comprendre et expliquer le comportement de ces applications, afin d'en optimiser les performances, constitue un défi majeur pour l'informatique : des *défis syntaxiques*, relatifs à l'observation, au traçage et à la visualisation de millions de processus décentralisés et asynchrones (*cf.* section 2.1), et des *défis sémantiques*, relatifs à l'extraction d'informations pertinentes à différentes échelles de temps (de la nanoseconde au millier de secondes) et différentes échelles d'espace (du processus à l'ensemble du système).

Ce chapitre s'intéresse plus particulièrement à la visualisation de performance. La section 7.1 montre que les outils couramment utilisés dans le domaine manquent de techniques de réduction de données adéquates. La section 7.2 propose alors d'appliquer la méthode d'agrégation présentée dans la partie précédente afin d'engendrer des représentations macroscopiques pertinentes pour l'analyse spatiale et temporelle des applications haute performance. La section 7.3 présente deux cas d'étude montrant que l'algorithme des partitions optimales permet de détecter – à moindre coût – les anomalies présentes dans les traces d'exécution. L'agrégation fournit ainsi des pistes

techniques pour l'optimisation des applications analysées. La section 7.4 montre que la méthode passe à l'échelle, en agrégeant la trace (artificielle) d'un million de processus. L'agrégation relève ainsi les défis syntaxiques et sémantiques du calcul haute performance.

7.1 Enjeux de l'agrégation pour la visualisation de performance

Les recherches en visualisation de performance consistent à développer des techniques de représentation graphique pour l'analyse exploratoire des systèmes de calcul [GZR+11] : grilles, réseaux pair-à-pair, supercalculateurs, informatique en nuage (*cloud computing*), *etc.* Elle vise notamment l'optimisation des applications exécutées via la détection de comportements irréguliers. Les outils de visualisation couramment utilisés dans le domaine (*e.g.*, JUMPSHOT-4, PAJÉ, PARAVER, TRIVA, VAMPIR, VITE, VIVA) proposent habituellement une analyse *post-mortem* de l'application : des traces d'activité sont collectées lors de l'exécution, puis visualisées à partir de représentions classiques de l'espace et du temps, telles que les diagrammes de Gantt [Wil03], ou de représentations moins classiques, telles que les graphes de ressources [SLV13] ou les *treemaps*[1] [Shn92]. Les figures 7.1 et 7.3 donnent des exemples de visualisation reposant sur ces représentations alternatives.



(a) 2170 machines (b) 25 clusters (c) 9 sites

FIGURE 7.1 – Graphe des ressources de la plate-forme GRID'5000 visualisée au niveau des machines (*hosts*), des clusters et des sites (extrait de [SLV13])

[1] Plusieurs traductions de ce terme ont été proposées dans la littérature : « diagrammes de répartition », « diagrammes d'occupation », ou encore « arborescences ». Nous préférons conserver dans la suite de ce chapitre le terme original « treemaps », au féminin, qui exprime mieux le fait qu'il s'agisse de représenter des arbres ou des hiérarchies.

Ces techniques de visualisation, cependant, souffrent de limitations syntaxiques et sémantiques. En effet, le rendu détaillé du niveau microscopique, en plus d'être très coûteux, est extrêmement difficile à analyser [SL12]. Plus grave encore, le désordre qui résulte de l'observation microscopique compromet la bonne interprétation de l'exécution. Les outils de visualisation doivent donc intégrer des techniques d'abstraction pour visualiser les données aux échelles qui intéressent l'utilisateur, tout en préservant l'interprétabilité des données et la sémantique du système. Certains outils ont recours à une *agrégation graphique*, lors du rendu de la visualisation (par exemple au niveau des pixels, pour PARAVER [LGM+05]), ou à une *agrégation de données* (par exemple au niveau des évènements, pour VIVA [SL12] et VAMPIR [BHJR10]). Ce processus de réduction est inévitable dans le cas de grandes traces, dans la mesure où les quantités d'information qu'elles contiennent sont supérieures à la quantité d'information qui peut être affichée à l'écran [SL12] : les données brutes sont alors transformées par un opérateur d'agrégation (moyenne, somme, médiane, *etc.* [EF10]) en s'appuyant sur la structure du système [SHN09]. L'agrégation vise ainsi à réduire la taille des données pour produire une représentation macroscopique cohérente des états et de la dynamique du système.

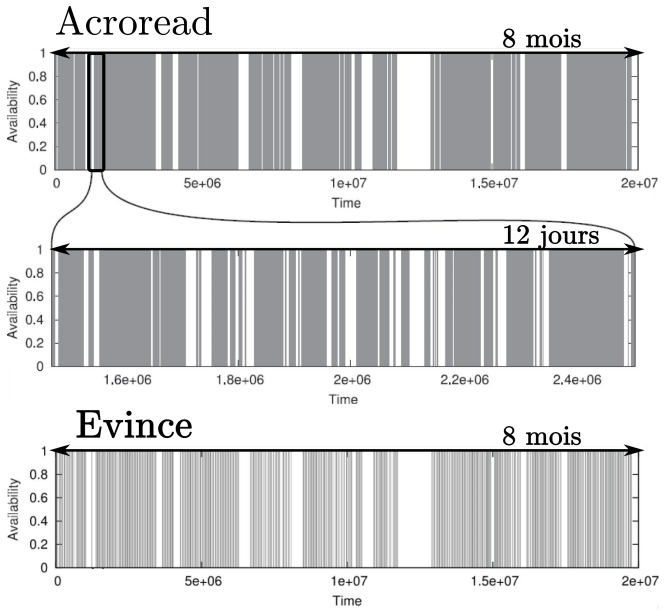


FIGURE 7.2 – Visualisation de la disponibilité d'un client BOINC au cours du temps (extrait de [SLV12]) : agrégation de *données* sur 12 jours et agrégation *graphique* selon deux logiciels de visualisation (ACROREAD et EVINCE)

Cependant, l'agrégation n'est pas un procédé anodin. Elle peut notamment engendrer une perte d'information dangereuse pour l'interprétation des données visualisées, en particulier lorsque les données agrégées sont fortement hétérogènes. Si elle n'est pas contrôlée, une telle transformation peut n'apporter aucune information pertinente, voire induire l'utilisateur en erreur.

Les frises temporelles de la figure 7.2 [SLV12] indiquent les périodes de disponibilité d'un client de la plate-forme de calcul distribué BOINC[2] (en gris, les périodes où le client est disponible, en blanc, les périodes où il ne l'est pas). Parce qu'il n'y a pas assez de place sur l'écran pour représenter toute l'information contenue dans une trace de 8 mois, une *agrégation de données* est nécessaire. La 1re et la 3e frise, respectivement réalisées avec les logiciels de visualisation de fichiers ACROREAD et EVINCE, présentent de telles agrégations temporelles. En examinant les détails sur une période de 12 jours (dans le cas d'ACROREAD, 2e frise), nous remarquons que le comportement microscopique est bien plus complexe que ce qui est effectivement visualisé. Pourtant, l'utilisateur ne dispose d'aucun moyen pour détecter et quantifier cette perte d'information. Plus grave encore, le logiciel de visualisation procède également à une *agrégation graphique* pour afficher les données très proches, notamment lorsque plusieurs informations sont superposées au niveau du pixel. Ainsi, le rendu diffère d'un outil à l'autre, en fonction des techniques de rendu implémentées (*cf.* 1re et 3e frises). L'interprétation des données dépend donc de l'outil utilisé pour la visualisation, ce qui compromet l'objectivité de la représentation.

Nous soutenons que, dans le domaine du calcul haute performance, le contrôle du procédé d'agrégation est un point crucial pour garantir l'interprétabilité des visualisations à grande échelle. Cependant, dans des travaux précédents, nous avons montré que les outils couramment utilisés pour la visualisation de performance (JUMPSHOT-4, PAJÉ, PARAVER, TRIVA, VAMPIR, VITE et VIVA), bien qu'ils implémentent des techniques d'agrégation sophistiquées, ne permettent pas de distinguer les agrégations utiles (surpression d'informations redondantes) des agrégations dangereuses pour l'analyse (perte d'informations importantes) [LPSVD12]. Afin de fournir des représentations macroscopiques proprement interprétables, nous proposons de doter ces outils de méthodes formelles et pratiques pour évaluer et contrôler la *qualité* des visualisations engendrées.

[2] *Berkeley Open Infrastructure for Network Computing*, une infrastructure permettant le partage de ressources de calcul, fondée sur la participation bénévole de machines connectées via Internet, pour la réalisation de projets de recherche divers et variés : http://boinc.berkeley.edu/.

7.2 Agrégation hiérarchique des traces d'exécution

Face aux limitations sémantiques des outils de visualisation de performance, ce chapitre applique la technique d'agrégation présentée dans la deuxième partie de cette thèse. L'objectif visé est la production de représentations macroscopiques pertinentes sur le plan syntaxique et sémantique pour l'analyse des applications parallèles. En particulier, nous cherchons à détecter les anomalies présentes à différents niveaux dans les traces d'exécution. Nous procédons pour cela à l'agrégation *spatiale* des traces et à leur visualisation via des treemaps *multirésolution*. L'approche est validée sur trois cas d'étude. Les deux premiers sont issus de traces d'exécution réelles (section 7.3). Le troisième, destiné à montrer l'efficacité de l'approche à grande échelle, s'intéresse à une trace engendrée de manière artificielle (section 7.4).

7.2.1 Applications et traces analysées

Les deux cas d'étude présentés dans la section suivante s'intéressent aux performances d'une application parallèle composées de *tâches de calcul*. Celles-ci sont réparties et exécutées sur la plate-forme GRID'5000, infrastructure d'expérimentation destinée à la recherche en informatique distribuée[3]. Cette plate-forme de calcul réunit actuellement plus de 2 200 processeurs, totalisant près de 7 900 cœurs, reliés par un réseau haute performance distribué géographiquement sur 9 sites en France et 2 sites à l'étranger. La répartition des tâches de calcul sur une plate-forme de cette envergure nécessite l'utilisation d'algorithmes de synchronisation distribués et dynamiques, chargés de contrôler la concurrence et d'optimiser la performance du système en équilibrant les charges de travail entre les processus. L'étude de ces algorithmes non-déterministes constitue un champ de recherche à part entière, reposant notamment sur l'analyse empirique de leurs résultats.

Définition 7.1. Nous notons Ω_p l'ensemble des processus exécutés sur la plate-forme pour une application donnée. Il s'agit du niveau microscopique de l'analyse spatiale, détaillant l'exécution de chacune des ressources de calcul.

[3] Voir le Wiki dédié https://www.grid5000.fr/ et l'article présentant la plate-forme en détail [BCC+06].

L'algorithme de répartition analysé dans la suite de ce chapitre repose sur KAAPI[4], une librairie pour la synchronisation des tâches et des flux de données dans les systèmes distribués. L'algorithme est d'abord chargé de distribuer les tâches de calcul entre les processus, puis d'équilibrer les charges de travail de manière dynamique lors de l'exécution. Pour procéder à l'évaluation de cet algorithme, nous récupérons des traces d'exécution contenant, pour chaque processus, le temps consacré au calcul lui-même (état RUN) et le temps durant lequel le processus, devenu inactif, tente de décharger les autres processus par le biais de requêtes de *vol de travail* (état STEAL).

Les temps passés par les processus dans chacun des deux états RUN et STEAL constituent les attributs du niveau de représentation microscopique. Ils sont co-dépendants dans la mesure où, sur la période analysée, les processus sont toujours dans un état ou dans l'autre. Ainsi, pour chaque processus, la somme de RUN et de STEAL est égale au temps d'exécution total du processus. Le cas d'exécution idéal correspond à une faible valeur de l'attribut STEAL pour tous les processus, indiquant que la grande majorité du temps d'exécution est dédiée au calcul. Nous sommes donc plus particulièrement intéressés par la visualisation de l'attribut STEAL.

Définition 7.2. Nous notons v_{STEAL} l'application de Ω_p dans $\mathbb{R}^+$ indiquant le temps passé par chaque processus dans l'état STEAL[5]. Il s'agit de l'attribut soumis à l'analyse.

7.2.2 Organisation hiérarchique de la plate-forme

La topologie du réseau de communication liant entre elles les ressources de la plate-forme GRID'5000 affecte l'exécution des applications. En effet, du fait de la distribution matérielle et géographique du système, les ressources de calcul sont plus ou moins proches en temps de communication : deux processus auront plus de facilité à communiquer s'ils appartiennent, dans l'ordre, à la même machine, deux machines au même cluster, à deux clusters du même site, *etc.* Ces propriétés du réseau sont donc liées à l'organisation hiérarchique de la plate-forme, étagée en niveaux de ressources : le niveau des

[4] *Kernel for Adaptative, Asynchronous Parallel and Interactive programming*, voir le site dédié http://kaapi.gforge.inria.fr/ et l'article présentant la libraire en détail [GBP07].

[5] En terme de *dénombrement* (*cf.* sous-section 3.2.1), l'attribut v_{STEAL} peut être interprété comme *la quantité de cycles d'horloge consacrés au vol de travail.* Ces unités atomiques étant de l'ordre de la nanoseconde, nous préférons une interprétation continue de ce dénombrement ($V = \mathbb{R}^+$).

processus, où les tâches sont effectivement exécutées, le niveau des *machines*, coordonnant plusieurs processus, le niveau des *clusters*, regroupant plusieurs machines, le niveau des *sites*, *etc.*

L'algorithme de répartition doit prendre en compte la topologie du réseau pour optimiser les performances du système. Nous faisons donc l'hypothèse que cette topologie permet d'expliquer – au moins en partie – le comportement des processus. En ce sens, deux processus relativement proches dans le réseau de communication sont supposés avoir une exécution similaire vis-à-vis de l'algorithme de répartition. En effet, dans la mesure où ils appartiennent au même contexte d'exécution, ils devraient avoir à peu près la même charge de travail. Les comportements hétérogènes témoignent alors d'une mauvaise répartition des charges, révélant ainsi d'éventuelles imperfections de l'algorithme. L'organisation hiérarchique sert donc de base sémantique pour l'agrégation de données (*cf.* section 5.2). Les agrégats représentent des groupes de ressources cohérentes vis-à-vis de la disposition syntaxique du système.

Définition 7.3. Nous notons $\mathcal{T}(\Omega_p)$ l'ensemble des parties de Ω_p correspondant aux processus regroupés par niveaux de ressources. $X \in \mathcal{T}(\Omega_p)$ peut désigner un processus (singleton), l'ensemble des processus coordonnés par une machine, l'ensemble des processus présents au sein d'un cluster, *etc.* Ces ensembles respectent la structure syntaxique du système et leur comportement est en partie expliqué par la nature du réseau. Ils constituent donc les parties admissibles de l'agrégation spatiale.
Nous notons $\mathfrak{P}_{\mathcal{T}}(\Omega_p)$ l'ensemble des partitions admissibles selon cette organisation hiérarchique. Il s'agit des représentations macroscopiques pertinentes pour l'analyse spatiale de l'algorithme de répartition.

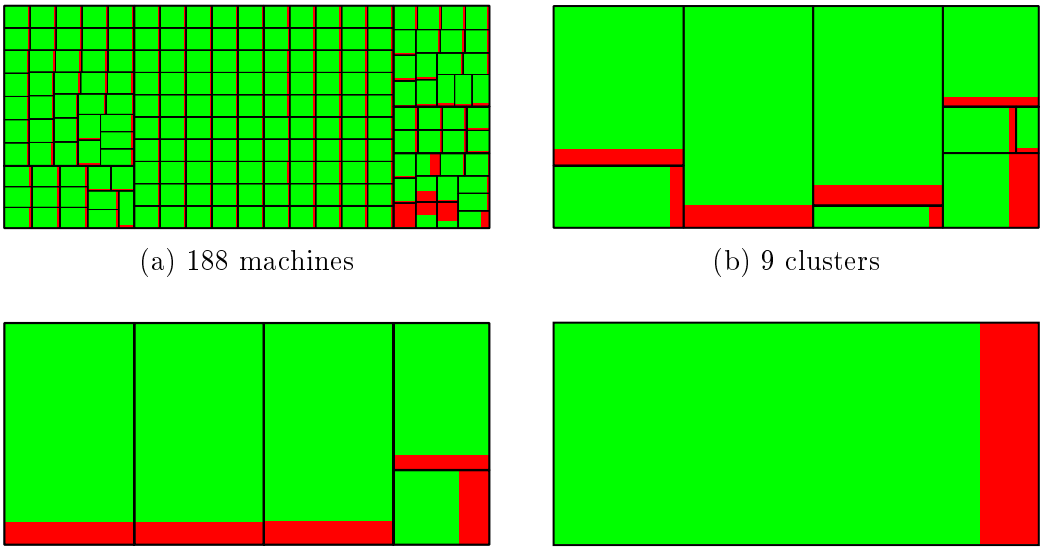
7.2.3 Outils de visualisation et représentations treemap

Nous utilisons les outils *open-source* PajéNG[6] et Viva[7], au sein desquels nous avons implémenté les mesures de qualité et l'algorithme des partitions optimales. Ces outils ont été choisis pour leur implémentation des représentations treemap, permettant de visualiser les traces qui nous intéressent.

Les treemaps sont particulièrement bien adaptées à la visualisation des

[6] Outil d'analyse implémentant des techniques de visualisation classiques. Voir https://github.com/schnorr/pajeng/ et l'article sur l'outil de visualisation Pajé dont PajéNG est une ré-implémentation en C++ [CSB00].

[7] Outil d'analyse reposant sur PajéNG et implémentant des techniques de visualisation alternatives telles que les treemaps [JS91, SHN12] et les graphes de répartition des ressources de calcul. Voir https://github.com/schnorr/viva/ et [SLV13].



(a) 188 machines (b) 9 clusters

(c) 5 sites (d) Agrégation totale

FIGURE 7.3 – Représentation treemap d'une application exécutée sur la plateforme GRID'5000 au niveau des machines (*hosts*), des clusters et des sites (extrait de [SHN12])

hiérarchies [Shn92]. Comme le montre la figure 7.3, les différents agrégats y sont représentés par des « boîtes imbriquées ». La treemap 7.3a permet de visualiser l'exécution au niveau microscopique : chaque rectangle représente un processus $x \in \Omega_p$ et les couleurs à l'intérieur des rectangles représentent les attributs des processus. Ici, il s'agit du temps passé dans l'état RUN ($v_{\text{RUN}}(x)$ en vert) et du temps passé dans l'état STEAL ($v_{\text{STEAL}}(x)$ en rouge). Ainsi, plus un processus est rouge, plus il a été en situation de vol de travail lors de l'exécution. Les trois autres treemaps sont engendrées par agrégation de la treemap microscopique à différents niveaux de la hiérarchie. Les surfaces correspondent à des ensembles de processus $X \in \mathcal{T}(\Omega_p)$. Les valeurs microscopiques des attributs y sont additionnées, indiquant la somme du temps passé dans chacun des états par les processus sous-jacents ($v_{\text{RUN}}(X)$ et $v_{\text{STEAL}}(X)$). La treemap 7.3d représente une seule valeur des attributs : la somme du temps passé par *tous* les processus dans chacun des états ($v_{\text{RUN}}(\Omega_p)$ et $v_{\text{STEAL}}(\Omega_p)$).

VIVA autorise deux procédures d'agrégation différentes : (1) une agrégation *par niveau*, comme illustré dans la figure 7.3 et (2) une agrégation *manuelle*, où les temps d'exécution sont agrégés sur demande de l'utilisateur pour chaque élément de la visualisation. Ainsi, il est possible de produire des représentations multirésolution en choisissant « à la main » les machines,

les clusters ou les sites à agréger. Afin de contrôler et d'automatiser cette procédure d'agrégation, nous avons implémenté les mesures de qualité et l'algorithme des partitions optimales (*cf.* chapitres 4 et 6) au sein du logiciel. Ainsi, VIVA fournit maintenant des indications sur la qualité des agrégats constitués. L'utilisateur peut alors distinguer les agrégats utiles à l'analyse (suppression d'information redondante) et ceux qui peuvent lui nuire (perte d'information). De plus, VIVA propose à l'utilisateur une nouvelle procédure d'agrégation *automatisée*, fournissant des représentations multirésolution en fonction du coefficient de compromis spécifié en entrée (*cf.* sous-section 4.3.3).

7.2.4 Mesures de qualité

À chaque partition de l'ensemble des processus $\mathcal{X} \in \mathfrak{P}_{\mathcal{T}}(\Omega_p)$, admissible selon l'organisation hiérarchique de la plate-forme, correspond une et une seule treemap. Afin de quantifier la perte d'information vis-à-vis de la treemap microscopique, et d'ainsi mesurer l'hétérogénéité des agrégats, nous utilisons la *divergence de Kullback-Leibler* $D(\mathcal{X})$ (*cf.* sous-section 4.3.2).

Il est également nécessaire de quantifier la complexité d'une représentation treemap pour appliquer l'algorithme des partitions hiérarchiques optimales. Dans le contexte de la visualisation de traces, nous devons trouver un critère graphique pour exprimer cette complexité. La granularité des treemaps peut être simplement définie par le nombre d'agrégats qui y sont visualisés. En effet, le calcul d'une treemap et son rendu graphique dépendent linéairement du nombre de rectangles représentés [Shn92]. Nous utilisons donc la *taille de la représentation* $T(\mathcal{X})$ pour quantifier la complexité (*cf.* sous-section 4.3.1). L'utilisation d'autres mesures de complexité, telles que l'*entropie de Shannon* (*cf.* annexe A), n'est pas adaptée aux représentations treemap dans la mesure où nous ne visualisons pas les unités atomiques sous-jacentes (cycles d'horloge dédiés au vol de travail), mais une agrégation préliminaire de ces unités (temps passé dans l'état STEAL).

Ainsi, la mesure de qualité combinée que l'on cherche à optimiser pour donner les « meilleures » représentations treemap des traces d'exécution analysées est la suivante (*cf.* sous-section 4.3.3) :

$$CQL_\alpha(\mathcal{X}) = \alpha \, \frac{\Delta Q(\mathcal{X})}{\Delta Q(\mathcal{P}_\top(\Omega_p))} - (1-\alpha) \, \frac{D(\mathcal{X})}{D(\mathcal{P}_\top(\Omega_p))}$$

où $\alpha \in [0,1]$ est le coefficient de compromis spécifié par l'utilisateur en fonction de la quantité de détails attendus.

Dans les sections suivantes, nous montrons que les représentations engendrées par notre méthode d'agrégation permettent de détecter à moindre coût les comportements irréguliers au sein de l'exécution et de les rapporter à la disposition syntaxique du système observé (section 7.3). L'approche répond alors aux défis de la visualisation de performance en passant à l'échelle du million de processus (section 7.4).

7.3 Détection d'anomalies dans les traces d'exécution

Cette section présente l'analyse spatiale de deux exécutions réelles de l'algorithme de répartition KAAPI. L'objectif est de détecter des anomalies en observant le temps passé par les processus dans l'état STEAL (en rouge dans les figures) et de les contextualiser vis-à-vis de la syntaxe du système (*i.e.*, l'organisation hiérarchique de GRID'5000). Nous montrons que l'algorithme des partitions optimales permet de détecter à moindre coût de telles anomalies.

7.3.1 Engorgement dans le réseau de communication

La trace analysée dans ce premier cas d'étude provient d'une application de 188 processus, chacun alloué à une machine de la plate-forme GRID'5000. Les 188 machines ainsi dédiées au calcul sont répartis sur 5 sites : Porto Alegre (13 machines), Bordeaux (25 machines), Toulouse, Rennes et Nancy (50 machines chacun).

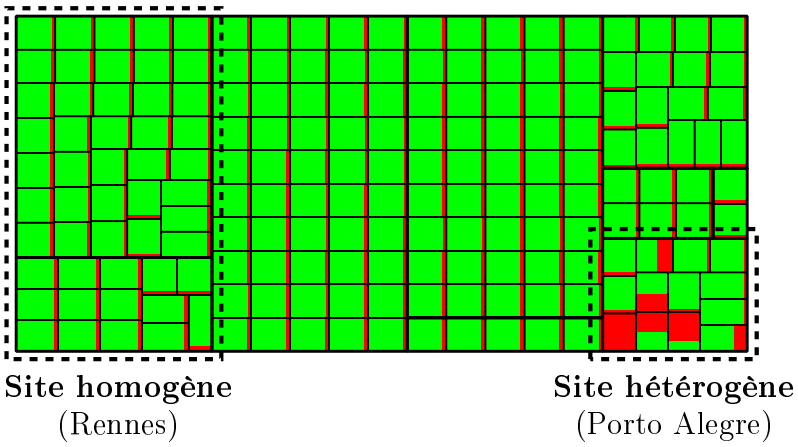


FIGURE 7.4 – Treemap microscopique (188 processus visualisés)

L'analyse de la treemap microscopique (figure 7.4) nous apprend que la grande majorité des processus a passé très peu de temps dans l'état STEAL, ce qui indique une bonne répartition des charges de travail. Cependant, *certains* processus du site de Porto Alegre (en bas à droite) ont passé beaucoup plus de temps que les autres à voler du travail, ce qui indique un éventuel problème dans l'algorithme de répartition. Cette anomalie peut être expliquée par une analyse technique plus approfondie : le site de Porto Alegre est connecté à la plate-forme par un réseau privé virtuel (*VPN*) maintenu via Internet. Du fait de ce statut particulier, la latence du réseau entre le site de Porto Alegre et les sites français est bien plus grande que dans le reste du réseau. Or, l'algorithme classique de répartition proposé par KAAPI ne prend pas en compte ce genre de propriétés locales : les processus envoient leurs requêtes pour le vol de travail de manière aléatoire, indépendamment des performances techniques du réseau de communication. Dans le cas d'un réseau hétérogène, comme celui de GRID'5000, il résulte des temps de vols de travail plus longs au niveau de la connexion VPN du site de Porto Alegre ; on parle alors d'*engorgement*[8].

La treemap microscopique n'est pas optimale dans la mesure où beaucoup d'information redondante y est représentée au sein des sites homogènes (par exemple le site de Rennes). L'analyse détaillée de tous les processus peut difficilement être généralisée à de très grands systèmes, faisant par exemple intervenir les 7 900 processeurs de la plate-forme GRID'5000. L'utilisateur peut alors visualiser la trace à un niveau d'abstraction supérieur pour simplifier l'analyse.

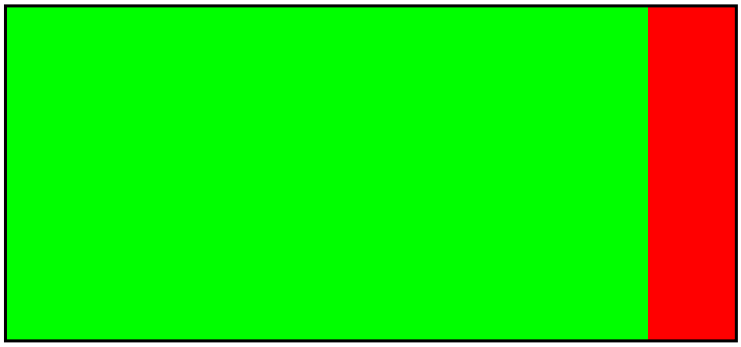


FIGURE 7.5 – Treemap entièrement agrégée (1 valeur visualisée)

[8] Pour plus de détails concernant l'analyse technique de ce cas d'étude, voir [SHN12]. Ici, l'objectif des représentations treemaps n'est pas d'expliquer *directement* le comportement des processus, mais de repérer des irrégularités afin d'informer l'utilisateur de la présence de zones problématiques au sein de l'exécution.

L'anomalie décrite précédemment ne peut bien évidemment pas être détectée à partir de la treemap entièrement agrégée (figure 7.5) dans la mesure où les différences de comportement entre les sites ne sont pas visualisées. La treemap agrégée au niveau des sites (figure 7.6) pourrait en revanche donner quelques indications. En effet, l'utilisateur remarque alors que le site de Porto Alegre a globalement passé plus de temps à voler du travail que les autres sites. Cependant, une telle représentation peut être mal interprétée. En particulier, l'utilisateur peut supposer que *tous* les processus du site de Porto Alegre ont volé du travail de manière inattendue. Il s'agit de l'*hypothèse de redistribution uniforme* (*cf.* interprétation 1 figure 7.7). L'utilisateur peut également penser que seulement *trois ou quatre* processus ont été entièrement inactifs (*cf.* interprétation 2). Dans les deux cas, il s'agit d'interprétations erronées qui peuvent nuire à l'analyse de l'application. En vérité, 7 processus ont eu un comportement similaire au reste de la plate-forme et 6 ont eu un comportement inattendu (*cf.* interprétation 3). Même si l'utilisateur fait ce genre d'hypothèse, il est impossible de déterminer *quels sont* les processus problématiques à partir de la donnée agrégée. La représentation ne fournit donc pas toute l'information pertinente pour décrire et expliquer l'anomalie détectée.

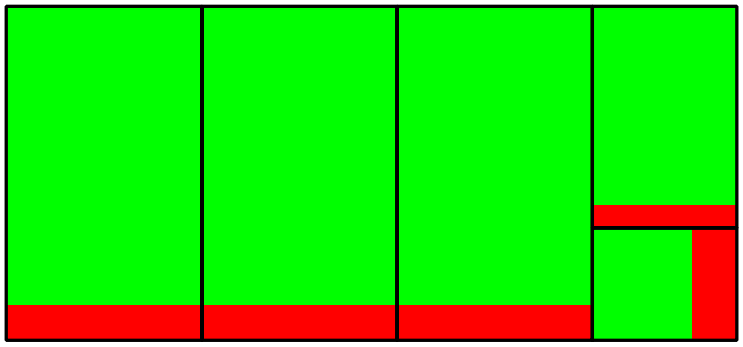


FIGURE 7.6 – Treemap au niveau des sites (5 sites visualisés)

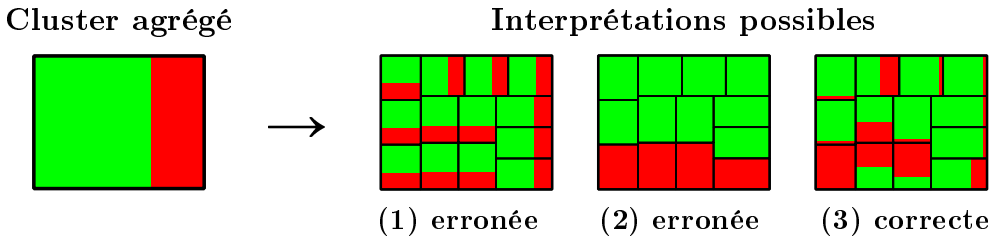


FIGURE 7.7 – Trois interprétations possibles d'une valeur agrégée au niveau d'un site

L'algorithme des partitions hiérarchiques optimales permet de supprimer l'information redondante tout en maximisant la quantité d'information relative à la représentation microscopique. Il constitue pour cela des treemaps *multirésolution*.

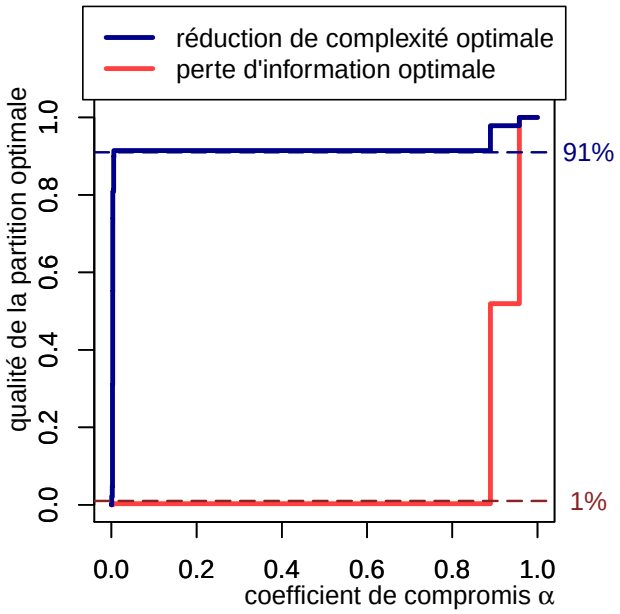


FIGURE 7.8 – Graphe de qualité des treemaps optimales en fonction du coefficient de compromis α

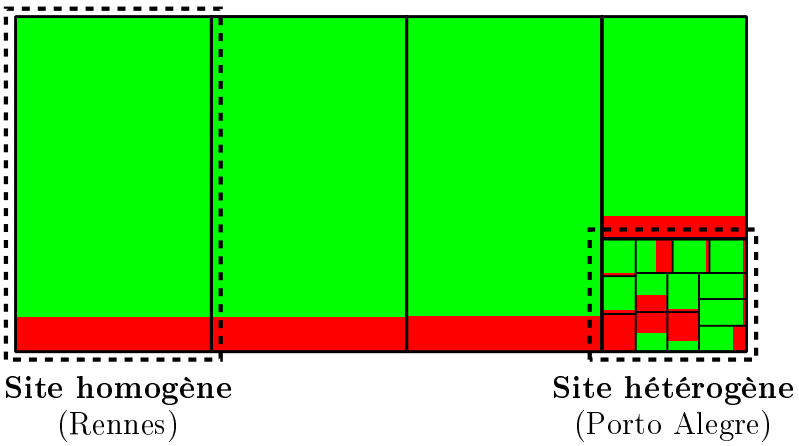


FIGURE 7.9 – Treemap optimale préservant au moins 99% de l'information microscopique

Le graphe des qualités optimales (figure 7.8) indique la réduction de complexité (en bleu) et la perte d'information (en rouge) associées aux treemaps optimales engendrées par l'algorithme en fonction du coefficient de compromis α spécifié par l'utilisateur (*cf.* sous-section 4.3.3). Il apparaît que la treemap microscopique peut être aisément simplifiée sans perdre beaucoup d'information (la réduction de complexité augmente considérablement pour de faibles valeurs de α, alors que la perte d'information reste très faible). Ainsi, lorsque α est inférieur à 0.88, la treemap optimale (figure 7.9) conserve 99% de l'information contenue dans la treemap microscopique et atteint 91% de la réduction de complexité maximale. Elle contient deux niveaux de représentation : l'exécution des sites homogènes est agrégée, alors que l'exécution du site de Porto Alegre est représentée au niveau des processus. Contrairement à la treemap agrégée au niveau des sites (figure 7.6), l'algorithme garantit à l'observateur que les sites agrégés *sont effectivement homogènes*. Celui-ci peut faire les bonnes hypothèses concernant le comportement des processus sous-jacents, sans procéder à une analyse plus détaillée de ces parties de la visualisation.

En supposant que l'organisation du système *explique* le comportement des individus et que l'hétérogénéité est ainsi l'indice de potentielles *anomalies*, les partitions optimales attirent l'attention sur les zones problématiques de l'exécution sans représenter celle-ci dans son intégralité.

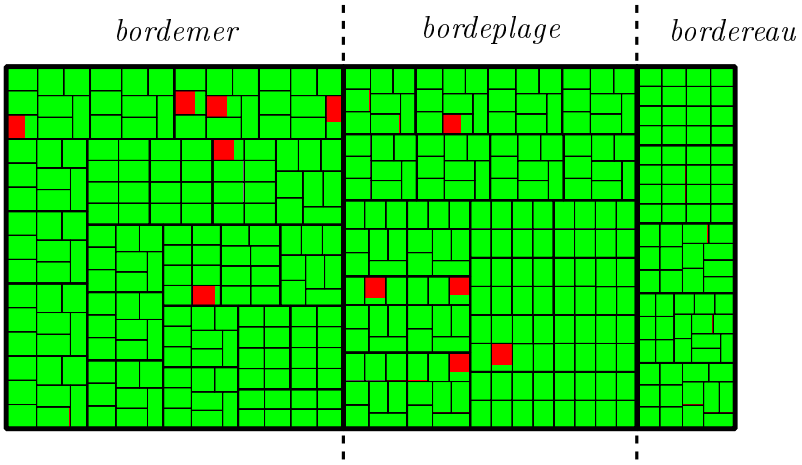


FIGURE 7.10 – Treemap microscopique (434 processus visualisés)

7.3.2 Détection d'anomalies multi-niveau

Ce deuxième cas d'étude présente une application de 434 processus exécutés sur 50 machines du site de Bordeaux. Celles-ci sont réparties en 3 clusters : 5 machines dans le cluster *bordereau*, 22 dans le cluster *bordeplage* et 23 dans le cluster *bordemer*. La treemap 7.10 représente l'ensemble des processus et le temps passé dans l'état RUN (en vert) et dans l'état STEAL (en rouge).

Comme dans le cas d'étude précédent, on suppose que l'hétérogénéité des comportements révèle des anomalies au sein de l'exécution. Dans la treemap présentée dans la figure 7.11 (rapport α inférieur à 0.76), les processus ayant un comportement homogène, au sein d'une même machine, sont agrégés avec une perte d'information négligeable. Ne sont détaillées que les machines dont au moins un des processus a passé plus de temps que la moyenne dans l'état STEAL. Ces machines problématiques méritent une attention particulière lors de l'analyse. De plus, le cluster *bordereau* est entièrement agrégé, indiquant qu'aucune irrégularité n'est détectée au sein des processus.

Cette représentation permet donc d'identifier au moins deux catégories d'objets : (1) les processus ayant passé beaucoup de temps à voler du travail (et les machines auxquelles ils appartiennent) ; (2) les clusters dont aucune machine n'a rencontré de telles difficultés. Une analyse plus approfondie du site de Bordeaux révèle que les machines du cluster *bordereau* ont quatre processeurs, tandis que celles des clusters *bordeplage* et *bordemer* n'en ont que deux [SHN12]. Il est fort possible que les anomalies soient expliquées par cette propriété syntaxique de la plate-forme.

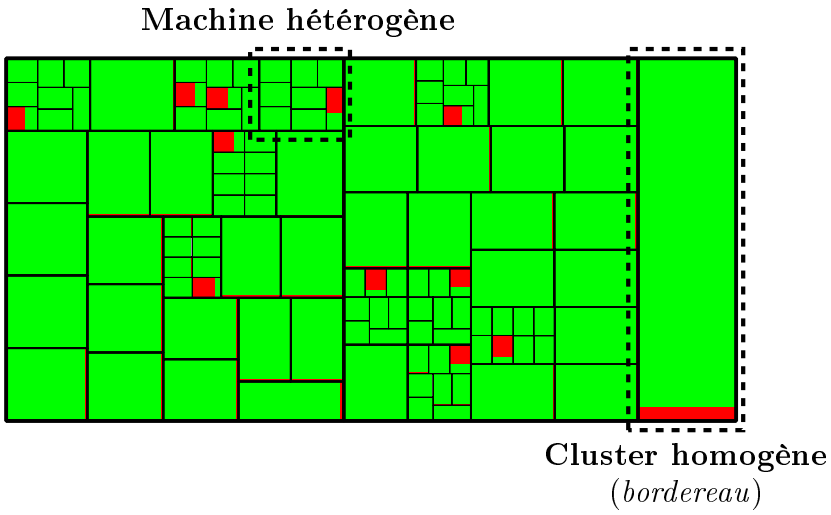


FIGURE 7.11 – Treemap optimale préservant au moins 99% de l'information

Les représentations optimales permettent de détecter les anomalies, de les interpréter à différentes échelles spatiales (processus, machines, clusters), de les situer au sein du système et de les mettre en correspondance avec les propriétés syntaxiques de la plate-forme.

7.4 Visualiser un million de processus

Ce troisième cas d'étude vise à montrer que l'algorithme des partitions optimales peut être appliqué à des systèmes de très grande taille. Mieux, il montre que l'agrégation est *inévitable* pour le passage à l'échelle. L'approche présentée dans cette thèse permet donc de visualiser des phénomènes qui ne pourraient être détectés par une méthode d'analyse classique.

Nous cherchons à visualiser une trace décrivant l'exécution d'un million de processus. Afin d'illustrer au mieux la méthode d'agrégation, nous travaillons sur une trace produite de manière artificielle. Le système décrit est organisé selon une hiérarchie comprenant 5 niveaux : 1 000 000 processus, 10 000 machines, 1 000 clusters, 100 super-clusters et 10 sites. Les processus connaissent deux états lors de l'exécution : VS0 et VS1 (respectivement représentés en jaune et en bleu dans les treemaps de cette section). Tous les processus ont un comportement plus ou moins similaire (VS0 varie uniformément entre 0.65 et 0.75 et nous avons VS0 + VS1 = 1), sauf un certain nombre dont le comportement diverge de la moyenne (VS0 est tiré aléatoi-

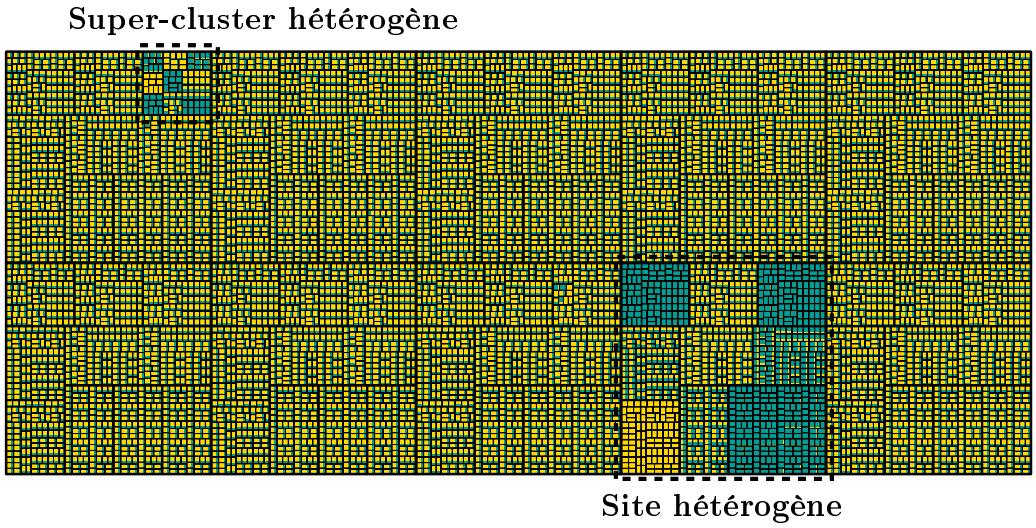


FIGURE 7.12 – Treemap au niveau des machines (10 000 machines visualisées)

rement entre 0 et 1). Pour souligner l'intérêt de notre approche, nous avons introduit ces comportements hétérogènes à chaque niveau de la hiérarchie. Il y a ainsi :

- *une machine* dont les 100 processus sont hétérogènes ;
- *un cluster* dont les 10 machines sont hétérogènes (mais les processus au sein de chacune de ces machines sont homogènes) ;
- *un super-cluster* dont les 10 clusters sont hétérogènes, *etc.*

Une treemap contenant un million d'éléments, en plus d'être extrêmement coûteuse à visualiser, ne peut simplement pas être représentée sur la surface d'une feuille A4 ou sur celle d'un écran d'ordinateur. La représentation microscopique étant impossible à visualiser, la treemap de la figure 7.12 représente la trace agrégée au niveau des 10 000 machines. Nous pouvons déjà observer deux zones hétérogènes de tailles différentes : *un site* contenant des super-clusters hétérogènes et *un super-cluster* contenant des clusters hétérogènes. De plus, s'il procède à une analyse minutieuse, l'utilisateur pourra trouver, noyé dans la complexité de la visualisation, *un cluster* contenant des machines hétérogènes. Pour le reste, et puisque nous travaillons ici au niveau des machines, il est strictement impossible de trouver *la machine* contenant des processus hétérogènes.

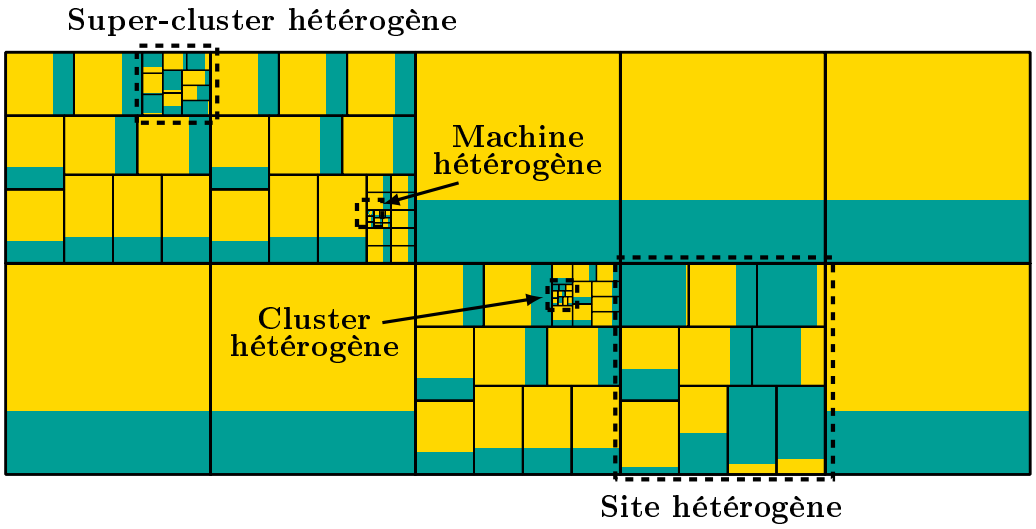


FIGURE 7.13 – Treemap optimale préservant au moins 95% de l'information microscopique

La treemap représentée dans la figure 7.13 est obtenue pour un coefficient de compromis α inférieur à 0.84. Elle contient 95% de l'information microscopique, mais atteint 99,98% de la réduction de complexité maximale. Ainsi, on ne gagne pas beaucoup à agréger plus, mais on dispose quand même de la majeure partie de l'information nécessaire à l'analyse. Ceci est dû au fait que l'information redondante relative aux 6 sites homogènes est agrégée, rendant la treemap beaucoup plus lisible. *Le cluster* contenant des machines hétérogènes – extrêmement difficile à repérer dans la figure 7.12 – est ici immédiatement détecté. En outre, la treemap agrégée est beaucoup moins coûteuse à manipuler. Au format vectoriel PDF, elle nécessite 50 fois moins d'espace mémoire que la treemap au niveau des machines : 446 000 octets pour stocker 10 000 éléments contre 8 900 octets pour stocker 190 éléments, ce qui correspond à la taille de la représentation T (linéaire par rapport au nombre d'agrégats représentés). Ceci constitue un gain considérable en ressources nécessaires à l'encodage et à la manipulation de ces représentations.

De plus, l'algorithme des partitions hiérarchiques optimales permet de conserver les détails microscopiques pour *la machine* contenant des processus hétérogènes (*cf.* figure 7.13). Il était strictement impossible de détecter cette anomalie à partir de la treemap agrégée au niveau des machines (*cf.* figure 7.12) dans la mesure où cette information avait été supprimée par l'agrégation de données. Ici, l'information est conservée et disponible pour l'analyse. Notons cependant que l'agrégation *graphique* empêche toujours de *visualiser* certains détails (*cf.* machine hétérogène de la figure 7.13). Le contrôle de l'agrégation permet néanmoins la *détection* de ces anomalies microscopiques, pouvant entraîner par la suite un zoom graphique de la part de l'utilisateur. L'agrégation doit donc être considérée comme un processus d'abstraction interactif entre l'information fournie par l'algorithme et l'observateur.

Une visualisation de l'ensemble des processus aurait peut-être permis de détecter l'anomalie, mais au prix d'une analyse extrêmement coûteuse : en particulier, la treemap microscopique nécessite 5 000 fois plus d'espace mémoire que la treemap multirésolution, mais elle ne fournit pas beaucoup plus d'information.

La représentation microscopique ne passant pas à l'échelle, l'algorithme des partitions optimales permet donc de visualiser des phénomènes qui ne pourraient être détectés *en pratique* à partir de la représentation microscopique ou de la représentation d'un seul niveau d'abstraction.

Cette section montre que la méthode de *détection* des anomalies passe à l'échelle. Cependant, la trace analysée ne correspond pas à un cas d'exécution réelle. Elle ne permet donc pas d'évaluer le passage à l'échelle de la méthode d'*explication* des anomalies (abordée dans la section précédente). En effet, l'hétérogénéité a été explicitement introduite en fonction de la structure (dans une machine, dans un cluster, *etc.*). Par conséquent, les anomalies sont trivialement expliquées par cette structure. Il apparaît donc nécessaire, en perspective de ce cas d'étude, de procéder à l'analyse d'une trace qui soit à la fois réelle *et* de grande taille, afin de montrer que la méthode de détection *et* la méthode d'explication des anomalies passent toutes les deux à l'échelle.

7.5 Bilan et perspectives

Au regard de la taille des applications développées par le calcul haute performance, l'agrégation de données semble inévitable pour le passage à l'échelle des techniques d'analyse. L'approche présentée dans cette thèse permet de garder le contrôle sur le procédé d'agrégation, afin de fournir à l'utilisateur le maximum d'information sur le système analysé, tout en réduisant le coût de la visualisation.

Contributions syntaxiques. En représentant les traces d'exécution selon le réseau de communication, l'organisation du système est prise en compte par le processus d'agrégation. Les relations syntaxiques, liées à la synchronisation des ressources de calcul, sont donc agrégées de manière cohérente et servent à expliquer les comportements observés. Le passage à l'échelle des techniques de visualisation est alors rendu possible par l'algorithme des partitions optimales. Il permet notamment de concentrer les ressources dédiées à la visualisation sur les zones cruciales de l'analyse, lorsque les relations syntaxiques ne sont pas homogènes (zones asynchrones indiquant de potentielles anomalies).

Les expériences présentées dans ce chapitre proposent ainsi de réduire le coût de la visualisation. Cependant, les difficultés syntaxiques limitent également l'étape de *collecte* des données, antérieure à l'étape de visualisation. En effet, les techniques présentées supposent l'enregistrement de traces d'exécution pour chacune des ressources du système. Or, le traçage de processus distribués et asynchrones induit de nombreuses difficultés techniques (collecte d'informations non-centralisées, datation d'évènements sans horloge globale [CMV01]). Bien que l'agrégation de traces microscopiques permette de simplifier la syntaxe du système lors de la visualisation, l'analyse de performance est néanmoins limitée par la collecte des données. La conclusion

de cette thèse (section 9.2) propose des pistes de recherche pour appliquer les techniques d'agrégation en amont, lors du processus de traçage, afin de réduire également le coût de l'étape de collecte.

Contributions sémantiques. Les apports de l'algorithme ne sont pas seulement d'ordre syntaxique. Même s'il a été possible en pratique de visualiser en détail un million de processus distribués et asynchrones, cela n'aurait que très peu de sens pour l'analyse. En proposant des abstractions spatiales pertinentes sur le plan informationnel, l'algorithme permet de procéder à une analyse multi-échelle de l'exécution. Les anomalies sont décrites à des granularités différentes : engorgement au niveau d'un cluster (7.3.1), difficultés d'ordonnancement au niveau des machines (7.3.2), détails de l'activité des processus, répondant ainsi aux difficultés sémantiques soulevées dans la section 2.1. De plus, la méthode met en évidence des phénomènes qui, en pratique, n'auraient pu être analysés à partir d'une sémantique microscopique (section 7.4).

Le reste de cette section présente des perspectives de recherche concernant la généralisation de la méthode d'agrégation pour la visualisation de performance.

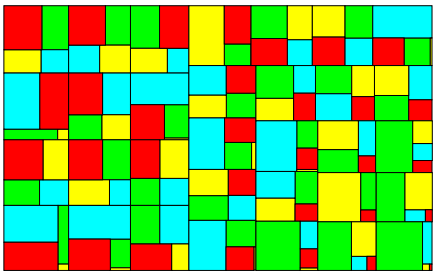
Hiérarchies non-spatiales. Dans ce chapitre, nous avons utilisé l'organisation hiérarchique de la plate-forme GRID'5000 pour visualiser et analyser les traces d'exécution. Dans le cas de systèmes décentralisés (*e.g.*, systèmes pair-à-pair) ou dont la hiérarchie ne contient que très peu de niveaux (hiérarchies « plates »), la visualisation peut néanmoins reposer sur d'autres propriétés syntaxiques du système.

Une hiérarchie peut notamment être construite par partitionnement hiérarchique (*hierarchical clustering* [SZG+96, MRS08, EF10]). Les ressources de calcul sont alors associées deux-à-deux, en fonction d'un critère de similarité, jusqu'à l'obtention d'un arbre binaire donnant la hiérarchie du système. Le critère de similarité fait intervenir certains attributs des ressources, traités comme des données *externes* (*e.g.*, puissance, disponibilité, réseau de communication, propriétaire de la ressource). Les attributs analysés (données *internes*) sont alors visualisés à partir de la hiérarchie engendrée à l'aide de ce critère. Dans ce cas, les abstractions ne sont pas spatiales ou topologiques, mais reposent sur les attributs externes utilisés pour le partitionnement. Ils servent donc à *expliquer* les attributs internes, supposés homogènes vis-à-vis de la hiérarchie. L'hétérogénéité révèle ainsi des comportements inattendus vis-à-vis des critères externes.

Organisations non-hiérarchiques. Comme indiqué en perspective du chapitre 5, l'agrégation peut s'appuyer sur une large classe d'organisations non-hiérarchiques. En particulier, l'agrégation selon un graphe (*cf.* section 5.5) permet de construire des abstractions cohérentes avec le réseau de communication de la plate-forme lorsque celui-ci n'est pas hiérarchisé. Ainsi, il est possible d'adapter la méthode d'agrégation aux systèmes pair-à-pair et aux grilles de calcul décentralisées. Notons cependant que, contrairement aux organisations hiérarchiques, l'algorithme d'agrégation selon un graphe n'a pas une complexité linéaire. Pire, la complexité dépend de la connectivité du réseau de communication : plus les ressources sont connectées, moins le procédé d'agrégation est contraint, plus le calcul des partitions optimales est coûteux.

Analyse de plusieurs attributs. Dans certains cas, plusieurs attributs doivent être examinés simultanément pour décrire l'exécution. Par exemple, supposons que les processus peuvent être dans k états possibles. Nous disposons alors de k attributs et de leurs représentations microscopiques respectives. Les treemaps permettent de visualiser ces représentations à l'aide d'un seul graphique (*cf.* figure 7.14). Cependant, lors de l'agrégation, une partition de Ω_p est nécessairement appliquée à *tous* les attributs. L'agrégation d'un attribut implique donc nécessairement l'agrégation des autres. Nous envisageons trois manières de procéder :

1. L'agrégation des k représentations microscopiques est réalisée séparément, engendrant k partitions différentes[9]. Les techniques de visualisation doivent alors être adaptées afin de « superposer » ces représentations agrégées disparates. Cela paraît difficile dans le cas des treemaps.



(a) Treemap microscopique (32 processus et 4 attributs)

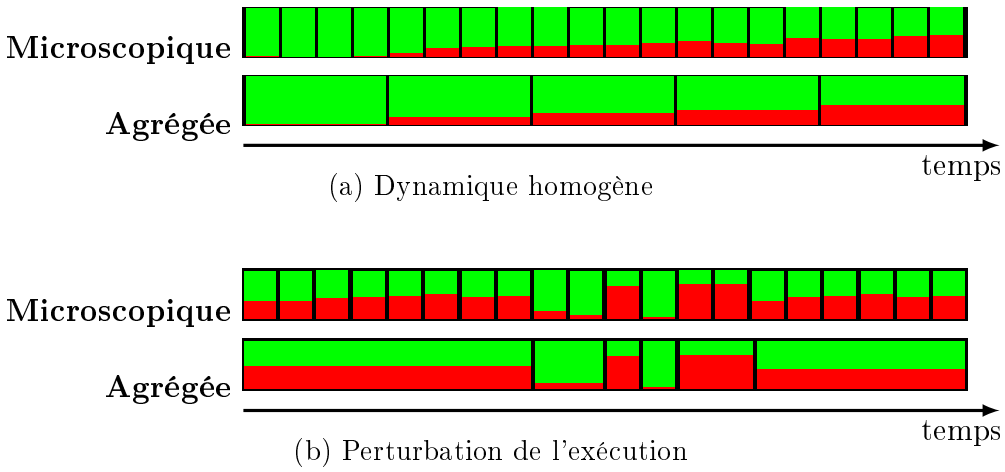
(b) Treemap macroscopique (valeurs totales des 4 attributs)

FIGURE 7.14 – Treemaps donnant les valeurs de 4 attributs (extrait de [SHN12])

[9] Cette approche est similaire à celle proposée par [WF94a] pour la production de *blockmodels* : l'agrégation y est réalisée séparément pour chacun des attributs.

2. Il est également possible de traiter les attributs comme un vecteur de taille k. Les mesures de qualité présentées dans cette thèse étant additives [Csi08], il est facile d'adapter les mesures de qualité en faisant la somme des mesures – éventuellement pondérée – pour chacune des k valeurs. L'algorithme engendre ainsi *une seule* partition optimale « en moyenne » pour les k attributs.
3. L'ensemble des attributs est considéré comme une dimension de l'analyse, constituée de k individus. Les attributs sont alors agrégés lorsqu'ils sont homogènes entre eux, réduisant ainsi le nombre de représentations microscopiques à visualiser[10]

Analyse temporelle des traces d'exécution. Les représentations manipulées dans ce chapitre sont entièrement agrégées dans le temps. Les traces d'exécution sont donc analysées selon leur seule dimension spatiale. Cependant, les outils de visualisation de performance proposent également de représenter la temporalité des systèmes, notamment à l'aide de diagrammes de Gantt [Wil03]. Selon cette dimension également, l'agrégation est nécessaire pour visualiser les dynamiques à différentes échelles (de la nanoseconde au millier de secondes). VIVA propose par exemple d'agréger l'activité des processus sur des tranches de temps microscopiques. La figure 7.15 donne des exemples de représentations temporelles multirésolution que l'on pourrait obtenir avec notre méthode d'agrégation.



(a) Dynamique homogène

(b) Perturbation de l'exécution

FIGURE 7.15 – Agrégation temporelle de la trace d'exécution d'un processus

[10] Cette approche est similaire aux techniques de statistique multivariée qui visent à réduire le nombre de variables explicatives (*e.g.*, analyse en composantes principales).

Cette approche temporelle peut donc être appliquée à la détection de perturbations dans les traces d'exécution. Parmi les millions d'évènements engendrés en quelques secondes, l'utilisateur aimerait repérer les périodes problématiques (*e.g.*, baisse de l'activité des processus) et expliciter les causes de ces anomalies. L'algorithme des partitions ordonnées optimales (*cf.* section 6.3) a par exemple été implémenté au sein du module OCELOTL[11] de la plate-forme d'analyse FRAMESOC [PMM12] (*cf.* figure 7.16). Les membres de ce projet évaluent actuellement son efficacité pour la détection de périodes d'exécution stables et de perturbations dans les traces d'applications multimédia [PDH+13].

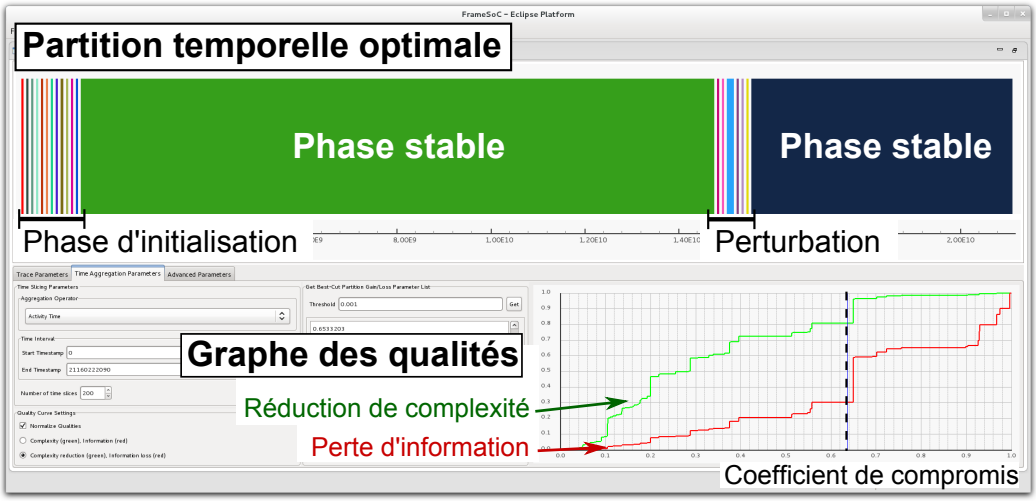


FIGURE 7.16 – Implémentation de l'algorithme des partitions ordonnées optimales au sein de l'outil OCELOTL

[11] https://github.com/dosimont/ocelotl

CHAPITRE 8

Agrégation de données médiatiques pour l'analyse des relations internationales

Dans le chapitre précédent, nous avons montré que l'algorithme des partitions optimales pouvait aider à la détection d'anomalies dans les traces d'exécution d'applications parallèles. Les systèmes étudiés dans ce domaine sont clairement définis, leur sémantique entièrement déterminée par la structure physique du réseau de communication et les phénomènes observés sont, pour la plupart, décrits de manière formelle et non-ambiguë par les experts. Ce second chapitre applicatif vise à montrer que notre approche peut être appliquée à des systèmes plus complexes sur le plan sémantique (*cf.* section 2.1). Nous nous orientons pour ce faire vers les sciences sociales et, plus particulièrement, vers les sciences politiques. L'objectif de l'application est l'analyse géographique et temporelle du système international via l'agrégation de données médiatiques.

La section 8.1 présente les problématiques scientifiques liées à l'analyse des données véhiculées par les *flux d'information médiatique* pour représenter les *relations internationales*. Nous montrons qu'il est nécessaire d'avoir recours à différents niveaux de représentation pour aborder la complexité sémantique du système. La section 8.2 propose d'appliquer notre méthode d'agrégation à la détection d'évènements de granularités variées au sein des flux d'information. La notion d'*évènement* est ainsi au cœur du processus d'abstraction. La section 8.3 présente deux cas d'étude consacrés à l'analyse géographique et temporelle d'évènements médiatiques. Nous montrons que l'approche satisfait les enjeux sémantiques de l'agrégation en engendrant des abstractions pertinentes pour l'analyse du système international.

8.1 Exploiter les flux d'information médiatique pour l'analyse du système international

Dans cette section, nous présentons la notion de « système international » et nous montrons que (1) un tel système peut être décrit et analysé à partir de la notion d'*évènement médiatique* et (2) une telle analyse nécessite de travailler à plusieurs niveaux de représentation afin de mettre en évidence des évènements de granularités différentes.

8.1.1 Analyse médiatique des relations internationales

Le *système international* désigne l'ensemble des entités politiques agissant et interagissant au niveau des nations : États, organisations internationales (OI), organisations non gouvernementales (ONG), entreprises multinationales, *etc.* Le *domaine* des Relations internationales est la branche des sciences politiques chargée des grandes questions relatives à ce système : quel est le rôle des entités internationales ? Comment rendre compte de l'organisation du système ? Quelles sont, en particulier, ses sources de stabilité et d'instabilité ? Comment décrire et expliquer les relations de pouvoir entre les nations ? Ce champ disciplinaire s'intéresse donc, en particulier, à la *notion* de « relations internationales », désignant l'ensemble des relations politiques, économiques, culturelles, géographiques, historiques, *etc.* qu'entretiennent les entités du système. Du fait de la disparité des entités et des relations considérées, les Relations internationales constituent un champ disciplinaire extrêmement complexe sur le plan sémantique (*cf.* section 2.1). À ce titre, de nombreuses disciplines des sciences sociales y participent : la géographie, la géopolitique, l'histoire, la sociologie, l'anthropologie, *etc.*

Dans cette thèse, nous nous intéressons à une méthode d'analyse particulière, fondée sur l'observation des *flux d'information médiatique*. Un flux est ici défini par la somme des informations produites par un média (presse, radio, télévision) et donnant une représentation particulière de l'actualité. Une hypothèse forte consiste à affirmer que les flux médiatiques permettent d'observer et de représenter le système international. Pour les géographes, un article [1] est à ce titre un « connecteur géographique élémentaire qui instaure une liaison entre le lieu de publication de l'article et le ou les lieux dont il est question dans l'article. » [G+11] Ainsi, un article publié par un flux médiatique appartenant à un pays A et relatant des évènements ayant eu

[1] Par analogie avec la presse écrite, nous parlons d'*article* pour désigner un ensemble d'informations juxtaposées transmises par un flux médiatique et visant à décrire une partie de l'actualité.

lieu dans un pays B est un indicateur de l'*attention médiatique* portée par A sur B [KV11]. Les flux témoignent ainsi des relations entre A et B selon l'hypothèse que *les médias parlent des pays avec lesquels leur pays d'origine entretient de fortes relations* [GR65, KV11][2].

À titre d'exemple, la figure 8.1 donne la part des articles parlant des différents pays du monde parmi tous ceux qui ont été publiés par deux journaux, de nationalités différentes : *Le Pays*, quotidien burkinabé (en bleu), et *Le journal de Montréal*, quotidien canadien (en orange). Cette carte donne ainsi des indications concernant la distribution spatiale de l'attention médiatique du Burkina Faso et du Canada. Elle permet notamment de mettre en évidence « des différences considérables dans la géographie des flux médiatiques » [G+11], révélant ainsi les positions très différentes des deux pays au sein du système international : par exemple, le journal burkinabé parle beaucoup plus des pays africains que le journal canadien, présumant ainsi de relations internationales privilégiées entre ces pays.

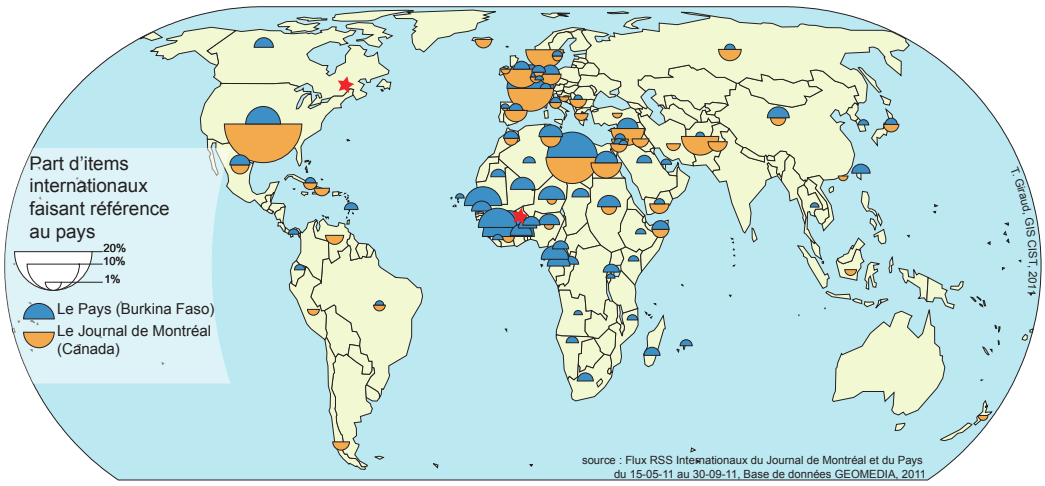


FIGURE 8.1 – Proportion des articles publiés par un journal burkinabé (en bleu) et un journal canadien (en orange) parlant des différents pays du monde (extrait de [G+11])

[2] Dans ces travaux, les facteurs expliquant une forte attention médiatique sont liés aux propriétés des évènements relatés (*e.g.*, intensité, fréquence, surprise) et aux relations entre le média et le pays où a lieu l'évènement (*e.g.*, distance géographique et culturelle, différences économiques entre les deux pays).

8.1.2 La notion d'évènement médiatique

Un *évènement médiatique* correspond à évènement réel fortement relaté par un flux d'information. Il témoigne ainsi de relations fortes entre deux pays du fait d'une situation particulière du système international. La détection de tels évènements constitue donc un aspect majeur de l'analyse médiatique des relations internationales. L'objectif est de mettre en évidence les points cruciaux pour l'analyse du système.

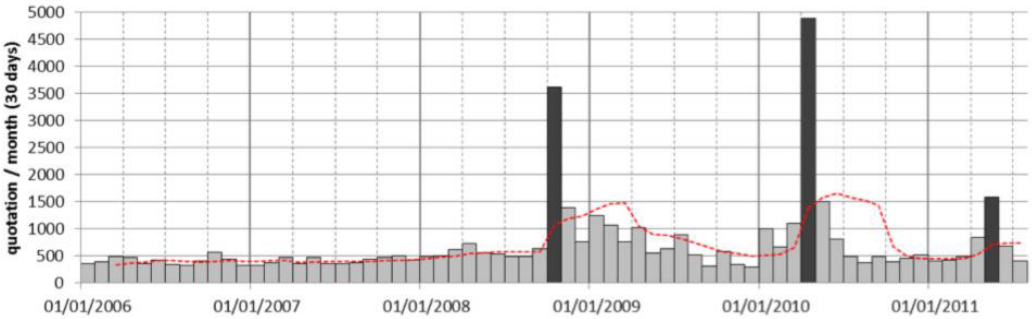


FIGURE 8.2 – Quantité d'articles parlant de l'Islande au cours du temps (base FACTIVA, extrait de [GGS11])

Un évènement médiatique peut être assimilé à un élément saillant du flux d'information. Formellement, il correspond donc à *une valeur inattendue de l'attention médiatique* [G+11]. Au niveau de l'analyse temporelle, il s'agit d'un brusque accroissement de l'intérêt porté sur un pays donné. La figure 8.2 donne par exemple le nombre d'articles mentionnant l'Islande, pour chaque mois de la période allant de janvier 2006 à juillet 2011, parmi l'ensemble des articles enregistrés dans la base de données FACTIVA[3]. Trois pics d'intérêt sont aisément détectés. Ils correspondent à des évènements ayant reçu une attention médiatique particulière (les deux éruptions volcaniques et la crise financière survenues durant la période d'observation [GGS11]). Notons, cependant, que ces accroissements de l'attention médiatique peuvent être détectés à différentes échelles temporelles :

- à l'échelle des jours : catastrophes climatiques sans répercussions humanitaires, discours de personnalités politiques, scrutins lors d'élections nationales, rencontres sportives ;
- à l'échelle des semaines ou des mois : soulèvements, affaires judiciaires de grande ampleur, campagnes présidentielles, tournois sportifs ;
- à l'échelle des années : crise de l'euro, conflits armés, printemps arabe.

[3] https://global.factiva.com

Ces accroissements concernent également plusieurs échelles géographiques : échelle *nationale* dans le cas d'une campagne présidentielle, échelle *internationale* dans le cas de conflits armés, échelle *mondiale* dans le cas de crises économiques touchant l'ensemble du système international. Pour les chercheurs en Relations internationales, tout comme dans les sciences sociales en général, l'utilisation de ces différents niveaux de description est monnaie courante. Ils constituent des abstractions pour l'analyse macroscopique des dynamiques sociales (*cf.* section 2.1).

Les évènements médiatiques sont de bons indicateurs pour la représentation du système international. Cependant, il est nécessaire de donner une définition multi-échelle de ces évènements, afin de fournir une sémantique pertinente pour l'analyse macroscopique du système. À ce titre, la méthode d'agrégation présentée dans la deuxième partie de cette thèse offre des outils d'abstraction adéquats.

8.2 Agrégation spatio-temporelle de l'attention médiatique

Cette section montre comment appliquer notre méthode d'agrégation à la détection d'évènements médiatiques macroscopiques. Au niveau microscopique, l'attention médiatique est modélisée par des *quantités de citations* (8.2.1) et un évènement médiatique par une *valeur inattendue* (8.2.2). Nous montrons comment utiliser la divergence de Kullback-Leibler pour détecter de telles valeurs (8.2.3) et nous explicitons la sémantique macroscopique qui leur est associée (8.2.4).

8.2.1 Représentation microscopique de l'attention médiatique

La représentation médiatique du système international repose sur l'extraction des informations contenues dans les articles publiés par un flux médiatique. Cette sous-section vise à formaliser le niveau de représentation microscopique à partir de trois *dimensions* d'analyse. Chacune dépend du type d'information extraite et est discrétisée pour former une *population* (*cf.* sous-section 3.2.1).

Dimension médiatique. La dimension médiatique est l'ensemble des *flux d'information médiatique* observés, correspondant aux différentes sources de l'information. Ils sont associés à plusieurs attributs : nationalité, localisation, langue, type de média, audience, fréquence, *etc.*

Définition 8.1. Nous notons Ω_f l'ensemble des flux d'information médiatique observés. Chaque article est associé à un unique flux $f \in \Omega_f$.

Dimension géographique. La dimension géographique est l'ensemble des *unités territoriales* retenues pour l'analyse spatiale de l'actualité. Cette dimension correspond à une discrétisation de l'espace géographique permettant de localiser les évènements relatés dans les articles.

Définition 8.2. Nous notons Ω_u l'ensemble des unités territoriales retenues pour la représentation géographique du système international. Un article est éventuellement associé à une (ou plusieurs) unité $u \in \Omega_u$ en fonction de son contenu.

Comme annoncé dans la section 2.2, nous prenons l'État comme niveau territorial de référence[4]. Ainsi, Ω_u désigne l'ensemble des 193 États Membres des Nations Unies[5]. Dans une perspective géographique, nous parlerons plutôt de « pays » afin d'insister sur la dimension spatiale de ces entités.

Dimension temporelle. La dimension temporelle correspond à une discrétisation du temps en *périodes d'observation* microscopiques. La datation des articles permet ainsi de préciser la temporalité des informations extraites.

Définition 8.3. Nous notons Ω_t l'ensemble des périodes d'observation servant à la représentation temporelle du système international. Chaque article est associé à une période $t \in \Omega_t$ selon sa date de publication.

Plusieurs granularités peuvent être envisagées pour discrétiser la dimension temporelle (heures, jours, semaines, mois, années). Dans les expériences qui suivent, nous choisissons le niveau des semaines afin de supprimer les variations liées aux cycles hebdomadaires et de nous concentrer sur des variations plus significatives, liées aux évènements médiatiques eux-mêmes.

[4] Nous soutenons ainsi une approche *réaliste* du système international [Bat09], selon laquelle les États sont les unités d'analyses privilégiées en Relations internationales.

[5] http://www.un.org/fr/members/

Notons que de nombreuses autres dimensions seraient utiles à l'analyse du système international. Il est par exemple intéressant de regrouper les articles en *catégories thématiques* à partir de leur contenu, afin de préciser le type d'évènement relaté : « politique », « économie », « conflit armé », « aide humanitaire », « sport », *etc.* Il est également possible d'introduire la notion d'*acteur de l'actualité* à partir de l'ensemble des protagonistes, physiques ou moraux, participant aux évènements relatés. Ce chapitre se concentre cependant sur les trois dimensions présentées ci-dessus. Elles fournissent déjà, à elles seules, une représentation microscopique complexe du système analysé.

Analyse tridimensionnelle. Les dimensions structurent la représentation microscopique de l'attention médiatique en trois axes d'analyse. Formellement, l'attention médiatique d'un flux donné pour un territoire donné durant une période donnée est mesurée par une *quantité de citations*. Les citations extraites des articles constituent donc les *unités atomiques* du processus d'observation (*cf.* section 3.2.1). Elles sont agrégées afin de constituer la représentation microscopique de référence, prenant la forme d'un cube de données tridimensionnel $\Omega_f \times \Omega_u \times \Omega_t$.

Définition 8.4. Pour chaque triplet (f, u, t) désignant un flux médiatique, une unité territoriale et une période d'observation, la valeur $v(f, u, t)$ correspond à la *quantité de citations* émises par le flux f, relatives au territoire u et situées pendant la période t. L'attribut v est interprété comme une mesure de l'attention médiatique.

Les données de la base GEOMEDIA[6]. Les expériences présentées dans ce chapitre exploitent l'instrument d'observation développé par le projet ANR CORPUS GEOMEDIA. Il s'agit d'un ensemble de capteurs médiatiques distribués géographiquement et collectant les articles de quotidiens en ligne grâce à la technologie RSS. Au 10 juin 2013, cette base de données contient 1 588 000 produits RSS[7] correspondant à des articles de presses publiés entre le 3 mai 2011 et le 10 juin 2013 par 131 journaux, de langue française ou anglaise, répartis dans 41 pays du monde. Les expériences présentées dans cette thèse exploitent principalement le flux RSS du *Guardian* consacré à l'actualité internationale[8]. Il a émis 59 234 produits RSS sur la pé-

[6] Observatoire des flux géomédiatiques internationaux (ANR-GUI-AAP-04), voir le blog dédié http://geomediatic.net/ et la proposition de projet [G+11].

[7] Ressource XML produite par un flux RSS. Dans le cas d'articles de presse, il contient diverses informations : titre, résumé, date de publication, auteur, *etc.* [G+11]

riode d'observation, soit une moyenne de 94 articles par jour. Chaque article est associé au pays $u \in \Omega_u$ lorsque celui-ci est cité dans le titre, le résumé ou le corps de l'article[9]. Dans le cas du *Guardian*, 77% des articles contiennent au moins une référence à l'un des 193 pays. En tout, 138 811 citations ont été extraites à partir du dictionnaire mis en place par les géographes [LPDV13].

Pour résumer, le niveau de représentation microscopique fait état de $|\Omega_f| = 131$ flux médiatiques (quotidiens en ligne), $|\Omega_u| = 193$ unités territoriales (pays membres des Nations Unies) et $|\Omega_t| = 90$ périodes d'observation (semaines allant du 4 mai 2011 au 20 janvier 2013). Le cube de données contient donc $|\Omega_f| \times |\Omega_u| \times |\Omega_t| \approx 2\ 280\ 000$ cellules. Dans ce chapitre, nous nous concentrons sur les 138 811 citations extraites des articles du *Guardian*, agrégées en une matrice de données contenant $|\Omega_u| \times |\Omega_t| \approx 17\ 370$ cellules, dont 74% sont non-nulles.

8.2.2 Détection d'évènements médiatiques

Les vecteurs du cube de données peuvent être spatialement ou temporellement visualisés afin de mettre en évidence les valeurs inattendues de l'attention médiatique. Il est alors nécessaire de déterminer quelles sont – à proprement parler – les *valeurs attendues*.

> Nous faisons l'hypothèse que les flux médiatiques sont homogènes dans le temps : on s'attend donc à ce que les quantités de citations soient constantes au cours du temps ou au moins proches d'une *attention médiatique globale*.

Variation temporelle de l'attention médiatique. Les flux observés n'ont pas nécessairement une activité constante. En particulier, la quantité d'articles publiés peut varier d'un jour à l'autre. L'hypothèse d'homogénéité concerne donc l'attention médiatique *relativement* à l'activité globale du flux. Nous parlons donc d'attention géographique *relative* pour désigner le rapport, sur une période de temps donnée, entre la quantité de citations *d'une unité territoriale* et la quantité de citations *de toutes les unités territoriales*.

[8] Quotidien britannique de la « presse de qualité ». *Cf.* la page consacrée au flux d'actualité internationale : http://www.guardian.co.uk/world.

[9] Il peut s'agir du nom du pays, des gentilés et adjectifs relatifs à ce pays, de sa capitale ou de ses villes principales, de ses principaux décideurs politiques, *etc.* L'analyse textuelle des produits RSS et l'extraction de l'information n'est pas la préoccupation de cette thèse. En particulier, la liste des mots-clés servant à l'extraction d'informations spatiales est sous la responsabilité scientifique des géographes et des spécialistes des média.

Définition 8.5. L'*attention géographique globale* du flux f pendant la période t, notée $v(f,t)$, est la quantité totale de citations sur cette période. L'*attention géographique relative* de l'unité territoriale u, notée $v_r(f,u,t|f,t)$[10], est le rapport entre l'attention médiatique concernant u et l'attention géographique globale :

$$v_r(f,u,t|f,t) = \frac{v(f,u,t)}{v(f,t)}$$

Exemple. L'attention médiatique du *Guardian* concernant les États-Unis était de **112 citations** la semaine du 17 octobre 2011 et de **132 citations** la semaine du 29 novembre 2012 – soit 1,2 fois plus. Cependant, il est important de préciser que le *Guardian* a effectué en tout **2 204 citations** la semaine du 17 octobre 2011 et seulement **1 249 citations** la semaine du 29 novembre 2012. Ainsi, l'attention géographique relative concernant les États-Unis était respectivement de **5,1% et 10,6% des citations** sur les deux semaines observées – soit 2 fois plus pour la semaine du 29 novembre 2012, durant laquelle a eu lieu l'élection présidentielle américaine.

Étant donnés un flux f et une unité territoriale u, le vecteur $(v_r(f,u,t|f,t))_{t\in\Omega_t}$ peut être visualisé par une série temporelle donnant la variation de l'attention géographique relative au cours du temps (*cf.* figure 8.3 pour le cas de la Grèce). Du fait de l'hypothèse d'homogénéité temporelle des flux, ces valeurs sont comparées à l'attention géographique relative *sur l'intégralité de la dimension temporelle* : $v_r(f,u,\Omega_t|f,\Omega_t)$. Il s'agit de la proportion des citations concernant u parmi l'intégralité des citations émises par f (2,5% dans le cas de la Grèce, *cf.* ligne en pointillés figure 8.3). Cette représentation permet de repérer des ruptures dans l'attention médiatique du *Guardian* et de mettre ainsi en évidence les évènements importants de l'actualité grecque selon le journal britannique.

Variation géographique de l'attention médiatique. Dans le cas d'une analyse géographique des évènements, nous parlons d'*attention temporelle relative* pour désigner le rapport entre la quantité de citations d'une unité territoriale *pendant une période donnée* et la quantité de citations de cette unité territoriale *sur l'intégralité de la période d'observation.* Selon l'hypothèse d'homogénéité temporelle des flux, l'attention temporelle relative devrait être proportionnelle à la taille de la période d'observation.

[10] Cette notation est à rapprocher de la notion de conditionnement en probabilité : $v_r(X|Y)$ est « l'attention médiatique de X *sachant* l'attention médiatique de Y ».

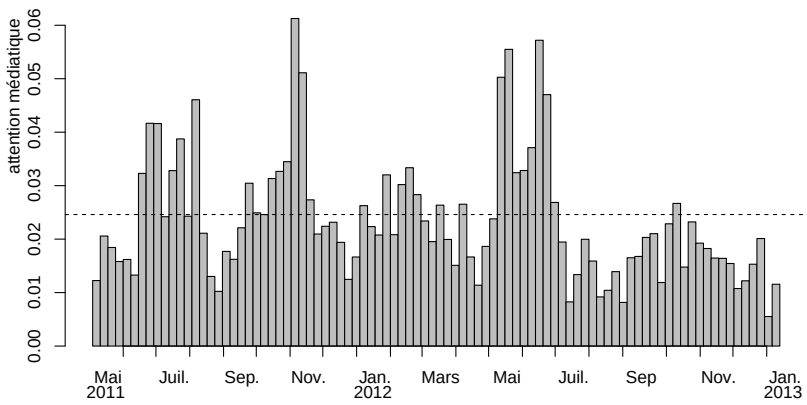


FIGURE 8.3 – Variation temporelle de l'attention géographique relative du *Guardian* concernant la Grèce (niveau hebdomadaire)

Définition 8.6. L'*attention temporelle globale* du flux f concernant l'unité territoriale u, notée $v(f, u)$, est la quantité totale de citations de cette unité territoriale. L'*attention temporelle relative* pendant la période t est le rapport entre l'attention médiatique pendant cette période et l'attention temporelle globale :

$$v_r(f, u, t|f, u) = \frac{v(f, u, t)}{v(f, u)}$$

Exemple. Toujours dans le cas du *Guardian*, l'attention médiatique concernant la Libye était de **111 citations** la semaine du 17 octobre 2011 – soit quasiment identique aux **112 citations** des États-Unis. Il est cependant important de noter que les États-Unis sont en général beaucoup plus cités que la Libye (**10 365 citations** contre **2 785 citations** sur l'intégralité de la période d'observation). Ainsi, l'attention temporelle relative était de **1,1% des citations** dans le cas des États-Unis et de **4,0% des citations** dans le cas de la Libye. Étant donné qu'une semaine représente 1/90e de la période d'observation totale, les valeurs attendues sont de l'ordre de 1,1%. L'attention concernant la Libye est donc 3,6 fois plus élevée que la moyenne, ce qui est expliqué par la mort du président libyen Mouammar Kadhafi le 20 octobre 2011.

Étant donnés un flux f et une période t, le vecteur $(v_r(f, u, t|f, u))_{u \in \Omega_u}$ peut être visualisé par une carte de citations. Celle présentée dans la figure 8.4 correspond à l'actualité du *Guardian* pendant le mois de juillet

2011. Comme cela correspond à 1/21e de la période d'observation totale, les valeurs devraient être proches de 4,7%. Les valeurs supérieures sont mises en évidence par la couleur rouge.

Un examen minutieux de cette carte permet d'identifier les régions qui ont bénéficié d'une attention médiatique inattendue durant le mois de juillet 2011 : au niveau de pays isolés (*e.g.*, attention de 6,3% en Thaïlande et de 15% au Guinée-Bissau) et au niveau de régions plus étendues (*e.g.*, attention médiatique des pays au nord de l'Amérique Centrale globalement forte – de 5,8% à 8,5% pour le Guatemala, Cuba, le Belize et le Nicaragua).

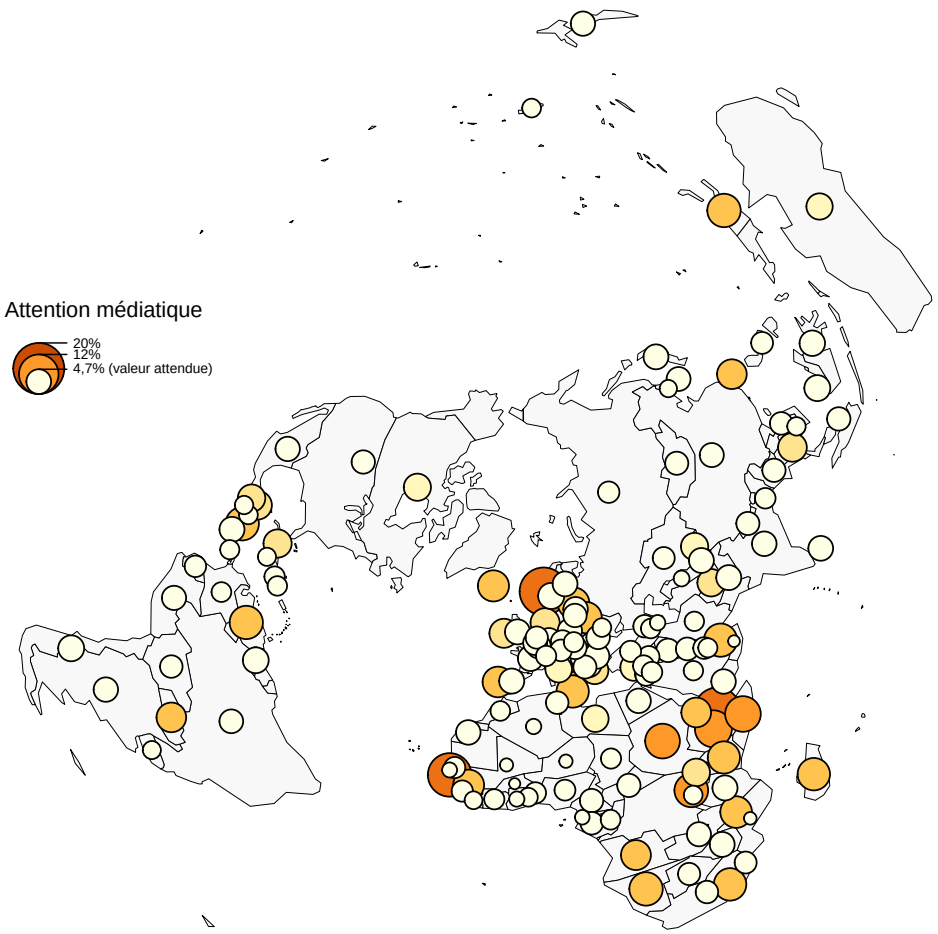


FIGURE 8.4 – Variation géographique de l'attention temporelle relative du *Guardian* pendant le mois de juillet 2011 (*cf.* zooms des figures 8.6 et 8.5 pour une meilleure lisibilité)

Cependant, la quantité d'information visualisée rend la lecture de la carte extrêmement difficile, en particulier pour les régions denses où l'attention médiatiques est particulièrement forte (*e.g.* Europe, Proche-Orient, Amérique centrale). Les figures 8.5 et 8.6 proposent des zooms sur les pays d'Afrique et d'Europe afin d'obtenir une meilleure lisibilité. Mais cette astuce ne permet pas d'avoir une vision globale de l'attention médiatique et le géographe doit alors procéder à une analyse parcellaire des données. Pourtant, la figure 8.4 contient énormément d'information redondante là où l'attention médiatique est homogène (*e.g.*, Asie, Amérique du nord, Amérique latine). Dans la section suivante, nous montrons que l'agrégation spatiale – représentant les évènements médiatiques à différentes échelles – améliore la lisibilité des cartes de citations tout en représentant le maximum d'information.

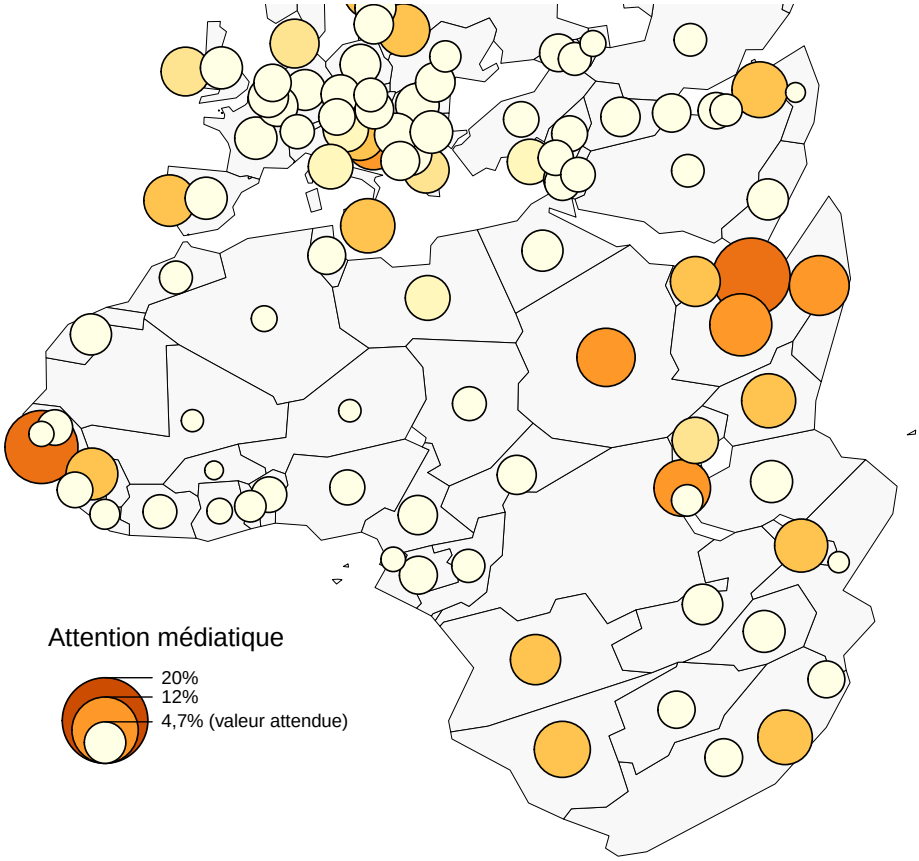


FIGURE 8.5 – Zoom sur l'attention temporelle relative du *Guardian* concernant les pays africains

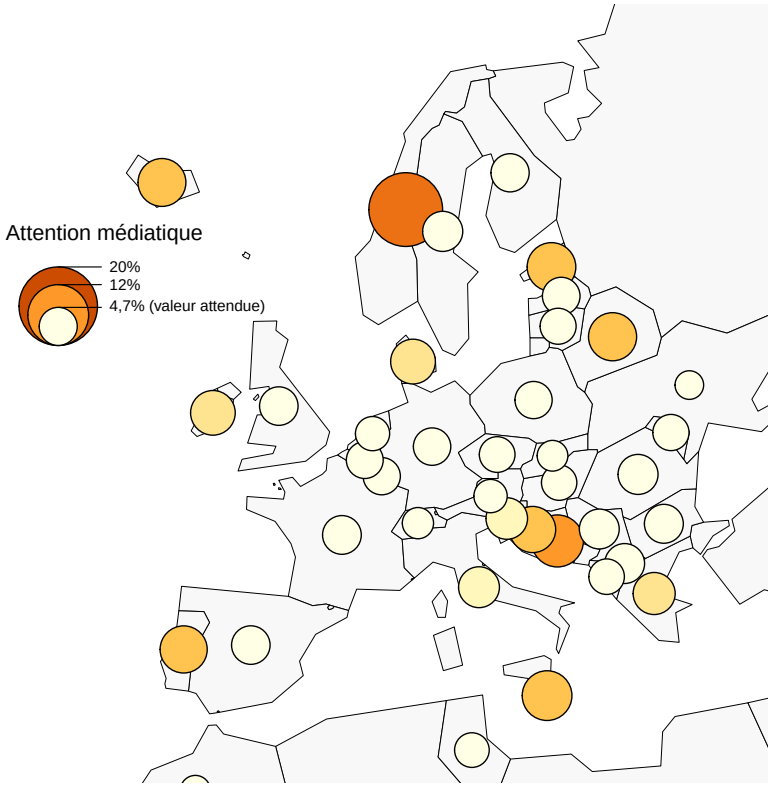


FIGURE 8.6 – Zoom sur l'attention temporelle relative du *Guardian* concernant les pays européens

Dans la section suivante, nous nous intéressons plus particulièrement à deux évènements de granularités différentes :

1. L'attention temporelle relative concernant la Norvège est de 18% (soit presque quatre fois la valeur attendue, *cf.* figure 8.6). Ceci est expliqué par les attentats survenus dans le pays le 22 juillet 2011 [11]. Il s'agit d'un évènement localisé au niveau national, c'est-à-dire d'un évènement microscopique selon la dimension géographique de référence.
2. L'attention temporelle relative concernant les pays de la Corne de l'Afrique (Rwanda, Soudan, Somalie, Éthiopie et Djibouti) varie de 9% à 16% (soit de deux à trois fois la valeur attendue, *cf.* figure 8.5). Ceci est expliqué par la crise alimentaire déclarée au début du mois de juillet 2011 dans cette région du monde[12]. Il s'agit donc d'un évènement étendu selon la dimension géographique, qui concerne une région du monde plus vaste que le seul niveau national.

[11] http://fr.wikipedia.org/wiki/Attentats_de_2011_en_Norv%C3%A8ge
[12] http://fr.wikipedia.org/wiki/Crise_alimentaire_de_2011_dans_la_Corne_de_l%27Afrique

8.2.3 Hypothèse de répartition non-uniforme

Contrairement au chapitre précédent, puisque l'attribut visualisé correspond à des quantités *relatives* d'unités atomiques, l'opérateur d'agrégation n'est pas ici une simple somme. En effet, l'attention (géographique ou temporelle) relative d'un agrégat *n'est pas* la somme des attentions relatives de ses individus. Étant donné un ensemble d'unités territoriales $U \subset \Omega_u$, les attentions temporelles *non-relatives* sont bel et bien sommées lors de l'agrégation :

Attention locale	**Attention temporelle globale**
$v(f, U, t) = \sum_{u \in U} v(f, u, t)$	$v(f, U) = \sum_{u \in U} v(f, u)$

En revanche, l'attention temporelle *relative* n'est pas sommée :

Attention temporelle relative

$$v_r(f, U, t | f, U) = \frac{v(f, U, t)}{v(f, U)} \neq \sum_{u \in U} v_r(f, u, t | f, u)$$

Il est possible de travailler à partir des quantités non-relatives, en utilisant une hypothèse de redistribution non-uniforme des unités (*cf.* section 3.2.3). Lors de l'*interprétation* des données agrégées, l'attention temporelle globale est utilisée pour redistribuer les citations (au lieu de la distribution uniforme) :

$$\forall u \in U, \quad p'(u) = \frac{v(f, u)}{v(f, U)} \quad \text{à la place de} \quad p'(u) = \frac{1}{|U|}$$

Pour tout $u \in U$, l'attention redistribuée selon la partie U est donc :

$$v_U(f, u, t) = \frac{v(f, u)}{v(f, U)} \, v(f, U, t)$$

Exemple. Voici l'attention médiatique du *Guardian* (f) concernant le Canada (can), le Mexique (mex) et les États-Unis (usa) durant la semaine du 23 avril 2012 (t) :

Attention locale			**Attention temporelle globale**		
$v(f, can, t)$	=	**11 citations**	$v(f, can)$	=	1 768 citations
$v(f, mex, t)$	=	**19 citations**	$v(f, mex)$	=	1 666 citations
$v(f, usa, t)$	=	**91 citations**	$v(f, usa)$	=	10 365 citations

Notons $W111$ l'ensemble des trois pays $\{can, mex, usa\}$[13]. Lors de l'agrégation, nous avons une attention locale de $v(f, W111, t) =$ **121 citations**. Selon l'hypothèse de redistribution uniforme, cette valeur agrégée est interprétée en supposant que les trois pays ont été cités **environ 40 fois chacun** (ce qui est loin d'être exact). En prenant en compte les attentions temporelles globales, nous remarquons que, sur la totalité de la période d'observation, le Canada capte en moyenne 13% des citations, le Mexique 12% et les États-Unis 75%. Une interprétation plus juste consiste donc à supposer que :

Attention redistribuée

$$\begin{array}{lclcl} v_{W111}(f, can, t) & = & 121 \times 13\% & = & \textbf{16 citations} \\ v_{W111}(f, mex, t) & = & 121 \times 12\% & = & \textbf{14 citations} \\ v_{W111}(f, usa, t) & = & 121 \times 75\% & = & \textbf{91 citations} \end{array}$$

Cette interprétation est beaucoup plus proche des véritables attentions locales. La connaissance des attentions médiatiques globales permet donc de mieux interpréter les valeurs agrégées.

Définition 8.7. Par conséquent, étant donnés un flux f, une semaine t et une partition des pays $\mathcal{X}$, la divergence de Kullback-Leibler est donnée par la formule suivante (*cf.* sous-section 4.3.2) :

$$D(\mathcal{X}) = \sum_{U \in \mathcal{X}} \sum_{u \in U} v(f, u, t) \log_2 \left(\frac{v(f, u, t)}{v(f, U, t)} \times \frac{v(f, U)}{v(f, u)} \right)$$

Il en va de même pour une partition $\mathcal{Y}$ de l'ensemble des semaines :

$$D(\mathcal{Y}) = \sum_{S \in \mathcal{Y}} \sum_{t \in S} v(f, u, t) \log_2 \left(\frac{v(f, u, t)}{v(f, u, S)} \times \frac{v(f, S)}{v(f, t)} \right)$$

[13] L'identifiant $W111$ est emprunté à la hiérarchie WUTS (*cf.* annexe C). Il désigne l'abstraction « Amérique du Nord », utilisée par les géographes pour parler conjointement des États-Unis, du Canada et du Mexique.

8.2.4 Sémantique des abstractions engendrées

Dimension géographique. Afin de constituer des agrégats de pays qui ont un sens pour l'analyse géographique du système international, nous souhaitons conserver certaines propriétés topologiques élémentaires de la population Ω_u telles que la relation de voisinage. Deux pays sont *voisins* lorsqu'ils partagent une frontière ou, éventuellement, lorsqu'ils disposent de voies maritimes directes. Un agrégat respectant cette propriété est une partie du graphe de voisinage constituée de nœuds connexes (*cf.* partitions admissibles selon un graphe, section 5.5).

Cependant, l'analyse des évènements et des relations internationales n'a pas uniquement recours à des explications d'ordre géographique. Le domaine des Relations internationales fait également intervenir des explications politiques, économiques, culturelles, historiques, *etc.* L'objectif est de modéliser également ces propriétés non-géographiques. Pour cela, nous exploitons la hiérarchie WUTS regroupant les pays du monde selon 5 niveaux d'analyse (*cf.* annexe C page 197). Cet outil d'abstraction géographique sert à produire des statistiques globales concernant les différentes régions du monde. La hiérarchie WUTS respecte la relation de voisinage (les agrégats sont des ensembles de nœuds connexes), mais s'appuie également sur des propriétés non-géographiques pour définir l'ensemble des parties admissibles. Les agrégats correspondent alors à des groupes de pays voisins partageant des caractéristiques politiques et culturelles [GD07].

Définition 8.8. Nous notons $\mathcal{W}(\Omega_u)$ l'ensemble des parties admissibles définies par la hiérarchie WUTS (*cf.* annexe C) et $\mathfrak{P}_{\mathcal{W}}(\Omega_u)$ l'ensemble des partitions admissibles de la dimension géographique selon cette hiérarchie (*cf.* section 5.2).

Dimension temporelle. L'agrégation d'une série temporelle nécessite également de conserver des propriétés élémentaires de Ω_t, notamment la relation d'ordre sur les périodes d'observation microscopiques.

Définition 8.9. Nous notons $<$ la relation d'ordre définie sur la population Ω_t. Les parties admissibles selon cet ordre sont constituées de périodes d'observations successives. Nous notons $\mathfrak{P}_{<}(\Omega_t)$ l'ensemble des partitions admissibles de la dimension temporelle selon cette relation d'ordre (*cf.* section 5.3).

8.3 Détection d'évènements médiatiques par agrégation

Cette section valide la méthode d'agrégation à partir de deux cas d'étude : le premier exploite les abstractions géographiques définies par la hiérarchie WUTS pour donner une représentation macroscopique des évènements médiatiques (8.3.1) ; le second exploite des abstractions temporelles (8.3.2). Dans les deux cas, nous montrons que l'algorithme des partitions optimales permet (1) de représenter de manière synthétique les données analysées, (2) d'attirer l'attention des valeurs remarquables et (3) de représenter ainsi le système international selon différents niveaux d'abstractions.

8.3.1 Agrégation et abstractions géographiques

Ce premier cas d'étude s'intéresse à l'actualité décrite par le *Guardian* durant le mois de juillet 2011. La figure 8.4 (page 135) donne la représentation microscopique de l'attention temporelle relative du flux d'information sous la forme d'une carte de citations. Nous y appliquons l'algorithme des partitions optimales en respectant la hiérarchie WUTS (*cf.* section précédente).

Le graphe des qualités présenté dans la figure 8.7 donne la réduction de complexité (en bleu) et la perte d'information (en rouge) associées aux partitions optimales engendrées par l'algorithme en fonction du coefficient de compromis spécifié par l'observateur. Pour $\alpha = 0$, nous obtenons la représentation microscopique (*cf.* figure 8.4). Celle-ci définit le niveau de représen-

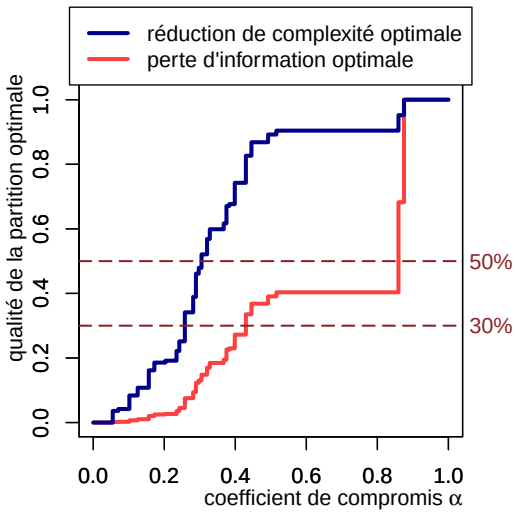


FIGURE 8.7 – Graphe de qualité correspondant à l'agrégation de la carte de citations de la figure 8.4

tation de référence et contient donc 100% de l'information disponible pour l'analyse. Cependant, elle est extrêmement difficile à lire dans la mesure où trop de données y sont représentées. Lorsque le coefficient α augmente, des agrégats géographiques sont choisis dans la hiérarchie WUTS pour simplifier la représentation de l'attention médiatique. Ainsi, l'observateur peut adapter la granularité des abstractions engendrées en fonction du niveau de détail désiré. Les figures 8.8 et 8.9 présentent deux partitions optimales choisies en fonction de la quantité d'information qu'elles contiennent. La carte de la figure 8.8 correspond à la partition la plus agrégée contenant au moins 50% de l'information microscopique (α inférieur à 0.85, *cf.* figure 8.7) et celle de la figure 8.9 au moins 70% de l'information microscopique (α inférieur à 0.42).

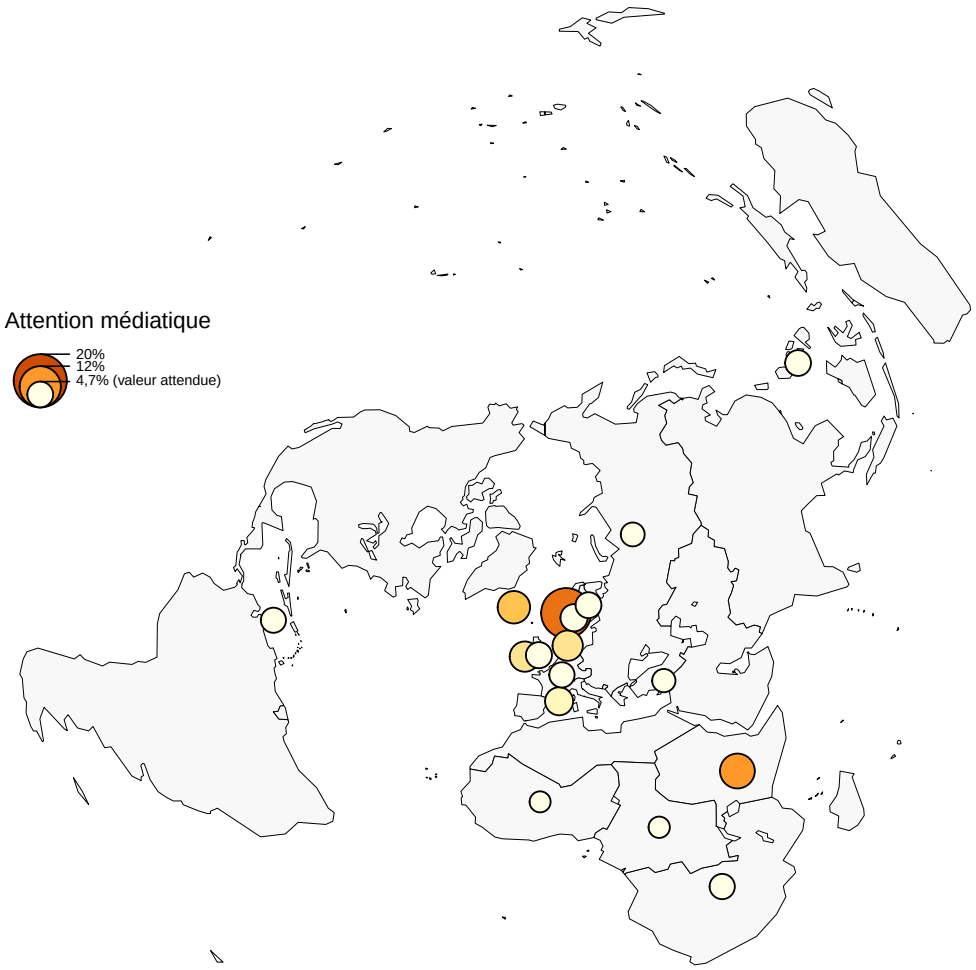


FIGURE 8.8 – Partition géographique optimale préservant au moins 50% de l'information microscopique (coefficient de compromis $\alpha \in [0.52, 0.85]$)

La carte de la figure 8.8 représente l'attention médiatique du *Guardian* concernant 17 agrégats géographiques. Deux d'entre eux (les Amériques et l'Asie-Pacifique), de très haut-niveau, ont une valeur proche de celle attendue selon l'hypothèse d'homogénéité temporelle des flux : aucun évènement de grande intensité n'a été relaté par le *Guardian* concernant cette région du monde. En revanche, deux évènements médiatiques sont immédiatement mis en évidence en Europe et en Afrique. Ils correspondent aux valeurs les plus inattendues et dont l'agrégation induirait une forte perte d'information (*cf.* accroissement de la perte pour $\alpha > 0.85$, figure 8.7).

1. La carte donne des détails concernant les pays d'Europe du Nord. Parmi eux, l'attention médiatique concernant la Norvège est très forte (18%, soit presque quatre fois l'attention médiatique attendue). L'observateur est donc immédiatement informé de l'occurrence d'un évènement de grande intensité au niveau national (les attentats du 22 juillet en Norvège, *cf.* page 137).
2. La carte donne également plus de détails concernant l'Afrique subsaharienne, mais à un niveau intermédiaire : les attentions médiatiques représentées concernent 4 régions mésoscopiques. L'observateur repère alors que l'attention médiatique concernant la Corne de l'Afrique est relativement élevée (8,6%, soit presque deux fois plus que la valeur attendue). Comme les détails du niveau national ne sont pas représentés pour cet agrégat, l'observateur considère que la valeur agrégée constitue, au moins en première analyse, une bonne approximation des valeurs sous-jacentes : *l'attention médiatique est uniformément élevée dans la Corne de l'Afrique.* Cet agrégat témoigne donc d'un évènement touchant une région géographique plus étendue que seul le niveau national. Il s'agit de la crise alimentaire déclarée en juillet 2011 dans les pays de la Corne de l'Afrique (*cf.* page 137).

L'algorithme des partitions hiérarchiques optimales permet donc de représenter les évènements médiatiques importants à différentes échelles géographiques.

La carte de la figure 8.9 est un peu plus détaillée que la carte précédente, notamment au niveau des pays d'Asie-Pacifique et des pays du Nord-Ouest de l'Afrique. D'autres évènements, d'intensité moindre, sont alors représentés :

3. Les violentes inondations qui ont débuté en Thaïlande vers la fin du mois de juillet 2011[14] (attention médiatique de 6,3%).

[14] http://fr.wikipedia.org/wiki/Inondations_de_2011_en_Tha%C3%AFlande#cite_note-TDG1-1

4. Le projet de coopération au développement qui a été engagé le 16 juillet entre l'Union européenne et la République de Guinée-Bissau (attention médiatique de 15%).

En revanche, la Corne de l'Afrique est toujours agrégée. Ce n'est que lorsqu'on souhaite représenter au moins 83% de l'information microscopique (α inférieure à 0.39) qu'on obtient les détails au niveau national de l'attention médiatique concernant cette région du monde.

> L'algorithme des partitions optimales engendre des abstractions en fonction du contexte d'analyse. Les évènements représentés et leur granularité dépendent donc des préférences de l'observateur.

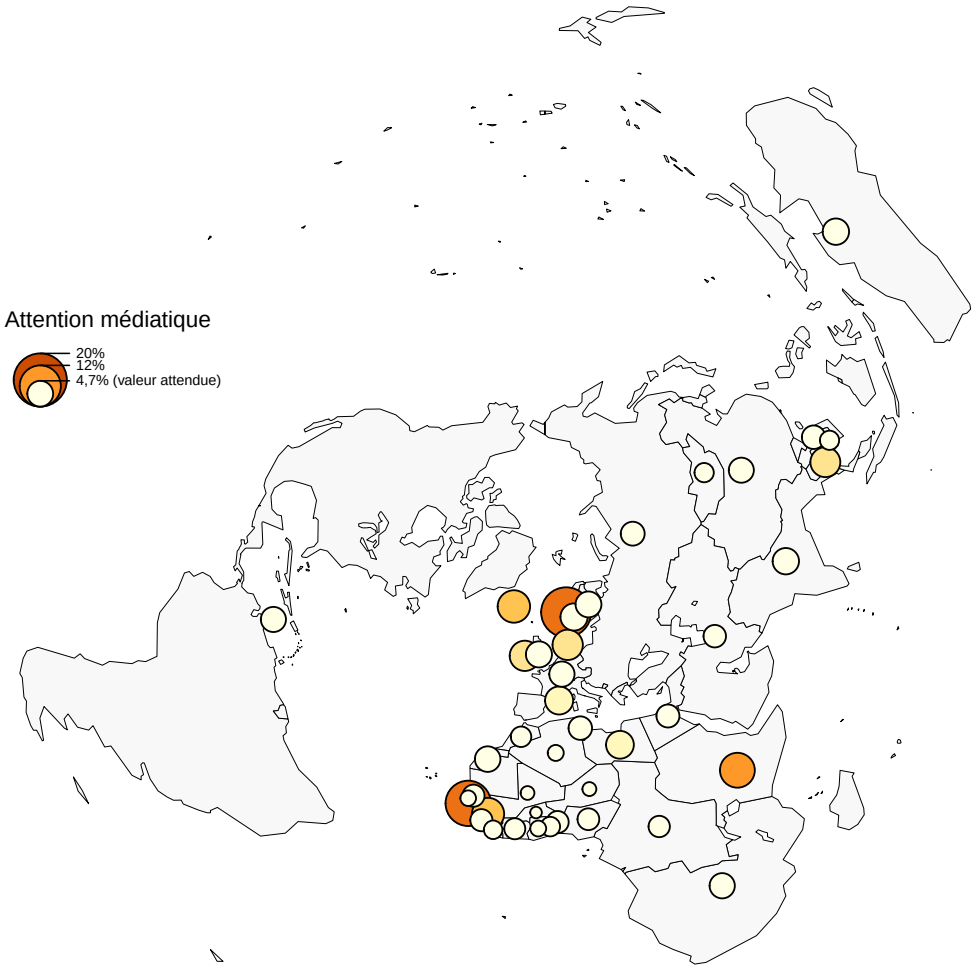


FIGURE 8.9 – Partition géographique optimale préservant au moins 70% de l'information microscopique (coefficient de compromis $\alpha \in [0.40, 0.42]$)

8.3.2 Agrégation ordonnée et abstractions temporelles

La même approche peut être appliquée à la dimension temporelle. Les évènements médiatiques ne sont plus identifiés en fonction de régions géographiques de forte attention médiatique, mais en fonction de périodes d'observation durant lesquelles l'attention prend des valeurs inattendues. Ces périodes, en rupture avec un état stable de l'attention médiatique, peuvent également être définies à plusieurs échelles (semaines, mois, années). L'algorithme des partitions optimales permet alors d'engendrer des représentations multirésolution pour l'analyse temporelle de l'actualité.

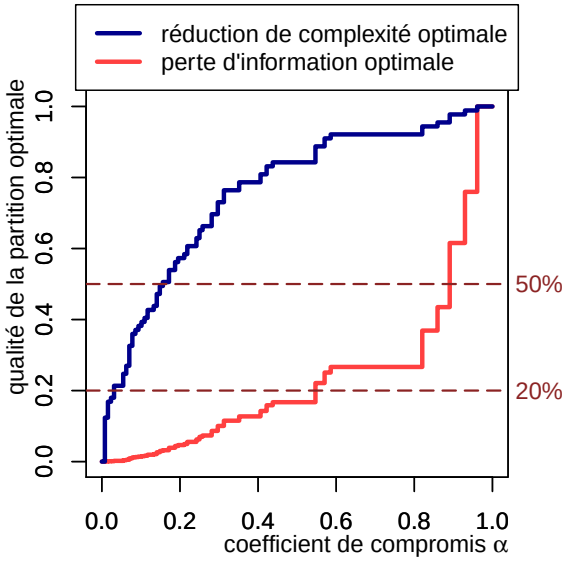


FIGURE 8.10 – Graphe de qualité correspondant à l'agrégation de la série de citations de la figure 8.11

Le graphe des qualités présenté dans la figure 8.10 permet de choisir le niveau de détail en contrôlant la réduction de complexité et la perte d'information de la partition engendrée. La série temporelle de la figure 8.11 donne la variation au cours du temps de l'attention géographique relative du *Guardian* concernant la Grèce. Elle définit donc l'information disponible au niveau de référence (niveau de représentation hebdomadaire). Les séries des figures 8.12 et 8.13 correspondent aux représentations les plus agrégées contenant respectivement au moins 80% et 50% de cette information microscopique (coefficient *alpha* respectivement inférieur à 0.44 et à 0.86, *cf.* figure 8.10).

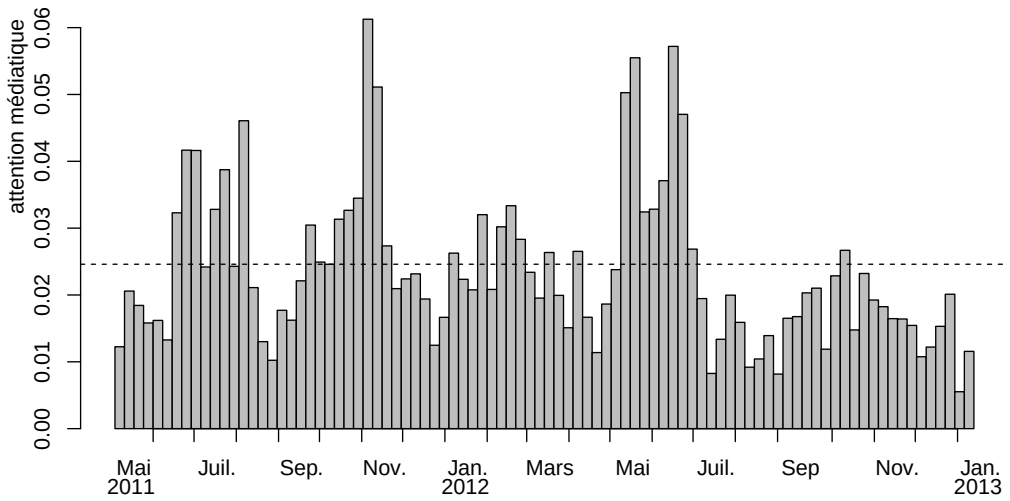


FIGURE 8.11 – Représentation microscopique de l'attention géographique relative du *Guardian* concernant la Grèce

La série temporelle présentée dans la figure 8.11 permet déjà de détecter des évènements importants dans l'actualité de la Grèce. La ligne horizontale en pointillés donne l'attention géographique relative sur l'intégralité de la période d'observation. Il s'agit du rapport entre *les citations de la Grèce* et *toutes les citations émises par le* Guardian *entre le 4 mai 2011 et le 20 janvier 2013*. Les pics d'attention dépassant cette valeur moyenne sont ainsi considérés comme la marque d'évènements médiatiques notables. Les périodes d'observation associées sont donc potentiellement importantes pour l'analyse. Nous nous concentrons sur trois évènements :

1. Le pic apparaissant au début du mois de novembre 2011 est expliqué par l'annonce, le 31 octobre, d'un référendum concernant la mise en place du plan d'austérité visant à réduire la dette publique grecque. Cette annonce provoque la surprise générale et est largement commentée par les média. Le 4 novembre, le ministre des Finances annonce l'abandon du référendum et le Premier ministre Georges Papandréou organise un vote de confiance du Parlement le soir même.
2. Le pic apparaissant au milieu du mois de mai 2012 est expliqué par l'échec des élections législatives du 6 mai, conclues le 16 mai par la mise en place d'un gouvernement intérimaire jusqu'à l'organisation de nouvelles élections.
3. Le pic apparaissant à la fin du mois de juin 2012 est expliqué par la tenue, le 17 juin, de ces nouvelles élections législatives[15].

[15] Voir la page Wikipédia dédiée à la crise de la dette publique grecque pour plus de

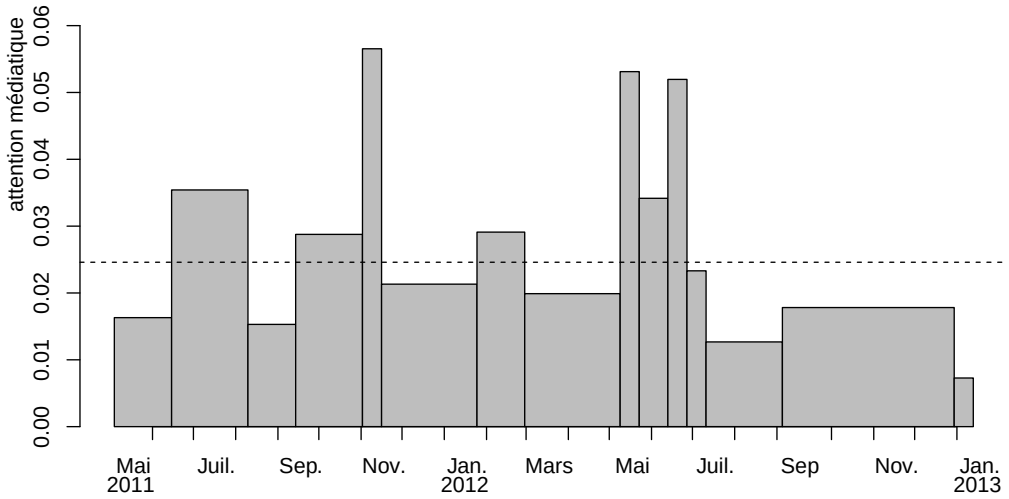


FIGURE 8.12 – Partition temporelle optimale préservant au moins 80% de l'information microscopique

La série temporelle de la figure 8.12 simplifie la série initiale en agrégeant des périodes d'observation consécutives (semaines) au sein desquelles l'attention médiatique est relativement homogène. Ne sont affichées que les attentions géographiques relatives des périodes agrégées. Comme elle contient moins de données, une telle représentation est beaucoup plus facile à manipuler, à visualiser et à interpréter. Elle contient cependant suffisamment d'information pour procéder à l'analyse de l'actualité grecque. En particulier, les trois évènements répertoriés précédemment sont toujours visibles. Ils sont même mis en évidence et aisément détectés par l'observateur. De plus, les variations entre les pics sont représentées de manière globale (*e.g.*, décroissance de l'attention médiatique après le troisième pic, en juillet et août 2012). Cette représentation permet donc de décrire l'actualité à partir de périodes de temps macroscopiques.

La série temporelle de la figure 8.13 donne une représentation encore plus synthétique des données. Seuls les pics les plus intenses sont représentés, correspondant aux évènements majeurs de l'actualité grecque.

1. Le premier pic, correspondant à l'annonce du référendum le 31 octobre 2011, puis à celle de son abandon le 4 novembre, est fortement mis en évidence par le processus d'agrégation.

détails concernant la chronologie de ces évènements :
https://fr.wikipedia.org/wiki/Crise_de_la_dette_publique_grecque.

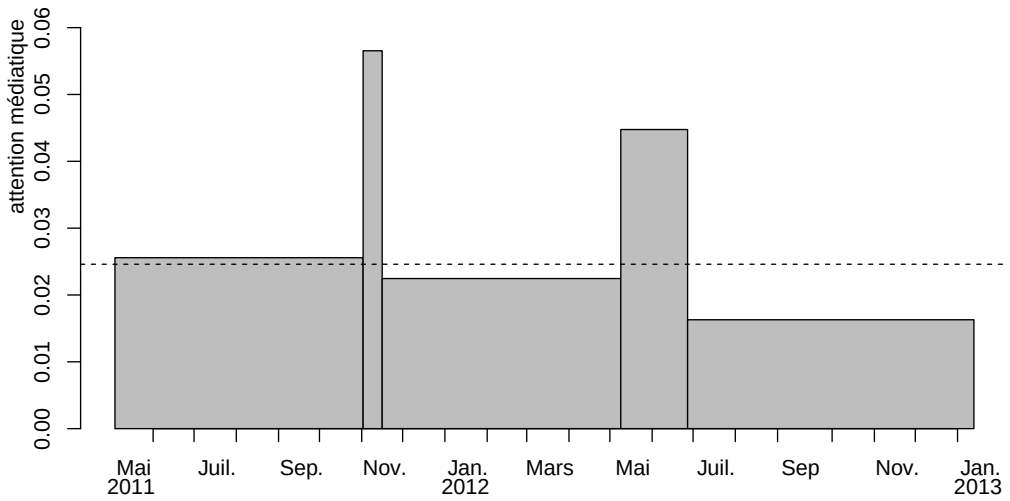


FIGURE 8.13 – Partition temporelle optimale préservant au moins 50% de l'information microscopique

2. Les pics de mai et de juin 2012, respectivement liés aux élections législatives du 6 mai et du 17 juin, sont agrégés entre eux et constituent maintenant une unique période d'observation de 7 semaines. Le pic ainsi formé est interprété comme relatif aux « élections législatives grecques de 2012 », sans donner le détail des évènements au niveau hebdomadaire. Il en résulte la création d'une abstraction temporelle pertinente pour décrire l'actualité grecque au niveau macroscopique.
3. Cette troisième série donne également une information globale intéressante : l'attention médiatique du *Guardian* concernant la Grèce a globalement diminué sur la période d'observation. Ceci pourrait être expliqué par la baisse de l'intérêt médiatique pour la crise grecque avec l'arrivée dans l'actualité de crises économiques d'autres pays européens tels que l'Espagne et l'Italie.

L'algorithme des partitions optimales permet donc de représenter les évènements médiatiques à plusieurs échelles de temps. Il met également en évidence les évolutions macroscopiques de l'attention médiatique. Si nous avions la profondeur et le recul temporel suffisants, nous pourrions ainsi identifier une période de temps macroscopique interprétée comme relative à « la crise de la dette publique grecque » représentée dans sa globalité.

8.4 Bilan et perspectives

L'une des difficultés majeures liées à l'analyse du système international réside dans la complexité sémantique des phénomènes observés (*cf.* section 2.1). Ces phénomènes peuvent en effet être rattachés à des entités de nature et de taille très différentes (*e.g.*, États, union d'États, organisations internationales, blocs géopolitiques) et leurs dynamiques interprétées selon des échelles de temps extrêmement variées (jours, mois, années, décennies). Une discussion centrale des Relations internationales s'intéresse justement aux niveaux de description appropriés pour l'analyse des dynamiques internationales [Bat09]. Nous soutenons l'idée que les difficultés sémantiques peuvent être résolues par la mise en place de processus d'abstraction appropriés et, en particulier, par des techniques d'agrégation (*cf.* chapitres 2 et 3). Dans le chapitre présent, nous avons donc exploité la méthode d'agrégation défendue dans cette thèse pour aborder les difficultés sémantiques liées à l'analyse *géographique* et *temporelle* du système international.

Premièrement, les mesures de qualité présentées dans le chapitre 4 permettent de s'assurer que les abstractions engendrées sont cohérentes avec les données microscopiques dont nous disposons pour l'analyse du système. Deuxièmement, les contraintes présentées dans le chapitre 5 garantissent que la sémantique de ces abstractions concorde avec les modèles géographiques utilisées dans le domaine des Relations internationales. À ce titre, nous avons également fourni dans [LPDV13] une palette d'outils pour l'évaluation des abstractions. Ces outils, qui n'ont pas été présentés en détail dans cette thèse, permettent notamment :

1. d'évaluer les abstractions géographiques en fonction du temps, afin de déterminer lorsqu'elles sont pertinentes pour l'analyse ;
2. d'évaluer les abstractions temporelles en fonction de l'espace, afin de déterminer l'échelle adéquate pour l'analyse des phénomènes ;
3. d'évaluer différents découpages de l'espace, afin de valider ou d'invalider les abstractions géographiques proposées dans le domaine[16].

Plus généralement, nous avons montré que l'algorithme des partitions optimales, présenté dans le chapitre 6, permet de combiner plusieurs abstractions afin de fournir une représentation globale du système international tout en détaillant les phénomènes important à plusieurs échelles d'espace et de temps. Ce chapitre montre donc que l'approche défendue dans cette thèse permet d'aborder la complexité sémantique du système international en engendrant des représentations macroscopiques. Nous pensons que ce résultat peut être généralisé à bien d'autres systèmes sociaux.

[16] *Cf.* la notion de *validation du biais externe* dans la sous-section 3.1.3.

Néanmoins, ce chapitre s'est concentré sur la conception d'abstractions unidimensionnelles géographiques *ou* temporelles. Cependant, les phénomènes internationaux peuvent être analysés selon de nombreuses autres dimensions et, également, de manière multidimensionnelle. Dans ce qui suit, nous proposons plusieurs pistes de recherche pour enrichir la sémantique macroscopique des évènements médiatiques.

Agrégation thématique. L'algorithme des partitions optimales agrège entre eux les évènements médiatiques proches dans le temps ou dans l'espace pour engendrer des abstractions géographiques ou temporelles. Cependant, de tels évènements ne sont pas nécessairement corrélés. Il est possible par exemple que deux pics médiatiques soient très proches dans le temps, mais qu'ils correspondent à des évènements de natures très différentes (*e.g.*, un évènement politique suivi d'un évènement climatique). Pour engendrer des abstractions cohérentes, le processus d'agrégation doit, dès lors, prendre en compte les différentes rubriques segmentant l'actualité.

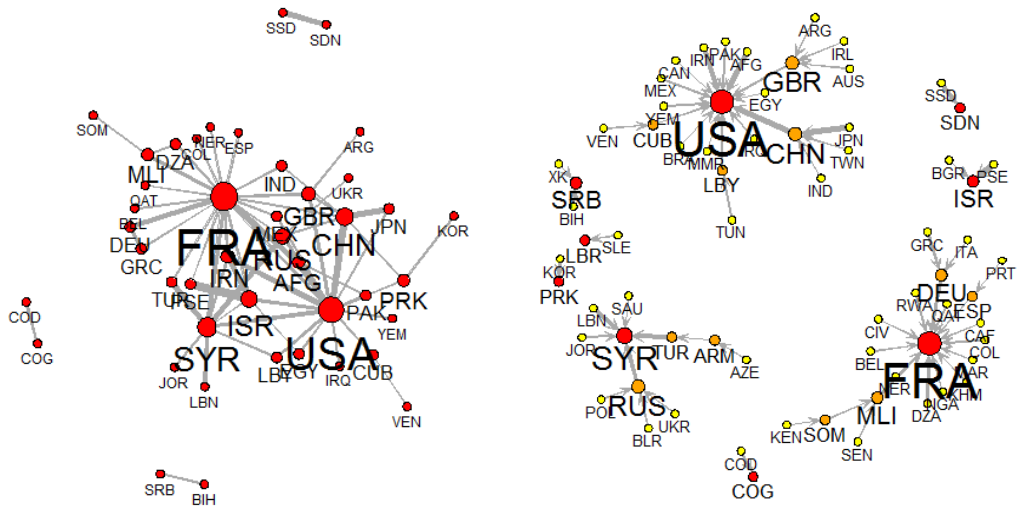
Nous proposons donc d'introduire une dimension *thématique* à la représentation de l'attention médiatique. L'objectif est de ranger les évènements détectés par thèmes (évènements politiques, culturels, économiques, climatiques, *etc.*). Dans la mesure où les catégories thématiques peuvent être plus ou moins générales ou spécialisées (le discours d'un homme politique peut par exemple être classé dans les catégories « discours politique », « meeting », « campagne présidentielle » ou « politique nationale »), la dimension thématique peut également être soumise à un processus d'agrégation. La définition des thèmes constituant la population microscopique, ainsi que la définition des partitions admissibles de cette population, relèvent du domaine de l'analyse textuelle et doivent être mises en place en étroite collaboration avec les sciences des média.

Notons cependant qu'il s'agit de mettre en place un processus d'abstraction par création de *concepts*, et non par création d'*objets* (*cf.* section 3.1). L'adaptation du processus d'agrégation à la création de concepts sera abordée plus en détail en perspective de cette thèse (*cf.* section 9.2).

Agrégation spatio-temporelle. Dans les expériences présentées dans ce chapitre, la dimension géographique et la dimension temporelle sont décorrélées. Nous définissons donc des abstractions géographiques pour une période donnée (*e.g.*, le mois de juillet 2011 dans la sous-section 8.3.1) ou des abstractions temporelles pour une région donnée (*e.g.*, la Grèce dans la sous-section 8.3.2). Cependant, le croisement des deux dimensions permet de définir un évènement comme *une région de l'espace-temps* ayant une forte atten-

tion médiatique. Les abstractions engendrées sont donc *à la fois* spatiales et temporelles. Pour cela, nous proposons d'appliquer la méthode d'abstraction à la population bi-dimensionnelle $\Omega_u \times \Omega_t$. Les parties admissibles sont des rectangles de la matrice de données correspondante, respectant la hiérarchie de Ω_u et l'ordre de Ω_t. L'ensemble des partitions admissibles est donc défini par le produit cartésien des ensembles des partitions admissibles des deux dimensions (*cf.* section 5.4).

Agrégation des co-citations. Une autre approche pour la représentation médiatique des relations internationales consiste à compter les *co-citations* des unités territoriales : deux pays sont co-cités lorsqu'ils apparaissent au sein d'un même article. La quantité de co-citations donne alors une indication concernant l'intensité des relations entre les deux pays, selon le point de vue d'un pays tiers (celui auquel appartient le flux médiatique). Par exemple, un article du *Times* parlant de l'intervention française au Mali donne un point de vue britannique sur les relations géopolitiques entre la France et le Mali. Le domaine des Relations internationales s'intéresse à ce genre d'information pour représenter notamment les *relations de domination politique* entre les différents pays du monde : un pays A est *dominé* par un pays B – au sein de la sphère médiatique – lorsque le pays B est très souvent cité et que le pays A est rarement cité seul [GGLP+13].



(a) Graphe des co-citations (b) Graphe de domination associé

FIGURE 8.14 – Graphe de co-citations et graphe de domination issus de la matrice des co-citations (extrait de [GGLP+13])

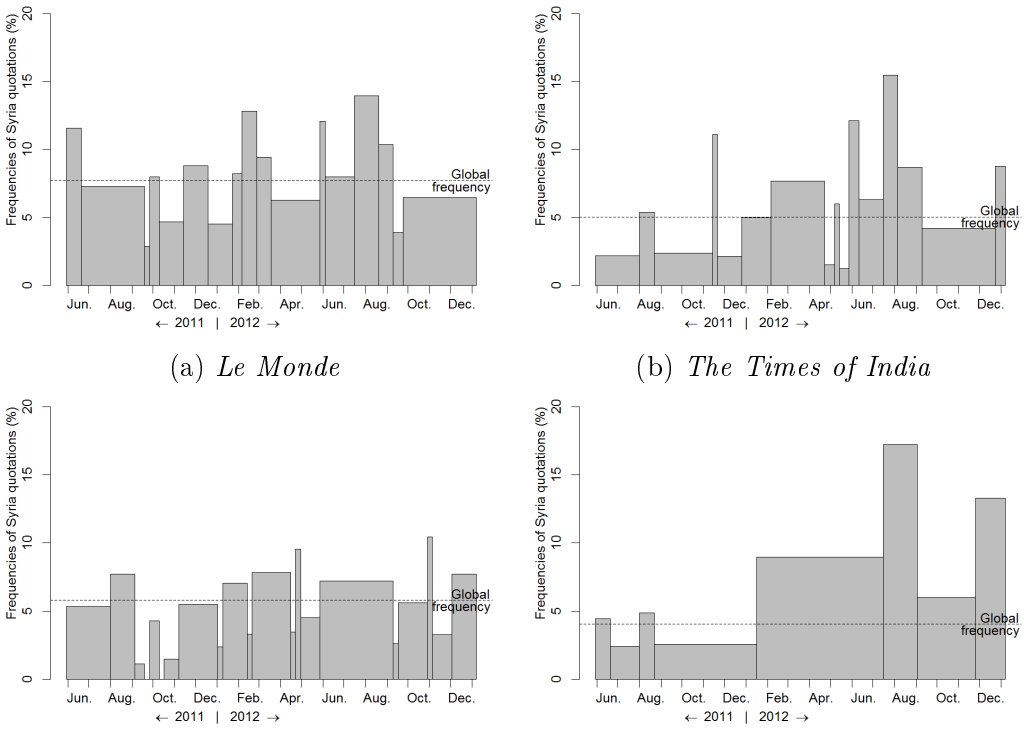
La conception d'abstractions à partir des co-citations nécessite d'agréger l'hyper-cube de données $\Omega_f \times \Omega_u \times \Omega_u \times \Omega_t$. L'*attention médiatique* $v(f, u_1, u_2, t)$ mesure alors la quantité de co-citations entre les unités territoriales u_1 et u_2 par le flux médiatique f pendant la période d'observation t. Le niveau de représentation microscopique peut être visualisé par un graphe de co-citations au sein duquel on recherche des valeurs inattendues (*cf.* figure 8.14). L'agrégation de l'hyper-cube peut respecter les mêmes contraintes géographiques que précédemment ou respecter la structure de graphe définie par la matrice des co-citations (section 5.5). Les abstractions constituées sont alors des groupes de pays associés de manière homogène dans l'actualité.

Agrégation des relations de causalité. Il est possible d'obtenir un profil temporel plus complet lors de l'extraction d'informations au sein des articles. Il est par exemple possible d'extraire les dates absolues (« le 11 mars 2011 »), les dates relatives (« il y a deux ans »), les durées (« pendant deux jours ») et les noms d'évènements (« catastrophe de Fukushima ») contenus dans le corps de l'article. Ces informations permettent de mieux caractériser la dynamique des évènements relatés, leurs relations de causalité, les effets de rappels, la variation de l'attention médiatique, *etc.*

La matrice des co-citations temporelles (lorsque qu'une date est citée par un article ou lorsque deux dates apparaissent dans le même article) peut être traitée de la même manière que la matrice des co-citations géographiques. Les abstractions ainsi constituées sont des groupes de dates associées de manière homogène dans l'actualité. Il est également possible de définir des relations de corrélations temporelles (un évènement est une cause potentielle d'un autre évènement lorsqu'ils sont temporellement liés dans un article). Les abstractions, engendrées en respectant l'ordre partiel ainsi défini (*cf.* section 5.5), sont donc des successions d'évènements liés causalement. D'autres exemples de structures plus complexes, croisant plusieurs dimensions spatiales et temporelles, sont proposées dans [LPDV11b].

Agrégation de la dimension médiatique. Dans ce chapitre, les expériences présentées ne font jamais intervenir plus d'un flux à la fois. Cependant, nous pensons que les mesures informationnelles présentées dans le chapitre 4 peuvent également être utilisées pour comparer et agréger les flux médiatiques entre eux. L'analyse de la dimension médiatique consiste alors : (1) à mesurer la redondance d'information entre deux flux, (2) à mesurer la « couverture médiatique » d'un ensemble de flux et (3) à classifier les flux en catégories homogènes. Ce dernier objectif consiste à partitionner et à agréger les flux médiatiques en fonction de leurs propriétés informationnelles. Comment com-

parer, dès lors, les séries temporelles associées à une même unité territoriale pour quatre flux différents ? (*cf.* figure 8.15) Comment comparer des cartes de citations, des matrices de co-citations, *etc.* ? Il est sans doute nécessaire de repenser la sémantique de l'agrégation lorsque celle-ci est appliquée aux flux médiatiques eux-mêmes.



(a) *Le Monde*

(b) *The Times of India*

(c) *The Financial Times*

(d) *The Washington Post*

FIGURE 8.15 – Variation temporelle de l'attention géographique relative de 4 flux concernant la Syrie (extrait de [GGLP+13])

CHAPITRE 9

Bilan et perspectives

Face à l'échec de la méthode analytique, cette thèse prétend participer au développement de l'approche systémique. En particulier, elle vise à l'édification d'une méthode scientifique générale pour l'*analyse macroscopique des grands systèmes*. Dans ce dernier chapitre, nous faisons le bilan des contributions apportées par ce travail de thèse, nous en exposons les hypothèses et les limites principales et donnons, enfin, quelques perspectives de recherche.

9.1 Contributions, limites et généralisation

La première partie a été consacrée à la définition d'un cadre méthodologique adéquat pour aborder les systèmes qui nous intéressent. Nous avons identifié les enjeux et les caractéristiques essentielles de la notion d'abstraction. Nous avons montré que l'*émergence épistémique* apporte les bases philosophiques suffisantes pour l'édification d'un tel cadre méthodologique en définissant les phénomènes macroscopiques comme le résultat d'un processus d'abstraction *subjectif* et *pragmatiste*. Nous avons ainsi montré que l'observateur, le contexte d'analyse et les objectifs visés ont une place prépondérante au sein de l'approche macroscopique. Nous avons mis en place un processus d'abstraction répondant à ces spécifications méthodologiques par la création d'*objets* macroscopiques. Le processus repose sur des techniques d'*agrégation* visant à engendrer des abstractions *spatio-temporelles* pour aborder la complexité syntaxique et sémantique des grands systèmes. Ce processus a été positionné vis-à-vis d'autres techniques d'analyse classiques telles que les techniques de *partitionnement*. Nous avons ainsi explicité les problématiques de recherche propres à l'agrégation de données.

Dans la deuxième partie, constituant la contribution théorique de la thèse, nous avons proposé des méthodes formelles pour le contrôle du processus d'agrégation. Nous avons montré qu'il est nécessaire d'*évaluer* et de *comparer* les représentations en fonction de leur adéquation avec le contexte d'analyse et les objectifs visés par l'observateur. Nous avons ainsi défini deux *mesures de qualité* cohérentes avec les objectifs de l'analyse macroscopique : la *réduction de complexité*, afin de garantir l'exploitabilité des représentations lors du passage à l'échelle, et la *perte d'information*, afin d'interpréter correctement les objets macroscopiques et de préserver les comportements microscopiques significatifs lors de l'agrégation. Nous avons donné un cadre formel générique pour évaluer, comparer et sélectionner les « meilleures » représentations à partir de mesures de qualité *combinées*. Cette méthode est générique et peut être appliquée à d'autres contextes et à d'autres objectifs dès lors que l'on est capable de quantifier les critères qui leurs sont associés.

Nous avons également montré qu'il est nécessaire, afin de conserver leur pouvoir explicatif, que les abstractions soient cohérentes avec les connaissances mobilisées par l'observateur pour analyser le système. Nous avons proposé un modèle de contraintes sur l'ensemble des *partitions admissibles*, plus expressif que les techniques communément utilisées par le partitionnement à base d'instances (*instance-level constrained clustering* [DB07]). Nous nous sommes limités à deux ensembles contraints, visant à conserver les propriétés *topologiques* des *hiérarchies constitutives* et des *relations d'ordre*, organisant respectivement les dimensions spatiales et temporelles des systèmes. Pour autant, cette méthode ouvre de nombreuses perspectives de recherche consistant à exprimer la syntaxe et la sémantique des abstractions en fonction du système observé et des « batteries explicatives » utilisées par l'observateur.

Afin d'automatiser le processus d'agrégation, nous avons apporté une solution algorithmique au *problème des partitions optimales*. L'algorithme ainsi proposé repose sur une propriété algébrique essentielle des mesures de qualité à optimiser : le *principe d'optimalité*. Initialement formulé dans [JSB+05], nous avons consolidé ce principe sur le plan logique et proposé une solution générique : l'*algorithme des partitions admissibles optimales*. Nous ne pensons pas que cet algorithme puisse être utilisé en dehors du principe d'optimalité. Pour des mesures de qualité quelconques, il faut donc en revenir aux heuristiques développées dans le domaine du partitionnement de données [HBV01]. Nous avons montré que la complexité spatiale et temporelle de l'algorithme proposé dépend des contraintes d'admissibilité et, plus spécifiquement, de la *relation de couverture* induite par ces contraintes. Nous avons explicité les classes de complexité associées à des implémentations spécialisées de l'algorithme : complexité *linéaire* dans le cas d'une organisation hiérarchique et complexité *quadratique* dans le cas d'un système ordonné.

D'autres implémentations spécialisées de cet algorithme générique pourront être développées sur le même principe (*e.g.*, dans le cas d'un ordre partiel ou d'un graphe). Ces implémentations permettront de préciser la complexité du problème des partitions optimale dans d'autres cadres d'analyse.

Dans la troisième partie, nous avons procédé à l'évaluation empirique du processus d'agrégation et des méthodes de contrôle proposées. L'objectif était de montrer que l'approche macroscopique est exploitable en pratique et qu'elle est applicable à une large classe de systèmes. Nous nous sommes concentrés sur deux domaines d'application : la visualisation de *systèmes de calcul* pour l'analyse de performance et la représentation du *système international* via la détection d'évènements médiatiques. Dans ces deux cadres d'analyse, nous avons montré que le processus d'agrégation permet de détecter et d'expliquer les comportements irréguliers apparaissant à plusieurs niveaux de granularité (spatiale et temporelle) et ce à moindre coût (afin d'assurer le passage à l'échelle). Il reste néanmoins important de procéder à une évaluation plus systématique du processus en le confrontant directement aux attentes des experts (analystes et géographes). Une telle évaluation fera sans doute apparaître des contextes, des objectifs d'analyse et des sémantiques qui n'ont pas été abordées en détail dans cette thèse. Nous pensons avoir néanmoins construit les outils suffisants pour adapter notre approche à ces nouveaux critères.

Enfin, nous pensons que l'approche macroscopique peut être généralisée à de nombreux domaines d'application, dès lors que les conditions suivantes sont vérifiées :

1. On dispose d'un *instrument d'observation* définissant le niveau microscopique de référence. Cet instrument localise les entités observées en fonction de dimensions discrétisées du système (attribut de dénombrement, *cf.* sous-section 3.2.1).

2. On dispose d'un *opérateur d'agrégation* et d'un *modèle de redistribution* des entités observées, permettant d'interpréter les données agrégées (dans le cas général, on pourra utiliser l'*hypothèse de redistribution uniforme*, *cf.* sous-section 3.2.3).

3. On est capable d'exprimer les critères internes par des *mesures de qualité* satisfaisant le principe d'optimalité (par exemple, des mesures *décomposables par un opérateur croissant*, *cf.* sous-section 6.2.2).

4. On est capable d'exprimer les critères externes par des *contraintes d'admissibilité* sur l'ensemble des partitions, induisant une implémentation spécialisée de l'algorithme dont la complexité temporelle est abordable vis-à-vis des systèmes qui nous intéressent.

9.2 Perspectives de recherche

Abstraction par définition de concepts. Le processus d'agrégation proposé dans cette thèse permet de définir des *objets* macroscopiques en adaptant la granularité spatiale et temporelle des objets représentés. Pour autant, nous pensons que l'agrégation peut également servir à la définition de *concepts* et qu'elle peut ainsi être exploitée comme un processus de *généralisation* (*cf.* sous-section 3.1.1). Par exemple, dans le cas de l'observation d'articles de presse (*cf.* chapitre 8), la dimension *thématique* permet de spécifier la nature des évènements relatés (*e.g.*, « politique », « économie », « culture »). L'agrégation consiste alors à généraliser ces catégories en champs thématiques homogènes.

Nous pensons que les mesures de qualité définies dans cette thèse sont adaptées à ce contexte. Cependant, la discrétisation de l'espace thématique en catégories microscopiques est une étape extrêmement délicate, qui nécessite le concours des experts et une réévaluation continue. Les contraintes d'admissibilité dépendent de la structure de l'espace thématique : *e.g.*, réseaux sémantiques, hiérarchie de concepts [Wil05]. Nous pensons donc que, pour adapter le processus d'agrégation aux dimensions non-physiques des systèmes, les difficultés majeures résident dans l'édification d'une structure sémantique pertinente. L'approche macroscopique peut également bénéficier de processus d'abstraction hybrides, mêlant concepts génériques et objets macroscopiques [SS77].

Agrégation selon un ensemble de parties quelconques. Les représentations macroscopiques engendrées par notre approche reposent sur le *partitionnement* des dimensions à agréger, c'est-à-dire sur des ensembles de parties *disjointes* et *recouvrantes*. Cependant, nous pouvons généraliser en nous intéressant aux représentations engendrées selon des ensembles de parties *quelconques*. Dans le cas de parties *non-disjointes*, certains individus sont représentés au sein de plusieurs agrégats et, dans le cas de parties *non-recouvrantes*, certains individus ne sont pas représentés (*cf.* figure 9.1).

L'objectif de cette généralisation est ainsi d'élargir l'ensemble des représentations possibles. Par exemple, dans le cas de l'agrégation de l'espace géographique, des unités territoriales partageant une portion de leur territoire peuvent néanmoins coexister au sein de la même représentation (*e.g.*, l'Himalaya et la Chine). Au contraire, les unités territoriales qui ne présentent aucun intérêt pour l'observateur peuvent être simplement écartées afin de simplifier la représentation (mécanisme de *sélection* des données). Les mesures de qualité doivent alors être adaptées à ce contexte.

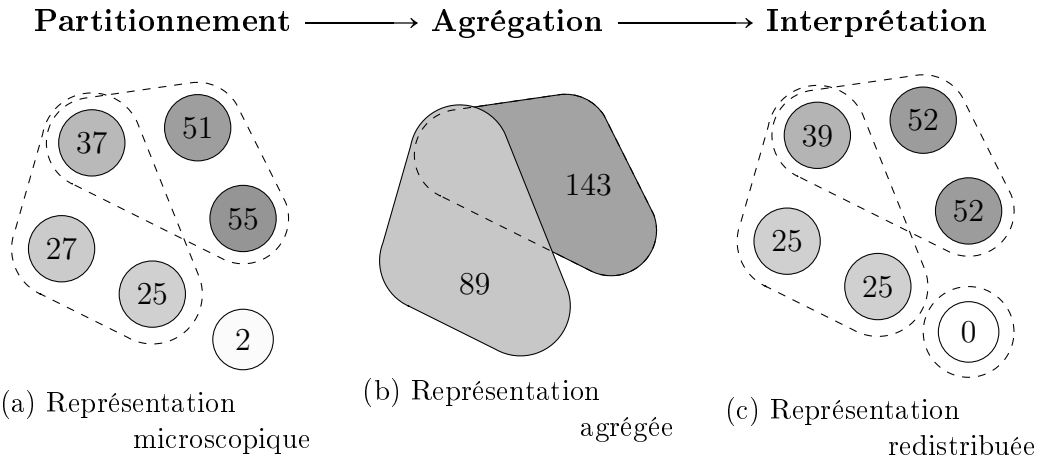


FIGURE 9.1 – Agrégation d'une population de 6 individus selon un ensemble de parties non-disjointes et non-recouvrantes

- La *taille des représentations* peut être définie comme dans la sous-section 4.3.1. Cependant, d'autres mesures de complexité peuvent être envisagées afin de quantifier et de minimiser les recouvrements entre agrégats (*e.g.*, mesures de redondance).

- Dans la mesure où l'observateur dispose de plus d'information concernant les individus appartenant à plusieurs agrégats et d'aucune information concernant ceux qui ne sont pas représentés, les mesures de *perte d'information* doivent être adaptées. Des hypothèses de redistribution peuvent par exemple être définies en fonction des mécanismes de recouvrement et de sélection des données (*cf.* figure 9.1c), afin de pouvoir utiliser la *divergence de Kullback-Leibler* telle que définie dans la section 4.3.2.

Les contraintes topologiques présentées dans le chapitre 5 peuvent également être adaptées à ce contexte. Cependant, l'ensemble des représentations admissibles est considérablement plus grand dans le cas de parties quelconques. Il apparaît donc nécessaire de vérifier que la complexité temporelle de l'algorithme des partitions optimales est toujours compatible avec le passage à l'échelle.

Observation macroscopique des grands systèmes. Le processus d'abstraction proposé dans cette thèse repose sur une hypothèse forte : nous disposons d'instruments d'observation capables de représenter le niveau microscopique. Dans le cas de systèmes de calcul (*cf.* chapitre 7), le résultat de l'observation microscopique consiste en des traces d'exécution (fonctions appelées, temps d'exécution, ressources consommées, *etc.*). Cependant, dans le cas d'applications parallèles *de grande taille*, l'étape de traçage est confrontée à plusieurs difficultés. Premièrement, du fait de la décentralisation des processus de calcul, il est nécessaire de disposer d'un instrument d'observation distribué, enregistrant le comportement des processus au niveau de chaque machine. Le nombre de « sondes » dépend donc de la taille de la plate-forme d'exécution. Deuxièmement, du fait de l'asynchronisme des processus, la juxtaposition des résultats du traçage distribué peut présenter des incohérences temporelles [CMV01]. Dans ces conditions, l'observation microscopique des grands systèmes passe difficilement à l'échelle.

La notion d'*observation macroscopique* consiste à engendrer une représentation macroscopique du système sans passer par l'observation détaillée et complète de ses parties [LP10] (*cf.* figure 9.2). Dans de précédents travaux, nous avons abordé cette approche dans le domaine des systèmes multi-agents en proposant une méthode d'agrégation distribuée et intégrée au sein même de l'exécution [LPDV11a]. L'agrégation *spatiale* est réalisée en plaçant dans l'environnement des *sondes macroscopiques* avec lesquelles les agents interagissent localement. Les données sont collectées, centralisées et agrégées au niveau de ces sondes. L'agrégation *temporelle* est réalisée à partir d'algorithmes de synchronisation distribués, tels que le *snapshot algorithm* [CL85],

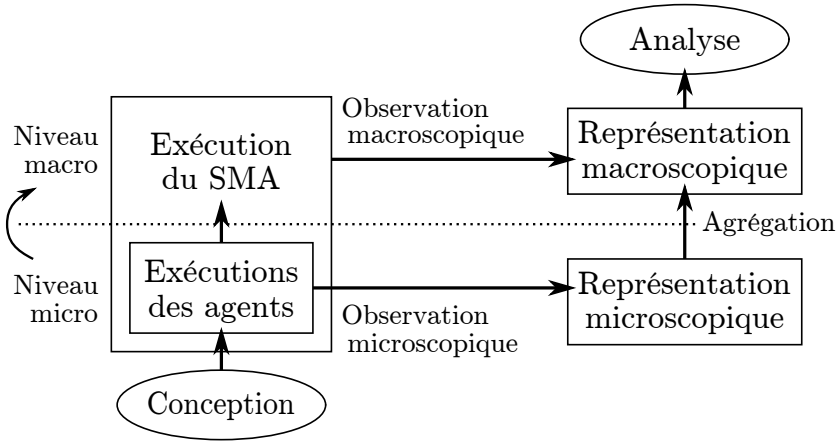


FIGURE 9.2 – Observation macroscopique des systèmes multi-agents (extrait de [LP10])

afin de produire un découpage du temps cohérent avec les « chaînes causales » constituant l'exécution [Mat89]. Les données sont alors synchronisées et agrégées en fonction des interactions locales entre les agents. De cette manière, les entités du système participent à leur propre représentation macroscopique.

L'intérêt d'une telle approche réside dans le fait que l'étape d'observation est réalisée par le système lui-même et s'adapte donc à sa complexité : le coût du processus d'agrégation est réparti dans l'espace et dans le temps, en fonction de l'activité microscopique des agents. Cependant, dans ces travaux préliminaires, les sondes macroscopiques sont placées – et le découpage temporel réalisé – de manière *ad hoc*, en fonction des phénomènes macroscopiques que l'on souhaite observer [LPDV11a]. En perspective de cette thèse, nous envisageons donc la possibilité d'incorporer l'algorithme des partitions admissibles optimales au sein de l'exécution, afin d'engendrer des représentations macroscopiques en fonction de critères internes *génériques* (réduction de complexité et perte d'information) et d'expliquer ces phénomènes en fonction de critères externes adaptés (structure causale de l'exécution).

La notion d'observation macroscopique aborde en réalité un problème épistémique dépassant le simple cadre de cette thèse : comment *observer* les grands systèmes ? Ce problème entraîne des questions méthodologiques (placement des sondes, fréquence d'acquisition des données) et techniques (algorithmes d'agrégations distribués, mesure de l'« effet de sonde ») non-triviales. Nous pensons néanmoins que l'approche défendue dans cette thèse constitue un point de départ pertinent pour aborder de telles questions.

BIBLIOGRAPHIE

[Aka73] Hirotogu AKAIKE. Information Theory and an Extension of the Maximum Likelihood Principle. *In* Samuel KOTZ et Norman L. JOHNSON, éditeurs. *Breakthroughs in Statistics. Volume 1 : Foundations and Basic Theory [1992]*, Springer Series in Statistics, Perspectives in Statistics, pages 610–624. Springer-Verlag, New York, 1973.

[Aka74] Hirotogu AKAIKE. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[ASZA11] Mehdi AGHAGOLZADEH, Hamid SOLTANIAN-ZADEH et Babak Nadjar ARAABI. Information Theoretic Hierarchical Clustering. *Entropy*, 13(2):450–465, 2011.

[Bat09] Dario BATTISTELLA. *Théorie des relations internationales*. Références Mondes. Les Presses de Science Po, Paris, 3e édition, 2009.

[BB04] Raymond BOUDON et François BOURRICAUD. *Dictionnaire critique de la sociologie*. Quadrige. Presses Universitaires de France, 7e édition, 2004.

[BCC+06] Raphaël BOLZE, Franck CAPPELLO, Eddy CARON, Michel DAYDÉ, Frédéric DESPREZ, Emmanuel JEANNOT, Yvon JÉGOU, Stéphane LANTÉRI, Julien LEDUC, Nouredine MELAB, Guillaume MORNET, Raymond NAMYST, Pascale PRIMET, Benjamin QUETIER, Olivier RICHARD, El-Ghazali TALBI et Iréa

TOUCHE. Grid'5000 : a large scale and highly reconfigurable Grid experimental testbed. *International Journal of High Performance Computing Applications*, 20(4):481–494, 2006.

[BD97] Éric BONABEAU et Jean-Louis DESSALLES. Detection and emergence. *Intellectica*, 25(2):85–94, 1997.

[Bed97] Mark A. BEDAU. Weak emergence. *Philosophical Perspectives : Mind, Causation, and World*, 11:375–399, 1997.

[Bed08] Mark A. BEDAU. Is Weak Emergence Just in the Mind ? *Minds and Machines*, 18(4):443–459, 2008.

[BHJR10] Holger BRUNST, Daniel HACKENBERG, Guido JUCKELAND et Heide ROHLING. Comprehensive Performance Tracking with Vampir 7. *In* Matthias S. MÜLLER, Michael M. RESCH, Alexander SCHULZ et Wolfgang E. NAGEL, éditeurs. *Tools for High Performance Computing 2009, Proceedings of the 3rd International Workshop on Parallel Tools for High Performance Computing*, pages 17–29. Springer-Verlag Berlin Heidelberg, 2010.

[Bis06] Christopher M. BISHOP. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[BPMG05] Fabio BOSCHETTI, Mikhail PROKOPENKO, Ian MACREADIE et Anne-Marie GRISOGONO. Defining and Detecting Emergence in Complex Networks. *In Proceedings of the 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES'05)*, volume 3684 de *Lecture Notes in Computer Science*, pages 573–580. Springer-Verlag Berlin Heidelberg, 2005.

[BT10] Daniel BEREND et Tamir TASSA. Improved bounds on bell numbers and on moments of sums of random variables. *Probability and Mathematical Statistics*, 30:185–205, 2010.

[Cha06] David J. CHALMERS. Strong and Weak Emergence. *In* Philip CLAYTON et Paul DAVIES, éditeurs. *The Re-Emergence of Emergence : The Emergentist Hypothesis from Science to Religion*. Oxford University Press, 2006.

[CL85] K. Mani CHANDY et Leslie LAMPORT. Distributed Snapshots : Determining Global States of Distributed Systems. *ACM Transactions on Computer Systems*, 3(1):63–75, 1985.

[CMV01] Jacques CHASSIN DE KERGOMMEAUX, Éric. MAILLET et Jean-Marc VINCENT. Monitoring Parallel Programs for Performance Tuning in Cluster Environments. *Parallel Program Development for Cluster Computing : Methodology, Tools and Integrated*, pages 131–150, 2001.

[CSB00] Jacques CHASSIN DE KERGOMMEAUX, Benhur de Oliveira STEIN et Pierre-Éric BERNARD. Pajé, an interactive visualization tool for tuning multi-threaded parallel applications. *Parallel Computing*, 26(10):1253–1274, 2000.

[Csi08] Imre CSISZÁR. Axiomatic Characterizations of Information Measures. *Entropy*, 10(3):261–273, 2008.

[Cui07] Qingguang CUI. *Measuring Data Abstraction Quality in Multiresolution Visualizations*. Mémoire de Master, Worcester Polytechnic Institute, 2007.

[Dar94] Vince DARLEY. Emergent Phenomena and Complexity. *In Proceedings of the 4th International Workshop on the Synthesis and Simulation of Living Systems*, Artificial Life IV, pages 411–416. MIT Press, 1994.

[DB07] Ian DAVIDSON et Sugato BASU. A Survey of Clustering with Instance Level Constraints. *In ACM Transactions on Knowledge Discovery from Data*, pages 1–41, 2007.

[DF94] Alexis DROGOUL et Jacques FERBER. Multi-Agent Simulation as a Tool for Modeling Societies : Application to Social Differentiation in Ant Colonies. *In* Cristiano CASTELFRANCHI et Eric WERNER, éditeurs. *Artificial Social Systems*, volume 830 de *Lecture Notes in Computer Science*, pages 2–23. Springer Berlin Heidelberg, 1994.

[DFP08] Jean-Louis DESSALLES, Jacques FERBER et Denis PHAN. Emergence in Agent-Based Computational Social Science : Conceptual, Formal, and Diagrammatic Analysis. *In* Ang YANG et Yin SHAN, éditeurs. *Intelligent Complex Adaptative Systems*, pages 255–299. IGI Global, Hershey, 2008.

[DMD06] Joris DEGUET, Laurent MAGNIN et Yves DEMAZEAU. Elements about the Emergence Issue : A Survey of Emergence Definitions. *ComPlexUs, Modelling in Systems Biology, Social, Cognitive and Information Sciences*, 3:24–31, 2006.

[DP02] Brian A. DAVEY et Hilary A. PRIESTLEY. *Introduction to Lattices and Order*. Cambridge University Press, Cambridge, UK, Seconde Edition, 2002.

[dR75] Joël de ROSNAY. *Le macroscope. Vers une vision globale*. Points Civilisation. Éditions du Seuil, Paris, France, 1975.

[Edm99] Bruce EDMONDS. What is Complexity ? – The philosophy of complexity *per se* with application to some examples in evolution. *In* Francis HEYLIGHEN et Diederik AERTS, éditeurs. *The Evolution of Complexity*, pages 1–18. Kluwer, Dordrecht, 1999.

[EF10] Niklas ELMQVIST et Jean-Daniel FEKETE. Hierarchical Aggregation for Information Visualization : Overview, Techniques, and Design Guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):439–454, 2010.

[FJ02] Mario A. T. FIGUEIREDO et Anil K. JAIN. Unsupervised learning of finite mixture models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(3):381–396, 2002.

[G+11] Claude GRASLAND *et al.* GEOMEDIA : Observatoire des flux géomédiatiques internationaux. Proposition de projet ANR-GUI-AAP-04, Agence Nationale de la Recherche (ANR Corpus), 2011.

[GBP07] Thierry GAUTIER, Xavier BESSERON et Laurent PIGEON. KAAPI : A thread scheduling runtime system for data flow computations on cluster of multi-processors. *In Proceedings of the 2007 International Workshop on Parallel Symbolic Computation (PASCO'07)*, pages 15–23. ACM New York, NY, USA, 2007.

[GD07] Claude GRASLAND et Clarisse DIDELON. Europe in the World – Final Report (Vol. 1). Rapport de recherche, ESPON project 3.4.1, 2007.

[GF12] Carlos GERSHENSON et Nelson FERNÁNDEZ. Complexity and Information : Measuring Emergence, Self-organization, and Homeostasis at Multiple Scales. *Complexity*, 18(2):29–44, 2012.

[GGLP+13] Timothée GIRAUD, Claude GRASLAND, Robin LAMARCHE-PERRIN, Yves DEMAZEAU et Jean-Marc VINCENT. Identification of International Media Events by Spatial and Temporal Aggregation of Newspapers RSS Flows. *In Proceedings of the 18th European Colloquium on Theoretical and Quantitative Geography (ECTQG'13)*, 2013.

[GGS11] Claude GRASLAND, Timothée GIRAUD et Marta SEVERO. Un capteur géomédiatique d'événements internationaux. *In Fonder les sciences du territoire. Proceedings of the 2011 International Conference of the International College of Territorial Sciences*, pages 184–190, Paris, France, 2011.

[GQLH12] Javier GIL-QUIJANO, Thomas LOUAIL et Guillaume HUTZLER. From Biological to Urban Cells : Lessons from Three Multilevel Agent-Based Models. *In* Nirmit DESAI, Alan LIU et Michael WINIKOFF, éditeurs. *Proceedings of the 13th International Conference on Principles and Practice of Multi-Agent Systems (PRIMA'10)*, volume 7057 de *Lecture Notes in Artificial Intelligence*, pages 620–635. Springer-Verlag Berlin Heidelberg, 2012.

[GR65] Johan GALTUNG et Mari Holmboe RUGE. The Structure of Foreign News : The Presentation of the Congo, Cuba and Cyprus Crises in Four Norwegian Newspapers. *Journal of Peace Research*, 2(1):64–91, 1965.

[GW99] Bernhard GANTER et Rudolf WILLE. *Formal Concept Analysis : Mathematical Foundations*. Springer-Verlag Berlin, Heidelberg, 1999.

[GZR+11] Qin GAO, Xuhui ZHANG, Pei-Luen RAU, Anthony MACIEJEWSKI et Howard SIEGEL. Performance Visualization for Large-Scale Computing Systems : A Literature Review. *In* Julie JACKO, éditeur. *Human-Computer Interaction. Design and Development Approaches*, volume 6761 de *Lecture Notes in Computer Science*, pages 450–460. Springer Berlin Heidelberg, 2011.

[HBV01] Maria HALKIDI, Yannis BATISTAKIS et Michalis VAZIRGIANNIS. On Clustering Validation Techniques. *Journal of Intelligent Information Systems*, 17(2-3):107–145, 2001.

[Huf52] David A. HUFFMAN. A Method for the Construction of Minimum-Redundancy Codes. *Proceedings of the Institute of Radio Engineers*, 40:1098–1101, 1952.

[JC97] M. R. Jean et Collectif IAD/SMA de AFCET/AFIA. Émergence et SMA. *In* Joël Quinqueton, Marie-Claude Thomas et Brigitte Trousse, éditeurs. *Actes des 5es Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents (JFIADSMA'97)*, pages 323–342. Hermès, 1997.

[JMF99] Anil K. Jain, M. Narasimha Murty et Patrick J. Flynn. Data Clustering : A Review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[JS91] Brian Johnson et Ben Shneiderman. Tree-Maps : a space-filling approach to the visualization of hierarchical information structures. *In Proceedings of the 2nd Conference on Visualization (VIS'91)*, pages 284–291. IEEE Computer Society Press, 1991.

[JSB+05] Brad Jackson, Jeffrey D. Scargle, David Barnes, Sundararajan Arabhi, Alina Alt, Peter Gioumousis, Elyus Gwin, Paungkaew Sangtrakulcharoen, Linda Tan et Tun Tao Tsai. An algorithm for optimal partitioning of data on an interval. *IEEE Signal Processing Letters*, 12(2):105–108, 2005.

[Kim99] Jaegwon Kim. Making Sense of Emergence. *Philosophical Studies*, 95(1-2):3–36, 1999.

[Kis07] Max Kistler. La réduction, l'émergence, l'unité de la science et les niveaux de réalité. *Matière première. Revue d'épistémologie et d'études matérialistes : Émergence et réductions*, pages 67–97, 2007.

[KL51] Solomon Kullback et Richard A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1): 79–86, 1951.

[Kol65] Andrey N. Kolmogorov. Three approaches to the quantitative definition of information. *Problems in Information Transmission*, 1(1):1–7, 1965.

[KV11] Ruud Koopmans et Rens Vliegenthart. Media Attention as the Outcome of a Diffusion Process–A Theoretical Framework and Cross-National Evidence on Earthquake Coverage. *European Sociological Review*, 27(5):636–653, 2011.

[Lau94] Alain LAURENT. *L'individualisme méthodologique.* Que sais-je ? Presses Universitaires de France, 1994.

[LGM+05] Jesús LABARTA, Judit GIMENEZ, E. MARTÍNEZ, P. GONZÁLEZ, Harald SERVAT, Germán LLORT et Xavier AGUILAR. Scalability of Visualization and Tracing Tools. *In Parallel Computing : Current & Future Issues of High-End Computing. Proceedings of the 2005 International Parallel Computing Conference (ParCo'05)*, volume 33, pages 869–876, 2005.

[LMO04] Tao LI, Sheng MA et Mitsunori OGIHARA. Entropy-Based Criterion in Categorical Clustering. *In Proceedings of the 21st International Conference on Machine Learning (ICML'04)*, pages 536–543, Banff, Canada, 2004.

[LP10] Robin LAMARCHE-PERRIN. Observation macroscopique pour l'analyse de systèmes multi-agents de trés grande taille. Mémoire de Master en informatique, Université Joseph Fourier, Grenoble, 2010.

[LP12] Robin LAMARCHE-PERRIN. Des collaborations possibles entre philosophie et Intelligence Artificielle. Mémoire de Master en philosophie, Université Pierre-Mendès-France, Grenoble, 2012.

[LPDV11a] Robin LAMARCHE-PERRIN, Yves DEMAZEAU et Jean-Marc VINCENT. Observation macroscopique et émergence dans les SMA de très grande taille. *In* Emmanuel ADAM et Jean-Paul SANSONNET, éditeurs. *Actes des 19e Journées Francophones sur les Systèmes Multi-Agents (JFSMA'11)*, pages 53–62. Éditions Cépaduès, 2011.

[LPDV11b] Robin LAMARCHE-PERRIN, Yves DEMAZEAU et Jean-Marc VINCENT. Organisation, agrégation et visualisation d'informations médiatiques. *In Fonder les sciences du territoire. Proceedings of the 2011 International Conference of the International College of Territorial Sciences*, pages 240–246, Paris, France, 2011.

[LPDV13] Robin LAMARCHE-PERRIN, Yves DEMAZEAU et Jean-Marc VINCENT. How to Build the Best Macroscopic Description of your Multi-agent System ? *In* Yves DEMAZEAU et Toru

ISHIDA, éditeurs. *Proceedings of the 11th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS'13)*, volume 7879 de *LNCS/LNAI*, pages 157–169. Springer-Verlag Berlin Heidelberg, 2013.

[LPSVD12] Robin LAMARCHE-PERRIN, Lucas M. SHNORR, Jean-Marc VINCENT et Yves DEMAZEAU. Evaluating Trace Aggregation Through Entropy Measures for Optimal Performance Visualization of Large Distributed Systems. Rapport de recherche RR-LIG-037, Laboratoire d'Informatique de Grenoble, Grenoble, France, 2012.

[LPVD12] Robin LAMARCHE-PERRIN, Jean-Marc VINCENT et Yves DEMAZEAU. Informational Measures of Aggregation for Complex Systems Analysis. Rapport de recherche RR-LIG-026, Laboratoire d'Informatique de Grenoble, Grenoble, France, 2012.

[Mat89] Friedemann MATTERN. Virtual Time and Global States of Distributed Systems. *Parallel and Distributed Algorithms*, 25:215–226, 1989.

[Moc09] Horatiu MOCIAN. Survey of Distributed Clustering Techniques. Mémoire de Master, Imperial College of London, 2009.

[MRS08] Christopher D. MANNING, Prabhakar RAGHAVAN et Hinrich SCHÜTZE. Hierarchical clustering. *In Introduction to Information Retrieval*, pages 377–401. Cambridge University Press, New York, NY, USA, 2008.

[OW06] Timothy O'CONNOR et Hong Yu WONG. Emergent properties. *In* Edward N. ZALTA, éditeur. *The Stanford Encyclopedia of Philosophy*. Stanford University, Winter 2006 Edition, 2006.

[PDH+13] Generoso PAGANO, Damien DOSIMONT, Guillaume HUARD, Vania MARANGOZOVA-MARTIN et Jean-Marc VINCENT. Trace Management and Analysis for Embedded Systems. *In Proceedings of the 7th International Symposium on Embedded Multicore SoCs (MCSoC'13)*. IEEE Computer Society Press, 2013.

[Pic04] Gauthier PICARD. *Méthodologie de développement de systèmes multi-agents adaptatifs et conception de logiciels à fonctionnalité émergente*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France, 2004.

[PMM12] Generoso PAGANO et Vania MARANGONZOVA-MARTIN. SoC-Trace Infrastructure. Rapport de recherche RT-0427, INRIA, 2012.

[PRT06] Riccardo M. PULSELLI, Carlo RATTI et Enzo TIEZZI. City out of Chaos : Social Patterns and Organization in Urban Systems. *International Journal of Ecodynamics*, 1(2):125–134, 2006.

[Rot64] Gian-Carlo ROTA. The Number of Partitions of a Set. *The American Mathematical Monthly*, 71(5):498–504, 1964.

[Saw01] R. Keith SAWYER. Simulating Emergence and Downward Causation in Small Groups. *In* Scott MOSS et Paul DAVIDSSON, éditeurs. *Multi-Agent-Based Simulation*, volume 1979 de *Lecture Notes in Computer Science*, pages 49–67. Springer Berlin Heidelberg, 2001.

[Sha48] Claude E. SHANNON. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423,623–656, 1948.

[Shn92] Ben SHNEIDERMAN. Tree visualization with Tree-maps : A 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1): 92–99, 1992.

[SHN09] Lucas M. SCHNORR, Guillaume HUARD et Philippe O. A. NAVAUX. Towards Visualization Scalability through Time Intervals and Hierarchical Organization of Monitoring Data. *In Proceedings of the 9th International Symposium on Cluster Computing and the Grid (CCGrid'09)*. IEEE Computer Society, 2009.

[SHN12] Lucas M. SCHNORR, Guillaume HUARD et Philippe O. A. NAVAUX. A hierarchical aggregation model to achieve visualization scalability in the analysis of parallel applications. *Parallel Computing*, 38(3):91–110, 2012.

[SL12] Lucas Mello SCHNORR et Arnaud LEGRAND. Visualizing More Performance Data Than What Fits on Your Screen. *In* Alexey CHEPTSOV, Steffen BRINKMANN, Michael M. RESCH et Wolfgang E. NAGEL, éditeurs. *Tools for High Performance Computing 2012*, pages 149–163. Springer, 2012.

[SLV12] Lucas M. SCHNORR, Arnaud LEGRAND et Jean-Marc VINCENT. Detection and Analysis of Resource Usage Anomalies in

Large Distributed Systems Through Multi-scale Visualization. *Concurrency and Computation : Practice and Experience*, 24(15):1792–1816, 2012.

[SLV13] Lucas M. SCHNORR, Arnaud LEGRAND et Jean-Marc VINCENT. Interactive Analysis of Large Distributed Systems with Scalable Topology-based Visualization. *In International Symposium on Performance Analysis of Systems and Software (ISPASS'13)*. IEEE Computer Society Press, 2013.

[SS77] John Miles SMITH et Diane C. P. SMITH. Database Abstractions : Aggregation and Generalization. *ACM Transactions Database Systems*, 2(2):105–133, 1977.

[Ste99] Achim STEPHAN. Varieties of Emergentism. *Evolution and Cognition*, 5(1):49–59, 1999.

[SZG+96] Doug SCHAFFER, Zhengping ZUO, Saul GREENBERG, Lyn BARTRAM, John DILL et Mark ROSEMAN. Navigating Hierarchically Clustered Networks Through Fisheye and Full-Zoom Methods. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 3(2):162–188, 1996.

[TB99] Luis TALAVERA et Javier BÉJAR. Integrating Declarative Knowledge in Hierarchical Clustering Tasks. *In* David J. HAND, Joost N. KOK et Michael R. BERTHOLD, éditeurs. *Proceedings of the 3rd International Symposium on Advances in Intelligent Data Analysis (IDA'99)*, volume 1642 de *Lecture notes in Computer science*, pages 211–222. Springer-Verlag Berlin Heidelberg, 1999.

[Uni07] UNITED NATIONS ENVIRONMENT PROGRAMME. *Global Environmental Outlook (GEO-4) : environment for development*, volume 4. UNEP, Nairobi, 2007.

[vB69] Ludwig von BERTALANFFY. *General System Theory : Foundations, Development, Applications*. George Braziller Inc., Revised Edition, 1969.

[vdV97] Gertrudis van de VIJVER. Emergence et explication. *Intellectica*, 25(2):7–23, 1997.

[VTR92] Francisco J. VARELA, Evan T. THOMPSON et Eleanor ROSCH. *The Embodied Mind : Cognitive Science and Human Experience.* The MIT Press, New Edition, 1992.

[WC00] Kiri WAGSTAFF et Claire CARDIE. Clustering with Instance-level Constraints. *In Proceedings of the 7th International Conference on Machine Learning*, pages 1103–1110, 2000.

[WF94a] Stanley WASSERMAN et Katherine FAUST. *Chapter 10. Blockmodels*, pages 394–424. Cambridge University Press, 1994.

[WF94b] Stanley WASSERMAN et Katherine FAUST. *Chapter 16. Stochastic Blockmodels and Goodness-of-Fit Indices*, pages 675–723. Cambridge University Press, 1994.

[Wil03] James M. WILSON. Gantt charts : A centenary appreciation. *European Journal of Operational Research*, 149(2):430–437, 2003.

[Wil05] Rudolf WILLE. Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies. *In* Bernhard GANTER, Gerd STUMME et Rudolf WILLE, éditeurs. *Formal Concept Analysis*, volume 3626 de *Lecture Notes in Computer Science*, pages 1–33. Springer Berlin Heidelberg, 2005.

Annexes

ANNEXE A

Agrégation de données et théorie de l'information

Cette section vise à présenter et à formaliser plusieurs mesures de la théorie de l'information permettant de caractériser le processus d'agrégation. Nous montrons notamment que la *divergence de Kullback-Leibler* (*cf.* sous-section 4.3.2) peut être interprétée comme :

1. une mesure de l'*irréversibilité* du processus d'agrégation ;
2. une mesure de la *perte de vraisemblance* de la représentation engendrée.

Notations. Soient Ω une population, $v(.)$ un attribut de dénombrement d'unités atomiques, $\mathcal{P}_\perp$ la partition microscopique de Ω et $\mathcal{X}$ une partition quelconque (*cf.* chapitre 3 pour le détail des notations utilisées dans cette annexe). Pour tout individu $x \in \Omega$,

$$p(x) = \frac{v(x)}{v(\Omega)}$$

est la probabilité qu'une unité, tirée de manière uniforme parmi les $v(\Omega)$ unités observées, soit associée à l'individu x. Pour toute partie $X \subset \Omega$, $p(X) = \sum_{x \in X} p(x)$ est la probabilité que cette unité soit associée à un individu de la partie X. La *distribution microscopique* des unités – associée à la partition microscopique $\mathcal{P}_\perp$ – est donnée par l'ensemble $(p(x))_{x \in \Omega}$ et leur *distribution agrégée* – associée à $\mathcal{X}$ – est donnée par l'ensemble $(p(X))_{X \in \mathcal{X}}$.

Quantité d'information contenue dans les distributions

L'*entropie de la distribution microscopique* $H(\mathcal{P}_\perp)$ mesure la quantité d'information qu'elle contient [Sha48]. Il s'agit du nombre optimal de bits d'information nécessaires en moyenne pour trouver l'individu associé à une unité (choisie de manière uniforme) :

$$H(\mathcal{P}_\perp) = -\sum_{x \in \Omega} p(x) \log_2 p(x)$$

L'*entropie de la distribution agrégée* $H(\mathcal{X})$ mesure également la quantité d'information qu'elle contient. Il s'agit du nombre optimal de bits d'information nécessaires en moyenne pour trouver la *partie* associée à une unité :

$$H(\mathcal{X}) = -\sum_{X \in \mathcal{X}} p(X) \log_2 p(X)$$

L'entropie de Shannon est parfois interprétée comme une *mesure de complexité* [Edm99, GF12]. En termes de ressources (*cf.* section 4.1), elle donne le nombre optimal de bits d'information nécessaires en moyenne pour *encoder* l'individu auquel est associé une unité choisie de manière uniforme (*cf.* par exemple le codage de Huffman [Huf52]). Ce qui nous intéresse alors, ce n'est pas de représenter les *quantités* d'unités associées aux individus (section 3.2), mais de représenter les *unités* elles-mêmes.

La *réduction d'entropie* (*cf.* sous-section 4.3.1) mesure alors la quantité d'information *économisée* par le processus d'agrégation, c'est-à-dire lors de l'encodage de la représentation macroscopique (*parties* auxquelles sont associées les unités) au lieu de l'encodage de la représentation microscopique (*individus* auxquels sont associées les unités) :

$$\Delta H(\mathcal{X}) = H(\mathcal{P}_\perp) - H(\mathcal{X})$$

Quantité d'information transmise par le processus d'agrégation

L'*information mutuelle* $I(\mathcal{P}_\perp, \mathcal{X})$ mesure la quantité d'information partagée par la distribution microscopique et la distribution agrégée [ASZA11]. En d'autres termes, elle mesure la quantité d'information *transmise* par le processus d'agrégation.

$$I(\mathcal{P}_\perp, \mathcal{X}) = \sum_{x \in \Omega} \sum_{X \in \mathcal{X}} p(x, X) \log_2 \frac{p(x, X)}{p(x)\, p(X)}$$

où $p(x, X)$ désigne la probabilité qu'une unité atomique soit à la fois associée à l'individu x (au niveau de la distribution microscopique) et à la partie X (au niveau de la distribution agrégée). Pour le processus d'agrégation défini dans le chapitre 3, toutes les unités associées à un individu x sont nécessairement associées à unique partie X (en d'autres termes, la distribution microscopique ne peut jamais être désagrégée) :

$$p(x, X) = \begin{cases} p(x) & \text{si } x \in X \\ 0 & \text{sinon} \end{cases}$$

On a donc :

$$\begin{aligned} I(\mathcal{P}_\perp, \mathcal{X}) &= \sum_{x \in \Omega} p(x) \log_2 \frac{p(x)}{p(x)\, p(X)} \\ &= -\sum_{X \in \mathcal{X}} p(X) \log_2 p(X) \\ &= H(\mathcal{X}) \end{aligned}$$

L'information mutuelle est donc égale à l'entropie de la représentation agrégée. Ainsi, toute l'information contenue dans la distribution agrégée est de l'information microscopique transmise par le processus d'agrégation : le processus d'agrégation *ne créé pas* d'information.

L'*entropie conditionnelle* mesure la quantité d'information relative à la distribution microscopique qui *n'est pas* contenue dans la distribution agrégée [ASZA11]. Il s'agit du nombre optimal de bits d'information nécessaires en moyenne pour trouver l'individu associée à une unité sachant la partie à laquelle elle est associée. En d'autres termes, l'entropie conditionnelle mesure la quantité d'information *perdue* par le processus d'agrégation.

$$\begin{aligned} H(\mathcal{P}_\perp | \mathcal{X}) &= H(\mathcal{P}_\perp) - I(\mathcal{P}_\perp, \mathcal{X}) \\ &= H(\mathcal{P}_\perp) - H(\mathcal{X}) \\ &= \Delta H(\mathcal{X}) \end{aligned}$$

L'entropie conditionnelle est donc égale à la réduction d'entropie.

Quantité d'information nécessaire pour renverser le processus

L'*information de désagrégation* $L(\mathcal{X})$ mesure la quantité d'information nécessaire pour retrouver la distribution microscopique à partir de la distribution macroscopique [LPVD12]. Il s'agit du nombre de bits d'information nécessaires en moyenne pour trouver l'individu associé à une unité lorsqu'on connaît la partie à laquelle elle est associée, mais qu'on *ne connaît pas* la distribution de probabilité interne de la partie. On suppose par exemple une *distribution uniforme* des unités au sein de la partie (*cf.* sous-section 3.2.3). En d'autres termes, l'information de désagrégation mesure la quantité d'information nécessaire pour *renverser* le processus d'agrégation lorsqu'on ne connaît que la distribution agrégée :

$$L(\mathcal{X}) = \sum_{X \in \mathcal{X}} p(X) \log_2 |X|$$

L'information de désagrégation peut également être interprétée comme une mesure de vraisemblance concernant *la probabilité de retrouver la distribution microscopique à partir de la distribution agrégée en tirant les unités uniformément au sein des parties.* Étant donnée une unité associée à une partie $X \in \mathcal{X}$, la probabilité d'associer cette unité à un individu $x \in X$ est $p'(x|X) = \frac{1}{|X|}$, où $|X|$ est la taille de la partie. La probabilité d'associer l'ensemble des $v(\Omega)$ unités aux « bons » individus est donc :

$$p'(\mathcal{P}_\perp|\mathcal{X}) = \prod_{X \in \mathcal{X}} \left(\frac{1}{|X|}\right)^{v(X)}$$

La *log-vraisemblance de la distribution agrégée* $\log \mathcal{L}(\mathcal{X}|\mathcal{P}_\perp)$ est alors :

$$\begin{aligned}
\log \mathcal{L}(\mathcal{X}|\mathcal{P}_\perp) &= -\log_2 \prod_{X \in \mathcal{X}} \left(\frac{1}{|X|}\right)^{v(X)} \\
&= \sum_{X \in \mathcal{X}} v(X) \log_2 |X| \\
&= v(\Omega)\ L(\mathcal{X})
\end{aligned}$$

L'information de désagrégation mesure donc la probabilité de désagréger correctement une distribution et, ainsi, de renverser le processus d'agrégation.

Quantifier l'irréversibilité du processus d'agrégation

On pourrait supposer que la quantité d'information nécessaire pour renverser le processus d'agrégation est égale à la quantité d'information perdue par le processus : on dira que le processus d'agrégation est *réversible* si et seulement si $L = \Delta H$. En d'autres termes, la quantité d'information économisée par le processus d'agrégation ($\Delta H(\mathcal{X})$) permet de retrouver la distribution microscopique ($L(\mathcal{X})$).

$$\begin{aligned}
L(\mathcal{X}) - \Delta H(\mathcal{X}) &= \sum_{X \in \mathcal{X}} p(X) \log_2 |X| + \sum_{x \in \Omega} p(x) \log_2 p(x) - \sum_{X \in \mathcal{X}} p(X) \log_2 p(X) \\
&= \sum_{X \in \mathcal{X}} \sum_{x \in X} (p(x) \log_2 |X| + p(x) \log_2 p(x) - p(x) \log_2 p(X)) \\
&= \sum_{X \in \mathcal{X}} \sum_{x \in X} p(x) \log_2 \left(\frac{p(x)|X|}{p(X)} \right) \\
&= D(\mathcal{X})
\end{aligned}$$

Nous retrouvons ainsi la *divergence* de Kullback-Leibler définie dans le cas de l'hypothèse de redistribution uniforme (*cf.* sous-section 4.3.2).

La divergence mesure donc la quantité d'information qui n'est pas récupérée lors qu'on redistribue uniformément les unités au sein des parties. Il s'agit du nombre de bits d'information *supplémentaires* nécessaires en moyenne pour trouver l'individu associé à une unité lorsqu'on utilise la distribution agrégée comme modèle de la distribution microscopique [KL51]. Or, nous avons : $D = L - \Delta H$. La divergence mesure donc la quantité d'information *supplémentaire* nécessaire pour renverser le processus d'agrégation si l'on dispose de la distribution agrégée *et* de l'information perdue par le processus. Ainsi, *le processus agrégation est réversible si et seulement si la divergence est nulle.* Dans ce cas, l'information perdue suffit à retrouver la distribution microscopique.

La divergence peut également être interprétée comme une mesure de vraisemblance concernant *la probabilité de retrouver la distribution microscopique en tirant les unités à partir de la distribution redistribuée* (*cf.* sous-section 3.2.3). Dans le cas de la distribution microscopique, la probabilité d'associer une unité à un individu $x \in \Omega$ est $p_{\mathcal{P}_\perp}(x) = p(x)$. La probabilité d'associer l'ensemble des $v(\Omega)$ unités aux « bons » individus est donc :

$$p(\mathcal{P}_\perp) = \prod_{x \in \Omega} p(x)^{v(x)}$$

La *log-vraisemblance de la distribution microscopique* $\log \mathcal{L}(\mathcal{P}_\perp)$ est donc :

$$\log \mathcal{L}(\mathcal{P}_\perp) = \sum_{x \in \Omega} v(x) \log_2 p(x)$$

Dans le cas de la distribution agrégée, la probabilité d'associer une unité à un individu $x \in X$ est $p_{\mathcal{X}}(x) = \frac{p(X)}{|X|}$. La probabilité d'associer l'ensemble des $v(\Omega)$ unités aux « bons » individus est donc :

$$p(\mathcal{X}) = \prod_{x \in \Omega} \left(\frac{p(X)}{|X|} \right)^{v(x)}$$

La *log-vraisemblance de la distribution agrégée* $\log \mathcal{L}(\mathcal{X})$ est donc :

$$\log \mathcal{L}(\mathcal{X}) = \sum_{x \in \Omega} v(x) \log_2 \left(\frac{p(X)}{|X|} \right)$$

On constate alors que :

$$\log \mathcal{L}(\mathcal{P}_\perp) - \log \mathcal{L}(\mathcal{X}) = v(\Omega)\ D(\mathcal{X})$$

La divergence est donc égale à la *réduction de vraisemblance* engendrée par le processus d'agrégation : $D(\mathcal{X}) = \Delta \log \mathcal{L}(\mathcal{X})$. En d'autres termes, la divergence mesure la diminution de la probabilité d'obtenir la distribution microscopique à partir de la distribution agrégée.

ANNEXE B

Algorithmes et complexité

Cette annexe présente en détail les implémentations spécialisées de l'algorithme des partitions admissibles optimales (chapitre 6) dans le cas de populations *hiérarchiques* (section B.1) et dans le cas de populations *ordonnées* (section B.2). Pour chacune de ces implémentations, nous explicitons :

1. les structures de données utilisées pour représenter et évaluer l'ensemble des parties admissibles ;
2. le pseudo-code de l'algorithme ;
3. une évaluation empirique de sa complexité temporelle.

Ces algorithmes ont été implémentés en C++ par Damien Dosimont, doctorant INRIA à l'Université Joseph Fourier. Leur code source est disponible à l'adresse suivante : https://github.com/dosimont/lpaggreg. La complexité temporelle a été évaluée en mesurant le temps d'exécution des algorithmes sur des jeux de données aléatoires de taille variable (la preuve théorique de la complexité est donnée dans la section 6.3). Voici les spécifications de la machine utilisée pour l'évaluation :

```
Processeurs Intel® Core™ i7 CPU 920 @ 2.67GHz × 8
Mémoire 7.8 Go DDR3
Disque dur 1 To SATA
Fedora Version 17 (BeefyMiracle) 64 bits
Noyau Linux 3.9.10-100.fc17.x86_64
```

B.1 Algorithme des partitions hiérarchiques optimales

L'*algorithme des partitions hiérarchiques optimales* (6.3.1) calcule une partition $\mathcal{X} \in \mathfrak{P}^{\alpha}_{\mathcal{T}}(\Omega)$ optimisant le compromis de qualité linéaire CQL_{α} (4.3.3) au sein de l'ensemble des partitions admissibles $\mathfrak{P}_{\mathcal{T}}(\Omega)$, défini selon la hiérarchie $\mathcal{T}(\Omega)$ (5.2). On se place dans le cas d'un attribut $v(.)$ *additif* (3.2.2) et dans le cas de l'*hypothèse de redistribution uniforme* (3.2.3). Nous utilisons la *réduction de taille* ΔT comme mesure de complexité (4.3.1) et la *divergence de Kullback-Leibler* D comme mesure de perte d'information (4.3.2).

B.1.1 Structures de données

La hiérarchie $\mathcal{T}(\Omega)$ est implémentée sous la forme d'un arbre de données : les *feuilles* représentent les individus $x \in \Omega$, les *nœuds* représentent les parties admissibles $X \in \mathcal{T}(\Omega)$ et la *racine* représente la population Ω prise dans son intégralité. Chaque nœud $X \in \mathcal{T}(\Omega)$ possède 7 étiquettes, engendrées et manipulées par l'algorithme :

1. *value* est la valeur $v(X)$ de l'attribut analysé. Au niveau des feuilles, ces valeurs constituent les données en entrée de l'algorithme. Au niveau des autres nœuds, elles constituent des variables internes servant au calcul de la divergence $D(X)$.
2. *size* est la taille de la partie X, c'est-à-dire le nombre d'individus qu'elle contient. Cette étiquette sert au calcul de la divergence $D(X)$ et de la réduction de taille $\Delta T(X)$.
3. *microInfo* est la somme des quantités d'information $-v(x)\log_2 v(x)$ associées aux individus $x \in X$. Cette étiquette sert au calcul de la divergence $D(X)$.
4. *sizeReduction* est la réduction de taille $\Delta T(X)$ associée à la partie X. Elle sert au calcul de la partition optimale.
5. *divergence* est la divergence de Kullback-Leibler $D(X)$ associée à la partie X. Elle sert également au calcul de la partition optimale.
6. *optimalCompromise* est le compromis de qualité CQL_{α} de la partition optimale sur X. Il sert au calcul de la partition optimale sur Ω.
7. *optimalCut* est une valeur booléenne indiquant s'il est optimal d'agréger les individus appartenant à la partie X. Cette étiquette donne une *coupe* de l'arbre et représente ainsi une partition hiérarchique optimale.

L'algorithme est constitué de 3 procédures récursives, consistant chacune à parcourir l'arbre en profondeur pour calculer les différentes étiquettes :

1. COMPUTEQUALITY($node$) calcule les étiquettes *value* (sauf si *node* est une feuille), *size*, *microInfo*, *sizeReduction* (non normalisée) et *divergence* (non normalisée) du nœud *node*.
2. NORMALIZEQUALITY($node, maxReduction, maxDivergence$) normalise les étiquettes *sizeReduction* et *divergence* (*cf.* sous-section 4.3.3) à partir des paramètres *maxReduction* et *maxDivergence*. Notons que ces paramètres prennent la valeur de *sizeReduction* et de *divergence* lorsque la procédure est exécutée sur la racine de l'arbre. Ils ne constituent donc pas des données en entrée de l'algorithme.
3. COMPUTEOPTIMALPARTITION($node, coeff$) calcule les étiquettes *optimalCompromise* et *optimalCut* en fonction du coefficient de compromis *coeff* spécifié en entrée (*cf.* sous-section 4.3.3).

Ainsi, si *root* est la racine de l'arbre, l'exécution successive de :

- COMPUTEQUALITY($root$)
- NORMALIZEQUALITY($root$, `null`, `null`)
- COMPUTEOPTIMALPARTITION($root, coeff$)

permet de calculer une partition hiérarchique optimale.

B.1.2 Pseudo-code de l'algorithme

Compute the quality of admissible parts

Require: A tree, representing the hierarchy, with a label *value* on each leaf, representing the value of the attribut of the corresponding individual.

Ensure: Each node of the tree has two labels, *sizeReduction* and *divergence*, respectively representing the non-normalized quality of the corresponding admissible part.

```
procedure COMPUTEQUALITY(node)

    if node has no child then                          ▷ Microscopic level
        node.size ← 1
        if node.value > 0 then
            node.microInfo ← −node.value ∗ log2(node.value)
        else
            node.microInfo ← 0
        end if

        node.sizeReduction ← 0
        node.divergence ← 0

    else                                               ▷ Other levels
        node.value ← 0
        node.size ← 0
        node.microInfo ← 0
        for each child of node do
            COMPUTEQUALITY(child)
            node.value ← node.value + child.value
            node.size ← node.size + child.size
            node.microInfo ← node.microInfo + child.microInfo
        end for

        node.sizeReduction ← size − 1
        if value > 0 then
            node.divergence ← microInfo − value ∗ log2(value/size)
        else
            node.divergence ← 0
        end if
    end if
end procedure
```

Normalize the quality of admissible parts

Require: A tree with labels *sizeReduction* and *divergence* on each node, representing the non-normalized quality of the corresponding part.

Ensure: The labels *sizeReduction* and *divergence* now represent the normalized quality of the corresponding part.

```
procedure NORMALIZEQUALITY(node, maxReduction, maxDivergence)

    if node is the root then
        maxReduction ← node.sizeReduction
        maxDivergence ← node.divergence
    end if

    node.sizeReduction ← node.sizeReduction/maxReduction
    node.divergence ← node.divergence/maxDivergence

    for each child of node do
        NORMALIZEQUALITY(child, maxReduction, maxDivergence)
    end for
end procedure
```

Compute an optimal hierarchical partition

Require: A tree with labels *sizeReduction* and *divergence* on each node, representing the normalized quality of the corresponding part, and a coefficient of compromise *coeff*.

Ensure: Each node of the tree has a Boolean label *optimalCut* representing the optimal partition (*optimalCut* = `true` when the individual of the corresponding part are aggregated and `false` when they are not).

```
procedure COMPUTEOPTIMALPARTITION(node, coeff)

    if node has no child then                          ▷ Microscopic level
        node.optimalCompromise ← 0
        node.optimalCut ← true

    else                                                ▷ Other levels
        macroCompromise ← coeff * node.sizeReduction
                                    −(1 − coeff) * node.divergence

        microCompromise ← 0
        for each child of node do
            COMPUTEOPTIMALPARTITION(child, coeff)
            microCompromise ← microCompromise
                                    +child.optimalCompromise
        end for

        node.optimalCompromise
                        ← max(microCompromise, macroCompromise)
        node.optimalCut ← (microCompromise < macroCompromise)
    end if
end procedure
```

B.1.3 Complexité temporelle

Dans la sous-section 6.3.1, nous avons montré que la complexité temporelle de l'algorithme des partitions hiérarchiques optimales est *linéaire* par rapport à la taille $|\Omega|$ de la population. L'évaluation empirique montre que le coefficient directeur dépend de la structure de l'arbre (*cf.* figure B.1). Nous avons réalisé une série d'exécutions dans le cas d'arbres binaires, ternaires et décimaux. Une régression linéaire donne : $temps = a\,|\Omega| + \epsilon$, avec :

- Arbre binaire : $a = 3,1 \times 10^{-4}$ millisecondes
- Arbre ternaire : $a = 2,0 \times 10^{-4}$ millisecondes
- Arbre décimaux : $a = 1,3 \times 10^{-4}$ millisecondes
- ϵ inférieur à 3 millisecondes dans chacun des trois cas

Les coefficients de détermination R^2 de ces régressions sont supérieurs à $0,998$, indiquant une très faible variance du temps d'exécution vis-à-vis de la relation exprimée. Ainsi, l'algorithme prend *une seconde* pour calculer la partition optimale d'une population contenant respectivement $3,2$ millions, $5,0$ millions et $7,7$ millions d'individus (suivant la structure de l'arbre).

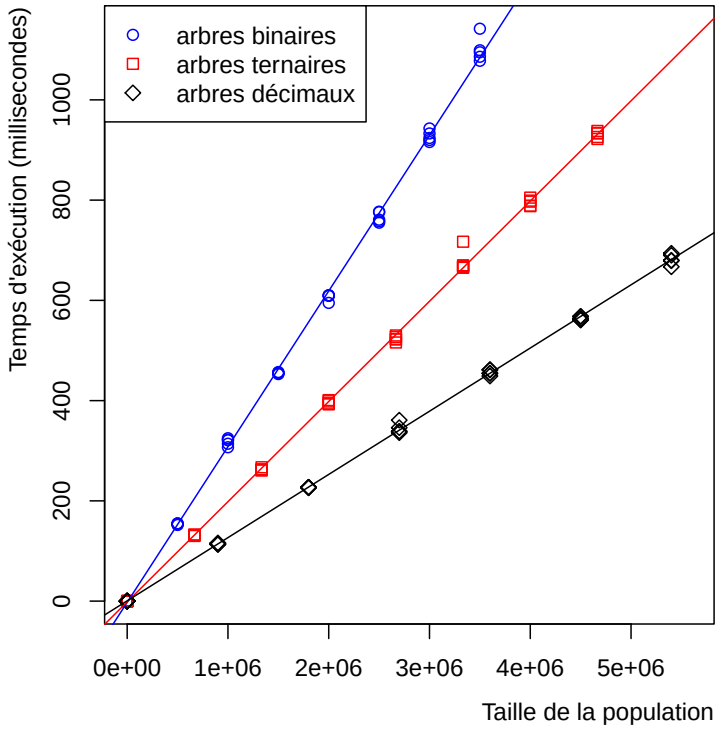


FIGURE B.1 – Temps d'exécution de l'algorithme des partitions hiérarchiques optimales

B.2 Algorithme des partitions ordonnées optimales

L'*algorithme des partitions ordonnées optimales* (6.3.2) calcule une partition $\mathcal{X} \in \mathfrak{P}^{\alpha}_{<}(\Omega)$ optimisant le compromis de qualité linéaire CQL_α (4.3.3) au sein de l'ensemble des partitions admissibles $\mathfrak{P}_{<}(\Omega)$, défini selon l'ordre total $<$ sur Ω (5.3). Comme pour l'algorithme précédent, on se place dans le cas d'un attribut $v(.)$ *additif* (3.2.2) et dans le cas de l'*hypothèse de redistribution uniforme* (3.2.3). Nous utilisons la *réduction de taille* ΔT (4.3.1) et la *divergence de Kullback-Leibler* D (4.3.2).

B.2.1 Structures de données

Soit une population $\Omega = \{x_0, \ldots, x_{n-1}\}$ de taille n, avec $x_i < x_j$ si et seulement si $i < j$. L'attribut $v(.)$ est implémenté par un vecteur *value* contenant n entiers tel que $value[i] = v(x_i)$ pour tout $i \in [\![0, n-1]\!]$. Chaque partie admissible $X_{i,j} = \{x_i, \ldots, x_{i+j}\} \in \mathcal{P}_{<}(\Omega)$ est représentée par un couple (i, j), où $i \in [\![0, n-1]\!]$ est l'index du premier individu de la partie et $j+1 \in [\![1, n-i]\!]$ est la taille de la partie ($j = |X_{i,j}| - 1$). Les mesures de qualité sont donc implémentées par des matrices de taille $n \times n$ où chaque cellule (i, j) représente la qualité de la partie correspondante. L'algorithme manipule 4 matrices :

1. $sumValue[i][j]$ est la valeur $v(X_{i,j})$ de l'attribut analysé. Pour $j = 0$, ces valeurs sont les données en entrée de l'algorithme. Pour $j > 0$, elles constituent des variables internes servant au calcul de la divergence D.
2. $microInfo[i][j]$ est la somme des quantités d'information $-v(x)\log_2 v(x)$ associées aux individus appartenant à la partie $X_{i,j}$. Cette matrice sert au calcul de la divergence D.
3. $sizeReduction[i][j]$ est la réduction de taille $\Delta T(X_{i,j})$ associée à la partie $X_{i,j}$. Cette matrice sert au calcul de la partition optimale.
4. $divergence[i][j]$ est la divergence de Kullback-Leibler $D(X_{i,j})$ associée à la partie $X_{i,j}$. Cette matrice sert également au calcul de la partition optimale.

Le vecteur *optimalCompromise* contient n entiers tels que, pour tout $j \in [\![0, n-1]\!]$, $optimalCompromise[j]$ est le compromis de qualité CQL_α de la partition optimale sur $\{x_0, \ldots, x_j\}$. Ce vecteur sert au calcul de la partition optimale. Celle-ci est représentée par un vecteur *optimalCut* contenant n entiers tels que, pour tout $j \in [\![0, n-1]\!]$, $optimalCut[j]$ est l'index du premier individu de la dernière partie de la partition optimale sur $X_{0,j} = \{x_0, \ldots, x_j\}$. En d'autres termes, $optimalCut[n-1] = k$ indique que

la partie $\{x_k, \ldots, x_{n-1}\}$ est agrégée et, si $k > 0$, $optimalCut[k-1]$ donne à son tour la dernière partie de la partition optimale sur $\{x_0, \ldots, x_{k-1}\}$.

L'algorithme est constitué de 3 procédures consistant chacune à manipuler et à parcourir ces matrices et ces vecteurs :

1. COMPUTEQUALITY($value$) calcule les matrices $sumValue$, $microInfo$, $sizeReduction$ (non normalisée) et $divergence$ (non normalisée).
2. NORMALIZEQUALITY($sizeReduction, divergence$) normalise les matrices $sizeReduction$ et $divergence$ (*cf.* sous-section 4.3.3).
3. COMPUTEOPTIMALPARTITION($sizeReduction, divergence, coeff$) calcule les vecteurs $optimalCompromise$ et $optimalCut$ en fonction du coefficient de compromis $coeff$ spécifié en entrée (*cf.* sous-section 4.3.3).

Ainsi, l'exécution successive de :

- COMPUTEQUALITY($value$)
- NORMALIZEQUALITY($sizeReduction, divergence$)
- COMPUTEOPTIMALPARTITION($sizeReduction, divergence, coeff$)

permet de calculer une partition ordonnée optimale.

B.2.2 Pseudo-code de l'algorithme

Compute the quality of admissible parts

Input : A vector representing the *value* of the attribut for each individual.
Output : Two matrices representing the non-normalized *sizeReduction* and *divergence* of each admissible part.

```
function COMPUTEQUALITY(value)
    n ← SIZEOF(value)
    sumValue ← NEWMATRIX(n, n)
    microInfo ← NEWMATRIX(n, n)
    sizeReduction ← NEWMATRIX(n, n)
    divergence ← NEWMATRIX(n, n)

    for i ∈ [[0, n − 1]] do                                  ▷ Microscopic level
        sumValue[i][0] ← value[i]
        if value[i] > 0 then
            microInfo[i][0] ← −value[i] ∗ log_2(value[i])
        else
            microInfo[i][0] ← 0
        end if
        sizeReduction[i][0] ← 0
        divergence[i][0] ← 0
    end for

    for j ∈ [[1, n − 1]] do                                  ▷ Other levels
        for i ∈ [[0, n − 1 − j]] do
            sumValue[i][j] ← sumValue[i][j − 1] + sumValue[i + j][0]
            microInfo[i][j] ← microInfo[i][j − 1] + microInfo[i + j][0]

            sizeReduction[i][j] ← j
            if sumValue[i][j] > 0 then
                divergence[i][j] ← microInfo[i][j]
                            −sumValue[i][j] ∗ log_2(sumValue[i][j]/(j + 1))
            else
                divergence[i][j] ← 0
            end if
        end for
    end for
    return (sizeReduction, divergence)
end function
```

Normalize the quality of admissible parts

Input : Two matrices representing the non-normalized *sizeReduction* and *divergence* of admissible parts.

Output : Two matrices representing the normalized *sizeReduction* and *divergence* of admissible parts.

```
function NORMALIZEQUALITY(sizeReduction, divergence)
    n ← SIZEOF(sizeReduction)

    maxReduction ← sizeReduction[0][n − 1]
    maxDivergence ← divergence[0][n − 1]

    for j ∈ [[0, n − 1]] do
        for i ∈ [[0, n − 1 − j]] do
            sizeReduction[i][j] ← sizeReduction[i][j]/maxReduction
            divergence[i][j] ← divergence[i][j]/maxDivergence
        end for
    end for

    return (sizeReduction, divergence)
end function
```

Compute an optimal ordered partition

Input : Two matrices representing the normalized *sizeReduction* and *divergence* of admissible parts and a coefficient of compromise *coeff*.

Output : A vector *optimalPartition* representing the optimal partition.

```
function COMPUTEOPTIMALPARTITION(sizeReduction, divergence, coeff)
    n ← SIZEOF(sizeReduction)

    optimalCompromise ← NEWVECTOR(n)
    optimalPartition ← NEWVECTOR(n)

    optimalCompromise[0] ← 0                          ▷ Initialize first subset
    optimalPartition[0] ← 0

    for j ∈ [[1, n − 1]] do                           ▷ Compute other subsets

        currentCut ← 0                                ▷ Case with no cut
        currentCompromise ← coeff ∗ sizeReduction[0][j]
                                        −(1 − coeff) ∗ divergence[0][j]

        for cut ∈ [[1, j]] do                         ▷ Compare with other cases
            compromise ← optimalCompromise[cut − 1]
                                    +coeff ∗ sizeReduction[cut][j − cut]
                                −(1 − coeff) ∗ divergence[cut][j − cut]

            if compromise > currentCompromise then
                currentCompromise ← compromise
                currentCut ← cut
            end if
        end for

        optimalCompromise[j] ← currentCompromise
        optimalPartition[j] ← currentCut
    end for

    return optimalPartition
end function
```

B.2.3 Complexité temporelle

Dans la section 6.3, nous avons montré que la complexité temporelle de l'algorithme des partitions ordonnées optimales est *quadratique* par rapport à la taille $|\Omega|$ de la population. La mesure du temps d'exécution (*cf.* figure B.2) montre que nous avons la relation suivante :

$$temps = a\,|\Omega|^2 + \epsilon$$

avec $a = 1,5 \times 10^{-4}$ millisecondes et ϵ inférieur à 1 milliseconde. Le coefficient de détermination R^2 est supérieur à $0,999$ indiquant une quasi-absence de variance du temps d'exécution vis-à-vis de cette formule. Ainsi, l'algorithme prend environ *une seconde* pour calculer la partition optimale d'une population ordonnée de 2500 individus.

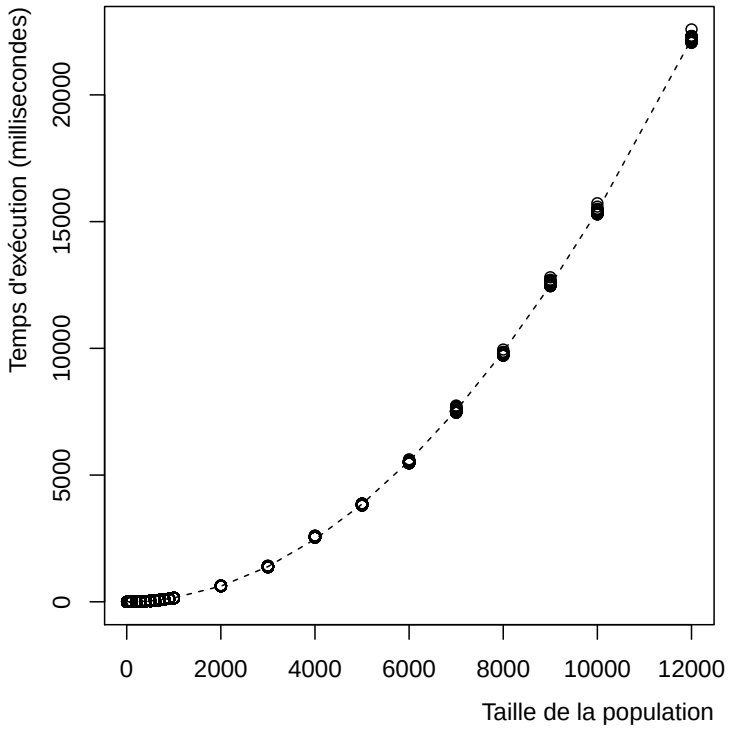


FIGURE B.2 – Temps d'exécution de l'algorithme des partitions ordonnées optimales

ANNEXE C

Hiérarchie WUTS

Cette annexe présente la hiérarchie WUTS utilisée en géographie pour construire des statistiques multi-niveaux concernant les différentes parties du globe [GD07]. Cette hiérarchie partitionne l'espace géographique en régions imbriquées, du niveau national au niveau continental. Elle conserve les propriétés de voisinage (les pays d'un agrégats forment un ensemble connexe) et également des propriétés non-spatiales : relations politiques, économiques, culturelles et historiques entretenues entre les différentes unités territoriales. Ainsi, les agrégats sont cohérents sur le plan *syntaxique* (organisation géographique des unités territoriales) et sur le plan *sémantique* (signification et interprétation des agrégats par les sciences sociales). Cette hiérarchie est utilisée dans le chapitre 8 pour agréger l'espace géographique. Les phénomènes macroscopiques ainsi observés peuvent être expliqués à partir des propriétés syntaxique et sémantique du système international.

La hiérarchie WUTS comporte 5 niveaux :

- le niveau des pays (WUTS5) ;
- un niveau contenant 36 micro-régions (WUTS4, *cf.* figure C.1) ;
- un niveau contenant 17 méso-régions (WUTS3, *cf.* figure C.2) ;
- un niveau contenant 7 macro-régions (WUTS2, *cf.* figure C.3) ;
- un niveau contenant seulement 3 régions (WUTS1, *cf.* figure C.4).

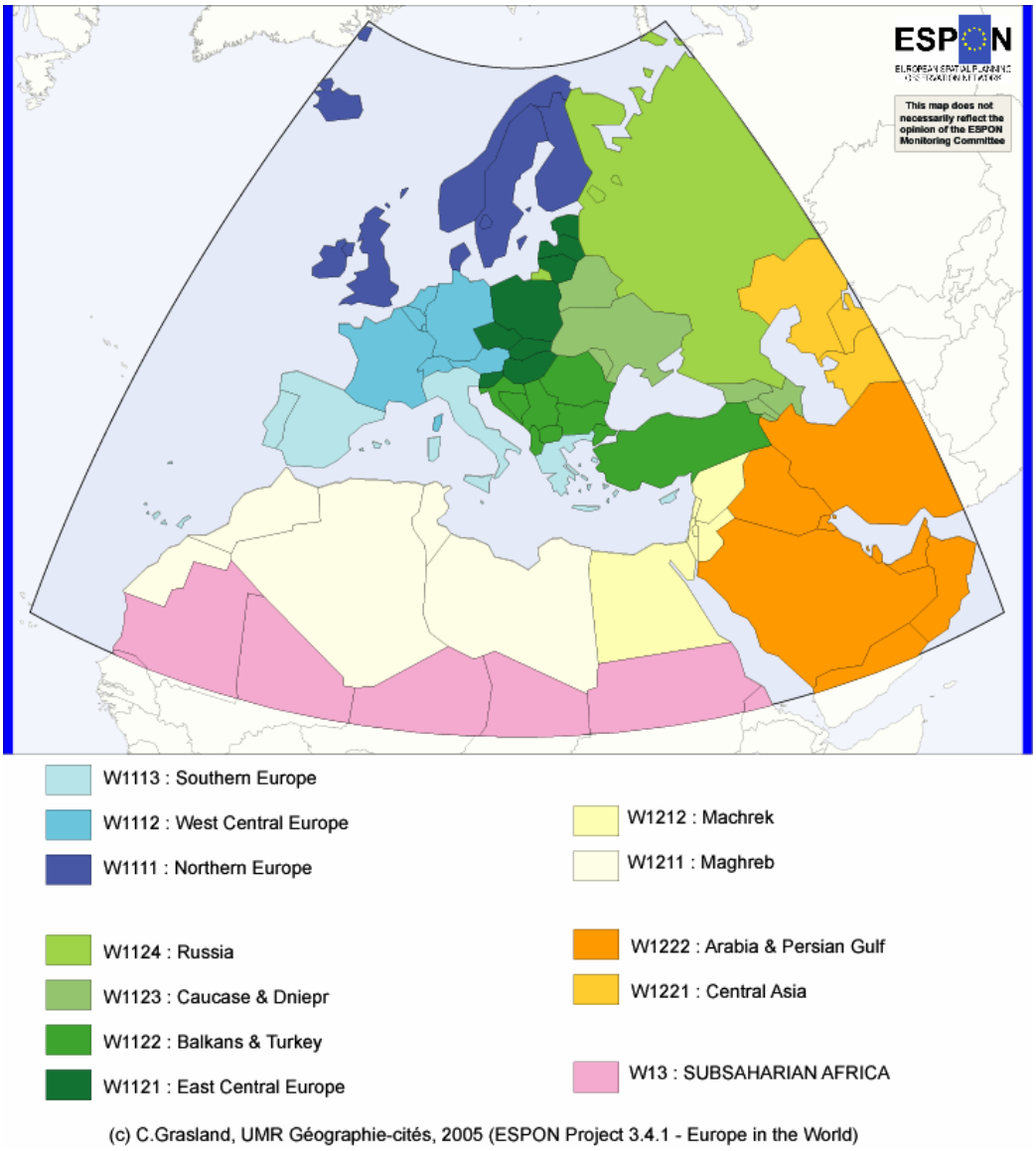


FIGURE C.1 – Deuxième niveau de la hiérarchie WUTS composé de 36 micro-régions : ne sont représentées ici que les 12 micro-régions de la zone européenne (*cf.* WUTS4, [GD07] page 100)

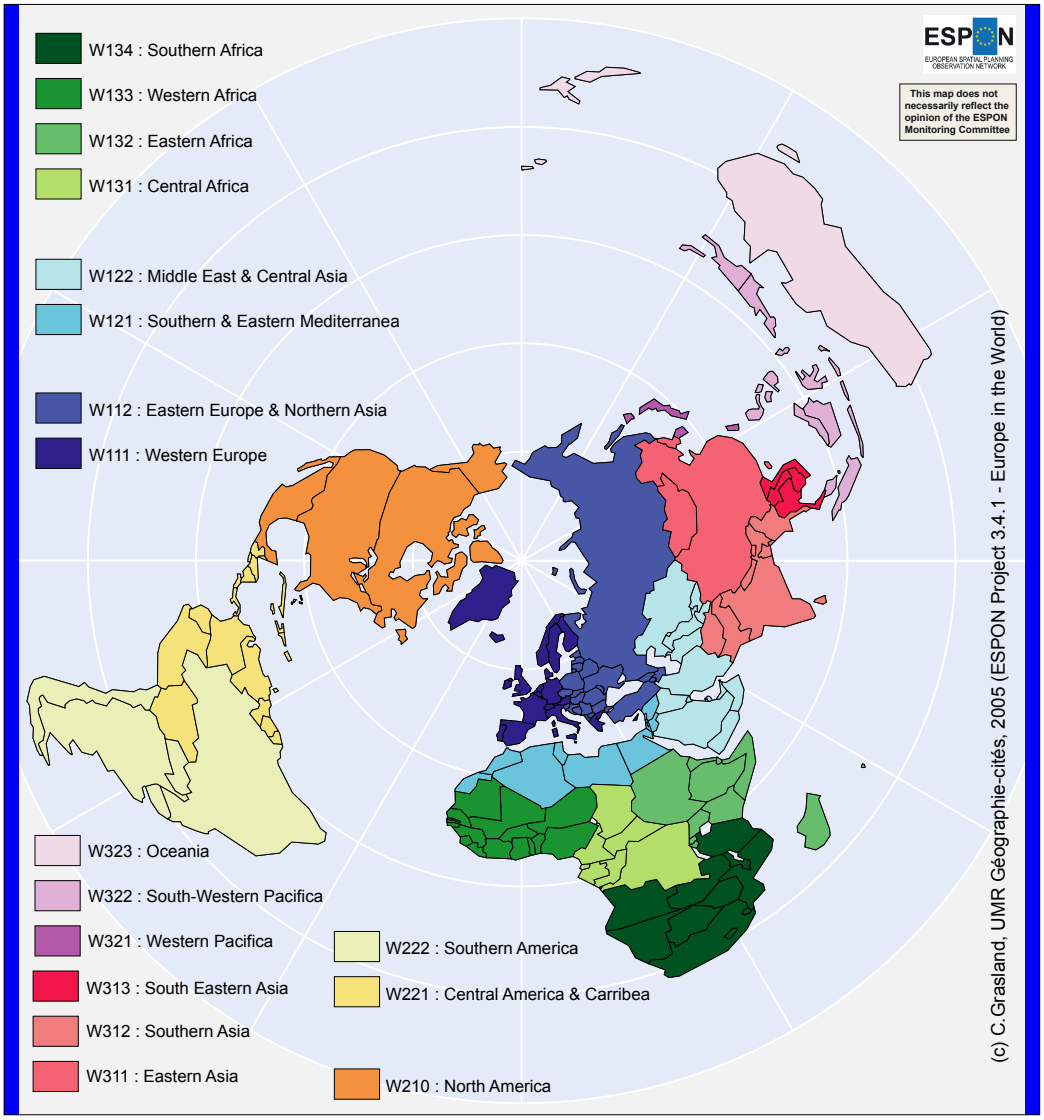


FIGURE C.2 – Troisième niveau de la hiérarchie WUTS composé de 17 mesorégions (*cf.* WUTS3, [GD07] page 98)

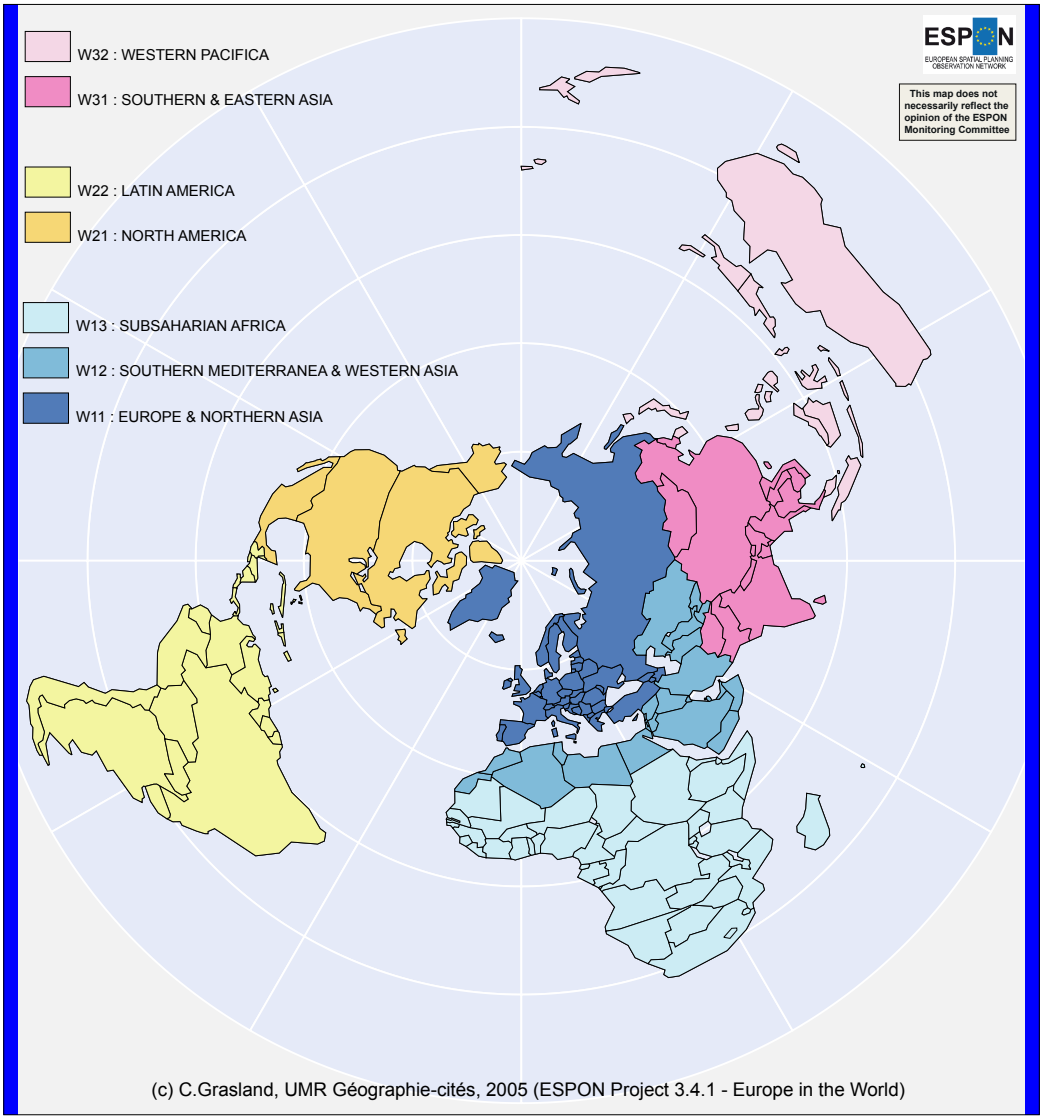


FIGURE C.3 – Quatrième niveau de la hiérarchie WUTS composé de 7 macro-régions (*cf.* WUTS2, [GD07] page 96)

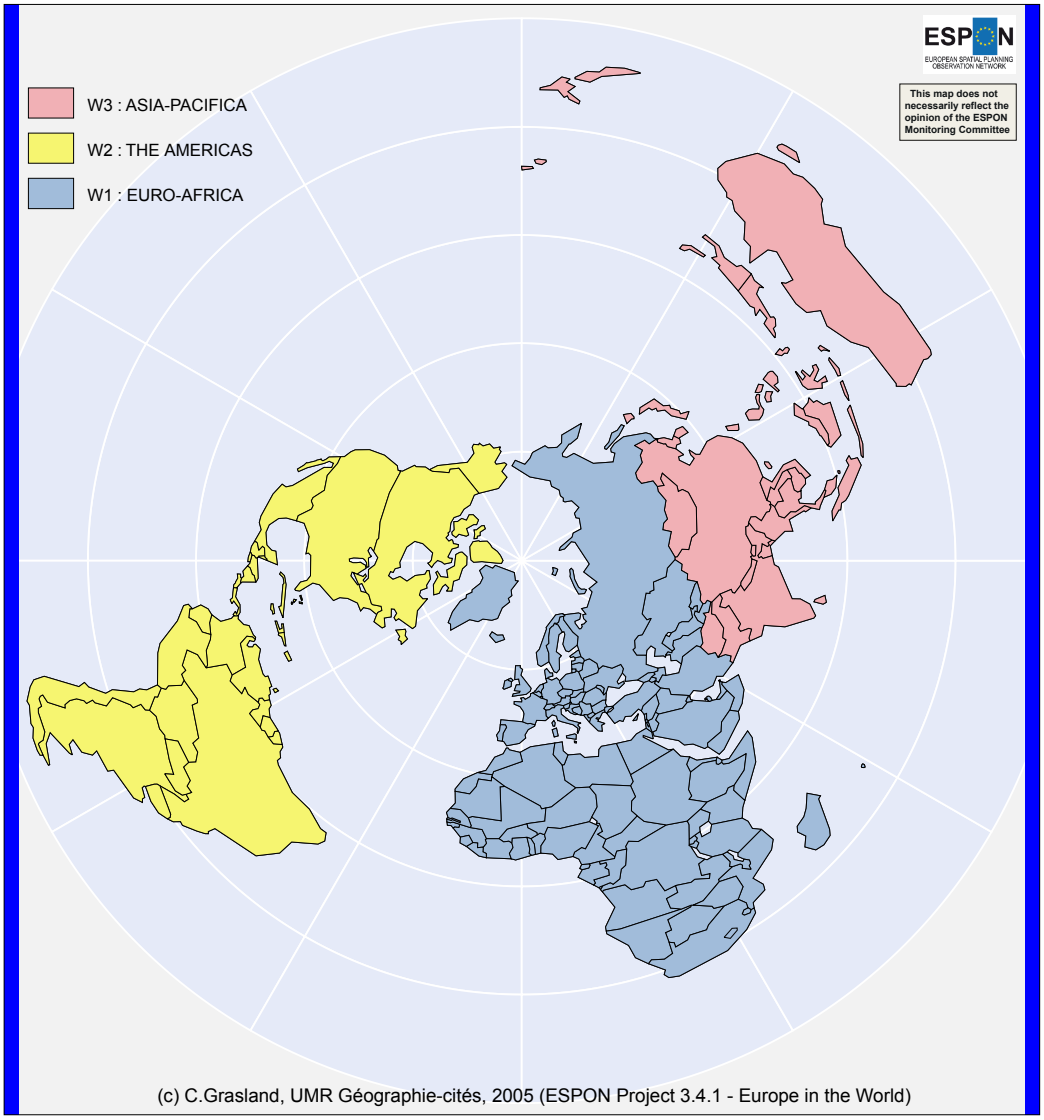


FIGURE C.4 – Cinquième niveau de la hiérarchie WUTS composé de seulement 3 régions (*cf.* WUTS1, [GD07] page 94)

Analyse macroscopique des grands systèmes

Émergence épistémique et agrégation spatio-temporelle

par **Robin LAMARCHE-PERRIN**

L'analyse des systèmes de grande taille est confrontée à des difficultés d'ordre *syntaxique* et *sémantique* : comment observer un million d'entités distribuées et asynchrones ? Comment interpréter le désordre résultant de l'observation microscopique de ces entités ? Comment produire et manipuler des abstractions pertinentes pour l'analyse macroscopique des systèmes ? Face à l'échec de l'approche analytique, le concept d'*émergence épistémique* – relatif à la nature de la connaissance – nous permet de définir une stratégie d'analyse alternative, motivée par le constat suivant : l'activité scientifique repose sur des *processus d'abstraction* fournissant des éléments de description macroscopique pour aborder la complexité des systèmes.

Cette thèse s'intéresse plus particulièrement à la production d'abstractions spatiales et temporelles par *agrégation de données*. Afin d'engendrer des représentations exploitables lors du passage à l'échelle, il apparaît nécessaire de contrôler deux aspects essentiels du processus d'abstraction. Premièrement, la complexité et le contenu informationnel des représentations macroscopiques doivent être conjointement optimisés afin de préserver les détails pertinents pour l'observateur, tout en minimisant le coût de l'analyse. Nous proposons des mesures de qualité (*critères internes*) permettant d'évaluer, de comparer et de sélectionner les représentations en fonction du contexte et des objectifs de l'analyse. Deuxièmement, afin de conserver leur pouvoir explicatif, les abstractions engendrées doivent être cohérentes avec les connaissances mobilisées par l'observateur lors de l'analyse. Nous proposons d'utiliser les propriétés organisationnelles, structurelles et topologiques du système (*critères externes*) pour contraindre le processus d'agrégation et pour engendrer des représentations viables sur les plans syntaxique et sémantique. Par conséquent, l'automatisation du processus d'agrégation nécessite de résoudre un problème d'optimisation sous contraintes. Nous proposons dans cette thèse un algorithme de résolution *générique*, s'adaptant aux critères formulés par l'observateur. De plus, nous montrons que la complexité de ce problème d'optimisation dépend directement de ces critères.

L'approche macroscopique défendue dans cette thèse est évaluée sur deux classes de systèmes. Premièrement, le processus d'agrégation est appliqué à la visualisation d'applications parallèles de grande taille pour l'analyse de performance. Il permet de détecter les anomalies présentes à plusieurs niveaux de granularité dans les traces d'exécution et d'expliquer ces anomalies à partir des propriétés syntaxiques du système. Deuxièmement, le processus est appliqué à l'agrégation de données médiatiques pour l'analyse des relations internationales. L'agrégation géographique et temporelle de l'attention médiatique permet de définir des évènements macroscopiques pertinents sur le plan sémantique pour l'analyse du système international. Pour autant, nous pensons que l'approche et les outils présentés dans cette thèse peuvent être généralisés à de nombreux autres domaines d'application.