



**UNIVERSITÉ
DE LORRAINE**

**BIBLIOTHÈQUES
UNIVERSITAIRES**

AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact bibliothèque : ddoc-theses-contact@univ-lorraine.fr
(Cette adresse ne permet pas de contacter les auteurs)

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

Influence Indépendante et Explicabilité de l'Exploration et de l'Exploitation dans les Métaheuristiques

THÈSE

présentée et soutenue publiquement le 14 Décembre 2022

pour l'obtention du

Doctorat de l'Université de Lorraine
(mention informatique)

par

Alexandre Bettinger

Composition du jury

Président : Vincent Chevrier

Rapporteurs : Laetitia Jourdan (PR, U.Lille)
Laurent Vercouter (PR, INSA Rouen)

Examineurs : Tassadit Amghar (MCF HDR, U.Angers)
Vincent Chevrier (PR, U.Lorraine)

Encadrants : Armelle Brun (PR, U.Lorraine)
Anne Boyer (PR, U.Lorraine)

Mis en page avec la classe thesul.

Remerciements

Ce travail est réalisé dans le cadre de la convention PEACE avec le soutien de la Direction du Numérique pour l'Éducation ainsi qu'avec le Rectorat français de l'Académie de Nancy-Metz et l'Université de Lorraine dans le cadre de l'incubateur numérique du Ministère de l'Éducation Nationale et de la Jeunesse.



Table des matières

Chapitre 1

Introduction générale

1.1	Positionnement et problématiques de recherche	7
1.1.1	Problématiques liées à l'influence de l'exploration et de l'exploitation	7
1.1.2	Problématiques liées à la réduction de données	8
1.1.3	Problématiques liées à l'explicabilité de l'E&E	9
1.1.4	Problématiques et contexte éducatif	9
1.2	Résumé des contributions	9
1.2.1	Influence de l'E&E	10
1.2.2	Réduction des données	11
1.2.3	Explicabilité de l'E&E	12
1.3	Plan du document	13

Chapitre 2

État de l'art

2.1	Métaheuristiques	15
2.1.1	Optimisation	16
2.1.2	Heuristiques	16
2.1.3	Métaheuristiques	16
2.1.4	MH et recommandation	17
2.1.5	MH hybrides et hyperheuristiques	18
2.1.6	Conclusion et ouvertures	18
2.2	Influence et Explicabilité de l'E&E	20
2.2.1	Contextes Organisationnels	21
2.2.2	Influence de l'E&E dans les MH	21
2.2.3	Influence de l'E&E dans les EA	25
2.2.4	Influence de l'E&E en Psycho-sociologie	31
2.2.5	Conclusion Générale	36

2.3	Réduction des données	36
2.3.1	Introduction	36
2.3.2	Méthodes de réduction des données	36
2.3.3	Focus sur l'échantillonnage	38
2.3.4	Directions de recherche	40
2.3.5	Conclusion	41
2.4	Manuels scolaires numériques	41
2.4.1	Introduction	41
2.4.2	Synthèse de l'état de l'art concernant les manuels scolaires numériques et concepts associés	42
2.4.3	Problèmes et ouvertures	43

<p>Chapitre 3 Contributions</p>

3.1	Influence de l'E&E	46
3.1.1	Processus d'influence de l'E&E indépendant de l'algorithme influencé	48
3.1.2	Nouveaux indicateurs pour l'influence et l'explicabilité de l'E&E	51
3.1.3	Opérateurs de variation pour l'influence de l'E&E	56
3.1.4	Conclusion	60
3.2	Réduction des données	61
3.2.1	Macro-processus de réduction	62
3.2.2	Fonctions de réduction	63
3.2.3	Réductions en séquences	71
3.2.4	Conclusion	73
3.3	Explicabilité de l'E&E	73
3.3.1	Indicateurs Hyper-Volumiques	75
3.3.2	Indicateurs Transitionnels	79
3.3.3	Classification Incrémentielle	82
3.3.4	Conclusion	82
3.4	Conclusion à l'ensemble des contributions	83

<p>Chapitre 4 Expérimentations</p>
--

4.1	Informations générales concernant les expérimentations	85
4.2	Macro-processus	88
4.3	Critères d'évaluation	88
4.3.1	Critères de performance	88

4.3.2	Critères de qualité	89
4.3.3	Critères temporels	89
4.4	Influence de l'E&E	90
4.4.1	Variations d'influence	90
4.4.2	Dataset MovieLens - Influence GA	90
4.4.3	Dataset MovieLens - Influence RA	93
4.4.4	Dataset Riiid - Influence GA	94
4.4.5	Dataset Riiid - Influence RA	95
4.4.6	Conclusion	97
4.5	Opérateurs de variation pour l'influence de l'E&E	97
4.5.1	Introduction	97
4.5.2	Résultats	97
4.5.3	Conclusion	98
4.6	Réduction des données	99
4.6.1	Dataset MovieLens - Réduction et GA	99
4.6.2	Dataset MovieLens - Réduction et RA	101
4.6.3	Dataset Riiid - Réduction et GA	102
4.6.4	Dataset Riiid - Réduction et RA	104
4.6.5	Conclusion	105
4.7	Réduction des données et influence de l'E&E	106
4.7.1	Dataset MovieLens - Réduction, Influence et GA	106
4.7.2	Dataset MovieLens - Réduction, Influence et RA	108
4.7.3	Dataset Riiid - Réduction, Influence et GA	109
4.7.4	Dataset Riiid - Réduction, Influence et RA	111
4.7.5	Conclusion	114
4.8	Réductions combinées	114
4.8.1	Dataset MovieLens - Réductions combinées - Référence GA	114
4.8.2	Dataset MovieLens - Réductions combinées - Référence RA	115
4.8.3	Conclusion	116
4.9	Explicabilité de l'E&E	117
4.9.1	Hyper-volumes	117
4.9.2	Transitions entre E&E	126
4.9.3	Classification incrémentielle	134
4.9.4	Conclusion	137

Chapitre 5

Discussions et travaux futurs

5.1	Synthèse	139
5.1.1	Rappel des questions de recherche	139
5.1.2	Rappel des contributions	140
5.1.3	Synthèse des conclusions des expérimentations	141
5.2	Discussion	142
5.2.1	Généralités	142
5.2.2	Influence de l'E&E	142
5.2.3	Réduction de données	143
5.2.4	Explicabilité de l'E&E	144
5.3	Travaux futurs	144
5.3.1	Influence de l'E&E	144
5.3.2	Réduction de données	145
5.3.3	Explicabilité de l'E&E	145
5.4	Conclusion	145

Annexes

Annexe A

Requêtes d'extraction MovieLens et Riid

Chapitre 1

Introduction générale

Une part importante des informations présentes dans cette section provient directement du site officiel du ministère français de l'éducation nationale et de la jeunesse [278].

L'École contribue au projet d'une société de l'information et de la communication pour tous. Elle forme les élèves à maîtriser ces outils numériques et le futur citoyen à vivre dans une société dont l'environnement technologique évolue constamment.

Une généralisation du numérique à l'École est permise par les espaces numériques de travail (ENT). Un ENT est un ensemble intégré de services numériques choisis et mis à disposition de la communauté éducative d'une ou plusieurs écoles ou d'un ou plusieurs établissements scolaires dans un cadre de confiance. Il constitue un point d'entrée unifié permettant à l'utilisateur d'accéder, selon son profil et son niveau d'habilitation, à des services et contenus numériques. Il offre un lieu d'échange et de collaboration entre ses usagers, et avec d'autres communautés en relation avec l'école ou l'établissement.

Depuis 2016, dans le cadre de l'action "Innovation Numérique et Excellence Éducative" (INEE) du programme d'investissement d'avenir (PIA2), les différents appels à projets pour lesquels les collectivités se sont fortement engagées ont permis d'équiper les écoles et les collèges.

Le ministère mène une politique de soutien au développement et à la diffusion de ressources numériques pédagogiques par l'intermédiaire de différents leviers comme le portail Éduthèque, les banques de ressources numériques éducatives, la plateforme ETINCEL, des ressources pour l'apprentissage de la programmation et de la pensée algorithmique, des ressources Édu-Up, l'animation de réseaux d'experts disciplinaires, la mise en place des travaux académiques mutualisés (TraAM).

Dans ce contexte global qui ne concerne pas seulement la France, les manuels scolaires, qui constituent un outil de base de l'apprentissage scolaire, longtemps exploités sous format papier, le sont aujourd'hui de plus en plus au format numérique car ils sont accessibles sur des plateformes d'apprentissage. Ce nouveau format permet de nombreuses interactions telles que la traditionnelle consultation et la réalisation d'exercices, mais également la modification ou l'ajout de ressources pédagogiques. Le format d'une ressource pédagogique peut varier (texte, exercice, vidéo, image, bande sonore). Grâce à ces plateformes, il est possible de disposer d'un ensemble de données de traces d'interactions des acteurs avec les ressources pédagogiques. Ces données n'étaient évidemment pas collectables avec des manuels scolaires au format papier, ce qui ouvre de nouvelles possibilités en matière d'analyse de la consommation des ressources pédagogiques mais aussi en matière de recommandation dans le cadre de l'analyse de l'apprentissage. La recommandation est le fait de filtrer de façon personnalisée des informations afin de cibler des éléments (*items*, ressources) susceptibles d'intéresser un ou plusieurs utilisateurs. L'analyse de

l'apprentissage ou *learning analytics* est la discipline consacrée à la mesure, la collecte et l'analyse basées sur des données des apprenants ou traces d'apprentissage dans le but de comprendre et d'optimiser l'apprentissage et le contexte [124]. Des techniques d'intelligence artificielle, et notamment la fouille de données, sont traditionnellement utilisées pour exploiter automatiquement les données collectées [212]. En *learning analytics*, cela permet notamment d'effectuer de la modélisation de l'apprentissage [91], de la modélisation dynamique des connaissances [5] [113] ainsi que de la personnalisation des apprentissages [347] [431].

Les manuels scolaires numériques peuvent fortement modifier les habitudes et les pratiques d'apprentissage des apprenants. Les apprenants interagissent toujours avec des ressources éducatives (leçons, exercices, évaluations) mais ces ressources sont numériques et l'automatisation de leur recommandation par les outils numériques modernes est une opportunité pour la personnalisation des apprentissages. L'objectif visé par ces outils est de maximiser les chances de progresser de chaque apprenant.

De plus, d'autres acteurs évoluent aux côtés des apprenants (les auteurs et éditeurs de ressources, les enseignants et même les institutions), ils ont des objectifs propres et partagés qui peuvent aussi entrer en considération dans la démarche de recommandation. Il est donc aussi possible de fournir des recommandations aux différents acteurs pour, par exemple, les assister dans leurs tâches professionnelles et mettre à leur disposition des informations sur l'activité des apprenants ou sur l'usage d'une ressource donnée ou d'un type de ressource. Cependant, les habitudes, comportements et surtout les objectifs varient d'un acteur à l'autre, ils ont donc besoin de recommandations personnalisées. Par exemple, nous pouvons proposer à un apprenant, des ressources à consulter pour faciliter ses démarches d'apprentissage et maximiser ses chances de réussite aux examens, en parallèle proposer à un enseignant des ressources facilitant l'organisation de ses cours ou des suggestions lui permettant de faire de la différenciation en fonction du profil des apprenants et proposer à un auteur des éléments lui permettant d'améliorer le contenu des ressources qu'il crée. Ces exemples ne sont qu'un aperçu de l'ensemble des possibilités offertes par la recommandation, ce qui vient aussi mettre en lumière le fait que les recommandations doivent être en cohérence avec l'ensemble des objectifs des utilisateurs. En effet, certains objectifs peuvent entrer en contradiction avec d'autres objectifs ou peuvent empêcher la réalisation d'autres objectifs.

Traditionnellement, la recommandation de ressources éducatives aux apprenants est assurée par les enseignants qui utilisent, entre autres, leurs compétences dans leurs domaines respectifs ainsi que leurs connaissances des niveaux d'apprentissage des apprenants et leurs capacités de compréhension, afin de trouver des compromis les menant à utiliser et à faire utiliser des ressources éducatives spécifiques. Il est entendu que cette démarche est longue et biaisée car elle repose sur des appréciations humaines incomplètes. L'arrivée de l'informatique et du traitement des données a permis la mise en place de processus d'automatisation de cette démarche, qui sont eux aussi biaisés. Ces processus touchent au domaine de la transformation des données, de l'optimisation, de l'apprentissage automatique, des mécanismes décisionnels ainsi que de la recommandation. Ces domaines sont très généraux, ils dépassent ainsi de loin le cadre applicatif des manuels scolaires numériques.

Les problématiques traitées dans ce manuscrit s'inscrivent dans le contexte général de la recommandation personnalisée.

1.1 Positionnement et problématiques de recherche

Rappelons que la recommandation est le fait de filtrer des informations afin de cibler des éléments (*items*, ressources) susceptibles d'intéresser un ou plusieurs utilisateurs. Dans le cadre des manuels scolaires numériques, les *items* sont des ressources éducatives (leçon, exercice, chapitre, vidéo et autres). Cette tâche peut être vue comme le traitement d'un vaste espace de recherche qui représente l'ensemble des recommandations possibles [108] [148] [272] [80]. Selon le contexte de la recommandation, une recommandation peut prendre différentes formes telles que des *items*, des ensembles d'*items* (*itemsets*) ou des séquences d'*items* [328]. Notons que les environnements de recommandation peuvent être soumis à de nombreux aléas et contraintes de recommandation.

Dans ce travail de thèse, nous nous intéressons à la recommandation d'*itemsets* (également appelés vecteurs ou solutions) [215] [260].

Nous présentons ci-dessous les problématiques de cette thèse.

1.1.1 Problématiques liées à l'influence de l'exploration et de l'exploitation

Dans de nombreux domaines d'intelligence artificielle, l'espace de recherche est souvent vaste car beaucoup d'*items* peuvent être recommandés et la littérature propose d'aborder ce problème comme un problème d'exploration et d'exploitation (E&E) de l'espace de recherche [94] [275] [414] [80]. Cet espace est encore plus grand lorsque l'on s'intéresse à la recommandation d'*itemsets*. Explorer signifie s'assurer que de nombreuses régions de l'espace de recherche sont examinées [94], afin de ne pas se retrouver piégé dans un optimum local. Exploiter signifie que si une région prometteuse de l'espace de recherche est trouvée, elle est examinée plus attentivement [94] pour voir si elle contient de bons optima locaux. Dans le cadre de la recommandation, un optimum local est un *itemset* (ou vecteur) considéré comme optimal par rapport à la zone de l'espace de recherche dans laquelle il se situe, sachant qu'il peut exister de meilleurs *itemsets* (optima locaux) dans d'autres zones de l'espace de recherche.

L'analyse de la littérature liée à la recommandation nous a permis de converger vers des contributions dans le domaine de l'influence de l'exploration et de l'exploitation (E&E) [42]. Nous avons choisi ce domaine car il présente des défis [94] [275] [414]. Dans ce travail, nous choisissons d'aborder le problème de la recommandation d'*itemsets* par l'optimisation et l'influence de l'E&E afin de trouver de meilleures recommandations au regard de critères de performance et de qualité.

La littérature souligne qu'entre autres domaines, le domaine des métaheuristiques (MH) aborde la problématique de la proposition de recommandations dans de vastes espaces de recherche [94]. Notons que dans la littérature, l'utilisation des MH n'est pas systématiquement référée à la recommandation mais la plupart du temps à l'optimisation qui est, à bien des égards, très proche de la recommandation. Nos travaux sur l'influence de l'E&E visent à contribuer à l'influence et à l'explicabilité de l'E&E dans le cadre de la recommandation par MH, afin d'améliorer les recommandations (au regard de la performance et de la qualité) ainsi que les processus d'influence de l'E&E et afin de renforcer la confiance envers les algorithmes.

Des études ont été menées pour influencer la dynamique de l'E&E afin d'assurer de meilleurs comportements de recherche et ainsi trouver de meilleures recommandations concernant des critères d'évaluation prédéfinis [94]. Influencer l'E&E implique de savoir quand, pourquoi et comment influencer [428]. En effet, l'E&E doivent être contrôlées régulièrement au travers d'une mesure et les opérations réalisées pour exercer l'influence doivent être justifiées. Notons ici que le contrôle de l'E&E et l'influence de l'E&E sont des mécanismes différents mais font tous deux partie du processus d'influence de l'E&E. Ces éléments contribuent à expliquer les comportements d'E&E afin de comprendre ce qui a conduit aux recommandations proposées [383].

Dans un cadre de recommandation, un utilisateur a des préférences explicites ou implicites sur les recommandations attendues, qui peuvent être représentées par des contraintes [371]. Certaines contraintes peuvent être gérées en amont lors de la sélection des données (extraction) ou en aval lors de la sélection des recommandations finales, comme par exemple pour les préférences du type "film d'humour" en recommandation de films. D'autres contraintes sont associées à des attributs d'*items*, par exemple, dans le cas de la recommandation d'un ensemble d'activités pour un week-end, le prix du logement ne doit pas être supérieur à 800€ et de préférence pas supérieur à 650€, le montant total pour l'ensemble du séjour devant être inférieur à 1500€. Afin d'illustrer cela en contexte de recommandation de ressources (éducatives ou autres), un *itemset* est constitué de ressources, une contrainte possible est le volume horaire maximum d'un *itemset* (volume temporel nécessaire pour consulter toutes les ressources de l'*itemset*), des traces numériques associées aux ressources servent à calculer les attributs des ressources (durée, popularité, difficulté et autres) et l'influence de l'E&E permet d'orienter la création des *itemsets* en fonction d'indicateurs tels que la complétion des contraintes ou encore le pouvoir d'E&E [432] qui utilisent les attributs des *items* pour décider si les *itemsets* générées sont suffisamment diversifiées par exemple.

Autrement dit, il faut être en mesure d'influencer la dynamique de l'E&E au sein d'un important ensemble d'*itemsets* afin de garantir un bon comportement de recherche et ainsi trouver de bons *itemsets* au regard des critères d'évaluation. De notre point de vue, les processus d'influence proposés dans la littérature ne garantissent pas une expression très impactante de l'influence.

Notre problématique est donc la suivante : Comment expliquer, mesurer, contrôler et influencer l'E&E ?

1.1.2 Problématiques liées à la réduction de données

La tâche de recommandation repose sur des données qui sont traitées par les algorithmes de recommandation. Les données portent sur les *items* et/ou les utilisateurs. Le jeu de données utilisé dans le cadre d'un processus de recommandation doit le plus souvent être transformé afin de s'assurer de sa qualité (redondances, bruits, incohérences, incomplétudes), de réduire sa taille et sa complexité [329]. L'objectif de cette transformation est d'améliorer les performances des traitements ultérieurs et la qualité des solutions/recommandations trouvées. A ces objectifs communs nous ajoutons notre objectif d'influencer indirectement le comportement d'E&E en proposant un espace de recherche transformé. La plupart de ces prétraitements sont des méthodes de réduction des données, nous avons réalisé un état de l'art sur les méthodes existantes en faisant un focus sur l'échantillonnage. Nous avons choisi l'échantillonnage car ce domaine présente des défis [254] [372] [416] [7] [285] [329]. Nous avons choisi de nous intéresser aux méthodes de réduction statistiquement conscientes car nous souhaitons conserver un maximum de caractéristiques des données, pour ne pas les dénaturer, tout en incluant des biais afin de faciliter l'atteinte de solutions difficilement accessibles. Nous avons donc besoin d'informations statistiques sur les données pour orienter le processus de réduction.

La problématique que nous abordons dans ce travail est : Quels processus d'échantillonnage proposer afin de réduire un jeu de données en conservant un maximum de ses caractéristiques dans le but de servir une démarche d'E&E pour la recommandation ?

A partir des précédentes problématiques énoncées, nous proposons la problématique suivante : L'utilisation conjointe de processus de réduction des données statistiquement conscients et d'un processus pour influencer l'E&E, permet-elle d'avoir un impact significatif sur les critères d'évaluation de la performance, de la qualité et du comportement d'E&E d'un algorithme de recherche ? Répondre à cette question per-

mettra de proposer des couples performants de fonctions de réduction et d'influence en fonction des cas d'exécution.

Enfin, nous abordons aussi un problème de séquençage de fonctions de réduction de données en tant que processus seul permettant l'optimisation et la recommandation. **Les problématiques associées sont : Un processus de séquençage de fonctions de réduction de données peut-il être utilisé seul pour la tâche d'optimisation et de recommandation ? Peut-il être révélateur de séquences compétitives de fonctions de réduction ?** L'idée ici est de comparer dans un premier temps les recommandations que peut fournir le processus de séquençage avec celles que peuvent fournir des MH classiques, afin d'éventuellement mettre à jour des cas d'exécution pour lesquels il est plus intéressant d'utiliser l'un ou l'autre. Dans un second temps, cela permettrait aussi de mettre en lumière les séquences les plus performantes en fonction des cas d'exécution.

1.1.3 Problématiques liées à l'explicabilité de l'E&E

Pouvoir expliquer la manière avec laquelle un algorithme explore et exploite l'espace de recherche demeure une ouverture importante de l'état de l'art [94] [275] [414]. Il est donc important de comprendre la manière avec laquelle l'E&E prennent place au sein des processus de recherche, c'est-à-dire de comprendre les transitions qui s'opèrent entre E&E [43], pour mieux gérer l'influence de l'E&E, pour décider d'utiliser ou non un algorithme dans un contexte d'exécution ou pour donner des informations compréhensibles à l'humain sur la manière dont un processus algorithmique s'est déroulé. En d'autres termes, expliquer le comportement d'E&E permettrait indirectement de parvenir à de meilleures solutions et permettrait de renforcer la confiance dans les résultats des algorithmes.

Nos problématiques sont donc les suivantes : Comment expliquer le comportement d'E&E ? Comment expliciter les transitions entre E&E ?

1.1.4 Problématiques et contexte éducatif

Les problématiques exposées jusqu'ici sont générales et peuvent être instanciées dans le cadre de la recommandation en milieu éducatif. En effet, nombre des processus de recommandation utilisés en milieu éducatif sont des processus d'E&E [333]. Il y a donc un potentiel besoin de les influencer. De plus, les données éducatives peuvent bénéficier des processus de réduction des données et le besoin d'explicabilité de l'E&E est lui aussi prégnant, en recommandation de ressources éducatives.

Par ailleurs, en contexte éducatif, ces problématiques peuvent être enrichies par des considérations liées au multi-acteur [280]. En effet, de nombreux acteurs peuvent être intégrés dans les processus de recommandation comme les enseignants, les apprenants ou les éditeurs de ressources. La prise en compte de données liées au multi-acteur peut notamment permettre de justifier certaines contraintes qui auront un impact direct sur les processus d'E&E.

1.2 Résumé des contributions

Nous avons choisi d'orienter nos contributions vers le domaine des MH car (1) elles sont transparentes et explicables [271] c'est-à-dire que leur lecture est claire et permet de comprendre comment les données sont collectées et traitées pour aboutir aux résultats, (2) elles sont performantes dans des environnements soumis à de nombreux aléas et contraintes et (3) elles sont particulièrement efficaces pour optimiser les décisions [432]. Dans le cadre des manuels scolaires

numériques, la démarche de recommandation peut ainsi bénéficier directement des trois points mentionnés ci-dessus.

Les contributions de ce travail de thèse se répartissent en trois grandes parties.

1.2.1 Influence de l'E&E

Nos contributions relatives à l'influence de l'E&E portent sur trois points. Premièrement, nous proposons un processus d'influence indépendant de l'algorithme influencé afin de favoriser la pleine expression de l'influence et de ne pas opérer de modifications structurelles et/ou paramétriques à l'intérieur de l'algorithme influencé. Deuxièmement, nous proposons une influence systématique de l'E&E, à chaque itération de l'algorithme influencé et tant que les *itemsets* ne respectent pas un indicateur d'E&E. Enfin, nous introduisons des indicateurs pour mesurer et représenter l'E&E sous de nouveaux angles.

De notre point de vue, les processus d'influence proposés dans la littérature ne garantissent pas une expression très impactante de l'influence. Une contribution de ce travail vise donc à donner plus de puissance au processus d'influence en concevant un processus indépendant de l'algorithme influencé.

Notons que l'indépendance du processus d'influence proposé fait qu'il peut être utilisé sur la plupart des MH, sans adaptation majeure.

De plus, nous proposons de nouveaux indicateurs pour mesurer et représenter le comportement d'E&E. Ils peuvent être utilisés au sein de la fonction d'influence. Comme le suggère la littérature, l'E&E mériteraient d'être représentées sous de nouveaux angles [94][414][275].

Le contrôle de l'E&E proposé est systématique car il est réalisé à chaque itération de l'algorithme de base mais aussi suite à toutes les variations du processus d'influence. Le processus d'influence proposé est évolutif, flexible et garantit la conformité finale des solutions trouvées par rapport aux résultats du contrôle de l'E&E. En effet, les variations d'influence sont opérées tant que la population n'est pas conforme à l'indicateur d'E&E, les bornes de l'indicateur évoluant au fil des itérations de la fonction d'influence et en fonction des valeurs prises par l'indicateur. Cela permet de garantir la conformité finale de la population à l'indicateur d'E&E. Dans la littérature, malgré la présence de processus évolutifs à l'intérieur du processus évolutif de base, il n'y a pas de recherche de conformité finale de l'indicateur d'E&E qui peut toujours être insatisfaisant suite aux variations d'influence.

Le processus d'influence proposé est indépendant et peut être greffé à n'importe quel algorithme, même non-évolutionnaire et il ne pose aucun problème de confrontation directe avec le processus algorithmique de base. Dans la littérature, les processus proposés agissent directement sur l'algorithme en opérant des modifications structurelles et/ou paramétriques. Nous supposons que cette forme de recherche de compromis n'impacte pas suffisamment l'E&E ou du moins ne les impacte pas d'une manière suffisamment indépendante. Nous supposons donc que l'impact sur l'E&E sera plus grand si l'influence est indépendante. Nous avons donc une hypothèse touchant à la force et à l'indépendance de l'impact sur l'E&E, ces deux éléments, distincts ou conjugués, pouvant mener à un comportement de recherche amélioré.

Nous proposons une liste importante et non exhaustive de structures possibles pour la réalisation des variations d'influence, laissant un choix intéressant en fonction du contexte d'implémentation.

Dans la littérature, la plupart des approches concernant la mesure et la représentation des comportements d'E&E sont basées sur l'appréciation de la diversité au sein de la population. Les indicateurs d'E&E proposés permettent de mesurer et de représenter le comportement d'E&E sous de nouveaux angles.

Nos travaux sur l'influence de l'E&E ont été l'occasion de proposer de nouveaux opérateurs de variations pour l'algorithmie évolutionnaire. Ces opérateurs peuvent être utilisés en tant qu'opérateurs d'influence de l'E&E mais aussi en tant qu'algorithmes évolutionnaires à part entière.

1.2.2 Réduction des données

Nous souhaitons que nos contributions répondent à certaines ouvertures mentionnées dans la littérature [254] [372] [416] [7] [285] [329].

Nos contributions consistent en un macro-processus et plusieurs fonctions pour la réduction d'un jeu de données par échantillonnage statistiquement conscient visant à réduire l'espace de recherche et à influencer indirectement l'E&E. Ces contributions discriminent les données en fonction d'informations représentatives. Cette approche apporte inévitablement des biais et nous faisons l'hypothèse que ces biais peuvent permettre d'atteindre des solutions difficilement accessibles.

De plus nous considérons qu'il est important de proposer des méthodes pouvant s'adapter à différents types de structures de données. Ces structures étant nombreuses, être suffisamment générique permet de toucher un grand nombre de domaines applicatifs et de recherche. Nous proposons donc des contributions utilisables sur la plupart des jeux de données.

Ensuite, ces contributions ont un fonctionnement explicable car elles se basent sur des informations statistiques qui peuvent être exposées simplement.

Ces contributions sont chainables au sein d'un macro-processus de réduction. Nous faisons l'hypothèse que plusieurs passes d'un même processus de réduction peut être bénéfique pour les traitements ultérieurs et que le mélange de processus de réduction peut lui aussi l'être.

Ces contributions ont pour objectif d'accélérer et d'améliorer les traitements ultérieurs des données en matière de performance, de qualité des solutions trouvées, d'E&E, d'analyse, de recommandation et de visualisation. En effet, une fonction de réduction modifie l'espace de recherche et peut donc avoir une influence indirecte sur le comportement d'E&E de l'algorithme. Nous souhaitons mettre en lumière cet impact de l'échantillonnage sur l'E&E à l'aide de critères d'évaluation.

Le macro-processus de réduction proposé est un chef d'orchestre qui chaîne les processus de réduction et qui décide quand cette chaîne de réduction doit s'arrêter. Chaque maillon de la chaîne de réduction travaille sur les données du maillon précédent afin de toujours plus réduire le jeu de données. Le macro-processus de réduction doit stopper la chaîne de réduction afin de répondre au compromis entre la taille et la représentativité des données. Cette approche simple et directe est pleinement explicable et suivant les processus algorithmiques des fonctions de réduction utilisées, peut présenter des temps de calcul raisonnables.

Par ailleurs, nous proposons un nouveau traitement algorithmique opérant un séquençage de fonctions d'échantillonnage. Ce traitement est utilisé seul pour une tâche d'optimisation et de recommandation et permet de cibler les séquences de fonctions de réduction les plus intéressantes. Cette contribution est avant tout conceptuelle. Le processus effectue différentes réductions d'un jeu de données, chaque version réduite étant issue d'une séquence de fonction de réduction. Pour une séquence, le jeu de données est réduit jusqu'à l'obtention d'une sélection finale ou recommandation ou sélection optimisée. Nous axons donc le processus d'optimisation et de recommandation sur la recherche d'une séquence de fonctions de réduction amenant une conscience statistique satisfaisante, passant par la discrimination brute d'une partie des données en fonction d'informations représentatives. Cette approche apporte inévitablement des biais mais nous misons sur ces biais pour atteindre des solutions difficilement accessibles. Cette contribution est

universelle car n'importe quelle fonction de réduction peut être introduite dans l'ensemble de départ. Elle est aussi non-supervisée car nous partons du principe que la supervision se caractérise souvent avec des temps de calcul élevés ainsi qu'avec une offuscation du processus de réduction, pouvant rendre son explicabilité impossible.

1.2.3 Explicabilité de l'E&E

Notre contribution relative à l'explicabilité de l'E&E consiste en de nouvelles approches permettant de donner des éléments d'information sur la manière avec laquelle une métaheuristique explore et exploite l'espace de recherche. Ces approches sont basées sur des indicateurs, issus d'hyper-volumes, de vecteurs de variation de présences ainsi que de résultats de classification, afin d'explicitier les transitions entre E&E. Par exemple, un indicateur donnant des informations concernant l'utilisation d'*items* spécifiques au cours des itérations et qui peut représenter graphiquement des marqueurs de transition entre E&E.

Nous proposons plusieurs indicateurs hyper-volumiques permettant d'obtenir des informations sur le comportement d'E&E d'une MH. A chaque itération du processus métaheuristique, ces indicateurs ainsi que d'autres indicateurs sont calculés et enregistrés au sein d'une variable de log.

Nous justifions nos contributions à travers les points présentés ci-après : (1) Le processus d'enregistrement des logs peut être greffé à la plupart des algorithmes ayant une course itérative. (2) Les indicateurs proposés peuvent être adaptés à de nombreuses métaheuristicques car bien que présentés ici dans un contexte d'algorithmie évolutionnaire [44], ils sont avant tout conceptuels. (3) Les indicateurs proposés donnent une représentation de l'E&E sous de nouveaux angles. (4) Les indicateurs proposés sont des marqueurs de différences inter-algorithme permettant de comparer les comportements d'E&E et de souligner les comportements marginaux.

Nous proposons aussi des indicateurs qui tentent de répondre à l'ouverture liée à la variété des transitions qui peuvent se produire entre E&E. Ces indicateurs sont basés sur des vecteurs de variations de présence et visent à donner des informations sur les évolutions de présence des *items*, les évolutions des variations de présence des *items*, les chaleurs de transition entre itérations et les classes de transition. Nous affirmons que ces contributions (1) sont adaptables à de nombreuses MH, (2) donnent une représentation de l'E&E sous de nouveaux angles permettant de renforcer l'explicabilité de l'E&E, la diversité étant l'angle de vue majeur dans la littérature, (3) montrent des différences inter-algorithmes permettant de comparer les comportements d'E&E et de souligner les comportements marginaux, (4) permettent d'expliquer les transitions entre E&E.

Par ailleurs, nous proposons un algorithme de classification d'*itemsets* intervenant à la fin de chaque itération d'une métaheuristique. Si lors de sa classification un *itemset* conduit à la création d'une nouvelle classe, nous considérons qu'il s'agit d'un *itemset* d'exploration et que l'itération au sein de laquelle il a été créé est une itération d'exploration. Les autres *itemsets* qui viennent s'ajouter aux classes existantes, sont considérés comme des *itemsets* d'exploitation et les itérations associées sont considérées comme des itérations d'exploitation. Nous justifions cette contribution à travers les points suivants : (1) Le processus de classification peut être greffé à la plupart des algorithmes ayant une course itérative. (2) L'approche proposée donne une représentation de l'E&E sous un nouvel angle. (3) L'approche proposée permet de marquer des différences inter-algorithme, de comparer les comportements d'E&E et de souligner les comportements marginaux.

1.3 Plan du document

Nous commençons ce document par un état de l'art des divers domaines liés à notre contexte ainsi qu'à nos problématiques ([Chapitre II](#)), à savoir, les domaines de la métaheuristique, de l'influence et de l'explicabilité de l'E&E, de la réduction de données et des manuels scolaires numériques. Pour continuer, nous présentons nos contributions ([Chapitre III](#)) concernant l'influence de l'E&E, la réduction de données ainsi que l'explicabilité de l'E&E. Nous poursuivons avec les expérimentations associées, leurs résultats et conclusions ([Chapitre IV](#)). Nous terminons ce document par des éléments de discussion permettant d'ouvrir des perspectives pour de futurs travaux ([Chapitre V](#)).

Chapitre 2

État de l'art

Notre contexte est pluriel car il s'intéresse à la recommandation, à la personnalisation, à la fouille de données, aux learning analytics, à la transformation des données, à l'optimisation, à l'apprentissage automatique, aux mécanismes décisionnels ainsi qu'aux manuels scolaires numériques. Ces domaines sont très généraux, ils dépassent ainsi de loin le cadre applicatif des manuels scolaires numériques.

Cet état de l'art s'intéresse aux domaines de la métaheuristique, de l'influence et de l'explicabilité de l'E&E, de la réduction de données et des manuels scolaires numériques. Nous présentons ces divers domaines car nos questions scientifiques y sont liées. Pour rappel, nos questions scientifiques sont les suivantes : Comment expliquer, mesurer, contrôler et influencer l'E&E ? Quels processus d'échantillonnage proposer afin de réduire un jeu de données en conservant un maximum de ses caractéristiques dans le but de servir une démarche d'E&E pour la recommandation ? L'utilisation conjointe de processus de réduction des données statistiquement conscients et d'un processus indépendant pour influencer l'E&E, permet-elle d'avoir un impact significatif sur les critères d'évaluation de la performance, de la qualité et du comportement d'E&E d'un algorithme de recherche ? Un processus de séquençage de fonctions de réduction de données peut-il être utilisé seul pour la tâche d'optimisation et de recommandation ? Peut-il être révélateur de séquences compétitives de fonctions de réduction ? Comment expliquer le comportement d'E&E ? Comment expliciter les transitions entre E&E ?

2.1 Métaheuristicques

Les problèmes modernes de recherche de solutions sont souvent complexes et liés à l'analyse de grands ensembles de données. Il existe des stratégies générales qui sont appliquées avec succès à des problèmes complexes mais, en réalité, il suffit souvent de trouver une solution approximative [160] comme dans le problème du sac à dos [260] qui consiste à trouver des combinaisons d'objets à insérer dans un sac à dos, le sac à dos ayant un volume maximal et chaque objet ayant un volume et un score d'utilité. Il s'agit donc d'un problème d'optimisation dont l'objectif est de trouver des solutions qui minimisent ou maximisent une fonction objectif. La fonction objectif est utilisée pour évaluer la qualité des solutions trouvées. De nombreux problèmes du monde réel peuvent être définis comme des problèmes d'optimisation comme le problème du voyageur de commerce [108] qui consiste à déterminer, étant donné plusieurs villes, le plus court circuit passant par chaque ville. L'ensemble des solutions possibles pour un problème donné peut être considéré comme un espace de recherche, ainsi les algorithmes d'optimisation sont souvent appelés algorithmes de recherche [217] [26]. Notre problème de recommandation de ressources éducatives

peut être traité comme un problème d'optimisation car les combinaisons de ressources éducatives à recommander peuvent être trouvées et optimisées à l'aide d'une MH afin de converger vers des combinaisons compétitives au regard de différents critères et contraintes.

2.1.1 Optimisation

Les problèmes d'optimisation peuvent être divisés en deux grandes catégories, exacts et approximatifs. Les algorithmes exacts trouvent la meilleure solution possible alors que les algorithmes approximatifs peuvent trouver la meilleure solution ou une solution satisfaisante. Nous ne nous intéressons pas aux algorithmes exacts car ils induisent souvent une charge de calcul supérieure aux algorithmes approximatifs et car, dans le cadre de notre problème, il est impossible de se prononcer sur l'exactitude d'une solution. En effet, dans le cadre de notre problème, une solution peut être exacte au regard de la fonction de fitness mais comme il est difficile de se prononcer en premier lieu sur l'exactitude de la composition de cette fonction, l'exactitude de cette solution est discutable. Les algorithmes approximatifs se divisent en deux catégories : les heuristiques (H) et les métaheuristiques (MH).

Les H et MH sont utilisées pour la recherche de solutions (E&E), pour la résolution des problèmes d'optimisation NP-difficiles (problème du voyageur de commerce, problème de la clique maximum, problème d'ordonnement de flux, problème P-Median) et aussi pour la sélection de caractéristiques dans la reconnaissance des motifs, pour le clustering automatique ou pour l'apprentissage automatique.

2.1.2 Heuristiques

L'heuristique (H) [217] fait référence à des techniques pour la résolution de problèmes, l'apprentissage et la découverte. Lorsqu'une recherche exhaustive est impossible, en raison de problèmes liés à la puissance de calcul, au temps d'exécution mais aussi à la consommation d'énergie, des méthodes heuristiques sont utilisées pour accélérer le processus de recherche d'une solution satisfaisante. Des exemples de cette méthode incluent l'utilisation d'une règle empirique, une supposition éclairée, un jugement intuitif ou le bon sens. En informatique, en intelligence artificielle et en optimisation mathématique, une heuristique est une technique conçue pour résoudre un problème plus rapidement lorsque les méthodes classiques sont trop lentes, ou pour trouver une solution approximative lorsque les méthodes classiques ne parviennent pas à trouver la solution exacte [217]. Ces algorithmes trouvent donc généralement une solution optimale, rapidement et facilement, et peuvent parfois trouver la meilleure solution.

2.1.3 Métaheuristiques

Les méthodes métaheuristiques (MH) sont des heuristiques intégrées dans des processus itératifs afin d'augmenter leurs capacités d'E&E [144]. Les MH sont dites stochastiques car elles utilisent de manière itérative des processus aléatoires. La grande majorité de ces méthodes sont utilisées pour résoudre des problèmes d'optimisation bien que le cadre général ne leur soit pas strictement dédié [103]. Les MH se sont avérées bien adaptées aux problèmes d'optimisation multi-objectifs [425] et mono-objectifs [362] car elles sont capables de capturer plusieurs solutions simultanément, tout en tenant compte des contraintes.

En informatique, une MH désigne une méthode de calcul qui optimise un problème en essayant de manière itérative d'améliorer une solution candidate par rapport à une mesure de qualité. Les MH peuvent rechercher dans de très grands espaces de solutions candidates, aussi appelés espaces de recherche. Cependant, les MH ne garantissent pas qu'une solution optimale

soit trouvée. Une MH [302], [163], [418] peut aussi être formellement définie comme un processus de génération itératif qui guide une heuristique subordonnée en combinant intelligemment différents concepts pour explorer et exploiter l'espace de recherche. Les MH peuvent être utilisées pour résoudre des problèmes complexes. D'autres termes ayant une signification similaire à celle de MH sont : *derivative – free*, recherche directe, boîte noire ou simplement optimiseur heuristique. Voici les propriétés qui caractérisent la plupart des MH : (1) Les MH sont des stratégies qui guident le processus de recherche pour explorer efficacement l'espace de recherche et trouver des solutions (quasi) optimales. (2) Les techniques MH vont de simples procédures de recherche locale aux processus d'apprentissage complexes. (3) Les MH sont approximatives et généralement non déterministes. (4) Les MH ne sont pas spécifiques à un problème.

Les MH peuvent faire peu d'hypothèses sur le problème d'optimisation en cours de résolution, elles peuvent donc être utilisables pour une variété de problèmes. Les problèmes d'optimisation qui ont attiré l'attention des approches MH sont variés, allant de simple à multi-objectif, continu à discret, contraint à non contraint. Résoudre ces problèmes n'est pas une tâche simple en raison de leur complexité.

Les MH sont divisées en deux catégories : *population – based* et *trajectory – based*.

2.1.4 MH et recommandation

Les problématiques d'optimisation prennent souvent place dans un environnement multi-critères et multi-objectifs. D'après [235], un système de recommandation multi-critère cherche à optimiser ses recommandations pour différents critères simultanément et ce à l'aide d'une fonction d'utilité. Il cherche aussi à exploiter les relations entre les critères. Il existe de nombreux critères [13] [207] [435] comme la précision, la sérendipité, la nouveauté, la couverture, la diversité. L'optimisation multi-objectifs permet de trouver des solutions satisfaisantes voire optimales lorsque la recherche exhaustive n'est pas possible [429]. Les ressources [431] [10] [39] [235] [432] [361] [430] donnent des définitions similaires au concept de "multi-objectifs" et des exemples d'applications aux systèmes de recommandation sont présentés dans [140] [330] [441].

Les tableaux de *digits* binaires sont une manière traditionnelle de représenter des *items* ou *itemsets* recommandés [260]. Chaque index d'un tableau correspond à un *item* recommandable. Les *digits* peuvent prendre la valeur 0 (non-recommandation de l'*item*) ou 1 (recommandation de l'*item*). Le nombre de *digits* à 1 correspond donc au nombre d'*items* recommandés. L'algorithme effectue itérativement des variations : génère, croise et transforme des tableaux, tout en essayant d'optimiser et de garder le meilleur d'entre eux au regard de contraintes prédéfinies et de critères d'évaluation.

Dans la littérature, l'utilisation des MH n'est pas systématiquement référée à la recommandation mais la plupart du temps à l'optimisation qui est, à bien des égards, très proche de la recommandation. En effet, en considérant le problème du sac à dos [260], nous pouvons affirmer que les solutions trouvées par les MH pour résoudre ce problème peuvent constituer des recommandations et qu'il est également possible d'ajouter des contraintes représentant des besoins spécifiques de l'utilisateur sans changer l'essence du problème d'optimisation et en apportant le fait que les solutions trouvées dans ce contexte constituent des recommandations personnalisées pour ce dernier utilisateur. Par conséquent, faire un état de l'art concernant l'utilisation des MH pour la tâche de recommandation doit prendre en considération toutes les contributions du domaine des MH. Nous avons trouvé deux revues intéressantes concernant les MH [108] [110] et quelques contributions spécifiques concernant l'utilisation des MH pour la recommandation [308] [208] [160] [380] [361]. Dans [361], un système de recommandation à optimisation multi-objectifs est proposé. Il génère des parcours d'apprentissage pour les apprenants à l'aide d'algorithmes

métaheuristiques, notamment l'algorithme génétique (Genetic Algorithm - GA) et l'algorithme d'optimisation des colonies de fourmis (Ant Colony Optimization - ACO). Concrètement, les éléments à recommander (cours) sont des nœuds dans un graphe, la métaheuristique est utilisée pour générer des chemins à travers le graphe et les meilleurs chemins trouvés sont recommandés. Dans [208], un système de recommandation hybride appliqué au jeu de données MovieLens est proposé. Concrètement, un algorithme de classification k-means est utilisé pour créer des clusters d'utilisateurs, chaque cluster ayant un centroïde. Ensuite, l'algorithme de colonie d'abeilles artificielles (Artificial Bee Colony - ABC) est utilisé pour optimiser les distances aux centroïdes des utilisateurs et les reclasser si nécessaire. Les recommandations données à un utilisateur sont directement liées à ses distances aux centroïdes, en effet des contenus liés aux utilisateurs similaires, au regard des centroïdes, sont recommandés. Dans [380], les auteurs proposent d'améliorer les recommandations d'itinéraires pour les touristes grâce à des algorithmes métaheuristiques. Concrètement, un algorithme de classification k-means est utilisé pour créer des clusters de points d'intérêt (Points Of Interest - POI) dans des zones géographiques. Pour chaque cluster, un algorithme génétique optimise l'itinéraire pour visiter tous les POI. Ces trois exemples confirment que, comme dit plus haut, la tâche de recommandation peut être considérée comme une sélection optimisée d'éléments dans un espace de recherche.

2.1.5 MH hybrides et hyperheuristiques

Les MH hybrides apportent des améliorations significatives par rapport aux MH classiques. L'objectif principal des algorithmes hybrides est de coupler les caractéristiques de différentes stratégies de recherche afin d'obtenir une synergie. L'hybridation la plus connue mélange les algorithmes basés sur la population avec la recherche locale [378].

Les hyperheuristiques (HH) se placent à un niveau supérieur car elles cherchent à activer la bonne (méta)-heuristique en fonction de l'espace de recherche et/ou des avancements du processus de recherche.

2.1.6 Conclusion et ouvertures

Les MH permettent de résoudre des problèmes d'optimisation difficiles et des problèmes de recherche complexes. Elles ne sont pas spécifiques à un problème et peuvent obtenir de meilleures solutions que les H. De plus, leur mise en œuvre est aisée et ne requiert pas d'informations contextuelles.

De nombreuses questions concernant les MH restent en suspens, constituant autant d'opportunités de recherche. Il est possible de comparer les MH afin d'aborder ces questions mais la comparaison ne permet jamais de tirer une vérité générale sur la supériorité d'une MH par rapport à une autre [439] [51] [32]. En effet, les performances d'une MH, tous critères considérés, varient en fonction de l'espace de recherche, des contraintes et plus généralement en fonction du traitement des données.

Plusieurs caractéristiques fondamentales permettent de décrire une MH à savoir : (1) La quantité de paramètres à traiter pour exécuter efficacement le processus d'optimisation. (2) Les étapes où l'algorithme peut équilibrer les efforts d'E&E. (3) La disponibilité de mécanismes de recherche locaux.

Concernant (1), la plupart des algorithmes MH de nouvelle génération ont un grand nombre de paramètres, ce qui est un inconvénient car afin d'obtenir des résultats de haute qualité dans des délais acceptables, les paramètres doivent être réglés pour la tâche d'optimisation. Les recherches ont progressé pour pallier cet inconvénient. Les algorithmes MH tels que SSO [96], SOS [82] et

TLBO [326] visent à utiliser moins de paramètres. En plus des tentatives qui ont été faites pour fournir des réglages de paramètres adaptatifs, une autre option consiste à développer des algorithmes MH sans paramètres. Cependant, cela demeure un problème ouvert. De plus amples informations sur les méthodologies expérimentales, les évaluations statistiques et le réglage des paramètres des MH peuvent être trouvées dans la littérature [36] [47] [269].

Concernant (2), fournir un bon équilibre entre les phases d'E&E du processus de recherche est un autre critère important pour la performance des algorithmes MH. De nombreux processus inspirés de l'évolution fournissent des mécanismes pour contrôler cet équilibre. De plus, les efforts pour contourner les optima locaux tout en explorant l'espace de recherche reste un domaine de recherche important. Le « redémarrage » est l'une des techniques actuelles largement utilisées pour pallier ce problème [110]. Aussi, le traçage des précédents optima locaux ou les techniques de clustering intelligent peuvent être de nouvelles directions de recherche prometteuses pour traiter le problème de la stagnation [110].

Concernant (3), le manque de mécanismes de recherche locale pouvant atteindre des optima locaux est un autre problème critique pour les algorithmes MH. En effet, ces mécanismes permettent de comprendre et d'améliorer les résultats d'un algorithme, ils garantissent aussi une amélioration des solutions candidates au cours des itérations successives. Cependant, la présence de ces mécanismes n'est pas indispensable, leur absence n'empêche pas une MH d'atteindre des résultats satisfaisants. Aussi, comprendre comment les approches MH obtiennent théoriquement des résultats satisfaisants est toujours une question de recherche ouverte.

Pour continuer, l'un des principaux inconvénients des algorithmes MH est qu'ils doivent estimer la valeur de fitness de chaque nouvelle solution qu'ils produisent. Les performances diminuent très rapidement lorsque la dimensionnalité du problème augmente et lorsque la complexité de l'évaluation de la fitness est élevée. La programmation dynamique ou le calcul parallèle peuvent être très efficaces pour calculer les fitness complexes [110].

Par ailleurs, les algorithmes MH hybrides sont une technologie émergente, et de nombreuses possibilités d'hybridation de MH n'ont toujours pas été abordées. La plupart des algorithmes hybrides et la plupart des HH rapportés obtiennent de meilleures solutions que les algorithmes MH classiques [110]. Il est important de noter ici que les études sur les HH visent à être indépendantes du problème et utilisables par des chercheurs non spécialistes dans ce domaine [110].

De plus, la plupart du temps, l'évaluation de la performance des MH est réalisée avec une analyse statistique en raison du manque de fondement théorique [83]. Il est donc nécessaire de fournir des méthodes de comparaison statistiquement valables. Conformément au théorème No Free Lunch, il n'est pas possible de s'attendre à ce qu'une MH fonctionne bien pour toutes les classes de problèmes d'optimisation [187] [410]. Une discussion approfondie sur les directions de recherche concernant la rigueur scientifique des MH peut être trouvée dans [363]. Les auteurs de [95] présentent douze règles pour l'évaluation des algorithmes MH dont les deux plus importantes sont : (1) Préférer un nombre égal d'évaluations de la fitness. (2) Examiner les problèmes sur lesquels l'algorithme proposé fonctionne bien. Ainsi, des études sur des tests de référence impliquant divers problèmes d'optimisation devraient être établies.

Une autre direction de recherche serait d'améliorer les structures internes des algorithmes MH actuels plutôt que d'en proposer de nouveaux qui sont similaires aux algorithmes existants. Il semble par exemple rester de nombreuses possibilités d'amélioration des opérateurs adaptatifs.

Enfin, le succès passé et actuel de ces algorithmes (H, MH et HH) leur garantit un riche avenir.

2.2 Influence et Explicabilité de l'E&E

L'E&E (Exploration et Exploitation) sont les deux pierres angulaires de la résolution de problèmes par la recherche [94]. Explorer signifie s'assurer que de nombreuses régions de l'espace de recherche sont examinées, afin de ne pas se retrouver piégé dans un optimum local. Exploiter signifie que si une région prometteuse de l'espace de recherche est trouvée, elle est examinée attentivement pour voir si elle contient un bon optimum local. L'E&E peuvent aussi être appelées respectivement "diversification" et "intensification" ou encore "recherche globale" et "recherche locale". Selon March [259], l'exploitation est définie comme le raffinement, le choix, la production, l'efficacité, la sélection, la mise en œuvre et l'exécution et l'exploration comme la recherche, la variation, la prise de risque, l'expérimentation, le jeu, la flexibilité, la découverte et l'innovation. Nous pouvons partir du principe que l'E&E sont associées à une certaine forme d'apprentissage et d'innovation [37] [159] [169]. Nous pouvons aussi caractériser l'exploitation uniquement comme l'utilisation des connaissances passées, c'est-à-dire une absence d'apprentissage [397] [399].

L'E&E sont donc en conflit par nature [94] car si beaucoup de temps est passé sur l'une, moins de temps est disponible pour l'autre. Eiben et Schippers [8] ont longtemps abordé la nécessité de trouver un équilibre entre E&E, considérant ces deux processus comme antagonistes. Malgré ce conflit apparent, l'E&E sont en réalité un continuum, leur interdépendance est au cœur de la question de savoir comment parvenir à un équilibre. En effet, l'exploitation conduit à des résultats immédiats et l'exploration promet des avantages futurs. Par conséquent, il est nécessaire d'explorer pour créer des opportunités à exploiter. La recherche d'un équilibre entre E&E est notamment importante en recommandation, contexte au sein duquel l'exploitation consiste en la recommandation de contenus (produits, films, ressources) pour lesquels il est attendu une grande satisfaction utilisateur, et où l'exploration consiste en la recommandation de contenus pour lesquels la satisfaction de l'utilisateur est incertaine et ce afin de recueillir des informations sur ses goûts.

Depuis les contributions fondatrices, des solutions à la fois contradictoires et complémentaires concernant le compromis entre E&E ont émergé [159]. Notons que nous avons choisi d'utiliser le terme "influence" pour regrouper les différentes appellations de l'état de l'art telles que "recherche d'équilibre", "recherche de compromis", "guidage".

Par ailleurs, concernant l'explicabilité, il existe deux angles de vue sur ce concept, à savoir (1) l'explicabilité des solutions et (2) l'explicabilité des comportements algorithmiques. L'explicabilité des solutions représente le caractère explicable des sorties d'un système ou algorithme comme par exemple le caractère explicable des recommandations proposées à un utilisateur. L'explicabilité des recommandations est cruciale pour que les utilisateurs comprennent ce qui leur est proposé. Ajouter une explication associée à une recommandation est connue pour augmenter la satisfaction et l'engagement des utilisateurs ainsi que la force de persuasion des recommandations [218] [132] [383], résultant d'une plus grande confiance et transparence dans les recommandations [132]. L'explicabilité des comportements algorithmiques représente le caractère explicable de la manière avec laquelle un système ou algorithme a atteint les solutions qu'il propose, comme le caractère explicable d'un comportement d'E&E. Dans le cadre de nos travaux, nous ne nous sommes pas intéressés à l'explicabilité des solutions mais à l'explicabilité de l'E&E. L'état de l'art actuel en matière d'explicabilité de l'E&E est inexistant, les seuls éléments permettant de donner des explications sur le comportement d'E&E étant les divers indicateurs, paramètres de contrôle, opérateurs et structures algorithmiques utilisés dans le cadre de l'influence de l'E&E.

Cet état de l'art repose sur la traduction ainsi que la synthèse de trois revues [414][94][275] qui abordent les problématiques de l'influence de l'E&E dans, respectivement, les domaines de la métaheuristique, de l'algorithmie évolutionnaire ainsi que de la psycho-sociologie. Nombre

de points abordés au sein de ces trois revues peuvent être généralisés à d'autres domaines de recherche ainsi qu'à d'autres domaines applicatifs.

2.2.1 Contextes Organisationnels

D'après et non limité à [368], une organisation, un système ou un algorithme de recherche, cherchant à mettre en place une influence de l'E&E, comprend une à n entités d'E&E étant chacune soit spécialisée dans l'exploration, soit spécialisée dans l'exploitation, soit non spécialisée c'est-à-dire opérant les deux. Les entités peuvent conserver leur spécialisation ad vitam. Un à m agents coordonnent et confrontent les entités d'E&E. Ils opèrent dans certains cas des basculements entre E&E. Les entités peuvent aussi décider à leur niveau quand opérer des basculements entre E&E. Les basculements s'opèrent en fonction de divers indicateurs comme des indicateurs de diversité ou des indicateurs temporels. Nous proposons Figure 2.1 pour résumer ces points.

De nombreuses questions se posent en fonction du contexte comme : (1) Comment assurer une bonne coopération entre les entités d'E&E? (2) Comment les entités peuvent-elles décider quand l'exploration ou l'exploitation sont appropriées? (3) Comment les exploitants peuvent-ils accéder à de nouvelles connaissances? (4) Comment gérer correctement les basculements entre E&E en fonction des indicateurs choisis?

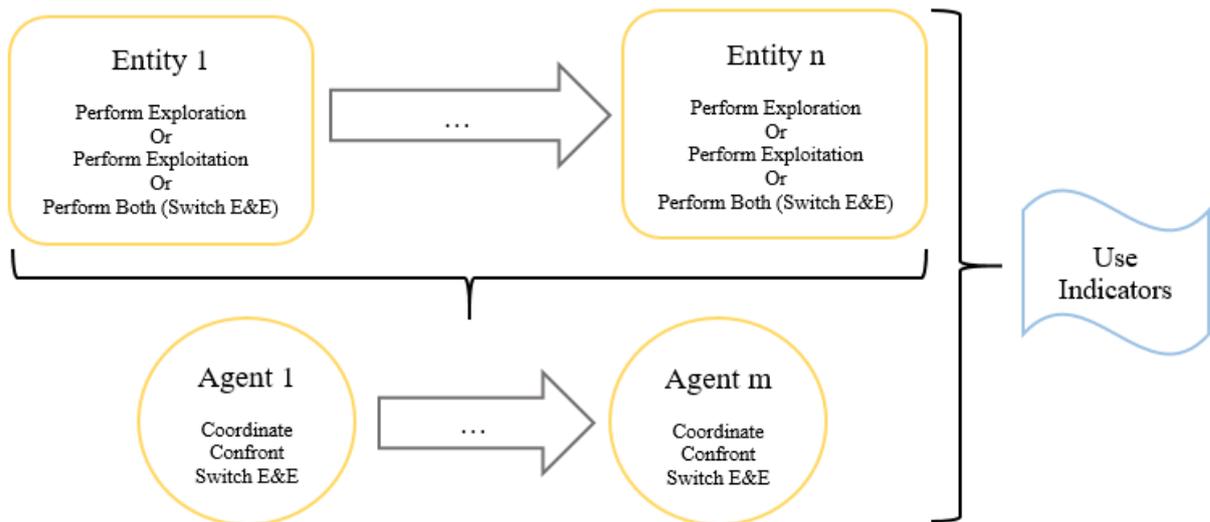


FIGURE 2.1 – Organisations de l'E&E

2.2.2 Influence de l'E&E dans les MH

Introduction

Les méthodes heuristiques font référence à des techniques pour la résolution de problèmes, l'apprentissage et la découverte où une recherche exhaustive n'est pas pratique. Elles sont utilisées pour accélérer le processus de recherche. Au cours des dernières décennies, une nouvelle famille d'algorithmes de recherche et d'optimisation a vu le jour, basée sur l'extension des méthodes heuristiques de base en les intégrant dans un cadre itératif augmentant leurs capacités d'E&E. Ces algorithmes ont reçu le nom de métaheuristiques (MH) [144], un aperçu des différentes méthodes existantes se trouve dans [51]. De nombreuses études montrent que les MH sont très utiles pour résoudre approximativement des problèmes d'optimisation difficiles dans

la pratique, car ils peuvent obtenir de bonnes solutions dans un laps de temps réduit. Recuit simulé (Simulated Annealing - SA), recherche tabu (Tabu Search - TS), algorithmes évolutifs (Evolutionary Algorithm - EA), optimisation des colonies de fourmis (Ant Colony Optimization - ACO) et recherche par dispersion (Scatter Search - SS) sont, entre autres, souvent répertoriées comme des exemples de MH classiques. Ils ont des antécédents historiques individuels et suivent des paradigmes et des philosophies différents. En tant que méthodes d'optimisation globale, les MH ont des capacités à la fois d'E&E, mais il a souvent été constaté qu'elles étaient piégées dans un optimum local plutôt que global. La raison principale est la difficulté à arbitrer correctement entre l'E&E. Dans une certaine mesure, concevoir un bon algorithme MH consiste à faire un bon compromis entre l'E&E. Malheureusement, il n'y a pas de réponse complète à cette question à ce jour.

E&E

Différentes explications aux termes exploration et exploitation ont été utilisées dans la littérature sur les MH. Tout d'abord, l'E&E peuvent être caractérisées comme une recherche globale et une recherche locale [231, 341, 320, 24]. Cependant, il n'est pas fiable de distinguer l'E&E uniquement par la portée de la recherche, car il n'y a pas de seuil incontesté pour définir l'échelle du local. Dans les algorithmes d'apprentissage automatique, l'E&E correspondent respectivement à l'acquisition et à l'utilisation des connaissances [420]. Certains chercheurs pensent que l'E&E peuvent être distinguées par l'intensité du caractère aléatoire [312]. L'E&E sont aussi considérées comme deux types de comportement dans l'acquisition d'informations sur des problèmes inconnus [79].

L'E&E sont également appelées diversification et intensification [51]. D'une manière générale, la diversification fait référence à la capacité de visiter de nombreuses et différentes régions de l'espace de recherche, et l'intensification fait référence à la capacité d'obtenir des solutions de haute qualité dans ces régions [248]. Un algorithme de recherche doit trouver un équilibre entre ces deux objectifs parfois contradictoires. La plupart des MH classiques ont plusieurs composants différents pour l'E&E comme les opérateurs génétiques ou les perturbations des distributions de probabilités. Ainsi, les composants E&E sont des opérateurs, des actions ou des stratégies de MH. Fournir un bon équilibre entre les composants E&E d'une MH est une tâche compliquée. En effet, la plupart des MH classiques affiche une nette tendance soit à l'exploration soit à l'exploitation. Le compromis entre l'E&E est toujours un sujet important dans les MH mais aussi dans d'autres domaines tels que l'apprentissage automatique, la modélisation et la prédiction. Le compromis entre l'E&E est important car il peut permettre de réduire les coûts de calcul et d'atteindre de meilleures solutions.

Difficulté du problème

La qualité du compromis entre l'E&E dépend de la difficulté d'optimisation du problème [79]. L'étude de cette difficulté peut être divisée en deux classes [55]. La première classe se concentre sur les propriétés de l'algorithme tandis que la seconde classe se concentre sur le problème lui-même et en particulier sur le paysage de fitness (Fitness Landscape - FL) du problème [291] [197] [56] [296] [168] [417] [239] [406] [245] [336] [57] [316][209].

Compromis entre E&E

Le compromis (tradeoff) entre l'E&E (T-E&E) est crucial pour les méthodes d'optimisation car il aide à réduire les coûts de calcul et à mettre en œuvre une procédure d'optimisation

efficace. Chaque MH a sa propre manière de gérer le T-E&E pour échapper aux optima locaux. Par exemple, un EA explore à l'aide de croisements et exploite à l'aide de mutations, un ACO explore à l'aide de phéromones et exploite à l'aide d'informations heuristiques.

Les études antérieures sur le T-E&E peuvent être classées en deux catégories : (1) Celle qui étudie des méthodologies qui relient le T-E&E au problème des bandits. Les statistiques sont l'outil le plus puissant pour analyser ces problèmes. (2) Celle qui étudie le contrôle direct du T-E&E.

Il existe de nombreuses méthodes différentes pour gérer le T-E&E dans la littérature : adaptative, hybride et interactive.

Approche Adaptative : Une MH adaptative peut changer sa configuration, régler un paramètre/opérateur particulier ou encore changer sa structure, tout au long du processus de recherche. Traditionnellement, des paramètres fixes sont appliqués aux MH et les configurations sont déterminées avant l'exécution, automatiquement ou manuellement. Cependant, les paramètres fixes et les configurations prédéfinies ne peuvent pas être appliqués à de nombreux problèmes, il peut aussi être souhaitable de les modifier en fonction du stade d'évolution lors de l'exécution. Les paramètres fixes et les configurations prédéfinies ne peuvent donc pas satisfaire les exigences des problèmes d'optimisation difficiles. Ainsi, ils doivent être gérés dynamiquement grâce à un contrôle par exemple, qui est la méthode la plus populaire. Les stratégies de contrôle des paramètres dans les MH sont classées en deux catégories, à savoir le contrôle déterministe et le contrôle adaptatif. Le contrôle déterministe modifie les paramètres selon des règles déterministes prédéterminées et n'utilise aucune rétroaction du processus d'évolution. Au contraire, le contrôle adaptatif (parfois appelé contrôle auto-adaptatif) est lié à une certaine forme de rétroaction de l'algorithme.

Divers MH adaptatives avec des configurations contrôlées ont été développées, et les résultats d'études empiriques ont montré leur succès et leur utilité. Trois questions doivent être abordées lors de la conception d'une MH adaptative [428] : (1) Quels objets faut-il contrôler et adapter ? (paramètres, opérateurs, populations) (2) Quelles informations sont utilisées pour activer/guider le processus d'adaptation ? Quand et comment exécuter l'adaptation ? Les informations les plus souvent utilisées sont les valeurs de fitness ou les distributions des populations. (3) Quelles méthodes sont utilisées pour contrôler les configurations ?

- *Objets d'Adaptation :*

- Paramètres de contrôle : Habituellement, les MH sont associées à un ensemble de paramètres de contrôle tels que la probabilité de croisement p_c et la probabilité de mutation p_m dans les GA. Ces paramètres de contrôle ont des effets significatifs sur les performances des algorithmes. Différents types de problèmes peuvent nécessiter différents réglages et ces paramètres ne sont pas indépendants mais ils interagissent les uns avec les autres. De plus, différentes étapes du processus d'exécution peuvent nécessiter des valeurs de paramètres différentes pour obtenir les meilleures performances. Par conséquent, il est plus approprié de contrôler de manière adaptative les valeurs des paramètres pendant le processus de recherche.
- Opérateurs évolutionnaires : La procédure d'une MH implique une séquence d'opérateurs évolutifs comme la mutation, le croisement et la sélection. Un opérateur peut généralement être implémenté de différentes manières. Différents opérateurs sont nécessaires pour différents états évolutifs et pour différents problèmes d'optimisation afin d'atteindre une performance idéale.
- Population : Les changements dynamiques de la taille de la population et de la topologie de la population sont deux formes principales de l'adaptation de la population. Maintenir la diversité de la population permet d'éviter une convergence prématurée.

- Structure algorithmique : Certains morceaux de code au sein de la MH peuvent être activés ou désactivés au cours de l'exécution.
- Autres : Par exemple, les MH parallèles sont constituées de plusieurs populations qui communiquent entre elles par un processus de migration. Les paramètres tels que le taux de migration, la taille de la migration et la topologie de communication peuvent être modifiés de manière adaptative au cours du processus de recherche.
- *Informations d'Adaptation* :
 - Facteurs déterministes : Le nombre de générations et le score de fitness sont deux facteurs déterministes couramment utilisés. Par exemple, le facteur d'échelle F de l'algorithme DE est diminué linéairement en fonction du nombre de générations.
 - Valeurs de fitness : Chaque individu maintient souvent ses propres paramètres qui sont contrôlés en fonction de la valeur de fitness individuelle. Dans d'autres schémas, les valeurs de fitness sont mesurées au niveau de la population et les paramètres sont également adaptés pour l'ensemble de la population [367].
 - Distribution de la population : La répartition de la population est souvent utilisée pour refléter sa diversité. Dans de tels cas, tous les membres de la population sont généralement associés aux mêmes paramètres adaptatifs. Un certain nombre de GA adaptatifs contrôlent la probabilité de croisement p_c ou la probabilité de mutation p_m sur la base de la diversité de la population. Dans ces approches, la diversité de la population peut être obtenue par des mesures telles que la distance de Hamming et d'autres informations statistiques.
 - Combinaison de la fitness et de la distribution : Par exemple, dans l'algorithme génétique adaptatif (Adaptive Genetic Algorithm - AGA), la population est partitionnée en clusters. Chaque cluster contient les individus ayant des vecteurs de composants similaires. Ensuite, p_c et p_m sont ajustées en considérant la taille relative du cluster contenant le meilleur individu et celle du cluster contenant le pire individu.
- *Méthodes d'Adaptation* :
 - Règles simples : L'adaptation est basée sur des règles simples définies en fonction de l'observation de caractéristiques d'exécution des algorithmes.
 - Co-évolution : Les méthodes de co-évolution utilisent des mécanismes de génération de paramètres et mettent ces paramètres dans un processus évolutif afin d'optimiser les valeurs qu'ils doivent prendre. Les méthodes auto-adaptatives utilisent la co-évolution.
 - Contrôle de l'entropie : L'entropie est utilisée pour mesurer l'incertitude avec une variable aléatoire. Dans les MH, la génération de nouveaux individus contient souvent des facteurs aléatoires et la probabilité de générer une certaine solution peut être analysée en fonction des informations sur la population. Par conséquent, l'entropie peut être utilisée pour analyser la population et ajuster les paramètres et les opérateurs.
 - Contrôle par ensemble flou (au sens mathématique) : L'utilisation du contrôleur flou dans les MH est assez simple. Les fonctions d'appartenance sont utilisées pour évaluer l'état actuel et des règles peuvent alors être proposées en fonction de l'état considéré. Enfin, les règles seront utilisées pour contrôler les paramètres [173].

Approche Hybride : L'hybridation est une technique qui combine une MH avec d'autres techniques d'optimisation. Différents auteurs ont souligné la nécessité d'hybridation des MH avec d'autres algorithmes d'optimisation, avec des techniques d'apprentissage automatique et autres. Certaines des motivations possibles de l'hybridation sont l'amélioration des performances, l'amélioration de la qualité des solutions obtenues, l'incorporation des MH dans le cadre d'un système plus large. Lozano et al. [248] ont regroupé les MH hybrides en trois catégories, à savoir : (1) MH hybrides collaboratifs qui sont basés sur l'échange d'informations entre différents MH, (2) MH hybrides intégratifs qui utilisent un algorithme subordonné d'un autre algorithme. (3) MH

avec composants évolutifs E&E qui cherchent à améliorer les MH classiques en transformant des composants E&E spécifiques. Il convient de mentionner que les MH combinant des techniques de conception expérimentales telles que la conception orthogonale, la conception uniforme et la méthode de Taguchi, entre autres, sont une direction de recherche intéressante [288][413].

Approche Interactive : Une MH interactive (Interactive Metaheuristic - IMH) utilise une interface homme-machine en tant que composant d'une tâche d'optimisation. L'opinion humaine est donc une composante du processus et permet une combinaison des capacités de recherche de la MH avec les connaissances expertes d'utilisateurs. C'est un domaine de recherche actif qui est cependant limité par la fatigue des utilisateurs. En effet, des évaluations demandées en trop grand nombre peuvent submerger ces derniers.

2.2.3 Influence de l'E&E dans les EA

Introduction

L'E&E sont souvent mal comprises par les praticiens et chercheurs en EA. Eiben et Schippers [120] ont fourni une première discussion sur l'E&E dans les EA. Leurs travaux ont soulevé plusieurs questions et ont démontré la nécessité de poursuivre les recherches. Blum et Roli [50] ont fourni différentes classifications des MH, ils ont décrit le fonctionnement interne de différentes MH et ont fourni une vue unifiée sur la diversification et l'intensification. L'E&E par Blum et Roli [50] se réfèrent à des stratégies à court terme liées au hasard, tandis que la diversification et l'intensification se réfèrent à des stratégies à moyen et long terme basées sur l'utilisation de la mémoire. Une autre classification des MH hybrides a été introduite par Talbi [377]. Liu et al. [241] ont également classé les EA en approches mono-processus et multi-processus en fonction de la façon dont l'équilibre entre l'E&E est atteint.

E&E dans les EA

Une opinion commune [120] affirme que l'espace de recherche est exploré par des opérateurs de croisement / mutation et exploité par des opérateurs de sélection. Cette opinion est très discutable. En effet, il a été montré [28] que les opérateurs de sélection peuvent contrôler le niveau d'exploration ou d'exploitation en faisant varier la pression de sélection. Par ailleurs, un opérateur de mutation explore car il modifie aléatoirement les individus et augmente ainsi la diversité d'une population, mais il exploite aussi car il conserve la plupart du matériel génétique existant. Pour continuer, un opérateur de croisement exploite car il combine deux parents ou plus pour générer une possible meilleure progéniture, mais il explore car il génère aussi des individus très différents de ceux existant. Il est donc impossible de lier directement un opérateur de sélection/croisement/mutation à l'exploration/exploitation.

De plus, la taille de la population impacte aussi l'E&E. Il est par exemple possible de diriger un processus d'évolution vers l'exploration ou l'exploitation par le redimensionnement de la population [166][358]. Avec une taille de population plus grande, l'espace de recherche est davantage exploré qu'avec une taille de population plus petite. C'est un moyen plus simple de maintenir la diversité, mais c'est souvent une solution insatisfaisante. De plus, la taille de la population influence également d'autres opérateurs.

La représentation impacte aussi l'E&E [193]. Les opérateurs de mutation et de croisement qui influencent l'E&E dépendent principalement de la représentation des individus. Il est important de savoir à quels niveaux les opérateurs fonctionnent [356]. La relation entre la représentation d'un individu et l'équilibre entre E&E est toujours mal comprise.

Une autre question importante est de savoir comment l'équilibre entre l'E&E est atteint. Les opérateurs sont capables de trouver un équilibre entre l'E&E en passant par des réglages appropriés de paramètres de contrôle comme les taux de croisement et de mutation. Dans les cas les plus extrêmes, ces réglages peuvent faire basculer l'EA dans une recherche aléatoire ou dans une recherche proche de l'escalade (hill climbing). Dans la littérature, les approches suivantes ont été essayées [244] : (1) Procéder à des essais et erreurs, qui est une méthode longue et fastidieuse, généralement réalisée de manière ad hoc. (2) Suivre des directives générales [102][166][343], qui sont souvent inapplicables à des cas spécifiques [355][436]. (3) Utiliser des EA sans paramètres [166][29], qui sont des approches robustes mais pour la plupart moins efficaces. (4) Utiliser les expériences d'applications similaires précédentes. (5) Identifier les caractéristiques des paysages de fitness par un classifieur afin de proposer de bons réglages des paramètres de contrôle [54]. (6) Faire une analyse statistique des interactions entre les paramètres de contrôle et de leurs effets sur les performances des algorithmes [98]. (7) Utiliser des modèles mathématiques, ce qui est souvent trop simple pour être réaliste ou trop difficile à comprendre par les utilisateurs ordinaires. (8) Considérer la recherche des meilleurs paramètres de contrôle comme un problème d'optimisation. Les exemples les plus connus sont l'approche méta-évolutive (évolution de l'évolution) [157], l'algorithme de course [46], l'optimisation séquentielle des paramètres [35], la méta-estimation de l'algorithme de distribution [290], l'hyper-heuristique [295], l'optimisation multi-objectifs [112] et l'apprentissage par renforcement [284]. Les travaux dans cette catégorie ont été étudiés et classés en méthodes d'échantillonnage, méthodes basées sur des modèles, méthodes de dépistage et algorithmes méta-évolutifs [119].

Pour continuer, le réglage des paramètres de contrôle dépend du problème [355][436]. Cela signifie qu'un bon rapport entre l'E&E dépend du problème. Par ailleurs, différentes valeurs pour les paramètres de contrôle peuvent être optimales à différentes étapes d'un processus d'évolution [117]. Une méthode de contrôle des paramètres peut être déterministe, adaptative ou auto-adaptative [117]. La catégorie déterministe ajuste les paramètres par des règles déterministes. La catégorie adaptative utilise la rétroaction d'un processus d'évolution pour contrôler les directions et les amplitudes des paramètres. La catégorie auto-adaptative code des paramètres dans les individus et opère une mutation et une recombinaison (évolution simultanée de l'évolution).

Une croyance répandue est que les EA devraient commencer par l'exploration pour progressivement basculer vers l'exploitation [130]. C'est généralement correct, mais une telle politique a tendance à rencontrer des difficultés lors de la résolution de certains problèmes [241][419]. Par conséquent, des approches (auto-) adaptatives ont été proposées. Elles vont de l'utilisation d'informations simples telles que les valeurs de fitness [166], le nombre de générations sans amélioration [118], la diversité [394] et l'entropie [335], à des techniques plus avancées, telles que les opérateurs génétiques adaptatifs et la sélection [172], la représentation adaptative [407] et les fonctions de fitness adaptatives [255].

Afin d'obtenir un bon rapport entre l'E&E, deux approches ont été classées : mono-processus et multi-processus [241]. Pour les approches mono-processus, des opérateurs indépendants (sélection, mutation, croisement) sont à l'œuvre. Pour les approches multi-processus, la coordination de différents opérateurs est à l'œuvre [121][131].

Enfin, il est important de noter que les réglages des paramètres de contrôle permettent le contrôle explicite des opérateurs mais seulement implicite de l'E&E.

Contrôle de l'E&E

Quand contrôler l'E&E ? L'E&E peuvent être contrôlées avant l'exécution par des réglages appropriés des paramètres de contrôle. Cependant, les problèmes d'optimisation sont variés et

nécessitent différentes quantités d'E&E. Étant donné qu'un problème est inconnu, les capacités de recherche pourraient être améliorées si les quantités d'E&E étaient modifiées de manière dynamique (pendant l'exécution). D'autre part, le contrôle des paramètres présente des avantages conceptuels par rapport au réglage des paramètres [16][62][241][305], l'E&E doivent donc être contrôlées dynamiquement.

À quelle occasion contrôler l'E&E ? Deux schémas existent : (1) Schéma déterministe avec pour exemple un contrôle toutes les k générations [176]. (2) Schéma adaptatif avec pour exemple un contrôle lorsque la meilleure fitness n'a pas changé pendant plusieurs générations [118] ou encore lorsque la diversité d'une population chute sous un certain seuil [349][394]. Il est nécessaire de contrôler l'équilibre entre l'E&E à différentes étapes afin d'obtenir de meilleurs résultats ou afin d'atteindre la convergence plus rapidement.

Comment contrôler l'équilibre de l'E&E ? L'équilibre entre l'E&E est implicite, son contrôle direct est donc difficile. De plus, avant même de le contrôler, il faut savoir le mesurer, ce qui demeure une question ouverte [45]. L'idéal serait de pouvoir clairement identifier ces deux phases afin de proposer des mesures directes, mais la réalité ne le permet pas et ainsi ce sont des mesures indirectes d'E&E qui sont principalement utilisées.

Diversité : La diversité peut être mesurée à trois niveaux : (1) Génotype (structurel / syntaxique / génotypique) correspondant aux différences entre les génomes au sein d'une population. (2) Phénotype (comportemental / sémantique / phénotypique) correspondant aux différences entre les valeurs de fitness au sein d'une population. (3) Complexe ou composite, correspondant à une combinaison des deux cas précédents [65].

Il est largement admis qu'une grande diversité dans une population contribue à la performance d'un EA [281]. Il faut noter que la diversité n'est que grossièrement liée à l'E&E. Une grande diversité n'est pas nécessairement obtenue par un bon rapport entre l'E&E. Une population diversifiée peut être obtenue par simple exploration. De plus, une population diversifiée ne contient pas forcément d'individus intéressants au regard de la fitness [253]. Une population diversifiée est une condition préalable à l'exploration afin d'éviter une convergence prématurée vers des optima locaux. D'un autre côté, promouvoir la diversité à toutes les étapes d'un processus évolutif peut être contre-productif car certaines phases demandent une forte exploitation. La relation entre la diversité et l'E&E est encore mal comprise, en particulier pour identifier les types (phénotypiques / génotypiques) et les quantités de diversité nécessaires à différents stades du processus évolutif [65].

Mesures de la diversité génotypique : Basée sur la différence : de nombreuses mesures peuvent être classées dans ce groupe, depuis le comptage de différents génotypes [226] et le comptage de neurones activés différemment représentant des régions de recherche particulières [18], jusqu'au comptage des fréquences d'allèles [102][99]. McPhee et Hopper [273] ont proposé une mesure simple, étendue par la suite [66].

Basée sur la distance : c'est probablement le type de mesure de la diversité le plus largement utilisé de nos jours. Différentes distances sont prises en compte comme la distance de Hamming [408] [349] [390], la distance de Minkowski (la distance euclidienne [270] et la distance de Manhattan sont des cas particuliers), la distance cosinus de similitude [128], la distance de modification [101], la distance au point moyen [394].

Basée sur l'entropie : l'entropie est une mesure très succincte de la diversité qui gagne en popularité. Elle représente la quantité de désordre de la population, où l'augmentation de l'entropie représente une augmentation de la diversité. Un avantage supplémentaire par rapport aux approches précédentes est que la distribution des valeurs est également incluse dans cette mesure, bien que la distribution précise soit inconnue. L'entropie s'est avérée être une mesure utile de la diversité génotypique [233][240][265][282].

Basée sur la probabilité : l'indice de diversité de Simpson fait partie de ce groupe [265][304]. L'indice de diversité de Simpson mesure la probabilité que deux individus, choisis au hasard dans un échantillon, appartiennent à la même espèce.

Basée sur les ancêtres/l'histoire : la mesure de la diversité est obtenue en comparant la population actuelle avec les populations des générations précédentes [273].

Mesures de la diversité phénotypique : Basée sur la différence : la mesure de diversité la plus simple basée sur la différence est le nombre de phénotypes différents [335]. D'autres contributions existent [192][250] [390].

Basée sur la distance : diverses mesures de distance, comme la distance euclidienne, peuvent être utilisées pour trouver des similitudes entre les individus au sein d'une population. Par exemple, la distance moyenne par rapport à d'autres individus au sein d'une population ou la distance moyenne entre les solutions non dominées et le vrai front de Pareto [437] ou encore la distance entre un individu et un individu non dominé [73]. Adra et Fleming [14] ont également introduit un indicateur de diversité. Un autre exemple a été présenté dans [394], où la distance au point moyen a été utilisée.

Basée sur l'entropie : l'entropie comme mesure de la diversité a été proposée pour la première fois par Rosca [335] et depuis lors, elle a été principalement utilisée comme mesure succincte de la diversité phénotypique [65][241].

Basée sur la probabilité : la mesure de la diversité de Simpson peut également être appliquée à la diversité phénotypique [304].

Approches pour équilibrer l'E&E

Une population diversifiée est une condition préalable à un bon équilibre entre E&E plutôt qu'une garantie. Cet équilibre peut également être atteint par d'autres moyens comme la fitness. Nous classons ci-après les approches concernant l'équilibre entre E&E.

Maintenance de la diversité : Il est difficile de déterminer une quantité utile de diversité. Une grande diversité est nécessaire pour échapper aux optima locaux, tandis qu'une faible diversité est nécessaire pour affiner les solutions. De nombreuses méthodes différentes, développées principalement dans les années 90, ont été utilisées pour le maintien de la diversité. L'étude de Mahfoud a été l'une des premières à élaborer une théorie globale de la diversité pour les GA [253], où les méthodes étaient classées comme non-niching et niching [253] [340]. Cette classification apparaît également dans d'autres travaux [387][438].

- Méthodes générales :

Basée sur la population : la diversité est maintenue par variation de la taille de la population, par élimination des doublons, par des techniques d'infusion, par les archives externes ou par la migration entre des sous-populations.

Basée sur la sélection : la diversité est maintenue en modifiant la pression de sélection ou les restrictions de remplacement.

Basée sur les croisements / mutations : la diversité est maintenue par des restrictions d'accouplement ou des opérateurs perturbateurs.

Hybride : la diversité est maintenue par des combinaisons de techniques sus-citées ou d'autres approches spécifiques.

- Méthodes non-niching :

Augmentation de la taille de la population : c'est l'une des approches les plus simples, mais plusieurs études [273] ont rapporté que l'augmentation de la taille de la population ne conduit pas toujours à une diversité accrue.

Élimination des doublons : en utilisant cette approche, un individu qui existe déjà dans une population est éliminé et un autre individu généré aléatoirement est inséré [73] [267].

Techniques d'infusion : de nouveaux individus sont insérés au hasard après un certain nombre de générations ou des techniques d'initialisation spéciales sont utilisées [234] [230] [155] [224] [156] [220] [325] [219] [322] [198] [150] [404]. Si le remplacement est fait en réponse à un manque de progrès ou à un changement dans un environnement dynamique, alors ces techniques peuvent être classées comme contrôle de diversité.

Archives externes : alors que les archives ont été utilisées principalement pour garder une trace de solutions non spécialisées [438] [437] [58], il y a également eu d'autres propositions d'utilisation d'archives externes lors de l'optimisation à objectif unique [426].

Migration entre sous-populations : des échanges d'individus sont effectués entre sous-populations. Les migrants peuvent par exemple remplacer les individus moins aptes, les individus choisis au hasard ou encore les individus les plus similaires. L'idée principale derrière cette approche est que différentes sous-populations peuvent maintenir différentes régions prometteuses de l'espace de recherche et donc maintenir la diversité [261]. La dynamique de l'E&E est en outre influencée par le nombre de sous-populations, la taille de ces sous-populations, les différences d'interconnexions entre les sous-populations, la fréquence de communication et la qualité des migrants [395] [20].

Modification de la pression de sélection : de nombreuses techniques ont été proposées pour éviter que la sélection ne soit biaisée en faveur d'individus très aptes. Deux des techniques les plus simples sont la sélection par classement [281] et la mise à l'échelle [157]. Certains mécanismes de sélection plus sophistiqués qui tentent de garantir que l'individu le plus apte ne sera pas toujours sélectionné et que l'individu le plus faible ne sera pas toujours rejeté sont brièvement décrits dans [266] [192] [78] [139].

Restrictions de remplacement : [102] a introduit le surpeuplement standard, pour maintenir la diversité, chaque individu nouvellement généré remplace l'individu existant le plus similaire d'une sous-population aléatoire. [247] a proposé un schéma de remplacement hybride où les individus faibles et contribuant peu à la diversité sont remplacés, favorisant ainsi l'exploitation (individus très aptes) et l'exploration (forte contribution à la diversité).

Restrictions d'accouplement : les individus sont autorisés à s'accoupler s'ils remplissent des conditions spéciales. Les exemples sont la prévention de l'inceste [122], les genres [334], l'accouplement distinct [105].

Opérateurs perturbateurs : par exemple, l'opérateur d'hyper-mutation [121] [87] [200]. Cependant, De Jong et Spears [104] ont souligné que la perturbation n'implique pas nécessairement une exploration utile.

Approches hybrides : concepts d'âge [141] [304], GA compacte [165] [227] et autres combinant différentes approches [257] [321].

- Méthodes niching :

Basée sur la fitness : partage de fitness (explicite [147] [188] et implicite [357]), compensation [315], compensation modifiée [354], regroupement [421].

Basé sur le remplacement : surpeuplement déterministe [253], surpeuplement probabiliste [277], sélection de tournois restreinte [164].

Basé sur la préservation : GA conservant les espèces [232].

Hybride : GA basé sur une population élitiste adaptative [236], ensemble d'algorithmes de niching [422].

Bien que la plupart des travaux réalisés en utilisant des méthodes non-niching et niching s'attendent simplement à une augmentation de la diversité et donc à un meilleur équilibre entre l'E&E, toute diversification n'est pas utile. À cet égard, les méthodes de niching peuvent être

considérées comme des méthodes de diversification plus fortes [134]. Il n'existe que quelques travaux théoriques sur les mécanismes de diversité [104] [133] [370] [134].

Contrôle de la diversité : La principale différence entre le maintien de la diversité et le contrôle de la diversité est que dans ce dernier cas, la diversité de la population, la fitness individuelle et/ou les améliorations de la fitness sont mesurées et utilisées comme rétroaction pour orienter un processus d'évolution vers l'exploration ou l'exploitation.

Diversité contrôlée et préservée par la sélection : La probabilité de survie peut être calculée sur la base de la diversité de la population, ou la diversité peut être incluse dans les fonctions de fitness qui orientent davantage le processus de sélection. Plusieurs approches entrent dans cette catégorie [286] [349] [73] [411] [41] [270] [14] [58].

Diversité contrôlée par le croisement et la mutation : L'idée est d'augmenter ou diminuer la probabilité de croisement et/ou de mutation après le calcul de la diversité de la population, de la fitness et/ou des améliorations de la fitness. Les approches diffèrent les unes des autres principalement sur la façon dont la diversité est calculée : explicitement par différentes mesures de diversité et/ou implicitement par la fitness ou les améliorations de la fitness [408] [366] [411] [202] [233] [270].

Diversité contrôlée par une population changeante : Après avoir mesuré la diversité de la population, la taille de la population ou la population seule est modifiée. Peu d'approches entrent dans cette catégorie [241] [335] [389].

Apprentissage de la diversité : La principale différence entre le contrôle de la diversité et l'apprentissage de la diversité est que, dans le premier cas, l'historique à court terme (par exemple, la population actuelle) est souvent utilisé pendant le calcul de la diversité, tandis que, dans le dernier cas, l'historique à long terme est utilisé en combinaison avec différentes techniques d'apprentissage automatique pour apprendre des domaines de recherche (non) explorés. Les approches d'apprentissage de la diversité les plus courantes sont (1) l'apprentissage culturel [97] [38] [365] [199], (2) les cartes auto-organisées [18] [216], (3) les arbres de partitionnement d'espace binaire [423] [86], (4) l'estimation de la distribution [289].

Autres approches directes : Les approches directes agissent comme s'il était possible de clairement différencier l'exploration de l'exploitation. Il faut donc garder à l'esprit que les phases d'E&E sont toujours imbriquées, par exemple, même si le processus est en phase d'exploration, un individu peut être généré en exploitant une région connue.

Trois catégories d'autres approches directes existent. La première utilise des sous-populations pour délimiter l'exploration de l'exploitation et certaines sous-populations sont utilisées pour une phase particulière (GA bi-population (bGA) [391], Forking GA [390], GA à décalage d'équilibre (Shifting Balance Genetic Algorithm - SBGA) [301], séparation aléatoire de la population en deux sexes [145]). La seconde utilise des déclencheurs pour provoquer une alternance de phases d'E&E (EA avec recherche locale également appelés EA hybrides [52] [237] [264] [6] [283], GA à recherche locale [279] ou algorithmes mémétiques [287] [222] [300] ; algorithme évolutif guidé par la diversité (Diversity Guided Evolutionary Algorithm - DGEA) [394] ; contrôle adaptatif de la topologie dans les GA cellulaires [16]). La troisième utilise des arbres d'ascendance pour mesurer explicitement l'E&E [93].

Conclusion et directions de recherche

Nombres de points abordés par [94] sont généralisables à d'autres domaines et en particulier au domaine de la MH. Les concepts liés à l'E&E ne sont pas bien compris, menant les recherches ainsi que les applications à prendre des raccourcis pour les aborder. Il reste donc encore de nombreux problèmes ouverts présentés ci-après.

-
- Définir formellement/mathématiquement les phases d'E&E ainsi que les transitions de l'une à l'autre.
 - Proposer des métriques pour la mesure directe de l'E&E. Actuellement, des mesures indirectes d'E&E sont principalement utilisées. Le contrôle de l'E&E en direct (pendant l'exécution) est difficile à réaliser. En proposant de nouvelles mesures directes d'E&E, il serait possible d'assigner un pouvoir d'E&E et ainsi de disposer d'une base de comparaison entre les algorithmes et/ou entre les opérateurs de variations. Ces comparaisons pourrait prendre place au sein de l'entièreté du spectre d'un problème donné. Ces mesures permettraient aussi de mieux contrôler l'E&E afin d'atteindre et de conserver un meilleur équilibre au cours d'une course évolutive. Il faudrait ensuite se demander si ces mesures directes d'E&E pourraient encore améliorer les performances des EA les plus compétitifs.
 - Définir formellement/mathématiquement l'équation d'équilibre de l'E&E, ce qui permettrait de mettre en lumière les éléments permettant d'atteindre l'équilibre.
 - Définir comment différents opérateurs (sélection, croisement, mutation) contribuent à l'équilibre dans le cadre d'approches diverses.
 - Définir les types d'opérateurs qui obtiennent de manière synergique les meilleurs résultats en fonction de l'approche.
 - Définir comment les propriétés de l'espace de recherche influencent l'E&E.
 - Définir comment le réglage des paramètres de contrôle influence l'E&E.
 - Définir des éléments d'explication de l'E&E pour divers acteurs et donc ayant des niveaux de complexité différents.
 - Définir le moment où l'équilibre doit être contrôlé.
 - Définir comment l'équilibre doit être contrôlé.
 - Réaliser des analyses sur différents paramètres de contrôle de l'E&E.
 - Réaliser des analyses sur différentes approches de maintien de la diversité, contrôle de la diversité et apprentissage de la diversité.

2.2.4 Influence de l'E&E en Psycho-sociologie

Introduction

De nombreuses décisions dans la vie des êtres vivants nécessitent un équilibre entre l'exploration de différentes options et l'exploitation de leurs récompenses [275]. L'exploitation consiste à tirer les meilleures récompenses de l'endroit où l'on se trouve (échelle locale) tandis que l'exploration consiste à considérer d'autres options (échelle globale) afin d'éventuellement obtenir de meilleures récompenses. Cet équilibre est communément appelé le compromis entre l'E&E. Il figure dans un large éventail de domaines de recherche comme l'apprentissage ou l'optimisation. Les définitions et processus entourant le comportement d'E&E sont souvent beaucoup plus complexes que ce qui vient d'être présenté. A ce titre, les compromis entre E&E sont considérés comme l'un des défis les plus fondamentaux dans notre compréhension du contrôle et du comportement adaptatifs [88].

L'analyse théorique des compromis entre E&E est compliquée à plusieurs égards. Premièrement, les concepts d'E&E et de compromis sont utilisés dans un large éventail de littératures impliquant différentes terminologies, méthodologies et perspectives. Cela rend difficile la synthèse des connaissances existantes en une vision cohérente et motive des désaccords dans les conclusions sur l'essence des comportements d'E&E. Deuxièmement, les compromis entre E&E peuvent dépendre d'un grand nombre de facteurs environnementaux, individuels et sociaux. La littérature documentant ces différents facteurs est une fois de plus abondante [88] [159]. Troisième-

mement, les comportements et compromis d'E&E ne sont pas toujours clairement identifiables. En effet, ils peuvent être décrits et considérés à différentes échelles spatiales et temporelles, ainsi que le long de différents continuum. Par conséquent, des comportements exploratoires à un niveau d'analyse peuvent être exploitants à un autre niveau et au sein d'un même niveau d'analyse les comportements peuvent avoir des composantes exploratoires et exploitantes.

De nombreuses approches de l'analyse du comportement décisionnel caractérisent différents scénarios de vie comme des représentations d'un compromis entre l'E&E : l'apprentissage par renforcement (Reinforcement Learning - RL) [88], la recherche de nourriture [90], le choix binaire risqué [151], l'apprentissage organisationnel [159]. Pour une revue, voir [186].

La ressource [275] intègre certains résultats de la littérature afin de mieux comprendre ce qu'est le compromis entre l'E&E, sous quelles formes il peut prendre place et propose de nouvelles perspectives théoriques. Plus précisément, cette ressource explore comment le compromis dépend de la conceptualisation de l'E&E, des facteurs environnementaux, sociaux et individuels, de l'échelle à laquelle l'E&E sont considérées, de la relation et des types de transitions entre les deux comportements, des objectifs du décideur. La ressource [275] illustre les complexités entourant ces concepts et met en évidence trois éléments nécessaires à une unification de la recherche en E&E : un continuum E&E, différents types de transitions entre ces deux états et le rôle des objectifs des agents.

Synthèse de l'état de l'art

Notions et définitions de l'E&E : Les définitions de l'E&E diffèrent sur au moins trois dimensions : les modèles comportementaux de l'agent, les valeurs et l'incertitude des choix, et les résultats obtenus à partir d'un choix [385]. Ces trois dimensions peuvent être mises en correspondance avec ce que fait un agent de recherche, avec les éléments sur lesquels l'agent fonde ses décisions et avec ce que l'agent retire de la recherche. Ces dimensions permettent l'analyse et la compréhension du comportement d'E&E ainsi que toutes les conclusions sur leurs compromis.

Les modèles comportementaux des agents représentent la dimension la plus couramment utilisée pour définir l'E&E [221] [293] [185] [151]. En général, un comportement est interprété comme exploratoire s'il n'est pas ciblé et s'il est variable dans le temps, et comme exploitant s'il est ciblé et stable dans le temps. Cependant, cette distinction n'est pas toujours claire car elle peut aussi dépendre des échelles spatiales et temporelles d'observation.

Les valeurs des choix et l'incertitude associée à la connaissance de ces valeurs sont principalement utilisées pour définir l'E&E dans la littérature RL. Dans certains modèles RL classiques, l'exploitation est définie comme le choix de l'option qui a la valeur subjective la plus élevée et l'exploration est définie comme le choix de toute autre option au hasard [374]. D'autres modèles de RL ont souligné l'importance de l'incertitude pour définir l'E&E [229] [23] [223].

Enfin, l'E&E ont été discutées relativement aux informations et récompenses obtenues par l'agent de recherche. Dans de nombreux domaines de recherche, l'exploration est supposée fournir à l'agent la possibilité d'apprendre et d'obtenir des informations, tandis que l'exploitation est supposée fournir des résultats explicites tels que des récompenses caloriques ou monétaires [88] [90] [180]. Cependant, la distinction entre information et récompense n'est pas toujours simple car les récompenses peuvent aussi fournir des informations sur la qualité de l'option sélectionnée [159] et même lorsqu'aucune récompense matérielle n'est obtenue, la recherche d'informations en elle-même peut être gratifiante [106] [152] [274].

Ainsi, les trois dimensions ne s'excluent pas et la plupart des conceptualisations de l'E&E reposent sur plusieurs d'entre elles. Cependant, les contributions respectives de ces dimensions

à une conceptualisation donnée ne sont pas toujours claires et cela peut être problématique si les dimensions considérées conduisent à des conclusions opposées [151] [152] [180] [181]. Il faut donc faire un examen de la façon dont une recherche particulière s'intègre dans une séquence de recherches au fil du temps, où les récompenses d'une recherche peuvent devenir des informations pour guider les recherches ultérieures et donc influencer ce qui est l'exploration par rapport à l'exploitation dans un modèle comportemental spécifique.

En plus d'affecter notre compréhension des comportements d'E&E, les hypothèses sur les dimensions sous-jacentes peuvent également influencer la nature du compromis considéré. Par exemple, une distinction entre l'E&E basée sur des modèles de comportement met en évidence les coûts et les avantages de rester sur une ressource par rapport au passage à une autre [75], tandis qu'une distinction basée sur les valeurs et l'incertitude des options de choix peut mettre l'accent sur la valeur ou le risque [229] [392]. En termes de résultats obtenus, un compromis E&E peut être très similaire à un compromis entre vitesse et précision [53] [84], ou précision et effort [311]. C'est le cas parce qu'une exploration plus longue demande plus de temps et d'efforts, mais conduit généralement à plus d'informations et, par conséquent, à de meilleurs choix. Cependant, voir [126] [142] [174] [228] pour une discussion sur l'effet less-is-more abordant les cas où disposer de moins d'informations peut être avantageux et, par conséquent, où aucun compromis ne peut exister.

Facteurs environnementaux, individuels et sociaux : La littérature nous indique au moins trois types de facteurs qui influencent les comportements et les compromis d'E&E : les facteurs environnementaux, individuels et sociaux.

Les facteurs environnementaux correspondent à des coûts et bénéfices associés à l'exploration et à l'exploitation, en termes d'informations et de ressources, ainsi qu'à des coûts de basculement entre les deux [398]. Par exemple, l'exploration est pertinente lorsque les ressources de l'option actuellement sélectionnée sont épuisées ou lorsque l'agent dispose d'informations insuffisantes sur l'état des options alternatives, augmentant ainsi la valeur de la recherche d'informations [75]. La prise en compte des antécédents d'exploration [183] peut aussi jouer un rôle dans l'appréciation des facteurs environnementaux.

Les facteurs individuels correspondent à des états propres à l'agent et touchent principalement aux aptitudes, aux limitations, à l'expérience ainsi qu'à de nombreux états cognitifs. Certains de ces facteurs sont dynamiques et peuvent changer entre différentes situations, d'autres sont plus stables [344].

Les facteurs sociaux correspondent à diverses informations que l'agent perçoit des autres agents avec qui il évolue. Par exemple, les compromis entre E&E peuvent être différents lorsqu'ils sont considérés au niveau du groupe plutôt qu'au niveau individuel [159] [90]. Deux facteurs clés sont la concurrence et l'information sociale. En général, la concurrence augmente les coûts d'exploration car elle crée le risque que d'autres agents exploitent la récompense avant l'agent [149] [384]. L'exploration peut être particulièrement utile dans de telles situations concurrentielles, car elle peut aider l'agent à trouver des options que les concurrents n'exploitent pas ou ne peuvent pas exploiter. La disponibilité de l'information sociale, en revanche, tend à diminuer les coûts d'exploration car elle permet à l'agent d'apprendre des décisions et des performances des autres et réduit ainsi le besoin de s'engager dans un comportement exploratoire risqué [396].

Enfin, les facteurs de différents types interagissent les uns avec les autres de multiples manières [184] [127].

Échelles spatiales et temporelles de l'E&E : Les échelles permettent la classification des comportements. En effet, les agents font des choix à différentes échelles spatiales et temporelles. Alors que certains compromis peuvent changer en fonction de l'échelle considérée, d'autres peuvent évoluer entre différents niveaux. Ainsi, les échelles et leurs niveaux affectent l'interpré-

tation des situations d'E&E [88] ce qui est cohérent avec l'idée de continuum [275].

Des exemples de la pertinence de l'échelle spatiale pour l'appréciation des transitions entre l'E&E sont : (1) la recherche de nourriture, où les agents prennent des décisions portant sur des parcelles de ressources [75] [369], (2) la mémoire humaine [182], (3) la sélection de l'habitat [268].

Des exemples de la pertinence de l'échelle temporelle pour l'appréciation des transitions entre l'E&E sont : (1) les prises de décisions rapides en environnement restreint [402], (2) la capacité d'apprentissage de l'agent [146] [107] [339] [191], (3) le problème à horizon infini [345] [191].

Par ailleurs, les compromis peuvent disparaître à certaines échelles car un agent peut les éviter [19] [25] [177] [314] [109] [249]. Cela n'empêche pas un compromis de réapparaître sur d'autres niveaux d'échelle, ce qui peut symboliser le fait que l'agent diffère la prise de décision afin d'obtenir plus d'informations. Des exemples d'évitement du compromis sont : (1) Le biais de statu quo [338], le biais d'omission [331] et l'heuristique par défaut [201] qui se réfèrent à des situations dans lesquelles les gens préfèrent des options qui ne provoquent aucun changement ou qui ne nécessitent aucune action manifeste [177] [178] [21] [25] [381]. (2) L'inertie de l'inaction [393] faisant référence à la tendance à omettre l'action lorsqu'une opportunité similaire et plus attrayante a été abandonnée. (3) La spécialisation de certains agents en exploration ou en exploitation afin de réduire les compromis E&E au niveau du groupe [90] [159].

Vers une théorie unificatrice des compromis entre E&E

Un Continuum E&E : L'E&E ne représentent pas nécessairement des comportements distincts nécessitant de faire des compromis, et il n'est pas nécessairement préférable de les considérer comme des choix explicites. Alors que de nombreuses conceptualisations du compromis supposent une distinction binaire, certaines revues soutiennent que l'E&E pourraient être mieux comprises comme les points finaux d'un continuum [88] [159]. Comme le montre [275], les extrémités de ce continuum correspondent à de l'exploration pure et à de l'exploitation pure. Entre les extrémités, tout un spectre de mélange entre E&E se retrouve, un agent peut ainsi s'engager à la fois dans l'E&E à chaque instant donné. La capacité de juger où se situe une action sur ce continuum pourrait être améliorée en considérant plus d'une dimension conceptuelle [30].

Transitions entre E&E : Deux types de transitions sont le plus souvent abordés dans la littérature : (1) la transition de l'exploration à l'exploitation [319] [135] [175] [125], (2) la transition de l'exploitation à l'exploration [88] [221] [182] [136] [317]. Dans la première transition, l'agent est confronté à la question de savoir combien de temps il doit continuer l'exploration et ainsi augmenter la quantité d'informations obtenues, et quand passer à l'exploitation et ainsi augmenter les chances d'obtenir des récompenses réelles. Des exemples sont : le satisficing [353] [135], les modèles heuristiques séquentiels d'arrêt optimal [345], le choix mutuel de partenaire [386], les modèles d'arrêt facultatif bayésiens [115], les modèles d'observateurs bayésiens [402], l'accumulation de preuves à un critère de seuil [67] [327] [400], et les modèles basés sur des architectures cognitives pour la prise de décision dynamique [151] [153]. Dans la deuxième transition, l'agent est confronté à la question de savoir combien de temps il doit continuer à exploiter une option actuelle et ainsi obtenir ses récompenses, et quand passer à l'exploration d'alternatives et augmenter ainsi ses chances de trouver de meilleures options ailleurs. Le modèle le plus important de ce compromis est sans doute le théorème de la valeur marginale [75], qui a longtemps tenté de formaliser un comportement optimal [369] [318] [317] [182] [409]. Une troisième variante de la transition a été discutée pour les situations sans direction claire. Par exemple, [259] décrit les compromis E&E dans le contexte organisationnel, qui nécessitent un équilibre constant entre E&E. Un modèle notable d'un tel équilibre est fourni par l'indice de Gittins [143] qui prédit si

un décideur doit exploiter ou explorer à chaque instant.

Par ailleurs, une transition peut être directe ou graduelle. Une transition directe correspond au passage brutal de l'exploration à l'exploitation et vice-versa. Une transition graduelle correspond au passage par des situations intermédiaires mélangeant E&E [40] [179] [374] [151].

Le rôle des objectifs des agents : En général, les objectifs pendant l'exploration sont axés sur l'obtention d'informations et donc sur la réduction de l'incertitude, tandis que les objectifs pendant l'exploitation sont axés sur l'obtention de récompenses [88] [152] [159] [185] [324] [374]. Les objectifs des agents peuvent affecter leur choix ainsi que la manière dont ils traitent les compromis possibles [154] [205] [206] [53] [84]. En retour, les résultats de ces choix peuvent affecter les objectifs et les comportements ultérieurs. Les objectifs peuvent aussi affecter l'interprétation d'un observateur du comportement d'E&E, et donc les conclusions qui sont tirées sur le compromis. Certains chercheurs ont suggéré que les comportements orientés vers un objectif pendant l'exploration peuvent reposer sur des mécanismes similaires à ceux requis pour l'exploitation [88] [151].

Conclusion

D'après [275], l'idée d'un compromis unique entre l'E&E représente une simplification exagérée. Au lieu de cela, il semble y avoir une famille de compromis [89] couvrant différents concepts et échelles, potentiellement traités par différents types de mécanismes. Ainsi, l'E&E sont mieux conceptualisées comme des points sur un continuum et sont omniprésentes à de nombreux niveaux d'abstraction. Elles se produisent à de multiples échelles dans l'espace et dans le temps. La prise de décisions quant au comportement à adopter sur ces différents niveaux peut reposer sur une heuristique connue ou nouvelle. La manière de choisir l'heuristique a donc son importance. La ressource [275] propose de nombreuses ouvertures présentées ci-après.

- Proposer une théorie unificatrice afin de couvrir les dimensions conceptuelles de l'E&E, y compris leurs interactions, contradictions et implications dans le cadre de l'explicitation des compromis. Cette théorie devrait être en mesure de rendre compte du fait que l'E&E peuvent exister sur un continuum.
- Capter les interactions entre les différents niveaux d'abstraction sur lesquels l'E&E se situent [376] et proposer une hiérarchisation des mécanismes de compromis à différentes échelles et à différents niveaux d'abstraction.
- Prendre en compte les effets sur l'E&E des objectifs des agents pour mieux évaluer les coûts et récompenses associés à l'E&E et ainsi atteindre les meilleurs compromis possibles en fonction des situations.
- Proposer une théorie unificatrice rendant compte de la variété des transitions qui peuvent se produire entre l'E&E. De telles transitions pourraient ouvrir de nouvelles perspectives sur le compromis.
- Explorer dans quelle mesure les mécanismes de compromis sont sensibles aux facteurs environnementaux, sociaux et individuels.

2.2.5 Conclusion Générale

Nous pouvons conclure que beaucoup d'études sur les compromis entre l'E&E ont été publiées, sans pour autant déboucher sur une réponse définitive [414]. La littérature se demande donc toujours comment séparer l'exploration de l'exploitation et comment ces deux notions coopèrent.

Le fait qu'il y ait un compromis à faire entre E&E n'implique pas de ne pas pouvoir poursuivre les deux activités, il est cependant plus ou moins facile de faire ce compromis. Il y a aussi un

manque de clarté dans ce qu'y est entendu exactement par équilibre entre E&E (quantité, ampleur ou autres).

Les problématiques liées à l'E&E ont été de plus en plus prises en compte au cours du temps et demeurent encore aujourd'hui très pertinentes [94]. Ainsi, toute contribution proposant une nouvelle manière d'explorer et d'exploiter l'espace de recherche est donc potentiellement pertinente. Il en va de même pour les contributions visant à expliquer, mesurer, contrôler, positionner et influencer l'E&E.

2.3 Réduction des données

Nous présentons dans cette section notre état de l'art concernant la réduction des données.

2.3.1 Introduction

Cet état de l'art repose sur la synthèse de plusieurs revues [254, 372, 416, 7, 285, 329] traitant de diverses méthodes d'échantillonnage et plus généralement de diverses méthodes de réduction des données dans de multiples domaines d'application.

A l'ère du Big Data, les volumes et complexités des données sont écrasants. Beaucoup de données sont de mauvaise qualité, à savoir, pas/peu pertinentes, redondantes, incomplètes, incohérentes et bruitées [49]. Cela influence les algorithmes de recherche en impactant leurs temps d'exécution, leurs capacités de généralisation et la précision de leurs prédictions. Les technologies traditionnelles de traitement sont ainsi inadaptées [70, 33, 81]. Dans ce contexte, les méthodes de prétraitement des données [137] [138], comme la réduction des données [306], sont fondamentales pour accélérer les algorithmes et potentiellement améliorer la qualité de leurs résultats d'analyse. La réduction des données est la transformation de l'information en une forme corrigée, ordonnée et simplifiée qui tente de conserver les informations les plus significatives. La réduction des données permet d'obtenir une amélioration du temps de calcul, de la précision, de l'adaptabilité et des capacités de généralisation des modèles d'apprentissage ainsi qu'une amélioration de l'explicabilité et de l'interprétabilité des résultats des algorithmes de recherche [138].

2.3.2 Méthodes de réduction des données

Nous allons présenter les différentes méthodes de réduction des données.

Échantillonnage et Rééchantillonnage dynamique

L'échantillonnage (Sampling) [254, 372, 416, 7, 285, 329] et le rééchantillonnage dynamique (Dynamic resampling) [359] [375] [238] [350] [440] [351] [352] [72] [292] sont des méthodes de réduction verticale (volume) qui consistent à extraire des données du jeu de données d'origine pour en créer un plus petit qui doit en être une version suffisamment représentative. Ces méthodes sont aisées à implémenter, conservent une bonne représentativité des données et en simplifient l'analyse. Cependant, ces méthodes peuvent provoquer des erreurs d'échantillonnage menant à une mauvaise représentativité et peuvent faire intervenir des biais statistiques abordés dans la littérature [323] [15] [170] [424]. Les domaines applicatifs sont variés [151] [254] [359] [350].

Suppression des redondances et des duplications

Cette méthode consiste en la suppression des éléments (ex : lignes) présents en plusieurs exemplaires ainsi que, pour chaque élément, des éléments lui étant trop similaires [263] [433], à

noter ici que la définition même de la similarité peut constituer un défi. Cela renforce la représentativité globale des occurrences, rend les données plus claires à interpréter et permet d'éviter certains biais statistiques. Cependant, il n'est pas souhaitable de supprimer les redondances et duplications dans tous les contextes en fonction des objectifs. Il peut aussi être difficile de poser les seuils de similarité permettant de prendre la décision de supprimer. Ces derniers points peuvent mener à des biais statistiques. La suppression des redondances et duplications concerne la quasi-totalité des domaines applicatifs, la recommandation en est un exemple [254] [416] [7] [329].

Compression

Il s'agit d'une réduction verticale (volume) par transformation d'une suite de bits en une suite de bits plus courte pouvant restituer les mêmes informations ou des informations proches [161] [12] [68] [4] [196]. Il existe deux types de compression, avec perte et sans perte. Une compression est dite sans perte si les données post-décompression sont identiques aux données d'origine, et avec perte si les données post-décompression sont différentes des données d'origine. Une compression sans perte peut être appliquée à tous types de données et vise à supprimer des répétitions dans les données tout en créant des informations permettant de retrouver les répétitions retirées. Les données sont donc les mêmes mais sont enregistrées avec plus de concision. Une compression avec perte est appliquée à des données perceptibles (images, sons, vidéos) et vise à supprimer des informations là où certains sens (vue, ouïe) les perçoivent peu. Par exemple, supprimer des informations dans les zones de contrastes qui sont peu distinguées par l'œil humain.

La compression réduit les coûts associés au mouvement des données et est un bon compromis entre la conservation de la qualité des données et l'efficacité des traitements. Cependant, la compression n'est pas adaptée à tout type de données, notamment en raison de la problématique de l'incompressibilité. Un autre exemple de domaine applicatif est celui des graphes de données.

Réduction de la vitesse

Cette méthode consiste en la réduction de la rapidité à laquelle les données affluent, c'est-à-dire en la réduction de la fréquence à laquelle elles sont générées, capturées et partagées [27] [63] [252]. Cela permet aux systèmes de mieux supporter les flux importants de données et les traitements associés mais cela peut provoquer des pertes d'informations à court/moyen/long termes ainsi que des décalages entre l'arrivée des informations et leur utilisation effective. Les domaines applicatifs sont ceux des systèmes devant absorber, traiter et restituer une importante quantité de données au cours du temps et souvent en temps réel, comme les plateformes web modernes ayant un important trafic et du contenu adaptatif.

Extraction de structures topologiques

L'analyse de données topologiques (Topological Data Analysis - TDA) vise à trouver des structures dans les données [360]. Il est possible de réduire les données en respectant ou conservant plus ou moins les structures topologiques extraites. Les méthodes associées comprennent le clustering, l'estimation de diversité, la réduction de dimension non linéaire, l'estimation de mode, l'estimation de crête et l'homologie persistante. Il faut choisir rigoureusement les méthodes d'extraction des structures topologiques en fonction des contextes. Un exemple de domaine applicatif est celui de l'analyse des données scientifiques.

Réduction de la dimensionnalité

Ce sont des méthodes de réduction horizontale consistant à sélectionner un ensemble de colonnes ou champs ou variables considérés comme représentatifs et qui sont donc conservés [303] [129] [195] [243] [294] [48] [262] [348] [342] [364] [346] [22] [17] [123]. Les autres sont supprimés. Ces méthodes consistent aussi en la création de nouvelles variables plus pertinentes sur la base de celles déjà présentes dans le jeu de données d'origine. Ces méthodes (1) mènent à une bonne représentativité, (2) aident à la suppression des redondances, (3) rendent la visualisation des données plus aisée, (4) réduisent la complexité spatiale, (5) aident à réduire le bruit dans les données et les rendent ainsi plus facilement interprétables, (6) atténuent le problème bien connu de “curse of dimensionality” et (7) améliorent, par la suppression des multi-colinéarités, l'interprétation des paramètres du modèle d'apprentissage. Cependant, ces méthodes peuvent (1) faire apparaître des biais statistiques, (2) écarter certaines données intéressantes, (3) trouver des corrélations entre les données qui ne sont pas toujours souhaitables et (4) donner des résultats décevants lorsque la moyenne et la covariance ne sont pas suffisantes pour caractériser le jeu de données. Les domaines applicatifs sont variés, ces méthodes sont courantes en apprentissage automatique.

Optimisation à l'intérieur des réseaux

Il s'agit de l'optimisation du stockage ou du déplacement des données à l'intérieur des réseaux [307] permettant de gérer des problématiques spécifiques aux réseaux et aux 6V du big data. Ces méthodes peuvent être complexes à mettre en place et à maintenir et peuvent être liées à de la suppression définitive de données. Les domaines applicatifs sont liés à la gestion des données sur les réseaux comme dans le cadre de l'internet des objets.

2.3.3 Focus sur l'échantillonnage

Pour rappel, les données utilisées dans le cadre d'un processus de recherche doivent le plus souvent être diagnostiquées et transformées afin de s'assurer de leur qualité (redondances, bruits, incohérences, incomplétudes), de réduire leur taille et leur complexité [329]. L'objectif est d'améliorer les performances des traitements ultérieurs et la qualité des solutions trouvées. Les éventuels échantillons finaux obtenus doivent pouvoir être analysés rapidement et avec un bon niveau de complétude dans le cadre d'un processus de recherche [92] [11]. A ces objectifs communs nous ajoutons notre objectif d'influencer indirectement le comportement d'E&E en proposant un espace de recherche transformé.

Dans le cadre de notre problématique de recommandation de ressources par des processus de recherche : (1) Nous utilisons la réduction horizontale par la sélection d'attributs car, dans la plupart des jeux de données, exploiter l'ensemble des attributs n'est pas utile et/ou possible en raison de la teneur de certains attributs qui n'apportent rien à l'objectif visé par le traitement des données et en raison du nombre d'attributs qui peut induire une charge de calcul trop importante. (2) Nous utilisons la réduction des redondances et duplications car, dans la plupart des jeux de données, il est nécessaire de réduire le bruit afin d'augmenter les chances de trouver des solutions satisfaisantes. En effet, le bruit dans les données peut par exemple multiplier les potentiels optima locaux d'un espace de recherche, ce qui augmente la probabilité qu'un processus de recherche se retrouve piégé dans des optima locaux moins intéressants que d'autres optima de l'espace de recherche. (3) Nous n'avons pas besoin de la compression des données car notre objectif n'est pas de faciliter le mouvement des données. (4) Nous ne nous intéressons pas à la problématique de la vitesse en considérant un système ayant déjà une fréquence de génération,

capture et partage étudiée. (5) Nous ne nous intéressons pas à l'extraction de structures topologiques car nous proposons des contributions dans le cadre de données ne le nécessitant pas. (6) Nous n'opérons pas dans un réseau. Il reste donc la réduction verticale que nous utilisons et pour laquelle nous proposons des contributions. Dans cet état de l'art, nous nous intéressons donc plus particulièrement aux méthodes d'échantillonnage.

L'échantillonnage a une longue histoire d'utilisation dans les bases de données [299] [298]. Presque toutes les analyses de données à grande échelle, en particulier l'apprentissage en profondeur, utilisent des méthodes d'échantillonnage [254] [332]. L'échantillonnage est une méthode de réduction verticale (volume) qui consiste à extraire des données du jeu de données d'origine pour en créer un plus petit qui doit en être une version suffisamment représentative. D'autres exemples de travaux touchant, entre autres, à l'analyse exploratoire des données, à l'estimation statistique, à la modélisation prédictive et à la sélection d'échantillons sont présentés dans [76] [258] [74] [158].

Nous proposons de diviser les méthodes d'échantillonnage en trois catégories, (1) les méthodes basées sur l'aléatoire, (2) les méthodes statistiquement conscientes, (3) les méthodes spécifiques.

Méthodes basées sur l'aléatoire

Ces méthodes fondent leurs décisions sur l'aléatoire. C'est notamment le cas du Simple Random Sampling [297] [162] [313]. Cette méthode fonctionne avec ou sans remise. Avec remise, chaque élément des données peut apparaître plusieurs fois dans l'échantillon. Sans remise, chaque élément des données peut apparaître au plus une fois dans l'échantillon. Ces méthodes ont plusieurs avantages : (1) Absences de biais car en raison de l'aléatoire chaque élément a la même probabilité d'être sélectionné. (2) Simplicité car les éléments sont sélectionnés au hasard, sans étapes supplémentaires. (3) Peu de connaissances requises des données. (4) Bonne représentation de l'ensemble des données. Cependant, ces méthodes ont aussi de nombreux inconvénients : (1) Pas de garantie que chaque groupe des données d'origine soit représenté équitablement. (2) Des biais peuvent exister, induits par des motivations humaines. (3) Une expertise humaine est nécessaire pour le traitement des données. (4) Les données d'origine doivent être suffisamment volumineuses. (5) Toutes les informations disponibles des données d'origine ne sont pas utilisées. Au regard de nos questions de recherche, l'utilisation de l'aléatoire n'est pas envisageable, ces méthodes ne nous intéressent donc pas dans le cadre de nos travaux.

Méthodes statistiquement conscientes

Ces méthodes fondent leurs décisions sur des analyses statistiques des données. C'est notamment le cas du Stratified Sampling [210] où les données d'origine sont divisées en groupes (strates) en utilisant par exemple une méthode de classification. De chaque groupe un échantillon aléatoire est tiré et ceux-ci sont ensuite combinés pour former l'échantillon final. Cela garantit que les éléments de chaque groupe sont considérés équitablement [388] [242] [31]. Par rapport au Simple Random Sampling, le Stratified Sampling offre une précision statistique plus élevée et réduit l'erreur d'échantillonnage. Il peut aussi permettre d'obtenir une taille d'échantillon plus petite tout en ayant la même précision. Cependant, les méthodes de Stratified Sampling sont soumises à la problématique de déséquilibre des classes [246] touchant au sous-échantillonnage et au sur-échantillonnage, le sous-échantillonnage pouvant entraîner une perte d'informations et le sur-échantillonnage pouvant entraîner un sur-ajustement des données. Un autre exemple de méthode statistiquement consciente est le Bootstrap Sampling [116] [61] où des échantillons sont créés itérativement par tirage avec remise. Des estimations statistiques, comme la mesure de

dispersion (variance, écart-type), sont évaluées sur les échantillons et la variabilité de ces évaluations est quantifiée par rapport au jeu de données d'origine. Cependant, l'amorçage sur de grands ensembles de données nécessite des coûts de calcul et de stockage élevés car il dépend du tirage répété d'échantillons de tailles comparables et du calcul des estimations à partir de tous ces échantillons. Au regard de nos questions de recherche, l'utilisation de la conscience statistique est pertinente, ces méthodes nous intéressent donc dans le cadre de nos travaux.

Méthodes spécifiques

Ces méthodes répondent à des problématiques spécifiques. C'est notamment le cas du Reservoir Sampling [401] où l'échantillonnage s'opère sans remise dans le cadre d'un ensemble de données dont la taille est indéterminée ou illimitée. Cette méthode reçoit des données d'un réseau et maintient un échantillon dans un tampon appelé réservoir. Elle a été largement utilisée dans des applications d'exploration de données à grande échelle [15] [100] [256]. Au regard de nos questions de recherche, l'utilisation de méthodes spécifiques est envisageable mais nous avons fait le choix de nous concentrer sur la conscience statistique car nous traitons des problèmes communs, ces méthodes ne nous intéressent donc pas dans le cadre de nos travaux.

2.3.4 Directions de recherche

Le domaine de la réduction de données et plus particulièrement de l'échantillonnage est ouvert et sujet à discussions. La littérature (voir les références de cette section) présente différents champs de contributions pour l'avenir, à savoir : (1) Proposer davantage de méthodes d'échantillonnage tenant compte des propriétés statistiques des données. Cependant, les méthodes d'échantillonnage statistiquement conscientes peuvent être coûteuses en calcul, un compromis intelligent doit donc être fait ici. (2) Proposer davantage de méthodes d'échantillonnage pour les jeux de données massifs et/ou de taille inconnue. (3) Proposer des méthodes permettant de comprendre les compromis entre la représentativité et la taille d'un échantillon. (4) Proposer une méthode d'échantillonnage universelle. (5) Proposer davantage de méthodes d'échantillonnage non supervisées. (6) Proposer davantage de méthodes hybrides de réduction des données. (7) Proposer des éléments explicatifs concernant le gain de performance et/ou de qualité obtenu suite à l'échantillonnage dans le cadre d'un processus de recherche et/ou d'apprentissage. (8) Proposer des éléments explicatifs concernant l'impact de l'échantillonnage sur l'E&E. L'échantillonnage modifie l'espace de recherche et a donc une influence indirecte sur le comportement d'E&E d'un algorithme de recherche. (9) Appliquer des méthodes d'échantillonnage dans des domaines peu ou pas exploités tel que le deep learning.

2.3.5 Conclusion

La gestion de la complexité amenée par le Big Data est un problème clé dont la recherche s'est emparé. La littérature révèle qu'il n'existe pas de méthode qui puisse gérer à elle seule ce problème en considérant les 6V du Big Data à savoir, volume, variété, vitesse, valeur, véracité et variabilité. La réduction des données est un domaine de recherche ouvert qui vise à améliorer les performances des algorithmes ainsi que la qualité des résultats proposés par ces derniers. Comme vu dans les sections précédentes, de nombreuses études sur la réduction des données ont été publiées, sans conduire à une réponse complète ou générale. Les domaines de la recommandation sont fortement concernés par ces problèmes. Des contributions proposant de nouvelles façons de réduire les données sont donc attendues. Nos questions de recherche en lien avec la réduction des données découlent de quelques ouvertures présentées ci-dessus.

2.4 Manuels scolaires numériques

2.4.1 Introduction

Les manuels scolaires sont des ensembles de ressources éducatives (leçons, exercices, évaluations) qui permettent d'étendre l'expérience d'apprentissage à des contextes où l'enseignant humain est absent. Avec les derniers développements en technologies interactives telles qu'Internet, la réalité virtuelle et augmentée, les techniques d'intelligence artificielle, les capteurs de données de l'Internet des objets, l'analyse des données volumineuses et le caractère omniprésent des appareils mobiles, le concept même de manuel scolaire mérite un réexamen. Les chercheurs qualifient les manuels scolaires numériques actuels de média enrichis [415], dynamiques [412], accessibles en temps réel via des réseaux [204], instructifs [434] et coïncidant avec certaines habitudes de lecture et certains styles d'apprentissage [77]. La lecture est un processus cognitif par lequel le lecteur construit une modélisation du texte pour le comprendre [214]. Plusieurs modèles informatiques sont à l'étude pour comprendre le comportement en lecture [214], qui tentent de déduire la compréhension du lecteur. Une tendance récente dans la recherche sur la modélisation des élèves consiste à incorporer les comportements de lecture pour représenter la compréhension des élèves [114] [189] [113] [382] [69] [310] [71]. L'objectif principal est d'apporter une adaptation du contenu en fonction de l'état des connaissances de l'élève, de son rythme d'apprentissage et de sa motivation, cette liste de facteurs n'étant pas exhaustive.

Les manuels en ligne adaptatifs sont l'une des technologies les plus anciennes de l'apprentissage Web personnalisé [171] [211] [405]. Ils reposent sur des méthodes de modélisation de l'apprenant et des connaissances. L'objectif principal est d'améliorer l'expérience d'apprentissage des apprenants en réduisant le temps total d'acquisition des compétences et en offrant des retours adaptatifs guidant les apprenants vers les contenus d'apprentissage les plus appropriés. Pour prendre en charge cette personnalisation, le système assure le suivi des performances des apprenants. Cela permet aussi de prédire le comportement des apprenants en fonction de leurs connaissances. La première génération de manuels adaptatifs [64] [60] [171] [211] se concentrait sur le comportement des apprenants en lecture pour les guider vers les pages les plus pertinentes en utilisant un support de navigation adaptative [64] [60] [171] [405] ou la recommandation [211]. Ces types de personnalisation reposaient sur une modélisation des connaissances où chaque page du manuel était associée à un ensemble de concepts présentés sur la page ainsi qu'aux concepts nécessaires à la compréhension de la page [64] [60].

2.4.2 Synthèse de l'état de l'art concernant les manuels scolaires numériques et concepts associés

Par le passé, les techniques dites "Adaptive Hypermedia" (AH) ont tenté de fournir des outils de navigation qui s'adapte à chaque étudiant, leur permettant de mieux localiser, reconnaître et comprendre l'information. Ces techniques ont fait leur preuve dans la mesure où elles ont permis d'améliorer la rétention d'information et l'efficacité générale de l'apprenant dans son processus d'apprentissage [9, 276, 203]. Malheureusement, ces techniques AH fonctionnent efficacement uniquement sur des corpus fermés de documents dont les contextes et méta-données ont été renseignés à la main au moment de la conception. Les corpus d'aujourd'hui, potentiellement l'ensemble des documents du Web ou du moins des amoncellements conséquents de documents au sein d'un organisme (université, lycée, collège), présentent de nouveau défis qui affectent la pertinence des techniques AH.

Dès 2005, le domaine de l'exploration de données éducatives [333] soulignait le fait que les systèmes éducatifs ont des caractéristiques spéciales qui nécessitent un traitement différent dans

le cadre du datamining. L'article [333] proposait déjà un inventaire des travaux en datamining sur les données d'apprentissage.

En 2015, la modélisation dynamique des connaissances occupe une place importante avec les notions de Knowledge Tracing et Knowledge Components [190]. Les études menées [5] montrent que les "KT-based model" sont capables de modéliser la lecture des manuels scolaires et présentent des performances prédictives supérieures aux modèles basés sur la régression logistique. Nous pouvons noter en 2016 l'article [213] qui propose dans ce contexte une modélisation dynamique des connaissances.

Toujours en 2015, l'article [403] s'intéresse aux environnements d'apprentissage interactifs et personnalisés basés sur le Concept Mapping permettant de représenter les relations entre les concepts d'apprentissage afin de mieux guider les élèves.

D'autre part, l'article [427] propose l'amélioration sémantique des manuels scolaires en ligne par des annotations sémantiques, dans le but d'améliorer les apprentissages et la personnalisation.

Dans cette même période, l'article [111] met en évidence la complexité des relations entre le guidage personnalisé et la modélisation de domaine ainsi que la possibilité d'utiliser la modélisation de domaine pour fournir un guidage personnalisé.

Nous devons aussi parler de la semi-supervision avec l'article [379] qui met en avant une approche semi-supervisée pour améliorer les techniques de datamining sur les ressources éducatives en ligne. Dans cette même veine, en 2017, l'article [225] s'intéresse à un ensemble de techniques semi-supervisées pour l'extraction des objectifs d'apprentissage et des prérequis.

En 2016, l'article [328] étudie la recommandation personnalisée de séquences de leçons par l'analyse des apprenants et des contenus éducatifs.

Par la suite, la communauté scientifique s'intéresse de plus en plus aux données cognitives et comportementales. C'est ainsi qu'en 2017, l'article [251] explique comment utiliser les graphes de connaissance et les données de comportement des étudiants pour la recommandation d'exercices personnalisés.

Pour continuer, l'apprentissage prend toujours plus de place dans le domaine avec notamment l'article [347] en 2018 qui ouvre la voie vers une expérience d'apprentissage personnalisée en utilisant le Reinforcement Learning. L'article [347] souligne aussi que la mise en œuvre du système et son déploiement dans des environnements d'apprentissage réels permettront une analyse comparative entre le système proposé et d'autres approches similaires. Cela vient insister sur le fait qu'il est souvent compliqué dans le domaine de trouver des cadres d'application réels avec des données réelles.

Enfin, l'année 2019 nous apporte plusieurs contributions intéressantes comme [212] qui propose une modélisation de l'étudiant avec extraction automatique des KCs pour les manuels adaptatifs. Elle présente quelques solutions possibles pour extraire automatiquement les KCs à partir d'un texte de grande taille.

Notons aussi l'article [309] abordant l'analyse de la visualisation pour l'exploration des relations de prérequis dans les manuels scolaires. Elle présente un ensemble de techniques de visualisation conçues pour aider les chercheurs et les analystes à mieux comprendre la question des dépendances de prérequis dans les manuels et à développer des stratégies plus puissantes pour l'extraction automatique.

Pour finir, l'article [59] fait la synthèse des récents développements dans le domaine des manuels électroniques en ligne et de l'analyse de la lecture. Il précise qu'en raison de l'absence de jeux de données réelles ou de leur petite taille dans d'autres cas, les dernières techniques d'apprentissage en profondeur (Deep Learning) n'ont pas encore été expérimentées dans le domaine.

2.4.3 Problèmes et ouvertures

Les contributions présentées [9, 276, 203, 403, 111, 213, 333, 379, 328, 427, 225, 212, 251, 309, 347, 190, 5, 59] soulignent de nombreux problèmes et ouvertures pour l'avenir de la recommandation de ressources éducatives.

Tout d'abord, les jeux de données liés à l'éducation sont quasi inexistantes ou de taille réduite ou ne comportent pas suffisamment de données intéressantes ou sont beaucoup trop bruités et inexacts. En conséquence, il n'y a toujours pas de véritable exploitation de méthodes deep learning. Les jeux de données sont aussi différents d'un organisme à l'autre. Il n'existe donc pas de solution algorithmique universelle au problème de la recommandation en éducation.

Ensuite, il est possible de représenter une bonne recommandation comme ayant (1) une bonne note de feedback fournie par l'apprenant ainsi que (2) un lien avec un bon résultat final obtenu par l'apprenant, à un examen par exemple. Or, la notation de la recommandation par l'élève est dans bien des contextes inexistante car elle n'est même pas prévue par le système. Concernant (2), il peut être difficile de faire le lien entre une recommandation et un résultat final.

De plus, beaucoup cherchent à améliorer la précision des algorithmes de recommandation et cela n'est pas forcément pertinent ni même utile. En effet, l'excès de personnalisation, par exemple, peut enfermer l'apprenant dans une bulle qui tue toute diversité et toute ouverture sur des sujets plus ou moins éloignés.

Pour continuer, les algorithmes reposent généralement sur des traitements lourds.

Par ailleurs, les recherches s'orientent toujours vers l'amélioration de méthodes spécifiques et ainsi peu de solutions hybrides sont proposées.

Nous pouvons ajouter que concernant la transparence, l'explicabilité, l'audit, la détection des biais et l'interprétabilité des algorithmes de recommandation, les recherches sont modestes. Certaines s'orientent toutefois vers le développement de méthodes qui aident à mieux comprendre ce que le modèle a appris ainsi que vers des techniques pour expliquer les prédictions individuelles.

Enfin, la plupart des papiers ne mentionnent pas du tout les problèmes liés à l'éthique (data privacy, sécurité, consentement) de l'utilisation des données d'apprentissage. Considérer ces problèmes de manière systématique semble aujourd'hui de plus en plus pertinent.

Chapitre 3

Contributions

Nous avons choisi de positionner nos contributions dans le domaine des MH car (1) elles sont transparentes et explicables [271] c'est-à-dire que leur lecture est claire et permet de comprendre comment les données sont collectées et traitées pour aboutir aux résultats, (2) elles sont performantes dans des environnements soumis à de nombreux aléas, (3) elles sont particulièrement efficaces pour optimiser les décisions [432] et (4) elles sont peu exploitées dans la littérature de la recommandation.

Dans de nombreux domaines d'intelligence artificielle, l'espace de recherche est souvent vaste car beaucoup d'*items* peuvent être recommandés et la littérature propose d'aborder ce problème comme un problème d'E&E de l'espace de recherche [94] [275] [414]. Cet espace est encore plus grand lorsque l'on s'intéresse à la recommandation d'*itemsets*.

L'espace de recherche peut être exploré par un processus d'E&E pour trouver de bonnes recommandations [80]. Explorer signifie s'assurer que de nombreuses régions de l'espace de recherche sont examinées [94], afin de ne pas se retrouver piégé dans un optimum local. Exploiter signifie que si une région prometteuse de l'espace de recherche est trouvée, elle est examinée plus attentivement [94] pour voir si elle contient de bons optima locaux. Dans le cadre de la recommandation, un optimum local est un *itemset* (ou vecteur) considéré comme optimal par rapport à la zone de l'espace de recherche dans laquelle il se situe, sachant qu'il peut exister de meilleurs *itemsets* dans d'autres zones de l'espace de recherche.

La littérature souligne qu'entre autres domaines, le domaine des métaheuristiques (MH) aborde la problématique de la proposition de recommandations dans de vastes espaces de recherche [94]. Nos travaux sur l'influence de l'E&E visent à contribuer à l'influence et à l'explicabilité de l'E&E dans le cadre de la recommandation par MH, afin d'améliorer les recommandations au regard de la performance et de la qualité, afin d'améliorer les processus d'influence de l'E&E et afin de renforcer la confiance envers les algorithmes.

Des études ont été menées pour influencer la dynamique de l'E&E afin d'assurer de meilleurs comportements de recherche et ainsi trouver de meilleures recommandations concernant des critères d'évaluation prédéfinis [94]. Influencer l'E&E implique de savoir quand, pourquoi et comment influencer [428]. En effet, l'E&E doivent être contrôlées régulièrement au travers d'une mesure et les opérations réalisées pour exercer l'influence doivent être justifiées. Notons ici que le contrôle de l'E&E et l'influence de l'E&E sont des mécanismes différents mais font tous deux partie du processus d'influence de l'E&E. Ces éléments contribuent à expliquer les comportements d'E&E afin de comprendre ce qui a conduit aux recommandations proposées [383].

L'analyse de la littérature liée à la recommandation nous a permis de converger vers des contributions dans le domaine de l'influence de l'E&E. Nous avons choisi ce domaine car il

présente des manques [94] [275] [414]. Dans ce travail, nous abordons le problème de la recommandation d'*itemsets* par l'optimisation et l'influence de l'E&E afin de trouver de meilleures recommandations au regard de critères de performance et de qualité.

Autrement dit, il faut être en mesure d'influencer la dynamique de l'E&E au sein d'un important ensemble de combinaisons afin de garantir un bon comportement de recherche et ainsi trouver de bonnes combinaisons au regard des critères d'évaluation. De notre point de vue, les processus d'influence proposés dans la littérature ne garantissent pas une expression très impactante de l'influence.

Dans les sections suivantes, nous présentons nos contributions dans le cadre d'un problème de recommandation d'*itemsets*. Nous avons choisi d'utiliser des algorithmes évolutionnaires parmi tous les mécanismes heuristiques itératifs disponibles. La raison principale de ce choix est la popularité des algorithmes génétiques. Rappelons que ces derniers exploitent généralement des tableaux de *digits*, ce qui est une façon traditionnelle de représenter des *items* ou des *itemsets* recommandables [260]. Chaque index dans un tableau correspond à un *item* recommandable. Les *digits* peuvent prendre la valeur 0 ou 1, où 0 correspond à la non-recommandation de l'*item* associé et 1 correspond à la recommandation de cet *item*. Le nombre de 1 – *digits* correspond au nombre d'*items* recommandés. L'algorithme effectue de manière itérative des variations : génère, croise et transforme des tableaux, en essayant d'optimiser et de conserver les meilleurs d'entre eux au regard de contraintes et de critères d'évaluation prédéfinis.

Afin de simplifier la compréhension des diverses notions utilisées, la Figure 3.1 propose un schéma récapitulatif.

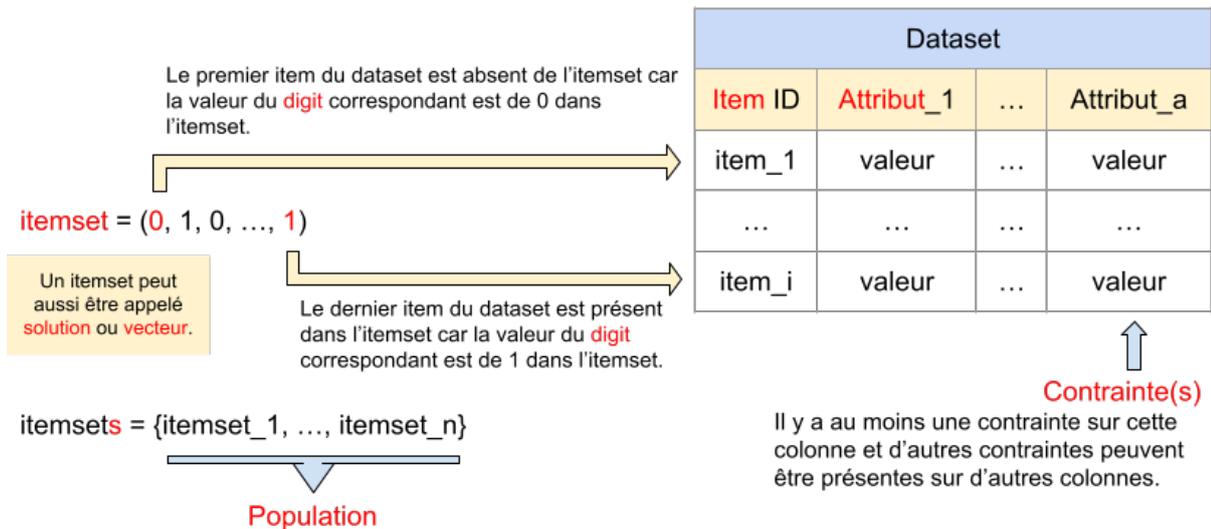


FIGURE 3.1 – Notions - Récapitulatif

3.1 Influence de l'E&E

Contrôler et influencer l'E&E permet d'orienter le processus de recherche soit vers plus d'exploration afin d'éviter qu'il se retrouve piégé dans un optimum local et afin de lui faire trouver des optima globaux, soit vers plus d'exploitation afin de faire en sorte qu'il trouve les meilleurs optima locaux au sein de régions intéressantes de l'espace de recherche.

Nous proposons un processus pour contrôler et influencer l'E&E d'un algorithme de recherche

(appelé algorithme influencé). De notre point de vue, les processus d'influence proposés dans la littérature ne garantissent pas une expression très impactante de l'influence car (1) ils dépendent fortement de l'algorithme influencé, (2) ils ne possèdent pas leurs propres opérateurs de variation, (3) ils ne garantissent aucune forme de respect des bornes de l'indicateur d'E&E à chaque itération de l'algorithme influencé. La première contribution de ce travail vise donc à donner plus de puissance au processus d'influence en concevant un processus indépendant de l'algorithme influencé. Nous présentons nos contributions dans le cadre des algorithmes évolutionnaires (Evolutionary Algorithm - EA) et des algorithmes par renforcement, en raison de leur popularité dans les problèmes de recherche. L'indépendance fait que l'influence peut être utilisée sur la plupart des MH, sans adaptation majeure. Le processus d'influence proposé est donc utilisable sur des algorithmes autres que les algorithmes évolutionnaires et par renforcement, que nous avons utilisé dans le cadre de nos expérimentations.

De plus, nous proposons de nouveaux indicateurs pour représenter le comportement d'E&E.

Nous justifions nos contributions à travers les points présentés ci-après. A en juger par l'état de l'art réalisé et en l'état actuel de nos connaissances, de telles contributions n'ont pas encore été proposées.

Le contrôle de l'E&E que nous proposons est systématique car il est réalisé à chaque itération de l'algorithme influencé mais aussi suite à toutes les variations du processus d'influence.

Dans la littérature, malgré la présence de processus évolutifs à l'intérieur du processus évolutif influencé, il n'y a pas de recherche de conformité finale de l'indicateur d'E&E qui peut toujours être insatisfaisant suite aux variations d'influence. Le processus d'influence proposé est évolutif, flexible et garantit la conformité finale de la population par rapport à l'indicateur d'E&E. En effet, les variations d'influence sont opérées tant que la population n'est pas conforme à l'indicateur d'E&E, les bornes de l'indicateur évoluant au fil des itérations de la fonction d'influence et en fonction des valeurs prises par l'indicateur. Cela permet de garantir la conformité finale de la population à l'indicateur d'E&E.

Dans la littérature, le processus peut chercher le compromis avec l'algorithme influencé, ne garantissant pas une pleine expression de l'influence. La littérature présente donc une dépendance du processus d'influence par rapport au processus algorithmique influencé. Nous pensons ici à l'optimisation de paramètres de l'algorithme influencé ou tout simplement au fait que le processus d'influence ne dispose pas de ses propres opérateurs de variation mais cherche simplement à contrôler ceux de l'algorithme influencé. Par exemple, les techniques méta-évolutives existantes sont orientées vers la méta-optimisation, c'est-à-dire l'optimisation de paramètres de l'algorithme influencé et non d'un indicateur ayant une existence dissociée de l'algorithme influencé, comme c'est le cas dans nos contributions. Dans la littérature, il existe toujours un lien avec l'algorithme influencé, qui prend la forme d'une rétroaction. Ce lien limite par définition les possibilités de traitement du processus d'influence. S'affranchir de cette dépendance permet de considérer des indicateurs à l'instant t de l'algorithme influencé et de les ajuster itérativement par rapport à des bornes qui elles aussi s'ajustent à cet instant t . Aucune rétroaction n'est nécessaire dans ce cas, seule la population est éventuellement modifiée et renvoyée à l'algorithme influencé, ce dernier n'accusant aucune modification structurelle et/ou paramétrique. Le processus d'influence proposé est indépendant et peut être greffé à n'importe quel algorithme, même non-évolutionnaire et il ne pose aucun problème de confrontation directe avec le processus algorithmique influencé. Ce que nous faisons est donc complètement différent car nous ne cherchons pas à optimiser l'algorithme influencé, nous faisons simplement un ajustement de notre indicateur d'E&E par variations supplémentaires successives de la population dans une fonction annexes qui n'a aucun lien avec l'algorithme influencé.

Dans la littérature, la plupart des approches sont basées uniquement sur l'appréciation de

Algorithm 1 Structures de variations

```

1: Variations Structure 1
2: WHILE  $e_e \leq e_{e\_min}$  or  $e_e \geq e_{e\_max}$  :
3:   WHILE  $e_e \leq e_{e\_min}$ 
4:     exploit(population)
5:     adjust( $e_{e\_min}$ ,  $e_{e\_max}$ )
6:     update_save_e_e()
7:   WHILE  $e_e \geq e_{e\_max}$ 
8:     explore(population)
9:     adjust( $e_{e\_min}$ ,  $e_{e\_max}$ )
10:    update_save_e_e()
11:
12: Variations Structure 2
13: WHILE  $e_e \leq e_{e\_min}$  or  $e_e \geq e_{e\_max}$  :
14:   IF  $e_e \leq e_{e\_min}$ 
15:     exploit(population)
16:     adjust( $e_{e\_min}$ ,  $e_{e\_max}$ )
17:     update_save_e_e()
18:   IF  $e_e \geq e_{e\_max}$ 
19:     explore(population)
20:     adjust( $e_{e\_min}$ ,  $e_{e\_max}$ )
21:     update_save_e_e()
22:
23: Variations Structure 3
24: WHILE  $e_e \geq e_{e\_max}$  :
25:   explore(population)
26:   adjust( $e_{e\_min}$ ,  $e_{e\_max}$ )
27:   update_save_e_e()
28: WHILE  $e_e \leq e_{e\_min}$  :
29:   exploit(population)
30:   adjust( $e_{e\_min}$ ,  $e_{e\_max}$ )
31:   update_save_e_e()

```

la diversité au sein de la population et il nous semble important d'apporter d'autres angles de vue. Les indicateurs d'E&E proposés permettent donc de mesurer et de représenter le comportement d'E&E sous de nouveaux angles comme ceux de la complétion des contraintes, des écarts temporels entre les phases d'E&E ou encore de la puissance d'E&E.

Ces contributions ont en partie fait l'objet d'une publication dans le cadre de la conférence GECCO 2022 [42].

3.1.1 Processus d'influence de l'E&E indépendant de l'algorithme influencé

Nous proposons un processus d'influence de l'E&E, évolutif, indépendant et utilisant des indicateurs d'influence. Le principe de fonctionnement de ce processus repose sur une fonction indépendante, dite fonction d'influence, qui est appelée plusieurs fois pendant l'exécution de l'algorithme influencé. L'ensemble des solutions de l'algorithme influencé, à l'instant t , est passé en paramètre de la fonction qui va transformer ces dernières itérativement en fonction de l'évo-

Algorithm 2 Ajustement des bornes de l'indicateur d'E&E

```

1: DEFINE FUNCTION adjust(e_e_min, e_e_max) :
2:   SET spread TO 0.05
3:   SET mean_e_e_indicador TO mean(e_e_indicador_values)
4:   SET e_e_min TO mean_e_e_indicador - spread
5:   SET e_e_max TO mean_e_e_indicador + spread

```

lution d'un indicateur d'influence au sein même de la fonction. Lorsque l'indicateur atteint une certaine valeur ou se repositionne dans un certain intervalle de valeurs (bornes), la fonction cesse les transformations et renvoie l'ensemble des solutions à l'algorithme influencé qui continue son exécution "normale" jusqu'au prochain appel de la fonction d'influence. Après chaque itération de la fonction d'influence, les bornes minimum et maximum de l'indicateur d'E&E sont ajustées en utilisant un paramètre (*spread*) qui est ajouté ou retiré à la valeur moyenne de l'indicateur d'E&E observée jusqu'à présent ([Algorithme 2](#)). Cela permet de replacer les valeurs des bornes en cohérence avec l'évolution de l'indicateur d'E&E et ainsi de converger vers une stabilisation autour des valeurs observées de l'indicateur d'E&E pour pouvoir stopper les itérations d'influence et donc sortir du processus d'influence. En effet, les variations d'influence sont opérées tant que l'indicateur est en dehors des bornes. Ce faisant, la fonction d'influence cherche à garantir qu'en fin d'itération de l'algorithme influencé, et suite aux variations d'influence, la population est bien conforme à l'indicateur d'E&E.

Notons que l'enrichissement apporté par le processus d'influence peut être perdu ou du moins détérioré si la fonction d'influence est appelée après la prise en compte des scores et des enregistrements des vecteurs finaux car l'itération suivante de l'algorithme influencé, par ses variations, peut impacter cet enrichissement. Nous avons donc fait le choix de positionner notre fonction d'influence avant la fin d'itération de l'algorithme influencé afin de ne pas perdre cet enrichissement. Nous donnons ainsi le "dernier mot" à la fonction d'influence. Aussi, ce positionnement dépend de l'algorithme à influencer ainsi que de la manière dont est construite la fonction d'influence (indicateur d'E&E, opérateurs de variations, bornes, règle d'ajustement, structure de la boucle itérative).

Un avantage important de cette approche est qu'elle est applicable avec n'importe quel indicateur, si tant est qu'il est possible de définir en amont les bornes minimum et/ou maximum de cet indicateur ainsi que la valeur du *spread* utilisée lors de l'ajustement des bornes. Pour déterminer ces informations, il est possible d'utiliser des valeurs arbitraires cohérentes, de réaliser des tests d'exécution ou encore de faire appel à un processus d'optimisation.

Enfin, cette démarche peut donner l'impression d'orienter l'algorithme vers un comportement aléatoire voire de détruire la démarche évolutionnaire influencée. Il est donc nécessaire de préciser qu'il n'en est rien en fonction des choix des opérateurs de variation d'influence, des indicateurs d'influence et des ajustements des bornes. C'est principalement le niveau de disruption amené par ces choix qui permet de juger de leur pertinence.

Soulignons de nouveau que dans la littérature, le processus d'influence a des caractéristiques qui, de notre point de vue, limitent son impact sur l'E&E. Il (1) dépend fortement de l'algorithme influencé, comme dans le cas de l'optimisation des paramètres ([Algorithme 3](#), [Figure 3.2](#)), (2) ne possède pas ses propres opérateurs de variation, (3) ne garantit aucune forme de respect des bornes de l'indicateur d'E&E à chaque itération de l'algorithme influencé. Rappelons donc de nouveau que nous proposons un nouveau processus d'influence qui (1) est dissocié de l'algorithme influencé, (2) a ses propres opérateurs de variation, (3) est appelé à chaque itération de l'algorithme influencé et (4) cherche à garantir la conformité des solutions aux bornes de l'indicateur

d'E&E à chaque itération de l'algorithme influencé. Ainsi, seuls les *itemsets* sont éventuellement modifiés et renvoyés à l'algorithme influencé, ce dernier ne présentant aucune modification structurelle et/ou paramétrique. Le processus d'influence proposé est donc générique.

L'indicateur mentionné ci-dessus est la mesure de l'E&E effectuée lors du contrôle. Nous proposons de l'utiliser pour décider de forcer l'exploitation, l'exploration ou les deux avec des variations supplémentaires. Un exemple d'indicateur d'E&E est la diversité des solutions trouvées jusqu'ici par l'algorithme influencé. L'approche que nous proposons est conçue pour utiliser n'importe quel indicateur, dès lors que des limites minimales et/ou maximales associées sont définies en amont, ainsi que la valeur du paramètre utilisé pour ajuster ces limites (plus de détails dans les sections suivantes). Ces valeurs peuvent être déterminées par des tests d'exécution, des valeurs arbitraires ou des informations statistiques de base concernant les données traitées.

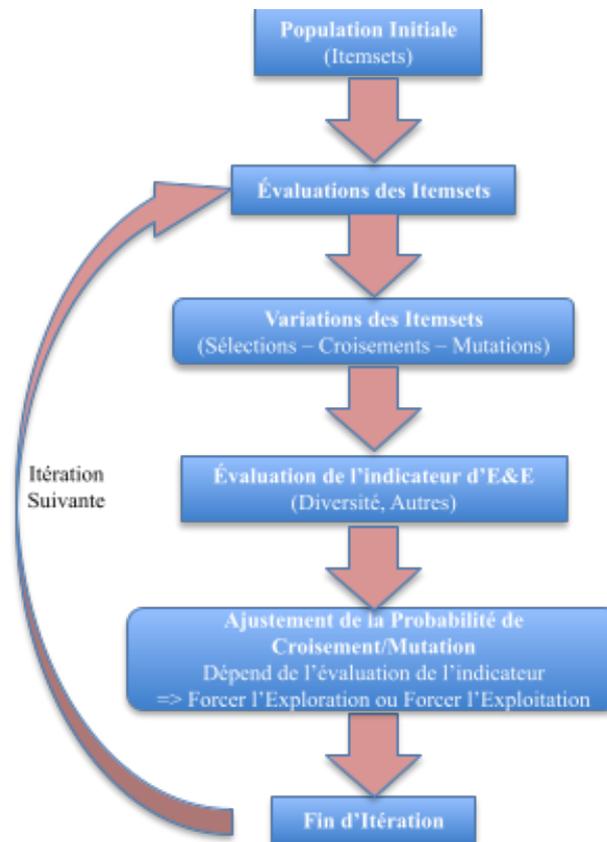


FIGURE 3.2 – Exemple d'Influence Traditionnelle de l'E&E

Le principe de fonctionnement du processus d'influence repose sur une fonction indépendante appelée systématiquement lors de l'exécution de l'algorithme influencé (Algorithme 4). Dans la littérature, l'influence n'est actionnée qu'une seule fois par itération alors que, juste après une décision d'influence, les éléments utilisés pour prendre cette décision pourraient permettre de décider à nouveau d'influencer. Nous proposons donc un contrôle systématique de l'E&E, à chaque itération de l'algorithme influencé mais également après chaque processus de variation d'influence c'est-à-dire tant que les *itemsets* ne respectent pas les limites de l'indicateur d'E&E. Le processus d'influence garantit ainsi la conformité des *itemsets* à l'indicateur d'E&E à chaque itération. Nous considérons que cela permet de renforcer l'impact de l'influence, le problème éventuel étant d'effectuer une influence trop impactante qui défigure ce que fait l'algorithme influencé en matière

Algorithm 3 Influence Traditionnelle de l'E&E

-
- 1: Étape 1 : Population Initiale
 - 2: Étape 2 : Évaluations des Itemsets
 - 3: Étape 3 : Variations des Itemsets (Sélections, Croisements, Mutations)
 - 4: Étape 4 : Évaluation de l'Indicateur d'E&E (ex : avec la diversité)
 - 5: Étape 5 : Ajustement de la Probabilité de Croisement/Mutation. Dépend de l'Évaluation de l'Indicateur. Vise à Forcer l'Exploration ou l'Exploitation
 - 6: Étape 6 : Fin d'Itération
 - 7: Étape 7 : Retour à l'étape 2
-

d'E&E. Nous devons également noter que le processus d'influence augmentera inévitablement le coût de calcul car il apporte des traitements supplémentaires.

Concrètement, toutes les solutions de l'algorithme influencé sont passées en paramètre de la fonction d'influence (Ligne 5), qui va les transformer (Lignes 19, 23) de manière itérative (Lignes 17, 18, 22) selon l'évolution de l'indicateur d'E&E (Ligne 11). Nous proposons une liste importante et non exhaustive de structures possibles pour la réalisation des variations d'influence (Algorithme 1), laissant un choix intéressant en fonction du contexte d'implémentation. Nous ne sommes pas en capacité de conseiller un choix de structure en fonction du contexte d'implémentation car il existe beaucoup de contextes d'implémentation très divers. Notons ici que la fonction *update_save_e_e()* recalcule l'indicateur d'E&E en utilisant des expressions d'indicateurs présentés dans la suite du document ou faisant partie de l'état de l'art. Notons ici que la fonction *calculer_ind(itemsets)* évalue l'indicateur d'E&E en utilisant des expressions d'indicateurs présentés en contributions ou faisant partie de l'état de l'art. Notons aussi que la fonction *maj_sauvegarder_ind()* recalcule l'indicateur d'E&E en utilisant des expressions d'indicateurs présentés en contributions ou faisant partie de l'état de l'art. Lorsque l'indicateur atteint une valeur prédéfinie ou se repositionne dans un certain intervalle de valeurs (Lignes 14, 15), la fonction d'influence arrête les transformations et renvoie toutes les solutions à l'algorithme influencé, qui continue son exécution jusqu'au prochain appel de la fonction d'influence (Ligne 5). Notons que les variations d'influence se font tant que l'indicateur est hors bornes (Lignes 17, 18, 22). Il est donc nécessaire de replacer les valeurs des bornes en fonction de l'évolution de l'indicateur d'E&E et ainsi converger vers une stabilisation autour des valeurs observées de l'indicateur d'E&E afin de pouvoir stopper les itérations d'influence. Pour cela, après chaque itération de la fonction d'influence, les bornes minimum et maximum de l'indicateur d'E&E sont ajustées (Lignes 20, 24) à l'aide d'un paramètre (Ligne 16) ajouté ou supprimé de la valeur moyenne de l'indicateur d'E&E observée jusqu'à présent (Algorithme 2). Notons que les fonctions *calculer_min_limit()*, *calculer_max_limit()* et *calculer_ajust_param()* retournent les valeurs initiales des bornes minimum et maximum de l'indicateur d'E&E ainsi que la valeur du paramètre d'ajustement. Nous ne proposons pas de structures types pour ces trois fonctions dans la mesure où elles peuvent prendre diverses formes allant du simple retour d'une valeur prédéterminée au retour d'une valeur issue d'une décision statistique plus ou moins complexe.

3.1.2 Nouveaux indicateurs pour l'influence et l'explicitabilité de l'E&E

Comme le suggère la littérature, l'E&E mériteraient d'être représentées sous de nouveaux angles [94][414][275]. C'est ce que nous entendons faire en proposant de nouveaux indicateurs. Ces indicateurs sont pensés pour mesurer et représenter les comportements d'E&E et pour être utilisés au sein de la fonction d'influence introduite précédemment. Nous présentons et analysons

Algorithm 4 EA Appelant la Fonction d'Influence

```

1: FUNCTION ea() # EA Traditionnel
2:   initialisation()
3:   WHILE(condition)
4:     ea_variations()
5:     influence(itemsets)
6:     fin_iteration()
7:   finalisation()
8:
9: FUNCTION influence(itemsets) # Influence de l'E&E
10:  # ind est l'indicateur d'E&E
11:  SET ind TO calculer_ind(itemsets)
12:  SET ind_valeurs TO []
13:  ADD ind TO ind_valeurs
14:  SET min_limit TO calculer_min_limit()
15:  SET max_limit TO calculer_max_limit()
16:  SET ajust_param TO calculer_ajust_param()
17:  WHILE ind <= min_limit or ind >= max_limit :
18:    WHILE ind <= min_limit
19:      exploiter(itemsets) # Opère des variations
20:      ajuster_limites()
21:      maj_sauvegarder_ind()
22:    WHILE ind >= max_limit
23:      explorer(itemsets) # Opère des variations
24:      ajuster_limites()
25:      maj_sauvegarder_ind()

```

ci-dessous ces indicateurs.

Potentiel d'amélioration

L'indicateur du potentiel d'amélioration (Improvement Potential - *IP*) cherche à évaluer le potentiel d'amélioration des *itemsets* d'une population. Si le potentiel est élevé, cela signifie qu'il y a encore beaucoup de solutions à haut potentiel non couvertes par la population.

Pour cela, l'indicateur exploite les meilleurs et les pires *digits* possibles au regard de la fitness afin de créer des tuples (*indice_digit, valeur_attendue*). La valeur attendue est soit 1 soit 0 car il s'agit de la valeur attendue dans un *itemset*. Le nombre total de tuples est nommé *tq*. Le nombre de *digits* dans un *itemset* qui ne respectent pas les valeurs attendues, appelé *bdq* (Equation 3.1), est évalué. Dans cette dernière équation, *itemset[n]* est la valeur du *digit* à l'indice *n* de l'*itemset* considéré, *expected_value_n* est la valeur attendue pour le *digit* d'indice *n* et *i* est le nombre de *digits*. Le potentiel est le rapport de *bdq* sur *tq* (Equation 3.2). Plus une solution (*itemset*) présente un *IP* fort, plus son potentiel d'amélioration est élevé car plus cette solution comporte des *digits* (*items*) dont les valeurs attendues ne sont pas les bonnes. Ainsi, cet indicateur permet de représenter l'E&E sous l'angle de vue du potentiel d'amélioration des solutions en permettant de statuer sur les niveaux d'E&E en fonction du potentiel d'amélioration des *itemsets* de la population.

$$bdq = \sum_{n=1}^i 1, \text{ if } itemset[n] \neq expected_value_n \quad (3.1)$$

$$IP = \frac{bdq}{tq} \quad (3.2)$$

Complétion des contraintes (CC)

L'indicateur *CC* évalue la complétion des contraintes des *itemsets* en traitant les contraintes comme des volumes à remplir. Pour évaluer cet indicateur, dans un premier temps, pour chaque *item* d'un *itemset* la moyenne des rapports des valeurs de chaque colonne sous contrainte de l'*item* par les contraintes maximales associées est calculée, ce qui représente la complétion des contraintes d'un *item* (*CC_item*). Un *item* est un tuple (*item_id*, *attribute*₁, ..., *attribute*_n) et une colonne est donc un attribut. La somme des *CC_item* d'un *itemset* donne sa complétion des contraintes (*CC_itemset*). L'indicateur *CC* est calculé en faisant la moyenne de l'ensemble des *CC_itemset* (Equation 3.4). L'ensemble des *CC_itemset* est noté *CC_itsets* (Equation 3.3), *cci* étant le nombre d'*itemsets* dans la population.

Plus *CC* est petit, plus il y a d'espace à combler parmi tous les *itemsets* car il est potentiellement possible d'ajouter plus d'*items* dans certains d'entre eux, tout en respectant les contraintes. Dans ce cas, il est possible d'affirmer que le processus d'E&E ne permettait pas jusqu'à présent d'insérer ces *items*. A contrario, plus *CC* est grand, moins il y a d'espace à combler parmi tous les *itemsets* et il est donc plus compliqué d'envisager l'ajout d'autres *items* dans certains d'entre eux, tout en respectant les contraintes. Ainsi, cet indicateur permet de représenter l'E&E sous l'angle de vue de la complétion des contraintes en permettant de statuer sur les niveaux d'E&E en fonction de la complétion des contraintes au sein des *itemsets* de la population.

$$CC_itsets = \{CC_itemset_1, \dots, CC_itemset_{cci}\} \quad (3.3)$$

$$CC = \overline{CC_itsets} \quad (3.4)$$

Liaisons inter-digits

Au cours du processus de recherche, des *digits* sont associés au sein des *itemsets*. Il est possible de considérer ces associations comme des éléments constitutifs de l'orientation de l'E&E. Nous représentons donc ici l'E&E sous la forme d'une matrice de liaisons, *D_i* étant un *digit*. L'exemple de matrice ci-dessous (Figure 3.3) montre qu'il y a neuf *digits* dans un vecteur. Un nombre au sein de la matrice représente le nombre de liaisons entre deux *digits* en considérant tous les vecteurs de la population. Il y a liaison entre deux *digits* s'ils sont tous les deux à 1 dans un même vecteur. Ainsi, une ligne de la matrice représente les nombres de liaisons entre le *digit* de la ligne et chaque autre *digit*.

Il est possible de considérer que cette matrice est une représentation de l'E&E, chaque valeur étant calculée en utilisant (Equation 3.5).

$$M(x, y) = \sum_{n=1}^i T((D_x, D_y))_n \quad (3.5)$$

T() est une fonction renvoyant 1 s'il y a liaison entre les deux *digits* du tuple passé en paramètre, c'est-à-dire si ce tuple est présent (ses deux *digits* ont la valeur 1) dans un vecteur

	D1	D2	D3	D4	D5	D6	D7	D8	D9
D1	-								
D2	26	-							
D3	4	36	-						
D4	48	8	9	-					
D5	71	89	22	83	-				
D6	11	7	12	53	1	-			
D7	58	22	16	27	6	4	-		
D8	86	2	68	75	57	81	3	-	
D9	96	45	73	91	46	64	65	58	-

FIGURE 3.3 – Exemple de matrice de liaisons

de la population. x et y représentent les coordonnées dans la matrice. i est le nombre de vecteurs dans la population. Seule la moitié de cette matrice par rapport à la diagonale nous intéresse, l'autre moitié étant identique symétriquement par rapport à la diagonale. Cette représentation de l'E&E permet d'évaluer l'indicateur el ou écart de liaisons. Pour cela, il faut évaluer la moyenne des valeurs de la matrice ($mean_m$) ainsi que la valeur maximale de la matrice (max_m). Il faut ensuite évaluer le ratio que représente la différence entre max_m et $mean_m$ par rapport à max_m (Equation 3.6).

$$el = \frac{max_m - mean_m}{max_m} \quad (3.6)$$

Du point de vue de l'algorithme, si el est trop faible, cela signifie qu'il n'y a pas de différences singulières entre les nombres de liaisons associés à chaque tuple et donc le comportement de l'algorithme surexploite potentiellement l'espace de recherche. Dans le cas contraire, l'algorithme surexplore potentiellement l'espace de recherche. Cependant, des exceptions inverses peuvent survenir marquant le fait que le choix de gestion fait par rapport à cela (bornes de l'indicateur) ainsi que le choix des opérateurs de variations sont donc déterminants pour mettre l'influence en place.

Il est possible de considérer un historique des matrices, c'est-à-dire qu'en fin de chaque itération de l'algorithme influencé, cette matrice est enregistrée afin d'avoir un historique de l'évolution de l'E&E.

Vector Links

Au cours du processus de recherche, des *digits* sont associés au sein des *itemsets*. Il est possible de considérer ces associations comme des éléments constitutifs de l'orientation de l'E&E. Est donnée ici une représentation de l'E&E à partir du nombre de liaisons possibles dans chaque vecteur. Dans chaque vecteur de la population, est calculé le nombre de liaisons possibles entre les $1 - digits$, un $1 - digit$ étant un *digit* ayant une valeur de 1. Les valeurs moyennes et maximales des nombres obtenus sont récupérées. Est calculé l'écart entre la moyenne et le maximum. La valeur de l'indicateur est la valeur en pourcentage de cet écart par rapport au maximum. Si l'écart est trop faible, il n'y a pas de différences singulières entre les nombres de liaisons associés à chaque vecteur et donc le comportement de l'algorithme surexploite potentiellement l'espace de recherche. Dans le cas contraire, l'algorithme surexplore potentiellement l'espace de recherche. Cependant, des exceptions inverses peuvent survenir marquant le fait que le choix de gestion fait par rapport à cela (bornes de l'indicateur) ainsi que le choix des opérateurs de variations sont donc déterminants pour mettre l'influence en place.

Il est possible de considérer un historique des nombres de liaisons possibles au fil des itérations de l'algorithme influencé.

Soit ndp le nombre de paires de 1 – *digit* possibles dans un vecteur. L'indicateur VL évalue dans quelle mesure ndp est réparti sur tous les vecteurs. Si la répartition est trop faible, il n'y a pas de différences singulières entre les valeurs de ndp , ce qui peut signifier que les vecteurs sont trop similaires, considérant qu'une MH a tendance à être de plus en plus dans l'exploitation au fil du temps. Par conséquent, l'algorithme surexploite potentiellement l'espace de recherche.

Toutes les valeurs ndp sont évaluées à partir de tous les vecteurs générés, ainsi que les valeurs ndp moyennes et maximales et la différence entre la moyenne et la valeur ndp maximale. La valeur de l'indicateur est le pourcentage représenté par cette différence divisée par le ndp maximal (Equation 3.7).

$$VL = \frac{\max(\{ndp\}) - \overline{\{ndp\}}}{\max(\{ndp\})} \quad (3.7)$$

E&E power

L'indicateur EEP représente la puissance d'E&E en tant que quantité d'exploration par rapport à l'exploitation. Nous considérons qu'à chaque fois qu'un *digit* est passé à 1 pour la première fois, cela constitue une exploration. Ainsi, une itération découvrant au moins un nouveau *digit* est une itération d'exploration et un vecteur contenant au moins un nouveau *digit* est un vecteur d'exploration. Les autres itérations et vecteurs sont dits d'exploitation.

Le nombre d'itérations d'exploration divisé par le nombre d'itérations d'exploitation ($\#eit$) est évalué ainsi que le nombre de vecteurs d'exploration divisé par le nombre de vecteurs d'exploitation ($\#evec$) et le nombre de *digits* nouvellement découverts divisé par le nombre de *digits* déjà découverts ($\#dnd$). EEP est la moyenne de ces trois éléments (Equation 3.8), il représente le ratio d'exploration comme la synthèse de ses valeurs.

$$EEP = \overline{(\#eit, \#evec, \#dnd)} \quad (3.8)$$

Un registre des *digits* non découverts peut aussi être utilisé pour définir formellement une phase d'E&E. Soit p une phase d'E&E pouvant prendre les valeurs $\{explore, exploit\}$. Soit ndd le registre des *digits* non encore découverts. Ce registre est mis à jour à chaque itération en enlevant des *digits* une fois qu'ils sont utilisés (passés à la valeur 1) pour la première fois à l'intérieur de vecteurs. Soit ie un marqueur de fin d'itération de l'algorithme influencé. $ndd(ie)$ correspond donc à ndd au marqueur de fin d'itération ie . Ainsi, p est définie comme dans Equation 3.9. Cette équation se lit ainsi : La phase à l'itération $ie + 1$ est une phase d'exploration si de la phase ie à la phase $ie + 1$ la taille du registre des *digits* non encore découverts a diminué. La phase à l'itération $ie + 1$ est une phase d'exploitation si de la phase ie à la phase $ie + 1$ la taille du registre des *digits* non encore découverts est restée inchangée.

$$p(ie + 1) = \begin{cases} explore, & \text{if } \overline{\overline{ndd(ie)}} > \overline{\overline{ndd(ie + 1)}} \\ exploit, & \text{if } \overline{\overline{ndd(ie)}} = \overline{\overline{ndd(ie + 1)}} \end{cases} \quad (3.9)$$

Temporalité

Nous proposons de représenter l'alternance des itérations d'E&E comme des écarts entre des valeurs temporelles. Concrètement, à chaque fois qu'une itération d'exploration a lieu, son *timer* est enregistré, il est ensuite possible en fin de macro-itération de calculer les écarts temporels entre ces *timers*. Les écarts temporels entre les instants de démarrage des itérations d'exploration sont évalués et font ressortir les écarts minimum (*ming*) et maximum (*maxg*), la différence entre

l'écart minimum et l'écart maximum (*DMMG*) (Equation 3.10), l'écart moyen (*AG*) (Equation 3.11). *DMMG* permet d'évaluer les écarts temporels les plus extrêmes entre les itérations d'exploration. *AG* représente implicitement la quantité d'itérations d'exploration qui représente implicitement leur régularité. La régularité attendue dépend du contexte. La comparaison de ces informations entre deux algorithmes permet par exemple de statuer sur le gain ou la perte de régularité de la distribution de l'exploration ou encore d'observer l'apparition de valeurs extrêmes. Notons qu'il est possible de faire de même avec les itérations d'exploitation.

$$DMMG = maxg - ming \quad (3.10)$$

$$AG = \overline{gaps} \quad (3.11)$$

3.1.3 Opérateurs de variation pour l'influence de l'E&E

Nos travaux sur l'influence de l'E&E ont également été l'occasion de proposer de nouveaux opérateurs de variations pour l'algorithmie évolutionnaire. Ces opérateurs peuvent être utilisés en tant qu'opérateurs d'influence de l'E&E (voir Section 3.1) mais aussi en tant qu'algorithmes évolutionnaires à part entière. Nous avons souhaité mesurer leur pertinence en tant qu'algorithmes évolutionnaires par rapport à un algorithme génétique classique, au regard de critères d'évaluation du comportement d'E&E et pour montrer leur potentielle utilité en tant qu'opérateurs d'influence de l'E&E.

Nous proposons donc de nouveaux algorithmes évolutionnaires générant des *itemsets*. Par souci de clarté et de concision, les éléments de pseudo-code présentés dans cette partie se concentrent uniquement sur la boucle itérative de l'EA et font abstraction d'éléments communs tels que la gestion de la mémoire ou la mise à jour des scores. De plus, ces éléments de pseudo-code sont volontairement présentés en langage naturel par souci de clarté car les explications présentées dans l'arborescence algorithmique sont plus compréhensibles. Les algorithmes ne sont donc pas présentés de manière traditionnelle car cela aurait été indigeste, trop long et trop brut. Nous avons donc opté, comme expliqué ci-dessus, pour une présentation synthétique des explications en arborescence algorithmique. Nous n'avons pas trouvé meilleur compromis.

Chaque opérateur est différent des autres de part les traitements qu'il effectue. Ces différences apportent des comportements d'E&E spécifiques.

Spycle

Cet algorithme utilise une expression mathématique permettant de modifier la vitesse de déplacement (*speed*) des *digits*. La vitesse est croissante et les déplacements alternent de droite à gauche en plusieurs cycles ($speed + cycle = spycle$). Le but est de garantir une exploration structurée autour d'une fonction et d'une vitesse variable ainsi qu'une exploitation suffisante. Cet algorithme est personnalisable en raison des possibilités multiples pour la fonction de vitesse. Une itération de cet algorithme exécute les éléments présentés dans Algorithme 5.

Struggle

Cet algorithme repose sur la sélection de plages aléatoires au sein des vecteurs. Des variations ciblées sont appliquées dans les cas de petites plages et des variations étendues sont appliquées, dans les cas de grandes plages. Le but est d'opérer des variations d'exploitation conditionnées par une décision d'exploration. En effet, les déterminations des plages en termes de lieux et de

Algorithm 5 Spycle

- 1: Augmenter la vitesse avec la fonction de vitesse choisie. Par exemple, si la fonction de vitesse est « square », la vitesse devient $vitesse * vitesse$.
 - 2: Changer le signe de la vitesse. Cela permet d'obtenir une direction alternant entre droite et gauche.
 - 3: Pour chaque vecteur de la population :
 - 4: Déplacer tous les $1 - digit$ vers la droite (vitesse positive) ou vers la gauche (vitesse négative), en utilisant la valeur de la vitesse. S'il y a déjà un $1 - digit$ là où l'on souhaite déplacer un $1 - digit$, c'est une collision et on ne fait rien (le $1 - digit$ considéré n'est pas déplacé). S'il y a un $0 - digit$ là où l'on souhaite déplacer un $1 - digit$, le $1 - digit$ considéré devient un $0 - digit$ et le $0 - digit$ où nous nous déplaçons devient un $1 - digit$.
 - 5: Ajouter un $1 - digit$ à gauche (vitesse négative) ou à droite (vitesse positive) de chaque $1 - digit$.
 - 6: Normaliser le vecteur.
 - 7: Fusionner le vecteur avec le vecteur le plus fort connu en mettant chaque $1 - digit$ du vecteur le plus fort dans le vecteur traité.
 - 8: Normaliser le vecteur.
-

Algorithm 6 Struggle

- 1: Pour chaque vecteur de la population :
 - 2: Sélectionner une plage aléatoire sur ce vecteur.
 - 3: Les deux meilleurs $0 - digits$ de la plage, d'après le score, deviennent des $1 - digits$.
 - 4: Les deux pires $0 - digits$ de la plage, d'après le score de contraintes, deviennent des $1 - digits$.
 - 5: Le pire $1 - digit$ de la plage, d'après le score, devient un $0 - digit$.
 - 6: Le meilleur $1 - digit$ de la plage, d'après le score de contraintes, devient un $0 - digit$.
 - 7: Normaliser le vecteur.
-

tailles sont variables ce qui garantit un bon comportement d'exploration et les variations en elles-mêmes sont très orientées vers l'exploitation. Une itération de cet algorithme exécute les éléments présentés dans [Algorithme 6](#).

Ramify

Cet algorithme repose sur des ramifications successives d'un *itemset* afin d'atteindre des *itemsets* compétitifs. L'algorithme ramifie un vecteur initial rempli de $1 - digits$. Les vecteurs fils obtenus sont à leur tour ramifiés jusqu'à obtenir des vecteurs acceptables au regard des contraintes. Ainsi, au sein d'une itération, la structure des vecteurs générés représente un arbre. Lors du processus de ramification présenté ci-dessous, il est aussi possible de retourner les sous-ramifications pour avoir un meilleur score mais un comportement différent, cela multiplie le temps d'exécution par plus de deux mais augmente la vitesse de convergence. Il y a un contrôle du nombre de ramifications qui permet de s'assurer que le nombre de ramifications ne dépasse pas un certain montant afin d'éviter une explosion du temps d'exécution. Ce montant est fixé arbitrairement ou en utilisant des connaissances sur le problème traité (caractéristiques, données, statistiques). Plus le nombre de ramifications est élevé, plus le temps d'exécution explose mais plus le score est meilleur. Cet algorithme ne gère pas la mémoire génétique de manière traditionnelle car il repart de zéro à chaque nouvelle itération, c'est-à-dire à partir d'un vecteur rempli

Algorithm 7 Ramify

- 1: Générer un vecteur rempli de 1 – *digits*.
 - 2: Ramifier le vecteur en deux ramifications tant qu’il ne respecte pas les contraintes maximales :
 - 3: Les deux ramifications deviennent des copies du vecteur.
 - 4: Transformer la moitié des 1 – *digits* de la 1ère ramification en 0 – *digits*, de manière aléatoire.
 - 5: Les 1 – *digits* du vecteur n’ayant pas été transformés en 0 – *digits* dans la 1ère ramification sont transformés en 0 – *digits* dans la 2ème ramification.
 - 6: Le vecteur devient la ramification ayant le meilleur score.
 - 7: Fusionner le vecteur (ramification finale) avec la ramification la plus forte connue jusqu’à présent, en mettant chaque 1 – *digit* de la ramification la plus forte dans le vecteur.
 - 8: Normaliser le vecteur.
-

Algorithm 8 Spasm

- 1: Pour chaque vecteur de la population :
 - 2: Macro Spasm :
 - 3: Sélectionner une plage aléatoire du vecteur.
 - 4: Les digits de la plage deviennent des 1-digits.
 - 5: Normaliser le vecteur.
 - 6: Micro Spasm 1 :
 - 7: Sélectionner aléatoirement un digit du vecteur.
 - 8: Le digit sélectionné ainsi que les digits autour de lui deviennent des 1-digits.
 - 9: Micro Spasm 2 :
 - 10: Sélectionner aléatoirement un digit du vecteur, en dehors de la plage utilisée pour le Macro Spasm.
 - 11: Le digit sélectionné ainsi que les digits autour de lui deviennent des 1-digits.
 - 12: Normaliser le vecteur.
-

de 1 – *digits*. Cela peut donner l’impression d’une destruction de mémoire et d’une perte d’informations. Cependant, la mémoire est présente lorsque la ramification finale est fusionnée avec la plus forte connue jusqu’à présent. Le but est de contrebalancer la partie exploitatrice amenée par cette fusion en effectuant de fortes explorations grâce aux retours à zéro lors des nouvelles itérations. Une itération de cet algorithme exécute les éléments présentés dans [Algorithme 7](#).

Spasm

Cet algorithme repose sur une variation initiale qui peut avoir un fort impact sur l’E&E, suivie par de plus petites variations. L’approche est dans un premier temps radicale car une plage complète du vecteur est transformée en 1 – *digits*. Cependant, il faut relativiser car la taille de la plage peut être très réduite. Les deux variations suivantes ajoutent un peu plus d’enrichissement mais autour de *digits* spécifiques. Le but est d’apporter de l’exploration au début de chaque itération puis d’ajouter quelques variations d’enrichissement. Les variations d’exploration ne peuvent pas effacer les bons *digits* trouvés par le passé car elles ne font que transformer des *digits* en 1 – *digits*. Une itération de cet algorithme exécute les éléments présentés dans [Algorithme 8](#).

Algorithm 9 Extossom

-
- 1: Pour chaque vecteur de la population :
 - 2: Sélectionner aléatoirement deux plages du vecteur et conserver la plus petite.
 - 3: Blossom 1 :
 - 4: Créer v1 en tant que copie du vecteur.
 - 5: Dans v1 et sur la plage sélectionnée, un digit sur deux devient un 1-digit. Le premier digit de la plage est le premier à être transformé en 1-digit.
 - 6: Blossom 2 :
 - 7: Créer v2 en tant que copie du vecteur.
 - 8: Dans v2 et sur la plage sélectionnée, un digit sur deux devient un 1-digit. Le deuxième digit de la plage est le premier à être transformé en 1-digit.
 - 9: Le vecteur devient le vecteur ayant le meilleur score entre v1 et v2.
 - 10: Chaque 0-digit du vecteur en dehors de la plage sélectionnée a une chance de devenir un 1-digit.
 - 11: Normaliser le vecteur.
-

Extossom

Cet algorithme repose sur des micro-variations et sélections destinées à garantir un mélange ainsi qu'une bonne représentativité des deux phases de l'E&E. Afin de servir une première étape d'exploration, cet algorithme sélectionne une plage aléatoire du vecteur considéré, applique deux variations distinctes sur cette plage et conserve le meilleur vecteur final obtenu. De plus, des variations supplémentaires sont appliquées en dehors de la plage. Cet algorithme est principalement exploratoire, néanmoins il effectue quelques choix concernant cette exploration et enrichit les solutions avec de légères variations. Nous supposons que cette combinaison apporte et soutient également l'exploitation. Une itération de cet algorithme exécute les éléments présentés dans [Algorithme 9](#).

U3S

Cet algorithme cherche à créer une synergie entre trois opérateurs de variations. L'opérateur "Surrounding Shot" est exploratoire car il sélectionne un *digit* aléatoirement et peut atteindre des zones non découvertes dans l'espace de recherche tout en exploitant autour du *digit* sélectionné. L'opérateur "Middle point symmetrical mutations" est principalement exploratoire mais peut parfois aussi exploiter l'espace de recherche. L'opérateur "Unexploited" permet de donner ou de redonner de l'importance à des *digits* absents de la population, il s'agit donc d'un opérateur exploratoire. Nous supposons que la combinaison de ces trois opérateurs est principalement exploratoire mais que les légers mouvements d'exploitation suffisent à garantir une convergence vers les meilleures zones de l'espace de recherche. Une itération de cet algorithme exécute les éléments présentés dans [Algorithme 10](#).

EEP

Cet algorithme repose sur la quantification de l'exploration par rapport à l'exploitation afin de forcer, au gré du processus, soit l'exploration, soit l'exploitation. Dans cet algorithme, des compteurs permettent de quantifier la part de l'exploration par rapport à l'exploitation. Une itération et un vecteur sont considérés comme exploratoires s'ils comprennent un ou plusieurs 1 – *digits* non utilisés jusqu'à présent. Un 1 – *digit* est exploratoire s'il n'a pas encore été utilisé

Algorithm 10 U3S

- 1: Surrounding Shot :
 - 2: Pour chaque vecteur de la population :
 - 3: Sélectionner aléatoirement un 1-digit.
 - 4: Sélectionner le 1-digit suivant le 1-digit sélectionné, en allant vers la droite.
 - 5: Les digits autour de ce dernier 1-digit sélectionné deviennent des 1-digits (droite et gauche).
 - 6: Middle point symmetrical mutations :
 - 7: Pour chaque vecteur de la population :
 - 8: Sélectionner l'index médian du vecteur.
 - 9: Sélectionner un index aléatoire du vecteur.
 - 10: Le digit situé à l'index sélectionné devient un 1-digit.
 - 11: Le digit situé à l'index symétriquement opposé à l'index sélectionné par rapport à l'index médian devient également un 1-digit.
 - 12: Unexploited :
 - 13: Pour chaque vecteur de la population :
 - 14: Créer zdi comme l'ensemble des index des digits étant à 0 dans tous les vecteurs de la population.
 - 15: Sélectionner aléatoirement un élément de zdi.
 - 16: Dans le vecteur, le digit à l'index correspondant à l'élément sélectionné devient un 1-digit.
 - 17: Retirer l'élément sélectionné de zdi.
 - 18: Pour chaque vecteur de la population :
 - 19: Normaliser le vecteur.
-

jusqu'à présent. Dans les cas contraires, les itérations, vecteurs et *digits* sont exploités. Les itérations, vecteurs et *digits* exploratoires et exploités sont comptés au fil des itérations. Le but est d'opérer des variations suivant le rythme des phases d'exploration. L'indicateur représente le montant de l'exploration par rapport à l'exploitation. Si l'exploration n'est pas suffisante, elle est forcée avec un opérateur spécifique et si elle est trop importante, l'exploitation est forcée avec un autre opérateur spécifique. Ces deux vérifications et variations éventuelles sont effectuées tant que l'indicateur est hors de ses limites. L'objectif est d'influencer directement l'E&E. Une itération de cet algorithme exécute les éléments présentés dans [Algorithme 11](#).

3.1.4 Conclusion

Nous avons présenté nos contributions en matière d'influence de l'E&E, à savoir, un processus d'influence de l'E&E indépendant de l'algorithme influencé, de nouveaux indicateurs pour l'influence et l'exploitabilité de l'E&E ainsi que des opérateurs de variation pour l'influence de l'E&E. L'impact de telles contributions sur l'E&E est fortement dépendant de l'espace de recherche traité et donc des données utilisées. Nous nous sommes donc naturellement intéressés à des modes de traitement des données susceptibles de transformer les espaces de recherche. Nous poursuivons donc avec la présentation de nos contributions portant sur la réduction des données.

Algorithm 11 EEP

-
- 1: Calculer la part des itérations d'exploration par rapport aux itérations d'exploitation.
 - 2: Calculer la part des vecteurs d'exploration par rapport aux vecteurs d'exploitation.
 - 3: Calculer la part des digits d'exploration par rapport aux digits d'exploitation.
 - 4: Calculer l'indicateur comme la moyenne des trois parts calculées avec remise des parts au même ordre de grandeur si besoin. Il est aussi possible de considérer une seule part.
 - 5: Définir le tableau (*ind_values*) pour enregistrer les valeurs de l'indicateur.
 - 6: Définir les limites initiales de l'indicateur, minimale et maximale.
 - 7: Définir l'écart de tolérance (spread) autour de la valeur moyenne de l'indicateur.
 - 8: Tant que l'indicateur \leq limite min :
 - 9: Pour chaque vecteur de la population :
 - 10: Forcer l'exploration en appliquant un opérateur de variation spécifique.
 - 11: Ajuster les limites de l'indicateur en calculant la moyenne de *ind_values* et en ajoutant et retirant le spread. Les limites deviennent les valeurs obtenues.
 - 12: Mettre à jour l'indicateur en le recalculant comme au début.
 - 13: Enregistrer l'indicateur dans *ind_values*.
 - 14: Tant que l'indicateur \geq limite max :
 - 15: Pour chaque vecteur de la population :
 - 16: Forcer l'exploitation en appliquant un opérateur de variation spécifique.
 - 17: Ajuster les limites de l'indicateur en calculant la moyenne de *ind_values* et en ajoutant et retirant le spread. Les limites deviennent les valeurs obtenues.
 - 18: Mettre à jour l'indicateur en le recalculant comme au début.
 - 19: Enregistrer l'indicateur dans *ind_values*.
 - 20: Pour chaque vecteur de la population :
 - 21: Normaliser le vecteur.
 - 22: Incrémenter les compteurs (itérations/vecteurs/digits exploratoires et exploités).
-

3.2 Réduction des données

Dans cette section nous nous intéressons à la réduction de données. Nous justifions nos contributions à travers les objectifs présentés ci-après.

- Tout d'abord, nous souhaitons que nos contributions soient capables d'accélérer et d'améliorer les traitements ultérieurs des données en matière de performance, de qualité des solutions trouvées par un algorithme de recherche, d'E&E, d'analyse, de recommandation et de visualisation. En effet, une fonction de réduction modifie l'espace de recherche et peut donc avoir une influence indirecte sur le comportement d'E&E de l'algorithme. Nous souhaitons mettre en lumière cet impact de l'échantillonnage sur l'E&E à l'aide de critères d'évaluation.
- De plus, nous souhaitons que nos contributions soient statistiquement conscientes car nous souhaitons discriminer les données en fonction d'informations représentatives. Cette approche apporte inévitablement des biais mais nous faisons l'hypothèse que ces biais permettront de mettre en valeur des solutions difficilement accessibles. En effet, nous présumons que les nouveaux espaces de recherche créés dans le cadre de la réduction de données sont propices à l'identification d'*itemsets* qui sont plus difficilement accessibles dans les espaces de recherche d'origine. Précisons ici qu'un *itemset* difficilement accessible peut par exemple être un *itemset* qui se situe à proximité d'optima qui l'empêchent systématiquement d'être sélectionné, ou encore un *itemset* qui présente une excellente com-

plétion de contraintes tout en ayant une fitness moins intéressante que d'autres *itemsets*. Il existe en réalité beaucoup de cas potentiels ne permettant pas l'identification et/ou la conservation de certains *itemsets* pouvant présenter des caractéristiques intéressantes du point de vue de l'optimisation et de la recommandation.

- Ensuite, nous souhaitons que nos contributions soient le plus universelles possible c'est-à-dire utilisables sur la plupart des jeux de données. Nous considérons ici qu'il est important de proposer des méthodes pouvant s'adapter à différents types de structures de données. Ces structures étant nombreuses, être suffisamment générique permet de toucher un grand nombre de domaines applicatifs et de recherche.
- Pour continuer, nous souhaitons que nos contributions soient non supervisées car nous partons du principe que la supervision se caractérise souvent avec des temps de calcul élevés ainsi qu'avec une offuscation du processus de réduction, pouvant rendre son explicabilité impossible.
- Enfin, nous souhaitons que nos contributions soient chaînables au sein d'un macro-processus de réduction car nous faisons l'hypothèse que plusieurs passes d'un même processus de réduction peut être bénéfique pour les traitements ultérieurs et car nous supposons que le mélange de processus de réduction peut lui aussi l'être.

Les problématiques associées à cet objectif de réduction sont les suivantes :

- Quels processus d'échantillonnage proposer afin de réduire un jeu de données en conservant un maximum de ses caractéristiques dans le but de servir une démarche d'E&E pour la recommandation ?
- L'utilisation conjointe de processus de réduction des données statistiquement conscients et d'un processus indépendant pour influencer l'E&E, permet-elle d'avoir un impact significatif sur les critères d'évaluation de la performance, de la qualité et du comportement d'E&E d'un algorithme de recherche ?
- Un processus de séquençage de fonctions de réduction de données peut-il être utilisé seul pour la tâche d'optimisation et de recommandation ? Peut-il être révélateur de séquences compétitives de fonctions de réduction ?

Pour rappel, dans le cadre de notre problématique de recommandation de ressources par des processus de recherche : (1) Nous utilisons la réduction horizontale par la sélection d'attributs. (2) Nous utilisons la réduction des redondances et duplications. (3) Nous n'avons pas besoin de la compression des données. (4) Nous ne nous intéressons pas à la problématique de la vitesse. (5) Nous ne nous intéressons pas à l'extraction de structures topologiques. (6) Nous n'opérons pas dans un réseau. Il reste donc la réduction verticale que nous utilisons et pour laquelle nous proposons des contributions. Ainsi, nous proposons un macro-processus et plusieurs fonctions pour la réduction d'un jeu de données par échantillonnage. Nous souhaitons que nos contributions permettent de répondre à certaines ouvertures mentionnées dans l'état de l'art [254] [372] [416] [7] [285] [329].

3.2.1 Macro-processus de réduction

Le macro-processus de réduction proposé est un chef d'orchestre qui chaîne les processus de réduction et qui décide quand cette chaîne de réduction doit s'arrêter. Chaque maillon de la chaîne de réduction travaille sur les données du maillon précédent afin de toujours plus réduire le jeu de données. Le macro-processus de réduction doit stopper la chaîne de réduction afin de répondre au compromis entre la taille et la représentativité des données. Cette approche simple et directe est pleinement explicable et suivant les processus algorithmiques des fonctions de réduction utilisées, peut présenter des temps de calcul raisonnables. Cette approche est aussi

générique de par sa nature macro.

Le macro-processus de réduction appelle une même fonction de réduction une ou plusieurs fois. Cependant, il serait aisé de le modifier afin qu’il puisse appeler des fonctions de réduction spécifiques en fonction d’un indicateur par exemple. Cela permettra d’obtenir un processus de réduction hybride dont l’avantage sera d’améliorer la représentativité des données en prenant en compte différentes propriétés statistiques et donc différents biais de représentation. Mis en commun, ces biais pourront mener à un meilleur compromis entre la taille et la représentativité tout en facilitant l’accès à certaines solutions dans le cadre d’un processus de recherche. Cependant, un processus de réduction hybride pourrait faire exploser le temps de calcul suivant la nature des fonctions de réduction utilisées. Il n’est pas non plus aisé de décider de l’ordre dans lequel appliquer des fonctions de réduction différentes et encore moins de décider des éventuels retours sur des fonctions déjà utilisées. Pour continuer, il n’est pas aisé de décider de la nature même des propriétés statistiques devant être prises en compte dans l’ensemble des fonctions de réduction à utiliser. Pour ces raisons, un processus de réduction hybride pourrait diminuer la représentativité des données.

Le macro-processus de réduction (Algorithme 12) procède à une ou plusieurs itérations de réduction (Ligne 8). Une itération de réduction appelle une fonction de réduction (Ligne 11) qui évalue chaque *item* (ou ligne ou ressource) du jeu de données individuellement au regard d’un indicateur afin de déterminer s’il peut y rester. Le processus passe à la prochaine itération de réduction tant que la taille du jeu de données final est supérieure à un certain pourcentage de la taille du jeu de données initial (Ligne 8), ce qui n’est pas standard. De plus, le processus s’arrête si une itération de réduction n’a pas permis de réduire le jeu de données (Ligne 12) ou si la réduction a été trop importante au regard d’un seuil (Ligne 14). Dans ce dernier cas, le jeu de données dans son état précédent l’itération de réduction actuelle est considéré comme étant le jeu de données final réduit (Ligne 16). Notons que les fonctions `compute_df_%_max()` et `compute_df_%_min()` retournent les valeurs des bornes minimum et maximum de la taille finale escomptée, en pourcentage, du jeu de données réduit. Nous ne proposons pas de structures types pour ces deux fonctions dans la mesure où elles peuvent prendre diverses formes allant du simple retour d’une valeur prédéterminée au retour d’une valeur issue d’une décision statistique plus ou moins complexe.

Nous proposons plusieurs fonctions de réduction que nous présentons dans les sections suivantes.

3.2.2 Fonctions de réduction

Fonction “Mean SPP Mean” - MSM

La fonction MSM utilise le SPP (“Score par Point de Pourcentage de complétion des contraintes”) et repose donc sur l’exploitation des contraintes présentes au sein d’un problème d’optimisation et/ou de recommandation. Les contraintes sont considérées comme étant définies sur les attributs sous forme d’intervalles (valeur minimale totale dans un *itemset*, valeur maximale totale dans un *itemset*).

Considérant une colonne d’un *item*, nous faisons correspondre au score de l’*item* la valeur de cette colonne (`column_value`) puis nous calculons une projection du score de l’*item* si la valeur de la colonne était la valeur maximale autorisée par la contrainte pour cette colonne. De cette façon, nous projetons par colonne un potentiel score de l’*item*, ce qui nous donne une information sur la valeur de cette colonne au regard de sa capacité à contribuer au score de l’*item*. Un `column_spp` représente donc le score de l’*item* que l’on obtiendrait si la colonne avait une

Algorithm 12 Reduce Macro

```

1: DEFINE FUNCTION reduce_macro() :
2:   SET df as the dataframe to reduce
3:   SET initial_df TO copy(df)
4:   SET df_initial_size TO len(df)
5:   SET stop TO False
6:   SET df_%_max TO compute_df_%_max()
7:   SET df_%_min TO compute_df_%_min()
8:   # Reduce WHILE df size is above a certain percentage of its initial size.
9:   WHILE len(df) > (df_%_max*df_initial_size) and stop is False :
10:    SET former_df TO copy(df)
11:    SET former_df_len TO len(df)
12:    reduce(df)
13:    IF len(df) EQUALS former_df_len :
14:      SET stop TO True
15:    IF len(df) < df_%_min*df_initial_size :
16:      SET stop TO True
17:      SET df TO copy(former_df)

```

complétion de contraintes de 100%, en suivant une relation de proportionnalité (Equation 3.12). Comme indiqué plus haut, la moyenne des *column_spp* d'un *item* est calculée (*mean_spp*) et donne donc une projection moyenne du score de cet *item*. Le *mean_spp* d'un *item* se calcule en faisant la moyenne des spp de chacune de ses colonnes (attributs) sous contrainte (*column_spp*) (Equation 3.13).

L'indicateur de cette fonction, nommé *global_mean_spp* ou *mean_spp_mean*, est la moyenne des *spp* moyens (*mean_spp*) de chaque *item* (Equation 3.14).

Seuls les *items* ayant un *mean_spp* au moins égal au *global_mean_spp* sont conservés. Cette décision repose sur l'idée que plus un *item* a un *mean_spp* élevé, plus il a du potentiel au regard de la relation de proportionnalité sus-mentionnée, et nous choisissons de considérer ce critère comme discriminant dans le cadre de l'échantillonnage car nous souhaitons créer un espace de recherche uniquement composé d'*itemsets* constitués d'*items* à fort potentiel. Nous présumons que cet espace de recherche est propice à l'identification d'*itemsets* qui sont plus difficilement accessibles dans l'espace de recherche d'origine. Reprécisons ici qu'un *itemset* difficilement accessible peut par exemple être un *itemset* qui se situe à proximité d'optima qui l'empêchent systématiquement d'être sélectionné, ou encore un *itemset* qui présente une excellente complétion de contraintes tout en ayant une fitness moins intéressante que d'autres *itemsets*. Il existe en réalité beaucoup de cas potentiels ne permettant pas l'identification et/ou la conservation de certains *itemsets* pouvant présenter des caractéristiques intéressantes du point de vue de l'optimisation et de la recommandation.

La fonction MSM, présentée dans Algorithme 13, considère ainsi chaque *item* du jeu de données (Ligne 6) afin de calculer son score (Ligne 9), ses *column_spp* (Lignes 11, 18) ainsi que son *mean_spp* (Ligne 21). Elle calcule ensuite le *global_mean_spp* (Ligne 28) afin de pouvoir sélectionner les *items* qui constitueront l'échantillon (Lignes 32, 33, 34).

$$column_spp = \frac{item_score * column_max_constraint}{column_value} \quad (3.12)$$

Algorithm 13 MSM

```

1: DEFINE FUNCTION reduce_mean_spp_mean() :
2:   SET df as the dataset to reduce
3:   SET all_spp TO []
4:   SET items_with_mean_spp TO []
5:   # For each item in df.
6:   FOR k IN range(len(df)) :
7:     SET item_spp_values TO []
8:     # Calculate the item score.
9:     r_score = score(df[k])
10:    # For each column of the item.
11:    FOR c IN range(len(df[k])) :
12:      # Check if the column is under constraints.
13:      SET under_c TO is_under_c(c)
14:      IF under_c is True :
15:        # Take the constraint max_value.
16:        SET c_max_val TO constraints[c]['max_value']
17:        # Calculate "spp"
18:        SET spp TO r_score/(df[k][c]/c_max_val)
19:        item_spp_values.append(spp)
20:      # Calculate the mean spp of the item.
21:      SET mean_spp TO mean(item_spp_values)
22:      # Associate it to the item.
23:      items_with_mean_spp.append('item' :df[k], 'mean_spp' :mean_spp)
24:      # Add the mean_spp to all_spp.
25:      all_spp.append(mean_spp)
26:      # Calculate the global mean spp.
27:      SET global_mean_spp TO mean(all_spp)
28:      # Recreate df with the items having a spp superior to the global mean spp.
29:      SET df TO []
30:      FOR k IN range(len(items_with_mean_spp)) :
31:        IF items_with_mean_spp[k]['mean_spp'] >= global_mean_spp :
32:          df.append(items_with_mean_spp[k]['item'])
33:      RETURN df

```

$$mean_spp = \overline{\{column_spp\}} \quad (3.13)$$

$$mean_spp_mean = \overline{\{mean_spp\}} \quad (3.14)$$

Fonction “Detailed SPP Mean” - DSM

La fonction DSM vise à créer un espace de recherche uniquement composé d'*itemsets* constitués d'*items* ayant au moins une colonne à fort potentiel. Dans la fonction DSM, la différence par rapport à MSM est que les colonnes sont considérées individuellement, c'est-à-dire que pour chaque colonne sous contrainte, le *column_spp* de chaque *item* est calculé, le *global_mean_spp* de tous les *items* pour cette colonne est calculé et les *items* ayant un *column_spp* au moins égal au *global_mean_spp* de la colonne sont conservés. Il s'agit donc d'une considération verticale, c'est-à-dire colonne par colonne, tandis que le “Mean SPP Mean” est une considération horizontale, c'est-à-dire *item* par *item*. Cette considération verticale repose sur l'idée qu'un *item* ayant au moins un *column_spp* élevé a du potentiel au regard de la relation de proportionnalité sus-mentionnée, et nous choisissons de considérer ce critère comme discriminant dans le cadre de l'échantillonnage car nous souhaitons créer un espace de recherche uniquement composé d'*itemsets* constitués d'*items* ayant au moins une colonne à fort potentiel. Nous présumons que cet espace de recherche est propice à l'identification d'*itemsets* qui sont plus difficilement accessibles dans l'espace de recherche d'origine. Ajoutons que le processus peut mener à l'ajout

Algorithm 14 DSM

```

1: DEFINE FUNCTION reduce_detailed_spp_mean() :
2:   SET df as the dataset to reduce
3:   SET final_items TO []
4:   # Change the structure of df.
5:   SET df_restructured TO []
6:   FOR k IN range(len(df)) :
7:     df_restructured.append('item':df[k],'score':0,'spp':0)
8:   SET df TO df_restructured
9:   # Calculate the score of each df item.
10:  FOR k IN range(len(df)) :
11:    SET df[k]['score'] TO score(df[k])
12:  # For each column.
13:  FOR c IN range(len(df[0]['item'])) :
14:    # All spp values to determine the mean.
15:    SET all_spp TO []
16:    # Check if the column is under constraints.
17:    SET under_c TO is_under_c(c)
18:    IF under_c is True :
19:      # Take the constraint max_value.
20:      SET c_max_val TO constraints[c]['max_value']
21:      # For each item in df.
22:      FOR k IN range(len(df)) :
23:        # Calculate "spp".
24:        SET spp TO df[k]['score']/(df[k]['item'][c]/c_max_val)
25:        SET df[k]['spp'] TO spp
26:        all_spp.append(spp)
27:      # Calculate the global mean spp.
28:      SET global_mean_spp TO mean(all_spp)
29:      # Put in final_items the items from df having a spp superior to the global mean spp.
30:      FOR k IN range(len(df)) :
31:        IF df[k]['spp'] >= global_mean_spp :
32:          # Check IF df[k]['item'] is not already present in final_items.
33:          IF df[k]['item'] NOT IN final_items :
34:            # Add in final_items.
35:            final_items.append(df[k]['item'])
36:  RETURN final_items

```

répété d'un même *item* au sein du jeu de données final car un même *item* peut être conservé dans le cadre du traitement de plusieurs colonnes. Il faut donc gérer ce cas afin d'éviter les duplications. Cette fonction de réduction (Algorithm 14) considère chaque colonne sous contrainte du jeu de données (Lignes 13, 18). Pour chaque colonne prise individuellement, le *column_spp* (Equation 3.12) (Ligne 24) de chaque *item* (Ligne 22) est calculé. Le *global_mean_spp* est ensuite calculé (Equation 3.15) (Ligne 28) en faisant la moyenne des *column_spp* calculés pour la colonne (*c*) en cours de traitement. Ensuite, les *items* ayant un *column_spp* au moins égal au *global_mean_spp* sont sélectionnés pour l'échantillon (Ligne 32) s'ils n'en font pas déjà partie (Ligne 34). La fonction passe ensuite à la colonne sous contrainte suivante (Ligne 13) et répète ce processus.

$$global_mean_spp_c = \overline{\{column_spp\}_c} \quad (3.15)$$

Fonction “Intra Values Stability” - IVS

L'indicateur de cette fonction repose sur le calcul de l'*IVS* de chaque *item* (*item_ivs*) qui correspond à la valeur de stabilité interne d'un *item*.

Les *items* ayant une *IVS* au moins égale à la moyenne de tous les *IVS* (*mean_ivs*) (Equation 3.18) sont conservés. Cette décision repose sur l'idée que plus les écarts observés au sein d'un *item* sont importants, plus celui-ci est intéressant à conserver de part les valeurs extrêmes

Algorithm 15 IVS

```

1: DEFINE FUNCTION reduce_intra_values_stability() :
2:   SET df as the dataset to reduce
3:   SET final_items TO []
4:   SET all_ivs TO []
5:   # Change the structure of df.
6:   SET df_restructured TO []
7:   FOR k IN range(len(df)) :
8:     df_restructured.append('item' :df[k], 'ivs' :0)
9:   SET df TO df_restructured
10:  # Calculate the ivs of each item.
11:  FOR k IN range(len(df)) :
12:    SET gaps TO []
13:    FOR c IN range(1, len(df[k]['item'])) :
14:      FOR c2 IN range(c+1, len(df[k]['item'])) :
15:        SET gap TO abs(df[k]['item'][c] - df[k]['item'][c+1])
16:        gaps.append(gap)
17:    SET ivs TO mean(gaps)
18:    SET df[k]['ivs'] TO ivs
19:    all_ivs.append(ivs)
20:  # Calculate the mean of all ivs.
21:  SET mean_ivs TO mean(all_ivs)
22:  # Put IN final_items all items having a ivs superior to mean_ivs.
23:  FOR k IN range(len(df)) :
24:    IF df[k]['ivs'] >= mean_ivs :
25:      final_items.append(df[k]['item'])
26:  RETURN final_items

```

qu'il est susceptible d'apporter. Nous choisissons de considérer ce critère comme discriminant dans le cadre de l'échantillonnage car nous souhaitons ici créer un espace de recherche constitué uniquement d'*itemsets* comprenant des *items* ayant des valeurs extrêmes. Nous présumons que cet espace de recherche est propice à l'identification d'*itemsets* qui sont plus difficilement accessibles dans l'espace de recherche d'origine.

L'*IVS* est calculée en faisant la moyenne (Equation 3.17) des écarts en valeurs absolues (gap) (Equation 3.16) entre chaque paire de colonnes (x, y) possibles au sein de l'*item*.

Cette fonction de réduction (Algorithme 15) considère chaque *item* (Ligne 11) afin de calculer ses gaps (Lignes 13, 14, 15) ainsi que la moyenne de ces derniers c'est-à-dire son *item_ivs* (Ligne 17). Ensuite, la moyenne de tous les *item_ivs* est calculée c'est-à-dire le *mean_ivs* (Ligne 21). Les *items* ayant un *item_ivs* au moins égal au *mean_ivs* sont sélectionnés pour l'échantillon (Lignes 24, 25, 26).

$$gap_{x-y} = |item_column_value_x - item_column_value_y| \quad (3.16)$$

$$item_ivs = \overline{\{gap\}} \quad (3.17)$$

$$mean_ivs = \overline{\{item_ivs\}} \quad (3.18)$$

Fonction “Mean Gap Toward Columns Means” - MGTCM

L'indicateur de cette fonction repose sur le calcul du *MGTCM* pour chaque *item* (*item_mgtcm*).

Les *items* ayant un *MGTCM* au moins égal à la moyenne (*mean_mgtcm*) (Equation 3.21) de tous les *MGTCM* calculés, sont conservés. Cela permet de donner de l'importance aux *items* les plus déviants par rapport aux moyennes des valeurs des colonnes et ainsi de conserver les valeurs extrêmes qu'ils sont susceptibles d'apporter. Nous choisissons de considérer ce critère

Algorithm 16 MGTCM

```

1: DEFINE FUNCTION reduce_mean_gap_toward_columns_means() :
2:   SET df as the dataset to reduce
3:   SET final_items TO []
4:   # Create a vector containing the mean value of each df columns.
5:   SET columns_means TO df.mean(axis=0)
6:   # Calculate the mean_gap of each item and save them in final_items.
7:   SET column_number TO len(df[0])
8:   SET all_mean_gaps TO []
9:   FOR k IN range(len(df)) :
10:    SET gaps TO []
11:    FOR c IN range(1, column_number) :
12:     SET gap TO abs(df[k][c] - columns_means[c])
13:     gaps.append(gap)
14:    SET mean_gap TO mean(gaps)
15:    final_items.append('item' :df[k], 'mean_gap' :mean_gap)
16:    all_mean_gaps.append(mean_gap)
17:   # Calculate the global mean gap.
18:   SET global_mean_gap TO mean(all_mean_gaps)
19:   # Recreate df with the items from final_items having a mean gap superior to global_mean_gap.
20:   SET df TO []
21:   FOR t IN range(len(final_items)) :
22:    IF final_items[t]['mean_gap'] >= global_mean_gap :
23:     df.append(final_items[t]['item'])
24:   RETURN df

```

comme discriminant dans le cadre de l'échantillonnage car nous souhaitons créer un espace de recherche uniquement composé d'*itemsets* constitués d'*items* ayant des valeurs extrêmes. Nous présumons que cet espace de recherche est propice à l'identification d'*itemsets* qui sont plus difficilement accessibles dans l'espace de recherche d'origine.

Le *MGTCM* correspond à la valeur moyenne (Equation 3.20) des écarts en valeurs absolues (gap) (Equation 3.19) de chaque colonne (x) d'un *item* avec la valeur moyenne de cette colonne dans le jeu de données.

Cette fonction de réduction (Algorithme 16) calcule la moyenne de chaque colonne (Ligne 5) puis considère chaque *item* (Ligne 9) afin de calculer ses gaps (Lignes 11, 12) ainsi que la moyenne de ces derniers (Ligne 14) c'est-à-dire son *item_mgtcm*. Ensuite, la moyenne de tous les *item_mgtcm* est calculée c'est-à-dire le *mean_mgtcm* (Ligne 18). Les *items* ayant un *item_mgtcm* au moins égal au *mean_mgtcm* sont sélectionnés pour l'échantillon (Lignes 22, 23, 24).

$$gap_x = |item_column_value_x - mean_column_x| \quad (3.19)$$

$$item_mgtcm = \overline{\{gap\}} \quad (3.20)$$

$$mean_mgtcm = \overline{\{item_mgtcm\}} \quad (3.21)$$

Fonction “Score Kings” - SK

L'indicateur de cette fonction repose sur le calcul du score de chaque *item*. Les *items* ayant un score inférieur à la moyenne de tous les scores sont conservés. Ainsi, cet indicateur permet de faire passer le score au second plan ce qui met en avant d'autres caractéristiques du jeu de données et notamment la compétitivité potentielle de la jonction d'*items* faibles. En effet, nous souhaitons créer un espace de recherche uniquement composé d'*itemsets* constitués d'*items* ayant des scores

Algorithm 17 SK

```

1: DEFINE FUNCTION reduce_score_kings() :
2:   SET df as the dataset to reduce
3:   SET final_items TO []
4:   SET all_s TO []
5:   # For each item, calculate its score and save it in final_items.
6:   FOR k IN range(len(df)) :
7:     SET s TO score(df[k])
8:     all_s.append(s)
9:     final_items.append('item' :df[k], 's' :s)
10:  # Calculate the mean_s.
11:  SET mean_s TO mean(all_s)
12:  # Recreate df with items having a score inferior to mean_s.
13:  SET df TO []
14:  FOR t IN range(len(final_items)) :
15:    IF final_items[t]['s'] < mean_s :
16:      df.append(final_items[t]['item'])
17:  RETURN df

```

faibles. Nous présumons que cet espace de recherche est propice à l'identification d'*itemsets* qui sont plus difficilement accessibles dans l'espace de recherche d'origine. Cette fonction de réduction (Algorithm 17) considère chaque *item* (Ligne 6) afin de calculer son score (Ligne 7) puis calcule la moyenne de tous les scores (Ligne 11). Les *items* ayant un score inférieur au score moyen sont sélectionnés pour l'échantillon (Lignes 14, 15, 16).

Fonction “Score Hills Kings” - SHK

Cette fonction associe différents compteurs à chaque *item*. Il est ensuite possible de sélectionner un *item* pour l'échantillon en fonction de diverses conditions simples ou composées utilisant les compteurs.

Les compteurs d'un *item* sont incrémentés en fonction des valeurs des colonnes de l'*item* par rapport à plusieurs agrégations portant sur l'ensemble des *items* du jeu de données, à savoir (1) un tableau contenant les valeurs moyennes de chaque colonne du jeu de données (*cm*), (2) un tableau contenant les valeurs minimum de chaque colonne du jeu de données (*cmin*), (3) un tableau contenant les valeurs maximum de chaque colonne du jeu de données (*cmax*), (4) un tableau contenant les points moyens (entre *cm* et *cmin*) de chaque colonne du jeu de données (*mpmin*) et (5) un tableau contenant les points moyens (entre *cm* et *cmax*) de chaque colonne du jeu de données (*mpmax*).

Les compteurs sont les suivants : nombre de colonnes au dessus de *cm*, nombre de colonnes en dessous de *cm*, nombre de colonnes au dessus de *mpmax*, nombre de colonnes en dessous de *mpmax*, nombre de colonnes au dessus de *mpmin*, nombre de colonnes en dessous de *mpmin*, nombre de colonnes au dessus de *cmin*, nombre de colonnes en dessous de *cmin*, nombre de colonnes au dessus de *cmax*, nombre de colonnes en dessous de *cmax*, nombre de colonnes entre *cmin* et *cmax*, nombre de colonnes entre *mpmin* et *mpmax*, nombre de colonnes entre *cm* et *mpmax*, nombre de colonnes entre *cm* et *mpmin*, nombre de colonnes entre *cm* et *cmin*, nombre de colonnes entre *cm* et *cmax*, nombre de colonnes entre *mpmax* et *cmax*, nombre de colonnes entre *mpmin* et *cmin*.

Il est possible de sélectionner un *item* pour l'échantillon en fonction de diverses conditions simples ou composées utilisant les compteurs. Par exemple, une condition simple serait “si le nombre de colonnes au dessus de *cm* est nul” et une condition composée serait “si le nombre de colonnes au dessus de *mpmax* n'est pas nul et si le nombre de colonnes entre *mpmax* et *cmax* n'est pas nul”. Les possibilités sont très nombreuses et nous en avons testé énormément, à un et deux

Algorithm 18 SHK

```

1: DEFINE FUNCTION reduce_score_hills_kings() :
2:   SET df as the dataset to reduce
3:   SET final_items TO []
4:   # Create the aggregations.
5:   SET columns_means TO df.mean(axis=0)
6:   SET columns_mins TO df.min(axis=0)
7:   SET columns_maxs TO df.max(axis=0)
8:   SET mean_points_max TO []
9:   SET mean_points_min TO []
10:  FOR k IN range(len(columns_means)) :
11:    SET mp_max TO mean([columns_means[k], columns_maxs[k]])
12:    mean_points_max.append(mp_max)
13:    SET mp_min TO mean([columns_means[k], columns_mins[k]])
14:    mean_points_min.append(mp_min)
15:  # Put in final_items the items respecting conditions regarding the counters.
16:  SET number_of_columns TO len(df[0])-1
17:  FOR t IN range(len(df)) :
18:    # Initialization of all counters.
19:    initialise_all_counters()
20:    # Increment counters for each column.
21:    FOR c IN range(1, len(df[0])) :
22:      increment_all_counters(t, c)
23:    # Use conditions to decide if the item has to be kept.
24:    IF counter_below_mppmin > 0 and counter_between_cm_cmin > 0 :
25:      final_items.append(df[t])
26:  RETURN final_items

```

éléments, pour la nullité et la positivité, afin d’en tirer les plus intéressantes en terme de réduction et de score obtenu mais aussi en fonction de nos autres critères d’évaluation. In fine, pour nos expérimentations nous avons utilisé la condition suivante : “si le nombre de colonnes en dessous de $mpmin$ n’est pas nul et si le nombre de colonnes entre cm_cmin n’est pas nul”. Nous souhaitons ici créer un espace de recherche uniquement composé d’*itemsets* constitués d’*items* répondant aux conditions choisies. Nous présumons que cet espace de recherche est propice à l’identification d’*itemsets* qui sont plus difficilement accessibles dans l’espace de recherche d’origine.

Cette fonction de réduction (Algorithme 18) commence par créer les agrégations portant sur l’ensemble des *items* (Lignes 5-14) puis considère chaque *item* (Ligne 17) afin d’incrémenter ses compteurs (Lignes 21, 22) (Algorithme 19) pour ensuite le sélectionner ou non pour l’échantillon (Ligne 25) en fonction d’une condition portant sur un ou plusieurs compteurs (Ligne 24).

Fonction “Constraints Score Kings” - CSK

L’indicateur de cette fonction repose sur le calcul du score de complétion des contraintes de chaque *item*. Les *items* ayant un score de complétion des contraintes inférieur à la moyenne de tous les scores de complétion des contraintes sont conservés dans le jeu de données final. Ainsi, cet indicateur permet de faire passer le score de complétion des contraintes au second plan ce qui met en avant d’autres caractéristiques du jeu de données et notamment la compétitivité potentielle de la jonction d’*items* faibles au regard de la complétion des contraintes. En effet, nous souhaitons créer un espace de recherche uniquement composé d’*itemsets* constitués d’*items* ayant des scores faibles au regard de la complétion des contraintes. Nous présumons que cet espace de recherche est propice à l’identification d’*itemsets* qui sont plus difficilement accessibles dans l’espace de recherche d’origine. Cette fonction de réduction (Algorithme 20) considère chaque *item* (Ligne 6) afin de calculer son score de complétion des contraintes (Ligne 7) puis calcule la moyenne de tous les scores de complétion des contraintes (Ligne 11). Les *items* ayant un score de complétion des contraintes inférieur au score moyen de complétion des contraintes sont sélectionnés pour

Algorithm 19 Increment all counters

```

1: DEFINE FUNCTION increment_all_counters(t, c) :
2:   IF df[t][c] > mean_points_max[c] :
3:     counter_above_mpmax += 1
4:   IF df[t][c] < mean_points_max[c] :
5:     counter_below_mpmax += 1
6:   IF df[t][c] > mean_points_min[c] :
7:     counter_above_mpmmin += 1
8:   IF df[t][c] < mean_points_min[c] :
9:     counter_below_mpmmin += 1
10:  IF df[t][c] > columns_mins[c] :
11:    counter_above_cmin += 1
12:  IF df[t][c] < columns_mins[c] :
13:    counter_below_cmin += 1
14:  IF df[t][c] > columns_maxs[c] :
15:    counter_above_cmax += 1
16:  IF df[t][c] < columns_maxs[c] :
17:    counter_below_cmax += 1
18:  IF df[t][c] > columns_means[c] :
19:    counter_above_cm += 1
20:  IF df[t][c] < columns_means[c] :
21:    counter_below_cm += 1
22:  IF df[t][c] > columns_mins[c] and df[t][c] < columns_maxs[c] :
23:    counter_between_cmin_cmax += 1
24:  IF df[t][c] > mean_points_min[c] and df[t][c] < mean_points_max[c] :
25:    counter_between_mpmmin_mpmmax += 1
26:  IF df[t][c] > columns_means[c] and df[t][c] < mean_points_max[c] :
27:    counter_between_cm_mpmmax += 1
28:  IF df[t][c] > mean_points_min[c] and df[t][c] < columns_means[c] :
29:    counter_between_cm_mpmmin += 1
30:  IF df[t][c] > columns_mins[c] and df[t][c] < columns_means[c] :
31:    counter_between_cm_cmin += 1
32:  IF df[t][c] > columns_means[c] and df[t][c] < columns_maxs[c] :
33:    counter_between_cm_cmax += 1
34:  IF df[t][c] > mean_points_max[c] and df[t][c] < columns_maxs[c] :
35:    counter_between_mpmmax_cmax += 1
36:  IF df[t][c] > columns_mins[c] and df[t][c] < mean_points_min[c] :
37:    counter_between_mpmmin_cmin += 1

```

l'échantillon (Lignes 14, 15, 16).

Résumé

Nous proposons de résumer certaines caractéristiques des fonctions de réduction dans le [Tableau 3.1](#) afin de permettre un positionnement de chaque fonction de réduction par rapport aux autres. Notons que toutes les fonctions de réduction ont le même objectif qui est de créer un espace de recherche (ER) propice à l'identification d'*itemsets* qui sont plus difficilement accessibles dans l'espace de recherche d'origine.

Comme mentionné plus haut, il semble possible d'effectuer la réduction d'un jeu de données en utilisant plusieurs fonctions de réduction mises en séquences. Nous présentons donc maintenant notre contribution en ce sens.

3.2.3 Réductions en séquences**Introduction**

Nous appelons séquence une suite de n éléments issus d'un ensemble de départ (où n est un entier naturel), chaque élément pouvant être présent jusqu'à n fois dans cette séquence. Ici, les éléments sont des fonctions de réduction de données. Par exemple, une séquence de 7 fonctions

Algorithm 20 CSK

```

1: DEFINE FUNCTION reduce_constraints_score_kings() :
2:   SET df as the dataset to reduce
3:   SET final_items TO []
4:   SET all_cs TO []
5:   # For each item, calculate its constraints score and save it in final_items.
6:   FOR k IN range(len(df)) :
7:     SET cs TO constraints_score(df[k])
8:     all_cs.append(cs)
9:     final_items.append('item' :df[k], 'cs' :cs)
10:  # Calculate the mean_cs.
11:  SET mean_cs TO mean(all_cs)
12:  # Recreate df with items having a cs inferior to mean_cs.
13:  SET df TO []
14:  FOR t IN range(len(final_items)) :
15:    IF final_items[t]['cs'] < mean_cs :
16:      df.append(final_items[t]['item'])
17:  RETURN df

```

TABLE 3.1 – Caractéristiques des fonctions de réduction

Fonction	Critère de sélection d'un item	Composition des itemsets de l'ER
MSM	Projection de la fitness en utilisant les contraintes	Items à fort potentiel au regard de la projection de la fitness
DSM	Projection de la fitness en utilisant les contraintes	Items ayant au moins une colonne à fort potentiel au regard de la projection de la fitness
IVS	Écart des valeurs entre les attributs	Items ayant des valeurs extrêmes
MGTCM	Écart des valeurs entre les attributs et les valeurs moyennes des colonnes associées	Items ayant des valeurs extrêmes
SK	Fitness	Items ayant des fitness faibles
SHK	Condition(s)	Items répondant aux conditions choisies
CSK	Score de complétion des contraintes	Items ayant des scores de complétion des contraintes faibles

est une suite de 7 fonctions contenant chaque fonction [0, 7] fois. L'ensemble des séquences qu'il est possible de constituer est le fruit du séquençage.

Nous proposons un processus effectuant différentes réductions d'un jeu de données, chaque version réduite étant issue d'une séquence de fonctions de réduction. Pour une séquence, le jeu de données est réduit jusqu'à l'obtention d'une sélection finale ou recommandation ou sélection optimisée. Nous axons donc le processus d'optimisation et de recommandation sur la recherche d'une séquence de fonctions de réduction amenant une conscience statistique satisfaisante, passant par la discrimination brute d'une partie des données en fonction d'informations représentatives. Cette approche apporte inévitablement des biais mais nous misons sur ces biais pour atteindre des solutions difficilement accessibles.

Cette contribution est universelle car n'importe quelle fonction de réduction peut être introduite dans l'ensemble de départ.

Description du processus

Le processus proposé ([Algorithme 21](#)) crée des séquences de fonctions de réduction (Ligne 5) en utilisant les fonctions de réduction déclarées (Ligne 4). Un exemple d'ensemble de fonctions déclarées serait (f_1, f_2, f_3) et quelques exemples de séquences seraient (f_1, f_3, f_3) , (f_2, f_2, f_2) , (f_2, f_3, f_1) . Pour chaque séquence de fonctions de réduction, le jeu de données d'entrée (df) (Ligne 3) est réduit en suivant l'ordre des fonctions de réduction présentes dans la séquence (Ligne 6). Le passage à la prochaine fonction de réduction se fait tant que les éléments restant dans df ne permettent pas de créer un vecteur qui respecte les contraintes. Si la dernière fonction de réduction est exécutée et qu'il faut encore réduire df , le processus retourne

Algorithm 21 Processus combi_reduc()

```

1: DEFINE FUNCTION combi_reduc() :
2:   Initialise candidate_vectors.
3:   Memorise the full dataset (df).
4:   Declare all reduction functions to be used.
5:   Create sequences of reduction functions.
6:   For each sequence, df is reduced following the order of the sequence.
7:     The reduction is done while remaining resources in df do not allow to create a vector
   that respects constraints.
8:     The reduction come back to the first function of the sequence if needed.
9:     The final vector obtained for each reduction sequence is saved in candidate_vectors,
   same for its score and associated reduction sequence.
10:    df is reinitialised at loop end.
11:    Mutations are performed on best candidate vectors.
12:    The best vector obtained is recommended, its score and associated reduction sequence are
   displayed.

```

à la première fonction de réduction de la séquence. Le vecteur final obtenu pour une séquence est sauvegardé dans l'ensemble des vecteurs candidats déclaré (Ligne 2), de même pour son score et sa séquence de fonctions de réduction. En fin d'itération, df est réinitialisé pour sa réduction par la prochaine séquence de fonctions de réduction. Ensuite, des mutations sont opérées sur les meilleurs vecteurs candidats (Ligne 11). Enfin, le meilleur vecteur obtenu est recommandé, son score et sa séquence de réduction associée sont affichés (Ligne 12).

3.2.4 Conclusion

Nous avons présenté nos contributions en matière de réduction des données, à savoir, un macro-processus de réduction des données, un ensemble de fonctions pour la réduction des données ainsi qu'un processus de réduction des données utilisant des séquences de fonctions de réduction des données. Avec ces deux premiers jeux de contributions portant sur l'influence de l'E&E ainsi que sur la réduction des données, nous nous sommes retrouvés confrontés au problème de l'explicabilité des processus d'E&E, problème très présent et ouvert dans la littérature. Nous nous sommes donc naturellement intéressé à des modes de représentation de l'E&E susceptibles d'éclairer les comportements d'E&E. Nous poursuivons donc avec la présentation de nos contributions portant sur l'explicabilité de l'E&E.

3.3 Explicabilité de l'E&E

Pouvoir expliquer la manière avec laquelle un algorithme explore et exploite l'espace de recherche demeure une ouverture importante de l'état de l'art [94] [275] [414].

Nous pouvons supposer qu'il est important de comprendre la manière avec laquelle l'E&E prennent place au sein des processus de recherche, c'est-à-dire de comprendre les transitions qui s'opèrent entre E&E, pour mieux gérer l'influence de l'E&E, pour décider d'utiliser ou non un algorithme dans un contexte d'exécution ou pour donner des informations compréhensibles aux humains sur la manière dont un processus algorithmique s'est déroulé. En d'autres termes, expliquer le comportement d'E&E permettrait indirectement de parvenir à de meilleures solutions et permettrait de renforcer la confiance dans les résultats des algorithmes.

Nos problématiques sont donc les suivantes : Comment expliquer le comportement d'E&E ? Comment expliciter les transitions entre E&E ?

Notre contribution consiste en de nouvelles approches permettant de donner des éléments d'information sur la manière avec laquelle une métaheuristique explore et exploite l'espace de recherche. Ces approches sont basées sur des indicateurs, issus d'hyper-volumes, de vecteurs de variation de présences ainsi que de résultats de classification, afin d'explicitier les transitions entre E&E. Par exemple, un indicateur donnant des informations concernant l'utilisation d'*items* spécifiques au cours des itérations et qui peut représenter graphiquement des marqueurs de transition entre E&E.

Nous proposons plusieurs indicateurs hyper-volumiques permettant d'obtenir des informations sur le comportement d'E&E d'une MH. A chaque itération du processus métaheuristique, ces indicateurs ainsi que d'autres indicateurs sont calculés et enregistrés au sein d'un log. Nous justifions nos contributions à travers les points présentés ci-après : (1) Le processus d'enregistrement des logs peut être greffé à la plupart des algorithmes ayant une course itérative. (2) Les indicateurs proposés peuvent être adaptés à de nombreuses métaheuristicques car bien que présentés ici dans un contexte d'algorithmie évolutionnaire [44], ils sont avant tout conceptuels. (3) Les indicateurs proposés donnent une représentation de l'E&E sous de nouveaux angles. (4) Les indicateurs proposés sont des marqueurs de différences inter-algorithme permettant de comparer les comportements d'E&E et de souligner les comportements marginaux.

Nous proposons aussi des indicateurs qui visent à répondre à l'ouverture liée à la variété des transitions qui peuvent se produire entre E&E. Ces indicateurs sont basés sur des vecteurs de variations de présence et visent à donner des informations sur les évolutions de présence des *items*, les évolutions des variations de présence des *items*, les chaleurs de transition entre itérations et les classes de transition. Nous affirmons que ces contributions (1) sont adaptables à de nombreuses MH, (2) donnent une représentation de l'E&E sous de nouveaux angles permettant de renforcer l'explicabilité de l'E&E, la diversité étant l'angle de vue majeur dans la littérature, (3) montrent des différences inter-algorithmes permettant de comparer les comportements d'E&E et de souligner les comportements marginaux, (4) permettent d'expliquer les transitions entre E&E.

Notons qu'en guise d'analyse critique, nous souhaitons rappeler qu'un indicateur est souvent fortement orienté dans un sens et qu'il ne suffit donc pas à lui seul à donner une représentation exhaustive des comportements d'E&E. Chacun des indicateurs proposés cherche donc à ouvrir une fenêtre, à poser un regard, en lien avec la thématique qu'il traite, ni plus, ni moins. La confrontation de différents regards peut permettre d'atteindre des niveaux de représentation satisfaisants.

Par ailleurs, nous proposons un algorithme de classification d'*itemsets* intervenant à la fin de chaque itération d'une métaheuristique. Si lors de sa classification un *itemset* conduit à la création d'une nouvelle classe, nous considérons qu'il s'agit d'un *itemset* d'exploration et que l'itération au sein de laquelle il a été créé est une itération d'exploration. Les autres *itemsets* qui viennent s'ajouter aux classes existantes, sont considérés comme des *itemsets* d'exploitation et les itérations associées sont considérées comme des itérations d'exploitation. Nous justifions cette contribution à travers les points suivants : (1) Le processus de classification peut être greffé à la plupart des algorithmes ayant une course itérative. (2) L'approche proposée donne une représentation de l'E&E sous un nouvel angle. (3) L'approche proposée permet de marquer des différences inter-algorithme, de comparer les comportements d'E&E et de souligner les comportements marginaux.

3.3.1 Indicateurs Hyper-Volumiques

Nos contributions consistent en plusieurs indicateurs hyper-volumiques (HVV, D%V, HVO, D%O, HVCO, HVCV) permettant d'obtenir des informations sur le comportement d'E&E d'une métaheuristique. La littérature ne présente aucun travaux utilisant ce type d'indicateur. A chaque itération du processus métaheuristique, ces indicateurs hyper-volumiques, des indicateurs présentés dans nos précédentes contributions ainsi que des indicateurs traditionnels de l'état de l'art [207] sont calculés et enregistrés au sein d'un log dont la structure est présentée dans [Tableau 3.2](#). Le choix de l'ensemble de ces indicateurs est arbitraire et vise à disposer d'informations pouvant être comparées et mises en relation dans le cadre de l'analyse et de l'explicabilité des comportements d'E&E.

Nous justifions nos contributions à travers les points suivants : (1) Le processus d'enregistrement des logs peut être greffé à la plupart des algorithmes ayant une course itérative. (2) Les indicateurs proposés peuvent être adaptés à de nombreuses métaheuristicques car bien que présentés ici dans un contexte d'algorithmie évolutionnaire [44], ils sont avant tout conceptuels. (3) Les indicateurs proposés donnent une représentation de l'E&E sous de nouveaux angles. (4) Les indicateurs proposés sont des marqueurs de différences inter-algorithme permettant de comparer les comportements d'E&E et de souligner les comportements marginaux.

Indicateur HVV

Cet indicateur permet de donner une représentation de l'amplitude de l'exploration multi-dimensionnelle de l'espace de recherche en termes de valeurs. HVV représente la part de l'hyper-volume de l'espace de recherche découvert par rapport à l'hyper-volume de l'espace de recherche total, en termes de valeurs ([Equation 3.22](#)). HVV signifie "Hyper-Volume in Values". HVVd et HVVmax signifient respectivement "Hyper-Volume in Values - Discovered" et "Hyper-Volume in Values - Maximum". HVVmax est calculé en multipliant entre elles les valeurs maximum possibles de chaque dimension (colonne) des données. HVVd est calculé en faisant de même mais avec seulement les éléments (lignes) découverts des données. Autrement dit, HVV permet de constater où en est HVVd par rapport à sa valeur maximale possible. Plus une métaheuristique est capable d'amener HVVd à sa valeur maximale, plus elle a de chance d'explorer suffisamment l'espace de recherche.

$$HVV = \frac{HVVd}{HVVmax} \quad (3.22)$$

Indicateur D%V

Cet indicateur est calculé pour chaque dimension des données et permet de donner une représentation des amplitudes des explorations mono-dimensionnelles de l'espace de recherche en termes de valeurs. D%V représente la part de la plus grande valeur découverte par rapport à la valeur maximale possible, pour une dimension ([Equation 3.23](#)). D%V signifie "Dimension Percent Value". k représente une dimension, MDV signifie "Maximum Discovered Value" et MPV signifie "Maximum Possible Value". MPV correspond à la valeur maximale possible d'une dimension et MDV correspond à la valeur maximale possible d'une dimension mais en considérant seulement les éléments (lignes) découverts des données. Autrement dit, les D%V permettent de constater où en sont les MDV par rapport à leur valeur maximale possible. Plus une métaheuristique est capable d'amener les MDV à leur valeur maximale, plus elle a de chance d'explorer suffisamment l'espace de recherche. Enfin, les valeurs de cet indicateur ne sont ni plus ni moins qu'une représentation

TABLE 3.2 – Log Structure

Attribut	Description
Iteration	Numéro de l'itération.
Vectors	Ensemble des vecteurs générés dans le cadre de cette itération.
Vectors Quantity	Quantité totale des vecteurs générés jusqu'à présent.
TV Score	Meilleur score de fitness parmi les vecteurs de <code>top_vectors</code> . Rappelons que <code>top_vectors</code> est l'ensemble des meilleures vecteurs générés au long d'une course évolutive.
AV Coverage	Couverture offerte par les vecteurs de <code>all_vectors</code> .
TVCC	Meilleur score de complétion des contraintes parmi les vecteurs de <code>top_vectors</code> . La complétion des contraintes est définie dans [44].
SimTV	Similarité moyenne des vecteurs générés dans le cadre de cette itération avec les vecteurs de <code>top_vectors</code> . Elle est calculée en faisant la moyenne des pourcentages de digits utilisés en commun entre tous les couples possibles (vecteur généré, vecteur de <code>top_vectors</code>).
Power	Rapport du nombre d'itérations d'exploration par le nombre d'itérations passées. Une itération est qualifiée d'exploratoire si elle a permis de découvrir de nouveaux digits c'est-à-dire si elle a permis de transformer un ou plusieurs 0-digit en 1-digit pour la première fois depuis le début de la course évolutive. Ainsi, par exemple, le passage des itérations sans augmentation du nombre d'itérations d'exploration signifie que l'algorithme est dans une phase d'exploitation.
HVV	Hyper-volume courant par rapport à l'hyper-volume maximum, en termes de valeurs.
D%V	Pour chaque dimension, valeur maximale découverte par rapport à la valeur maximale possible.
HVO	Hyper-volume courant par rapport à l'hyper-volume maximum, en termes d'occurrences.
D%O	Pour chaque dimension, nombre d'occurrences découvertes par rapport au total d'occurrences.
HVCO	Concentration hyper-volumique occurrentielle.
HVCV	Concentration hyper-volumique en valeur.

éclatée par dimension de HVV (Equation 3.24), “n” étant le nombre de dimensions considérées des données.

$$D\%V_k = \frac{MDV_k}{MPV_k} \quad (3.23)$$

$$HVV = \prod_{k=1}^n D\%V_k \quad (3.24)$$

Indicateur HVO

Cet indicateur permet de donner une représentation de l'amplitude de l'exploration multi-dimensionnelle de l'espace de recherche en termes d'occurrences. L'indicateur HVO représente la part de l'hyper-volume de l'espace de recherche découvert par rapport à l'hyper-volume de l'espace de recherche total, en termes d'occurrences (Equation 3.25). HVO signifie “Hyper-Volume

in Occurrences”. HVOD et HVOMax signifient respectivement “Hyper-Volume in Occurrences - Discovered” et “Hyper-Volume in Occurrences - Maximum”. HVOMax est calculé en multipliant entre eux les nombres de valeurs distinctes de chaque dimension (colonne) des données. HVOD est calculé en faisant de même mais avec seulement les éléments (lignes) découverts des données. Autrement dit, HVO permet de constater où en est HVOD par rapport à sa valeur maximale possible. Plus une métaheuristique est capable d’amener HVOD à sa valeur maximale, plus elle a de chance d’explorer suffisamment l’espace de recherche.

$$HVO = \frac{HVOD}{HVOMax} \quad (3.25)$$

Indicateur D%O

Cet indicateur est calculé pour chaque dimension des données et permet de donner une représentation des amplitudes des explorations mono-dimensionnelles de l’espace de recherche en termes d’occurrences. Cet indicateur représente la part du nombre d’occurrences distinctes découvertes par rapport au nombre total d’occurrences distinctes, pour une dimension (Equation 3.26). D%O signifie “Dimension Percent Occurrences”. “k” représente une dimension, DDO signifie “Distinct Discovered Occurrences” et DPO signifie “Distinct Possible Occurrences”. DPO correspond au nombre de valeurs distinctes d’une dimension et DDO correspond au nombre de valeurs distinctes d’une dimension mais en considérant seulement les éléments (lignes) découverts des données. Autrement dit, les D%O permettent de constater où en sont les DDO par rapport à leur valeur maximale possible. Plus une métaheuristique est capable d’amener les DDO à leur valeur maximale, plus elle a de chance d’explorer suffisamment l’espace de recherche. Enfin, les valeurs de cet indicateur ne sont ni plus ni moins qu’une représentation éclatée par dimension de HVO (Equation 3.27), “n” étant le nombre de dimensions considérées des données.

$$D\%O_k = \frac{DDO_k}{DPO_k} \quad (3.26)$$

$$HVO = \prod_{k=1}^n D\%O_k \quad (3.27)$$

Indicateur HVCO

HVCO signifie “Hyper-Volumetric Concentration in Occurrences” et représente la concentration dans l’espace de recherche, en se basant sur les occurrences de *digits* ainsi que sur les occurrences dimensionnelles. Une concentration (C) peut s’exprimer sous la forme d’une quantité (qty) par rapport à un volume (V) (Equation 3.28). Concernant la quantité, nous la calculons comme la moyenne entre Mq et Tq, Mq étant la moyenne du nombre d’utilisations des *digits* utilisés et Tq étant le nombre d’utilisations du *digit* le plus utilisé. Concernant le volume, nous proposons ici d’utiliser HVO qui représente bien l’évolution du volume exploré de l’espace de recherche par rapport au volume total possible, en se basant sur les occurrences. Le remaniement de l’équation obtenue mène à l’expression finale (Equation 3.29). HVCO représente donc la concentration hyper-volumique occurrenceielle et permet de donner une représentation de la saturation de l’espace de recherche, en se basant sur les occurrences. Autrement dit, HVCO permet de constater la charge, en occurrences de *digits*, par rapport à l’hyper-volume en occurrences. Notons que mean(Mq, Tq) et HVOD sont variables tandis que HVOMax est constant.

Aussi, $\text{mean}(Mq, Tq)$ peut croître, stagner et décroître tandis que HVOD peut croître et stagner. Ainsi, différentes combinaisons de variations peuvent être rencontrées, nous résumons leurs significations dans [Tableau 3.3](#).

$$C = \frac{qty}{V} \quad (3.28)$$

$$HVCO = \frac{\overline{(Mq, Tq)}HVOMax}{HVOD} \quad (3.29)$$

TABLE 3.3 – HVCO Variations Combos Significations

HVCO	$\overline{(Mq, Tq)}$	HVOD	Significations Possibles
↗	↗	↗	<ul style="list-style-type: none"> • Les digits utilisés sont en moyenne plus utilisés. • Le digit le plus utilisé est encore plus utilisé ou un autre digit devient plus utilisé que lui. • Une ou plusieurs occurrences sont découvertes dans une ou plusieurs dimensions.
↗	↗	→	<ul style="list-style-type: none"> • Les digits utilisés sont en moyenne plus utilisés. • Le digit le plus utilisé est encore plus utilisé ou un autre digit devient plus utilisé que lui. • Aucune occurrence n'est découverte, quelle que soit la dimension.
↗	↘	↗	<ul style="list-style-type: none"> • Cas Impossible
↗	↘	→	<ul style="list-style-type: none"> • Cas Impossible
↗	→	↗	<ul style="list-style-type: none"> • Cas Impossible
↗	→	→	<ul style="list-style-type: none"> • Cas Impossible
↘	↗	↗	<ul style="list-style-type: none"> • Les digits utilisés sont en moyenne plus utilisés. • Le digit le plus utilisé est encore plus utilisé ou un autre digit devient plus utilisé que lui. • Une ou plusieurs occurrences sont découvertes dans une ou plusieurs dimensions.
↘	↗	→	<ul style="list-style-type: none"> • Cas Impossible
↘	↘	↗	<ul style="list-style-type: none"> • Les digits utilisés sont en moyenne moins utilisés ce qui correspond à l'utilisation de nouveaux digits venant baisser cette moyenne. • Le nombre d'utilisations du digit le plus utilisé diminue. => Impossible • Une ou plusieurs occurrences sont découvertes dans une ou plusieurs dimensions.
↘	↘	→	<ul style="list-style-type: none"> • Les digits utilisés sont en moyenne moins utilisés ce qui correspond à l'utilisation de nouveaux digits venant baisser cette moyenne. • Le nombre d'utilisations du digit le plus utilisé diminue. => Impossible • Aucune occurrence n'est découverte, quelle que soit la dimension.
↘	→	↗	<ul style="list-style-type: none"> • Le nombre d'utilisations moyen des digits utilisés ne change pas. • Le nombre d'utilisations du digit le plus utilisé ne change pas. • Une ou plusieurs occurrences sont découvertes dans une ou plusieurs dimensions.
↘	→	→	<ul style="list-style-type: none"> • Cas Impossible
→	↗	→	<ul style="list-style-type: none"> • Cas Impossible
→	↗	↗	<ul style="list-style-type: none"> • Les digits utilisés sont en moyenne plus utilisés. • Le digit le plus utilisé est encore plus utilisé ou un autre digit devient plus utilisé que lui. • Une ou plusieurs occurrences sont découvertes dans une ou plusieurs dimensions.
→	↘	→	<ul style="list-style-type: none"> • Cas Impossible
→	↘	↗	<ul style="list-style-type: none"> • Cas Impossible
→	→	→	<ul style="list-style-type: none"> • Le nombre d'utilisations moyen des digits utilisés ne change pas. • Le nombre d'utilisations du digit le plus utilisé ne change pas. • Aucune occurrence n'est découverte, quelle que soit la dimension.
→	→	↗	<ul style="list-style-type: none"> • Cas Impossible

Indicateur HVCV

HVCV signifie "Hyper-Volumetric Concentration in Values" et représente la concentration dans l'espace de recherche, en se basant sur les valeurs. Une concentration (C) peut s'exprimer sous la forme d'une quantité (qty) par rapport à un volume (V) ([Equation 3.28](#)). Concernant la quantité, nous la calculons comme la moyenne entre Mq et Tq, Mq étant la moyenne des valeurs moyennes des *digits* utilisés et Tq étant la valeur moyenne du *digit* utilisé ayant la meilleure valeur moyenne. Concernant le volume, nous proposons ici d'utiliser HVV qui représente bien l'évolution du volume exploré de l'espace de recherche par rapport au volume total possible, en se basant sur les valeurs. Le remaniement de l'équation obtenue mène à l'expression finale ([Equation 3.30](#)). HVCV représente donc la concentration hyper-volumique en valeur et permet de donner une représentation de la saturation en valeur de l'espace de recherche. Autrement dit, HVCV permet de constater à quel point le volume exploré est chargé en valeur. Notons que $\text{mean}(Mq, Tq)$ et HVVd sont variables tandis que HVVmax est constant. Aussi, $\text{mean}(Mq, Tq)$ peut croître, stagner et décroître tandis que HVVd peut croître et stagner. Ainsi, différentes

combinaisons de variations peuvent être rencontrées, nous résumons leurs significations dans [Tableau 3.4](#).

$$HVCV = \frac{\overline{(Mq, Tq)}HV Vmax}{HV Vd} \quad (3.30)$$

TABLE 3.4 – HVCV Variations Combos Significations

HVCV	$\overline{(Mq, Tq)}$	HVVd	Significations Possibles
↗	↗	↗	<ul style="list-style-type: none"> • Un ou plusieurs digits sont découverts, augmentant la moyenne des valeurs moyennes des digits utilisés. • Un digit est découvert, ayant une valeur moyenne supérieure aux autres. • Un ou plusieurs digits sont découverts, ayant une ou plusieurs valeurs dimensionnelles supérieures aux autres.
↗	↗	→	<ul style="list-style-type: none"> • Un ou plusieurs digits sont découverts, augmentant la moyenne des valeurs moyennes des digits utilisés. • Un digit est découvert, ayant une valeur moyenne supérieure aux autres. • Les valeurs dimensionnelles maximales découvertes ne changent pas.
↗	↘	↗	• Cas Impossible
↗	↘	→	• Cas Impossible
↗	→	↗	• Cas Impossible
↗	→	→	• Cas Impossible
↘	↗	↗	<ul style="list-style-type: none"> • Un ou plusieurs digits sont découverts, augmentant la moyenne des valeurs moyennes des digits utilisés. • Un digit est découvert, ayant une valeur moyenne supérieure aux autres. • Un ou plusieurs digits sont découverts, ayant une ou plusieurs valeurs dimensionnelles supérieures aux autres.
↘	↗	→	• Cas Impossible
↘	↘	↗	<ul style="list-style-type: none"> • Un ou plusieurs digits sont découverts, diminuant la moyenne des valeurs moyennes des digits utilisés. • Le digit utilisé ayant la meilleure valeur moyenne voit cette dernière diminuer. => Impossible • Un ou plusieurs digits sont découverts, ayant une ou plusieurs valeurs dimensionnelles supérieures aux autres.
↘	↘	→	<ul style="list-style-type: none"> • Un ou plusieurs digits sont découverts, diminuant la moyenne des valeurs moyennes des digits utilisés. • Le digit utilisé ayant la meilleure valeur moyenne voit cette dernière diminuer. => Impossible • Les valeurs dimensionnelles maximales découvertes ne changent pas.
↘	→	↗	<ul style="list-style-type: none"> • La moyenne des valeurs moyennes des digits utilisés ne change pas. • Le digit utilisé ayant la meilleure valeur moyenne ne change pas. • Un ou plusieurs digits sont découverts, ayant une ou plusieurs valeurs dimensionnelles supérieures aux autres.
↘	→	→	• Cas Impossible
→	↗	→	• Cas Impossible
→	↗	↗	<ul style="list-style-type: none"> • Un ou plusieurs digits sont découverts, augmentant la moyenne des valeurs moyennes des digits utilisés. • Un digit est découvert, ayant une valeur moyenne supérieure aux autres. • Un ou plusieurs digits sont découverts, ayant une ou plusieurs valeurs dimensionnelles supérieures aux autres.
→	↘	→	• Cas Impossible
→	↘	↗	• Cas Impossible
→	→	→	<ul style="list-style-type: none"> • La moyenne des valeurs moyennes des digits utilisés ne change pas. • Le digit utilisé ayant la meilleure valeur moyenne ne change pas. • Les valeurs dimensionnelles maximales découvertes ne changent pas.
→	→	↗	• Cas Impossible

3.3.2 Indicateurs Transitionnels

Les indicateurs transitionnels que nous proposons ont pour objectif de répondre à l'ouverture de l'état de l'art liée à l'explicabilité de la variété des transitions qui peuvent se produire entre E&E. Ces indicateurs sont basés sur des vecteurs de variations de présence et visent à donner des informations sur les évolutions de présence des *items*, les évolutions des variations de présence des *items*, les chaleurs de transition entre itérations et les classes de transition.

Ces contributions (1) sont adaptables à de nombreuses MH, (2) donnent une représentation de l'E&E sous de nouveaux angles permettant de renforcer l'explicabilité de l'E&E, la diversité étant l'angle de vue majeur dans la littérature, (3) montrent des différences inter-algorithmes permettant de comparer les comportements d'E&E et de souligner les comportements marginaux, (4) permettent d'expliquer les transitions entre E&E.

Afin de représenter les différentes transitions entre E&E ayant lieu lors de l'exécution d'une MH, nous proposons d'utiliser l'information transitionnelle la plus fine à notre disposition, à savoir les valeurs des *digits*.

Notons que, pour l'ensemble des indicateurs présentés dans cette section, de plus amples informations, concernant leurs utilisations pour l'explicabilité de l'E&E, sont présentées dans les

expérimentations correspondantes.

Ces contributions ont en partie fait l'objet d'une publication dans le cadre de la conférence KES 2022 [43].

Présences et variations

Nous proposons de combiner les évolutions des valeurs prises par les *digits* ou tout simplement leur présence et leur absence tout au long du processus de recherche, pour constituer des marqueurs comportementaux.

Pour commencer, nous enregistrons les quantités de présence à la fin de chaque itération, c'est-à-dire le nombre de fois où chaque *digit* a une valeur à "1" parmi tous les *itemsets* de la population. Nous obtenons ainsi des vecteurs de présence (vp), dont un exemple est $(0,0,0,2,0,1,0,0,0,0,0,0,3,0,0,1,1)$. Dans cet exemple, la quatrième valeur (2) signifie que la quatrième ressource du jeu de données est sélectionnée/recommandée dans deux *itemsets* parmi tous les *itemsets* de la population. Pour continuer, nous proposons de représenter les marqueurs de transition de tous les *digits* par un vecteur $varp$, représentant le marqueur de transition d'une itération à l'autre. Cette transition représente des passages plus ou moins prononcés entre E&E et rend donc compte de l'évolution de l'E&E. Il est ensuite possible d'analyser l'évolution des vecteurs $varp$ au cours des itérations pour expliquer le comportement d'E&E grâce aux transitions qu'il opère. Un vecteur $varp$ est défini comme la soustraction de deux vecteurs vp (vp_j et vp_i de deux itérations i et j , qu'elles soient consécutives ou non), pour obtenir la variation de présence de chaque *digit* entre deux itérations. On obtient ainsi le vecteur de variation de présence $varp_{ij}$ (Equation 3.31) dont un exemple est $(+0,-2,+0,+3,+1,+0,+0,+1,+0,-2,-1)$. Notons qu'un vecteur $varp$ peut contenir des valeurs nulles et qu'il peut apparaître plusieurs fois au cours des itérations.

$$varp_{ij} = vp_j - vp_i \quad (3.31)$$

Présences sommées

Nous proposons de représenter la somme des quantités de présence des *digits* tout au long du processus de recherche, c'est-à-dire, à la fin d'une itération, le vecteur (vps) de la somme du nombre de fois où chaque *digit* est présent dans les populations en considérant toutes les itérations précédentes. Ces vps permettent d'évaluer l'évolution de l'utilisation des *digits* au cours des itérations de la MH. Plus une valeur est élevée au sein d'un vps , plus le *digit* associé a été utilisé au cours des itérations couvertes par ce vps .

Le vps d'une itération i (vps_i) est calculé en additionnant son vp avec tous les vp des itérations précédentes (Equation 3.32).

$$vps_i = \sum_{n=1}^i vp_n \quad (3.32)$$

Variations de présence sommées

Nous proposons de représenter les variations de présence sommées tout au long du processus de recherche, c'est-à-dire, pour une fin d'itération, le vecteur ($vvps$) de la somme des $varp$ en considérant toutes les itérations précédentes. Ces $vvps$ fournissent des informations qui contribuent à évaluer l'évolution des variations de présence des *digits* au cours des itérations de la

MH. Plus une valeur est élevée au sein d'un *vvps*, plus le digit associé a varié en présence au cours des itérations couvertes par ce *vvps*.

Pour une itération i , le vecteur des variations de présence sommées ($vvps_i$) est formé en additionnant les valeurs absolues de tous les *varp* des itérations précédentes (Equation 3.33).

$$vvps_i = \sum_{n=1}^i |varp_n| \quad (3.33)$$

Chaleur de transition

Nous représentons la chaleur de transition avec $th+$, $th-$ et th .

$th+$ représente le fait que des *digits* sont découverts marquant ainsi une exploration ou que des *digits* déjà découverts sont plus utilisés marquant ainsi une exploitation. $th-$ représentent le fait que des *digits* déjà découverts et peu utilisés sont moins utilisés, marquant ainsi la fin d'une exploration ou que des *digits* déjà découverts et très utilisés sont moins utilisés, marquant ainsi la fin d'une exploitation. Notons ici que les interprétations possibles pour $th+$ et $th-$ se situent sur un spectre beaucoup plus large que celui des différents exemples que nous avons présentés.

Sur cette base, nous proposons de représenter la chaleur de transition (th) tout au long du processus de recherche. Elle contribue à donner des informations sur la force de transition entre les itérations. Une valeur th élevée signifie que de nombreux changements ont eu lieu entre les itérations alors qu'une valeur th faible signifie que peu de changements ont eu lieu entre les itérations. Ces changements concernent le comportement d'E&E.

Il est possible de représenter le th à l'échelle d'un *digit*. Plus il est élevé, plus le *digit* en question oscille entre sélection et non-sélection et donc plus il représente un point d'indécision dans le processus de recherche.

Nous définissons $th+$ et $th-$ comme étant respectivement la somme des valeurs positives et la somme des valeurs absolues négatives d'un *varp*.

Pour une itération i , th est calculé en additionnant les valeurs absolues de $varp_i$ (Equation 3.34).

$$th_i = \sum_{n=1}^{dim(varp_i)} |varp_i[n-1]| \quad (3.34)$$

Classification des vecteurs de transition

Afin d'identifier des classes de transition et ainsi d'identifier, par exemple, les moments où les transitions se produisent à travers le processus de recherche et par conséquent de révéler quel type de *varp* est associé à un comportement d'E&E spécifique, nous proposons de regrouper les vecteurs *varp*. Cette classification automatique peut être effectuée en utilisant la similarité entre les valeurs des vecteurs *varp* ou en utilisant les valeurs th . Notons ici que deux vecteurs *varp* peuvent avoir des valeurs th similaires tout en étant radicalement différents d'un point de vue transitionnel. Notons aussi que cette classification peut être effectuée en utilisant d'autres critères.

Nous proposons d'associer des informations à chaque classe, à savoir son identifiant, l'ensemble de ses vecteurs *varp*, son *varp* moyen, sa quantité de vecteurs *varp*, sa moyenne globale th , sa moyenne positive th ($th+$), sa moyenne négative th ($th-$), le nombre moyen d'itérations séparant deux de ses vecteurs *varp*. Ces éléments peuvent être représentés graphiquement et analysés. L'identifiant de classe est un nombre. L'ensemble des vecteurs *varp* d'une classe est

l'ensemble des vecteurs *varp* appartenant à cette classe. Une classe C_j possède une quantité de vecteurs *varp* notée $|C_j|$. Le *varp* moyen d'une classe est $varpmean_j$ dont les valeurs représentent les moyennes par dimension des valeurs des vecteurs *varp* de cette classe (Equation 3.35). La quantité de vecteurs *varp* dans une classe est le nombre de vecteurs *varp* appartenant à cette classe. La moyenne *th* d'une classe est la moyenne *th* des vecteurs *varp* appartenant à cette classe. La moyenne *th+* d'une classe est la moyenne *th+* des vecteurs *varp* appartenant à cette classe. La moyenne *th-* d'une classe est la moyenne *th-* des vecteurs *varp* appartenant à cette classe. Le nombre moyen d'itérations séparant deux vecteurs *varp* d'une classe est le nombre moyen de x itérations de la MH qui se sont écoulées entre l'apparition de deux vecteurs *varp* de cette classe. Plus clairement, cela signifie qu'en moyenne, cette classe transitionnelle apparaît toutes les x itérations.

$$varpmean_j = \sum_{m=1}^{|C_j|} \frac{varp_m}{|C_j|} \quad (3.35)$$

3.3.3 Classification Incrémentielle

Notre contribution consiste en un algorithme de classification d'*itemsets* intervenant à la fin de chaque itération d'une métaheuristique. Si lors de sa classification un *itemset* conduit à la création d'une nouvelle classe, nous considérons qu'il s'agit d'un *itemset* d'exploration et que l'itération au sein de laquelle il a été créé est une itération d'exploration. Les autres *itemsets* qui viennent s'ajouter aux classes existantes, sont considérés comme des *itemsets* d'exploitation et les itérations associées sont considérées comme des itérations d'exploitation.

Cette contribution présente les avantages suivants : (1) Le processus de classification peut être greffé à la plupart des algorithmes ayant une course itérative. (2) L'approche proposée donne une représentation de l'E&E sous un nouvel angle. (3) L'approche proposée permet de marquer des différences inter-algorithme, de comparer les comportements d'E&E et de souligner les comportements marginaux.

Le fonctionnement est présenté dans [Algorithme 22](#). La variable *vectors_classes* (Ligne 2), contient l'ensemble des classes de la classification de l'itération précédente. Une classe contient l'ensemble des vecteurs appartenant à cette classe. La variable *ee_phases* (Ligne 3) contient le type E&E de chaque itération, l'index correspondant au numéro de l'itération. Le nombre de classes avant classification est sauvegardé (Ligne 5) puis le vecteur paramètre est classifié (Ligne 7) en utilisant les informations déjà présentes dans *vectors_classes*. A l'issue de la classification, le nombre de classes peut avoir augmenté, ce nombre est donc sauvegardé (Ligne 9). Si ce nombre a augmenté (Ligne 12), le type E&E de l'itération passée en paramètre est modifié (Ligne 13).

Le principe consiste donc à identifier les transitions entre E&E grâce à l'apparition d'*itemsets* provoquant l'apparition de nouvelles classes d'*itemsets*. Notons qu'un *itemset* peut provoquer l'apparition d'une nouvelle classe tout en intégrant une classe existante. Nous supposons qu'il est possible de trouver des correspondances entre les transitions répertoriées par la classification incrémentielle et les changements notables d'autres indicateurs au cours des itérations. Nous supposons aussi que la classification incrémentielle est capable de répertorier des transitions que les autres indicateurs ne parviennent pas à révéler.

3.3.4 Conclusion

Nous avons présenté nos contributions en matière d'explicabilité de l'E&E, à savoir, des indicateurs hyper-volumiques et transitionnels ainsi qu'une classification incrémentielle. Ces contri-

Algorithm 22 Add To Vectors Classes

```

1: DEFINE FUNCTION add_to_vectors_classes(vector, iteration_num) :
2:   global vectors_classes
3:   global ee_phases
4:   # Starting length of vectors_classes.
5:   SET vc_len_start TO len(vectors_classes)
6:   # Classify the vector.
7:   classify(vector)
8:   # Ending length of vectors_classes.
9:   SET vc_len_end TO len(vectors_classes)
10:  # If a new class was created, it is an exploration iteration.
11:  # Save this information into ee_phases.
12:  IF vc_len_end > vc_len_start :
13:    SET ee_phases[iteration_num] TO "Exploration"

```

butions ne font qu'effleurer toute la richesse et la complexité des possibilités en matière d'explicitabilité de l'E&E.

3.4 Conclusion à l'ensemble des contributions

Nous avons présenté nos contributions en matière d'influence de l'E&E, de réduction des données ainsi que d'explicitabilité de l'E&E. Nous achevons ainsi la présentation de nos contributions dans le cadre de cette thèse, en stoppant notre cheminement face à l'océan des potentielles contributions futures dans ces domaines.

Nous allons maintenant présenter nos expérimentations associées à ces diverses contributions ainsi que leurs résultats.

Chapitre 4

Expérimentations

Le début de ce chapitre présente des informations générales relatives aux expérimentations menées, au macro-processus ainsi qu'aux critères d'évaluation utilisés. Ensuite viennent les présentations des résultats et analyses des expérimentations suivies de leurs conclusions. Les expérimentations reprennent les contributions présentées précédemment, à savoir (1) l'influence de l'E&E, (2) les opérateurs de variation pour l'influence de l'E&E, (3) la réduction des données, (4) l'utilisation conjointe de la réduction des données et de l'influence de l'E&E, (5) les réductions combinées des données et (6) l'explicabilité de l'E&E.

4.1 Informations générales concernant les expérimentations

Nos expérimentations visent à comparer un ensemble de cas d'exécution, c'est-à-dire à comparer les résultats obtenus par des algorithmes étant exécutés en utilisant divers procédés présentés en contributions. Ces comparaisons nous permettent de statuer sur la pertinence de nos contributions.

Nos expérimentations ont été réalisées sur les jeux de données (datasets) « MovieLens 25M » [167] et « Riiid » [1]. « MovieLens 25M » [167] est très utilisé dans la littérature de la recommandation. Il se compose d'évaluations et de *tags* donnés par des utilisateurs sur des films. Il contient 25 millions d'évaluations et 1 million de *tags* donnés par 162 000 utilisateurs sur 62 000 films. L'ensemble de données comprend également 15 millions de scores de pertinence pour 1 129 *tags*. Riiid Labs, un fournisseur de solutions en IA pour l'éducation, met à disposition EdNet [85] qui est la plus grande base de données ouverte au monde pour l'enseignement de l'IA contenant plus de 100 millions d'interactions d'étudiants. Ils ont organisé un concours nommé 'Riiid! Answer Correctness Prediction' [1] sur kaggle. Ce concours reposait sur l'utilisation du jeu de données Riiid.

Nous choisissons deux datasets touchant des thématiques différentes afin de mettre en avant la portabilité de nos approches et contributions.

Concernant le jeu de données MovieLens, nous choisissons arbitrairement d'utiliser les attributs numériques suivants : la note moyenne, la quantité de notes et la pertinence moyenne des *tags*. Quelques exemples de requêtes d'extraction sont présentés dans [Algorithme 25](#), [Algorithme 26](#), [Algorithme 27](#). Concernant le jeu de données Riiid, nous nous intéressons arbitrairement à la partie du jeu de données relative aux questions et nous choisissons d'utiliser les attributs numériques suivants : le pourcentage de réponses correctes, le temps moyen passé sur les questions précédentes et le pourcentage de questions précédentes qui étaient accompagnées d'explications. Quelques exemples de requêtes d'extraction sont présentés dans [Algorithme 28](#),

Algorithme 29, Algorithme 30.

Les objectifs et les contraintes des recommandations ont également été déterminés arbitrairement, en veillant à apporter suffisamment de diversité pour chacun d'eux, afin d'obtenir des résultats représentant de nombreux cas d'exécution différents et donc afin de renforcer la validité des résultats.

Une exécution d'un algorithme concerne une variation de contrainte (un jeu de bornes minimum et maximum associées à certains attributs des données), un objectif de recommandation ou profil (des préférences, des informations sur les utilisateurs), un cas d'exécution (influencé ou non, avec ou sans réduction, autres) et une macro-exécution (l'algorithme est exécuté plusieurs fois afin de moyenniser les résultats obtenus). Les expérimentations comprennent six variations de contraintes, six objectifs de recommandation, plusieurs cas d'exécution et trente boucles de macro-exécution, pour un total de plusieurs milliers d'exécutions, chacune comprenant mille itérations. Ces quantités ont été choisies suite à un ensemble de tests initiaux qui nous ont permis de les poser afin qu'elles soient suffisantes pour permettre aux algorithmes d'atteindre la convergence et de continuer leur exécution en post-convergence dans le but d'y observer des résultats. Ces quantités sont aussi voulues suffisantes pour moyenniser divers résultats et ainsi obtenir une représentation d'un ensemble de cas d'exécution.

Pour calculer la fitness d'un *itemset*, les valeurs des attributs des *items* composant cet *itemset* sont sommées dans une fonction de *fitness* à objectif unique afin de simplifier les comparaisons et d'économiser du temps de calcul. C'est un procédé courant dans la littérature.

Nos expérimentations ont été réalisées en utilisant l'infrastructure de calcul Grid'5000 [34], soutenue par un groupe d'intérêt scientifique hébergé par Inria et incluant le CNRS, RENATER ainsi que plusieurs Universités et organisations.

Nous utilisons deux algorithmes de référence dans nos expérimentations, un algorithme génétique (GA) et un algorithme de renforcement (RA). L'utilisation de ces deux algorithmes de référence doit permettre de souligner l'adaptabilité des contributions proposées. L'algorithme génétique de référence utilise un opérateur de croisement de type "*one point*" et un opérateur de mutation de type "*bit string*". Sa structure a déjà été présentée dans les contributions. L'algorithme par renforcement choisi en référence est traditionnel [2] [373] (Figure 4.1, Algorithme 23), sa structure globale est similaire à celle de l'algorithme génétique. Q et N représentent respectivement le tableau des scores de renforcement associés à chaque *digit* et le tableau du nombre de fois où chaque *digit* a pris la valeur 1. N et Q sont mis à jour de manière traditionnelle, chaque *digit* modifié d voit sa valeur correspondante incrémentée de 1 dans N (Equation 4.1) et par la formule traditionnelle dans Q (Equation 4.2).

$$N[d]_{+} = 1 \tag{4.1}$$

$$Q[d]_{+} = 1/N[d] * (reward - Q[d]) \tag{4.2}$$

Des tests statistiques traditionnels de type *ranksum* sont effectués afin de s'assurer que les valeurs des critères d'évaluation (présentés plus bas), qui sont des moyennes sur toutes les macro-itérations, proviennent bien de distributions différentes par rapport à celles de l'algorithme de référence.

Dans les tableaux de résultats présentés ci-après, les lignes % ↗, % → et % ↘ font respectivement référence aux pourcentages de résultats des critères d'évaluation qui sont augmentés, inchangés et diminués par rapport à l'algorithme de référence (ou *baseline*), en tenant compte de tous les cas d'exécution. Pour compléter ces informations, nous effectuons des tests statistiques représentés par la ligne %*Pass* qui indique le pourcentage de tests réussis en tenant compte de

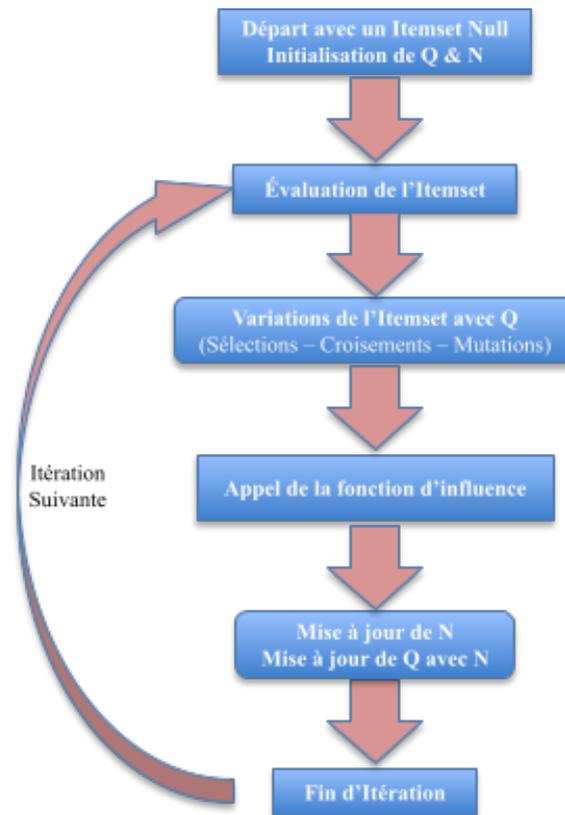


FIGURE 4.1 – RA appelant la fonction d'influence

tous les cas d'exécution. Précisons que l'hypothèse nulle d'un test statistique est que les valeurs des critères d'évaluation, de l'algorithme de référence et de l'une de ses versions sous approche (influence, réduction, autres), proviennent de la même distribution. Si l'hypothèse nulle est rejetée, c'est un succès ($p\text{-value} < 0,05$) et sinon c'est un échec ($p\text{-value} \geq 0,05$). Nous considérons arbitrairement qu'un critère d'évaluation permet de tirer des conclusions fiables si son %Pass est supérieur à 50%, ce qui correspond à une majorité de succès.

Nous avons aussi calculé plusieurs indicateurs de variations par rapport à la référence afin de mesurer l'impact des approches. Dans les tableaux de résultats, les lignes $\overline{\%Var}$, $|\overline{\%Var}|$ et $\overline{\%VarUP}$ font référence aux pourcentages moyens de variations par rapport à la référence, respectivement globales (tenant compte du signe de la variation), absolues (ne tenant pas compte du signe de la variation) et en augmentation (seules les variations positives sont prises en compte). Notons que pour calculer ces pourcentages, seuls les cas significatifs ont été retenus, c'est-à-dire les cas pour lesquels l'utilisation de l'approche est clairement profitable par rapport à la référence, au regard des scores de convergence et des scores maximum. Le seuil choisi pour retenir les cas significatifs a été déterminé de façon arbitraire.

Enfin, il faut noter que quelques éléments présentés en contributions ne se retrouvent pas dans les expérimentations en raison de leur peu de pertinence au regard des résultats obtenus. Nous les avons cependant conservés en tant que contributions car nous pensons qu'ils peuvent malgré tout être utiles dans d'autres contextes.

Algorithm 23 RA appelant la fonction d'influence

```

1: FUNCTION ra()
2:   initialisation()
3:   WHILE(condition not reached)
4:     ra_variations_with_Q()
5:     influence(itemsets)
6:     update_N_Q()
7:     iteration_end()
8:   finalisation()

```

4.2 Macro-processus

Le macro-processus adopté dans ces expérimentations s'articule autour de différentes étapes (Figure 4.2) dont les choix de conception respectifs sont arbitraires. Dans un premier temps, les données d'entrée brutes sont récupérées, correspondant à l'objectif de recommandation qui est construit à partir des informations de profil. Ces données se présentent sous la forme de tuples (*item* id, attribut 1, . . . , attribut n).

Les données sont normalisées. Nous choisissons d'associer chaque attribut à une contrainte "min" ou "max". S'il doit être maximisé, nous choisissons de laisser ses valeurs telles quelles et s'il doit être minimisé, ses valeurs prennent le complément à 1 pour représenter le fait que plus la valeur est petite, plus elle aura d'importance lorsque la somme se fera dans la fonction de *fitness*. Les contraintes sont définies sur les attributs sous forme de tuples (valeur minimale totale dans un *itemset*, valeur maximale totale dans un *itemset*). Des poids sont associés aux attributs, un poids représente la contribution moyenne d'un attribut dans la somme totale de toutes les valeurs d'attribut considérées. Un score d'*item* est défini comme la somme pondérée de ses attributs. Un score d'*itemset* est défini comme la somme de ses scores d'*items*.

L'algorithme est exécuté trente fois (macro-itérations). Lors d'une exécution, les vecteurs générés sont enregistrés dans *all_vectors* et les meilleurs vecteurs sont enregistrés dans *top_vectors*. Enfin, des sélections sont effectuées pour afficher les recommandations finales (à partir de *top_vectors*) et les valeurs moyennes des critères d'évaluation sont renvoyées pour toutes les exécutions. Divers éléments de visualisation sont enregistrés en conséquence.

4.3 Critères d'évaluation

D'après [432] [235], un système de recommandation vise à optimiser ses recommandations pour différents critères simultanément et ce à l'aide d'une fonction d'utilité ou fonction objectif. Les critères les plus courants [13] [207] [435] [3] [337] [194] sont l'accuracy, la serendipity, la novelty, le coverage et la diversity.

Nous décidons d'évaluer nos contributions en considérant plusieurs critères liés à la performance, à la qualité et au temps qui ont été introduits en contribution et qui sont proposés dans la littérature [207][194].

4.3.1 Critères de performance

Les critères de performance présentés dans cette section sont présents dans la littérature. Comme mentionné dans la section 4.2, un score d'*item* est défini comme la somme pondérée de ses valeurs d'attribut normalisées et un score d'*itemset* est défini comme la somme des scores de



FIGURE 4.2 – Macro-processus des expérimentations

ses *items*.

Convergence Iterations (CI) : Nombre moyen d'itérations nécessaires pour atteindre les points de convergence.

Max Score Iterations (MSI) : Nombre moyen d'itérations nécessaires pour atteindre les scores maximum.

Convergence Score (CS) : Score maximum moyen atteint sur les points de convergence.

Max Score (MS) : Score maximum moyen atteint.

Execution Time (ET) : Temps d'exécution moyen.

4.3.2 Critères de qualité

All Vectors Coverage (AVC) : Couverture prenant en compte tous les vecteurs générés. Représente le pourcentage de *digits* qui sont au moins une fois à 1 en considérant tous les vecteurs générés. Si chaque *digit* est au moins une fois à 1 dans l'ensemble des vecteurs considérés, la couverture est totale. Plus la couverture est élevée, plus le processus de recherche a trouvé de nombreux *items*. Ce critère est présent dans la littérature.

Constraints Completion (CC) : Pourcentage moyen de complétion des contraintes. Il représente dans quelle mesure les *items* recommandés remplissent les contraintes, ce qui peut être lié à la qualité des recommandations. Ce critère n'est pas présent dans la littérature.

E&E Power (EEP) : Part de l'exploration par rapport à l'exploitation. Voir les contributions pour plus de détails. Ce critère n'est pas présent dans la littérature.

4.3.3 Critères temporels

Min-Max Gap (DMMG) : Différence des écarts temporels minimum et maximum entre deux itérations d'exploration. Ce critère n'est pas présent dans la littérature.

Average Gap (AG) : Écart temporel moyen entre deux itérations d'exploration. Ce critère n'est pas présent dans la littérature.

Algorithm 24 Opérateurs de Variation d’Influence

```

1: FUNCTION explorer(itemsets)
2:   SET remove_quantity TO 0.25 * len(itemsets)
3:   SET itemsets TO sorted(itemsets)
4:   FOR k IN range(remove_quantity) :
5:     itemsets.pop()
6:   FOR k IN range(remove_quantity) :
7:     itemsets.append(generate_itemset())
8:
9: FUNCTION exploiter(itemsets)
10:  FOR p IN range(len(itemsets)) :
11:    SET done TO False
12:    WHILE done is False :
13:      SET rand TO rand(0, len(itemsets[p])-1)
14:      IF itemsets[p][rand] EQUALS 0 :
15:        SET itemsets[p][rand] TO 1
16:      IF constr_max_check(itemsets[p]) is False :
17:        SET done TO True # End WHILE loop.
18:        SET itemsets[p][rand] TO 0 # Cancel

```

4.4 Influence de l’E&E

4.4.1 Variations d’influence

Pour effectuer les variations d’influence, nous utilisons deux opérateurs, présentés dans [Algorithme 24](#).

Un opérateur force l’exploration (Ligne 1) et un autre opérateur force l’exploitation (Ligne 9). Ces opérateurs sont utilisés dans le processus d’influence (voir contributions) et peuvent être changés en fonction des besoins expérimentaux.

L’opérateur d’exploration est un opérateur connu de la littérature. Il est prévu ici de forcer l’exploration en supprimant les pires *itemsets* (Lignes 2, 4, 5) et en les remplaçant par de nouveaux générés aléatoirement (Ligne 7). Nous faisons l’hypothèse que ces modifications amènent des *items* inconnus correspondant à des régions inexplorées de l’espace de recherche. L’opérateur d’exploitation est développé dans le cadre de nos travaux. Il est destiné à forcer l’exploitation en conservant tous les *itemsets* et en faisant des mutations (Ligne 15) tant que les contraintes sont respectées (Lignes 12, 16, 17). Cela permet de conserver l’exploitation actuelle de l’espace de recherche, tout en faisant des tentatives d’amélioration du score qui renforcent l’exploitation.

Ces variations d’influence sont effectuées tant que l’indicateur d’influence ne respecte pas les bornes qui lui sont associées. Nous insistons aussi sur le fait que ces opérateurs de variations peuvent être changés, l’état de l’art en comportant un nombre conséquent.

4.4.2 Dataset MovieLens - Influence GA

Les résultats sont présentés dans [Tableau 4.1](#).

TABLE 4.1 – MovieLens - Influence sur GA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	62	65	40	63	100	50	44	2	82	77
% →	2	0	56	36	0	50	48	13	0	17
% ↘	37	35	4	1	0	0	8	85	18	5
%Pass	17	20	56	68	100	57	52	89	78	96

TABLE 4.2 – MovieLens - Influence sur GA - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	24	4	18	18	45	7	-1	-25	17	11
$ \overline{\%Var} $	27	7	18	18	45	7	2	25	83	33
$\overline{\%VarUP}$	33	8	18	18	45	12	3	No Data	113	40

Critères de performance

CI et *MSI* augmentent dans respectivement 62% et 65% des cas. Ainsi, l'influence augmente généralement le nombre d'itérations nécessaires pour atteindre la convergence et les scores maximum, ce qui est un inconvénient compte tenu du temps d'exécution mais qui est aussi un avantage vis-à-vis du problème de la convergence prématurée [122]. Cette conclusion doit être vue à la lumière de tests statistiques (%Pass) qui ne rejettent pas l'hypothèse nulle. Par conséquent, ces critères d'évaluation ne permettent pas de tirer des conclusions fiables dans le cadre de ces expérimentations.

CS et *MS* sont soit inchangés soit améliorés par l'influence. *CS* est inchangé dans 56% des cas mais obtient un important ratio d'augmentation (40% des cas). *MS* est augmenté dans 63% des cas et obtient un taux de stagnation significatif de 36% (des cas). Nous pouvons conclure que les fonctions d'influence conduisent à des scores au moins égaux à ceux de l'algorithme de base et dans de nombreux cas sont capables de les améliorer. Ces résultats sont validés par des tests statistiques. En effet, la plupart d'entre eux rejettent l'hypothèse nulle donc ces critères d'évaluation permettent de tirer des conclusions fiables dans le cadre de ces expérimentations.

Comme attendu, *ET* est augmenté par l'influence car des traitements supplémentaires sont apportés par le processus d'influence, ce qui est validé par les tests statistiques.

Critères de qualité

AVC est inchangé (50% des cas) ou augmente (50% des cas) par l'influence. Une augmentation d'*AVC* signifie que davantage de dimensions (*items*) sont utilisées dans le processus de recherche. Ces résultats sont validés par les tests statistiques, en effet la plupart d'entre eux rejettent l'hypothèse nulle.

CC reste inchangé (48% des cas) ou augmente (44% des cas) par l'influence, ce qui montre que le processus d'influence proposé contribue à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est ainsi améliorée par l'influence. De plus, ces résultats confirment que l'explicabilité de l'E&E est améliorée dans le sens où ils soulignent

la capacité du processus d'influence à remplir les volumes de contraintes et par conséquent à atteindre des régions spécifiques et des optima de l'espace de recherche. Ces résultats sont validés par les tests statistiques.

EEP diminue (85% des cas) par l'influence, ce qui signifie que les fonctions d'influence ont tendance à diminuer la part de l'exploration par rapport à l'exploitation et donc à renforcer l'exploitation. Ces résultats contribuent à l'explicabilité de l'E&E en montrant les grandes orientations du processus de recherche, induites par l'influence. Ces résultats sont validés par les tests statistiques.

Critères temporels

Nous pouvons constater que le processus d'influence permet d'augmenter les valeurs des critères temporels dans 82% des cas pour *DMMG* et dans 77% des cas pour *AG*. Cela signifie que la distribution temporelle des itérations d'exploration devient plus clairsemée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus prononcées. Cependant, dans certains cas, nous observons l'effet inverse (18% et 5% des cas), qui correspond à des phases d'exploration rapprochées avec des écarts temporels extrêmes réduits. Ces résultats sont validés par les tests statistiques, en effet la plupart d'entre eux rejettent l'hypothèse nulle.

Ces expérimentations confirment que les indicateurs temporels expliquent comment l'influence affecte la régularité de l'E&E.

Conclusion

Les critères étudiés dans cette section montrent un impact positif du processus d'influence avec des valeurs de ratio significatives pour la plupart des critères : deux critères de performance (*CS*, *MS*), tous les critères de qualité (*AVC*, *CC*, *EEP*) et tous les critères temporels (*DMMG*, *AG*).

En ce qui concerne les critères de performance, le processus d'influence permet à l'algorithme d'atteindre de meilleurs scores au détriment du nombre d'itérations et du temps d'exécution. Cela peut être ou non un problème en fonction des cas d'exécution. En ce qui concerne les critères de qualité, le processus d'influence fait que l'algorithme de référence évalue plus de solutions, ce qui correspond à une valeur de *CC* plus élevée dans certains cas. Du point de vue de la recommandation, nous rappelons ici que les préférences des utilisateurs sont encodées dans les contraintes, en conséquence *CC* porte des informations sur la qualité de la recommandation. Étant donné que la valeur de *CC* reste dans presque tous les cas influencés au moins égale à la valeur de *CC* des cas non influencés, nous pouvons conclure que le processus d'influence que nous proposons permet de trouver de meilleures recommandations. Par ailleurs, le processus d'influence a un impact significatif sur *EEP* en favorisant l'exploitation. En ce qui concerne les critères temporels, le processus d'influence conduit à un impact notable sur la régularité de l'E&E.

Pour résumer, nous pouvons affirmer que le processus d'influence a un impact positif sur le comportement de recherche. Nous notons qu'une analyse détaillée des résultats montre que certains cas d'influence sont plus performants que d'autres. Nous notons également que l'influence proposée n'a pas un impact positif dans tous les contextes, ce qui était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes. Nous soulignons également que les résultats confirment que le processus d'influence n'induit pas de comportements aléatoires car les scores restent au moins égaux à la référence.

De plus, nous confirmons que les indicateurs d'influence proposés aident à fournir des explications concernant l'E&E. En effet, ils représentent les comportements d'E&E sous de nouveaux angles comme le montrent *CC*, *EEP*, *DMMG* et *AG*.

Par ailleurs, les résultats de [Tableau 4.2](#) montrent des variations notables pour *CI*, *CS*, *MS*, *ET*, *AVC*, *EEP*, *DMMG* et *AG*. Ces variations viennent souligner l'impact de l'approche.

Finalement, nous pouvons conclure que ce travail est un pas en avant pour les processus d'influence de l'E&E ainsi que pour la représentation et l'explicabilité de l'E&E.

4.4.3 Dataset MovieLens - Influence RA

Dans cette section nous nous intéressons à l'autre algorithme de référence, l'algorithme de renforcement. Les résultats sont présentés dans [Tableau 4.3](#).

TABLE 4.3 – MovieLens - Influence sur RA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	19	30	34	28	100	71	31	19	89	71
% →	0	1	37	47	0	27	40	6	1	28
% ↘	81	69	29	25	0	2	28	76	10	0
%Pass	45	38	54	58	100	67	39	79	93	95

TABLE 4.4 – MovieLens - Influence sur RA - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	-3	-2	5	5	38	2	0	0	55	No Data
$ \overline{\%Var} $	9	2	5	5	38	2	1	3	55	No Data
$\overline{\%VarUP}$	11	1	5	5	38	4	2	4	55	No Data

Critères de performance

CI et *MSI* diminuent majoritairement par l'influence dans 81% et 69% des cas respectivement. Cela signifie que l'influence diminue généralement le nombre d'itérations nécessaires pour atteindre la convergence et les scores maximum, ce qui est un avantage compte tenu du temps d'exécution mais qui est aussi un inconvénient concernant le problème de convergence prématurée.

CS et *MS* sont inchangés (37%, 47% des cas), améliorés (34%, 28% des cas) et diminués (29%, 25% des cas) par l'influence. Ainsi, un nombre important de cas d'influence conduisent à des scores au moins égaux à ceux de l'algorithme de référence.

Comme attendu, *ET* est raisonnablement augmenté par l'influence, dans tous les cas.

Critères de qualité

AVC est inchangé (27% des cas) ou augmente (71% des cas) par l'influence. Une augmentation de *AVC* signifie que davantage de dimensions (*items*) sont utilisées dans le processus de

recherche.

CC reste inchangé (40% des cas), augmente (31% des cas) et diminue (28% des cas) par l'influence, ce qui montre que le processus d'influence proposé contribue dans de nombreux cas à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est améliorée dans ces cas.

EEP diminue (76% des cas) par l'influence, ce qui signifie que les fonctions d'influence ont tendance à diminuer la part de l'exploration par rapport à l'exploitation et donc à renforcer l'exploitation.

Critères temporels

Les valeurs de *DMMG* et *AG* sont augmentées par l'influence (89% et 71% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus clairsemée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus prononcées.

Conclusion

Ces expérimentations confirment les conclusions tirées pour l'algorithme génétique. Le processus d'influence a un impact positif, ce qui confirme la généralité de l'approche proposée.

Par ailleurs, les résultats de [Tableau 4.4](#) montrent des variations notables pour *CI*, *CS*, *MS*, *ET* et *DMMG*. Ces variations viennent souligner l'impact de l'approche.

4.4.4 Dataset Riid - Influence GA

Pour confirmer la généralité du processus d'influence proposé, nous effectuons des évaluations similaires sur un second jeu de données (Riid). Les résultats sont présentés dans [Tableau 4.5](#).

TABLE 4.5 – Riid - Influence sur GA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	44	67	69	61	100	34	31	1	72	74
% →	1	1	28	38	0	66	67	33	2	18
% ↘	55	32	3	1	0	0	2	65	26	8
%Pass	27	17	64	72	100	33	52	77	62	95

TABLE 4.6 – Riid - Influence sur GA - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	29	5	15	15	49	2	2	-24	171	36
$ \overline{\%Var} $	31	5	15	15	49	2	2	24	210	47
$\overline{\%VarUP}$	34	6	15	15	49	5	5	No Data	324	70

Critères de performance

CI diminue majoritairement par l'influence dans 55% des cas et *MSI* augmente majoritairement par l'influence avec 67% (des cas). Ainsi, l'influence diminue généralement le nombre d'itérations nécessaires pour atteindre la convergence, ce qui est un avantage compte tenu du temps d'exécution mais qui est aussi un inconvénient concernant le problème d'une possible convergence prématurée.

CS et *MS* augmentent (69%, 61% des cas) par l'influence. Dans 28% des cas pour *CS* et dans 38% des cas pour *MS*, les scores sont égaux à ceux de l'algorithme de référence.

Comme attendu, *ET* est augmenté par l'influence, dans tous les cas.

Critères de qualité

AVC est inchangé (66% des cas) ou augmente (34% des cas) par l'influence. Une augmentation de *AVC* signifie que davantage de dimensions (*items*) sont utilisés dans le processus de recherche.

CC reste inchangé (67% des cas) et augmente (31% des cas) par l'influence, ce qui montre que le processus d'influence proposé contribue dans de nombreux cas à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est améliorée dans ces cas.

EEP diminue (65% des cas) par l'influence, ce qui signifie que les fonctions d'influence ont tendance à diminuer la part de l'exploration par rapport à l'exploitation et donc à renforcer l'exploitation.

Critères temporels

Les valeurs de *DMMG* et *AG* sont augmentées par l'influence (72% et 74% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus clairsemée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus prononcées.

Conclusion

Ces expérimentations confirment les conclusions tirées pour le jeu de données MovieLens, bien que certains pourcentages diffèrent par rapport au jeu de données MovieLens. Le processus d'influence a un impact positif, ce qui confirme la généralité de l'approche proposée.

Par ailleurs, les résultats de [Tableau 4.6](#) montrent des variations notables pour *CI*, *CS*, *MS*, *ET*, *EEP*, *DMMG* et *AG*. Ces variations viennent souligner l'impact de l'approche.

4.4.5 Dataset Riiid - Influence RA

Nous analysons maintenant les expérimentations réalisées sur le jeu de données Riiid portant sur l'influence de l'E&E de RA. Les résultats sont présentés dans [Tableau 4.7](#).

Critères de performance

CI diminue majoritairement par l'influence avec 81% (des cas) et *MSI* diminue majoritairement par l'influence avec 73% (des cas). Ainsi, l'influence diminue généralement le nombre d'itérations nécessaires pour atteindre la convergence, ce qui est un avantage compte tenu du temps d'exécution mais qui est aussi un inconvénient concernant le problème de convergence prématurée.

TABLE 4.7 – Riiid - Influence sur RA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	18	26	42	37	92	58	23	6	48	28
% →	0	1	25	34	2	41	50	2	1	40
% ↘	81	73	33	29	6	0	26	92	51	32
%Pass	40	37	54	57	90	52	37	84	43	70

TABLE 4.8 – Riiid - Influence sur RA - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	-2	-1	7	5	2	2	2	-9	-11	-10
$ \overline{\%Var} $	13	3	7	5	3	2	2	9	24	15
$\overline{\%VarUP}$	12	2	7	5	3	3	3	No Data	12	7

CS et *MS* augmentent (42%, 37% des cas) par l'influence. Dans 25% des cas pour *CS* et dans 34% des cas pour *MS*, les scores sont égaux à ceux de l'algorithme de référence et pour le reste des cas les scores sont inférieurs à ceux de l'algorithme de référence.

Comme attendu, *ET* est augmenté par l'influence (92% des cas).

Critères de qualité

AVC est inchangé (41% des cas) ou augmente (58% des cas) par l'influence. Une augmentation de *AVC* signifie que davantage de dimensions (*items*) sont utilisés dans le processus de recherche.

CC reste inchangé (50% des cas), diminue (26% des cas) et augmente (23% des cas) par l'influence, ce qui montre que le processus d'influence proposé contribue dans de nombreux cas à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est améliorée dans ces cas.

EEP diminue (92% des cas) par l'influence, ce qui signifie que les fonctions d'influence ont tendance à diminuer la part de l'exploration par rapport à l'exploitation et donc à renforcer l'exploitation.

Critères temporels

Les valeurs de *DMMG* et *AG* sont augmentées par l'influence (48%, 28% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus clairsemée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus prononcées. Les valeurs de *DMMG* et *AG* sont aussi diminuées par l'influence (51%, 32% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus dense et que les différences temporelles extrêmes entre ces itérations ont tendance à être moins prononcées.

Conclusion

Ces expérimentations confirment les conclusions tirées pour le jeu de données MovieLens. Le processus d'influence a un impact positif dans de nombreux cas et il est source de modifications dans le comportement d'E&E. Ces résultats confirment donc la généralité de l'approche proposée.

Par ailleurs, les résultats de [Tableau 4.8](#) montrent des variations notables pour *CI*, *CS*, *MS*, *EEP*, *DMMG* et *AG*. Ces variations viennent souligner l'impact de l'approche.

4.4.6 Conclusion

Les critères de performance, de qualité et d'E&E étudiés montrent des impacts positifs du processus d'influence dans de nombreux cas d'exécution. L'approche proposée permet d'influencer le comportement d'E&E et les indicateurs d'influence proposés aident à fournir des explications concernant l'E&E. En effet, ils représentent les comportements d'E&E sous de nouveaux angles.

Les résultats obtenus dépendent des cas d'exécution, ainsi l'influence proposée n'a pas un impact positif dans tous les contextes.

Les résultats confirment que le processus d'influence n'induit pas de comportements aléatoires.

Les résultats concernant les variations par rapport à la référence montrent des variations notables, ce qui vient souligner l'impact de l'approche.

Les résultats sont confirmés dans différents contextes (datasets, algorithmes, contraintes et autres), ce qui confirme la généralité de l'approche proposée.

Nous pouvons conclure que ce travail est un pas en avant pour les processus d'influence de l'E&E ainsi que pour la représentation et l'explicabilité de l'E&E.

4.5 Opérateurs de variation pour l'influence de l'E&E

4.5.1 Introduction

Ces expérimentations visent à juger de la pertinence des opérateurs de variations proposés au regard des capacités d'E&E. Le jeu de données utilisé est MovieLens. Nous n'avons pas jugé nécessaire d'effectuer des expérimentations sur le jeu de données Riid car les résultats obtenus sur MovieLens permettent de juger de la pertinence des opérateurs de variations proposés au regard des capacités d'E&E. Nous ne nous intéressons qu'aux critères d'évaluation ayant un lien avec le comportement d'E&E. De plus, pour chaque couple "objectif de recommandation / variation de contraintes" nous exécutons tous les algorithmes proposés en contribution. Un algorithme génétique classique nous sert de référence de comparaison, les tests statistiques ont été réalisés par rapport à ce dernier.

4.5.2 Résultats

Les résultats sont présentés dans [Tableau 4.9](#).

Qualité

AVC stagne (50% des cas) et diminue (41% des cas). Une diminution de l'AVC signifie que moins de dimensions (*items*) sont utilisées dans le processus de recherche, ce qui peut être considéré comme négatif car cela montre des difficultés à explorer l'espace de recherche par

TABLE 4.9 – MovieLens - Opérateurs de variation pour l'influence de l'E&E

	Qualité		Temporel	
	AVC	EEP	DMMG	AG
% ↗	9	67	44	52
% →	50	4	6	21
% ↘	41	28	51	27
%Pass	47	96	76	87

rapport à la référence. Ce critère d'évaluation ne permet pas de tirer des informations fiables car seulement 47% des tests statistiques rejettent l'hypothèse nulle.

EEP augmente (67% des cas) et diminue (28% des cas). Les cas de diminution signifient que les opérateurs proposés diminuent la part que représente l'exploration par rapport à l'exploitation et donc qu'ils renforcent potentiellement l'exploitation. Les cas d'augmentation signifient que les opérateurs proposés augmentent la part que représente l'exploration par rapport à l'exploitation et donc qu'ils renforcent potentiellement l'exploration. Ces résultats contribuent à expliquer l'E&E en montrant les grandes orientations du processus de recherche en matière d'E&E. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 96% des tests statistiques rejettent l'hypothèse nulle.

Temporalité E&E

DMMG augmente (44% des cas) et diminue (51% des cas). AG augmente (52% des cas), stagne (21% des cas) et diminue (27% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus clairsemée ou plus compactée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment les opérateurs proposés impactent la régularité de l'E&E. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 76% et 87% des tests statistiques rejettent l'hypothèse nulle.

4.5.3 Conclusion

Nous pouvons conclure que les opérateurs proposés favorisent l'exploration et l'exploitation en fonction du cas d'exécution et impactent la régularité de l'E&E. Ils ont donc un impact sur le comportement de recherche. Notons qu'une analyse détaillée des résultats montre que certains opérateurs proposés sont plus impactant que d'autres. Nous notons également que les opérateurs proposés n'ont pas d'impact dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes.

Les opérateurs proposés provoquent donc des comportements d'E&E spécifiques, ce qui les rend potentiellement intéressants dans le cadre d'une démarche d'influence de l'E&E.

Les opérateurs proposés pourraient donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation, toujours au regard de la performance, de la qualité et de l'E&E.

4.6 Réduction des données

Nous présentons dans cette section les résultats de nos expérimentations sur la réduction des données. Toutes les sous-sections montrent des résultats menant à des conclusions similaires.

4.6.1 Dataset MovieLens - Réduction et GA

Les résultats sont présentés dans [Tableau 4.10](#).

TABLE 4.10 – MovieLens - Réduction sur GA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	44	24	43	47	3	61	15	0	52	5
% →	6	6	21	19	7	39	35	10	6	18
% ↘	50	71	36	34	90	0	50	89	41	77
%Pass	50	52	79	81	93	62	74	88	87	89

TABLE 4.11 – MovieLens - Réduction sur GA - Variations / Baseline

	Performance					Qualité		
	CI	MSI	CS	MS	ET	AVC	CC	EEP
$\overline{\%Var}$	29	-22	41	40	-71	13	-4	-88
$ \overline{\%Var} $	65	28	41	40	71	13	5	88
$\overline{\%VarUP}$	103	8	41	40	No Data	20	4	No Data

Performance

CI augmente (44% des cas) et diminue (50% des cas) donc la réduction augmente et diminue le nombre d'itérations requises pour atteindre la convergence, ce qui est respectivement négatif et positif pour le temps d'exécution et potentiellement négatif si cela mène à des convergences prématurées. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 50% des tests statistiques rejettent l'hypothèse nulle.

MSI est principalement diminué (71% des cas) donc la réduction diminue le nombre d'itérations requises pour atteindre les scores maximum. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 52% des tests statistiques rejettent l'hypothèse nulle.

CS et MS augmentent (43%, 47% des cas) et stagnent (21%, 19% des cas). La réduction permet donc principalement d'augmenter les scores atteints en convergence ainsi que les scores maximum. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 79% et 81% des tests statistiques rejettent l'hypothèse nulle.

ET diminue (90% des cas) principalement donc la réduction diminue les temps d'exécution. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 93% des tests statistiques rejettent l'hypothèse nulle.

Qualité

AVC augmente (61% des cas) et stagne (39% des cas). La réduction permet donc de plus facilement explorer l'espace de recherche mais c'est logique et attendu car l'espace de recherche diminue lors des opérations de réduction. Ce critère d'évaluation permet de tirer des informations fiables car 62% des tests statistiques rejettent l'hypothèse nulle.

CC stagne (35% des cas) et diminue (50% des cas) donc la réduction ne contribue pas à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est ainsi diminuée. De plus, ces résultats améliorent l'explicabilité de l'E&E car ils soulignent l'incapacité du processus de recherche à remplir les volumes de contraintes et par conséquent à atteindre des régions spécifiques et des optima de l'espace de recherche. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 74% des tests statistiques rejettent l'hypothèse nulle.

EEP diminue (89% des cas) principalement ce qui signifie que la réduction diminue la part que représente l'exploration par rapport à l'exploitation et donc qu'elle renforce potentiellement l'exploitation. Ces résultats contribuent à expliquer l'E&E en montrant les grandes orientations du processus de recherche en matière d'E&E. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 88% des tests statistiques rejettent l'hypothèse nulle.

Temporalité E&E

DMMG augmente (52% des cas) et diminue (41% des cas). AG diminue (77% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus compactée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment la réduction impacte la régularité de l'E&E. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 87% et 89% des tests statistiques rejettent l'hypothèse nulle.

Conclusion

Les résultats présentés ci-dessus montrent un impact positif du processus de réduction. Cependant, la fonction de réduction optimale à utiliser dépend du cas d'application et donc de la structure de l'espace de recherche et des contraintes posées.

Concernant les critères de performance, la réduction permet d'atteindre de meilleurs scores en moins d'itérations et dans des temps d'exécution moindres. En ce qui concerne les critères de qualité, la réduction permet l'évaluation de solutions plus diverses mais elle peine à conserver de bonnes complétions de contraintes. Nous rappelons ici que les préférences des utilisateurs sont encodées dans les contraintes, par conséquent CC porte des informations sur la qualité de la recommandation. Comme CC diminue, nous pouvons conclure que la réduction ne permet pas de trouver de meilleures recommandations. Par ailleurs, d'après EEP, la réduction favorise l'exploitation de l'espace de recherche. Concernant les critères temporels, la réduction impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

En résumé, la réduction a un impact sur le comportement de recherche. Notons qu'une analyse détaillée des résultats montre que certaines fonctions de réduction sont plus performantes que d'autres. Nous notons également que la réduction n'a pas d'impact positif dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes.

Par ailleurs, les résultats de [Tableau 4.11](#) montrent des variations notables pour *CI*, *MSI*, *CS*, *MS*, *ET*, *AVC* et *EEP*. Ces variations viennent souligner l'impact de l'approche.

La réduction des données pourrait donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation, toujours au regard de la performance, de la qualité et de l'E&E.

Ces résultats sont prometteurs pour nos futurs travaux sur la réduction de données par échantillonnage statistiquement conscient ainsi que sur l'influence de l'E&E.

4.6.2 Dataset MovieLens - Réduction et RA

Pour montrer l'adaptabilité des processus de réduction proposés, nous choisissons de les appliquer à un algorithme de renforcement traditionnel. Les résultats sont présentés dans [Tableau 4.12](#).

TABLE 4.12 – MovieLens - Réduction sur RA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	38	31	41	43	1	68	23	0	46	0
% →	12	13	25	23	13	31	35	14	19	64
% ↘	50	56	33	34	87	1	42	86	35	35
%Pass	35	20	75	76	87	58	72	85	74	86

TABLE 4.13 – MovieLens - Réduction sur RA - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	24	-4	32	29	-68	38	-1	-51	3	No Data
$ \overline{\%Var} $	54	7	32	29	68	38	4	51	75	No Data
$\overline{\%VarUP}$	108	6	32	29	No Data	52	5	No Data	107	No Data

Performance

CI augmente (38% des cas) et diminue (50% des cas) donc la réduction augmente et diminue le nombre d'itérations requises pour atteindre la convergence, ce qui est respectivement négatif et positif pour le temps d'exécution et potentiellement négatif si cela mène à des convergences prématurées. Ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 35% des tests statistiques rejettent l'hypothèse nulle.

MSI augmente (31% des cas) et diminue (56% des cas) donc la réduction augmente et diminue le nombre d'itérations requises pour atteindre les scores maximum. Ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 20% des tests statistiques rejettent l'hypothèse nulle.

CS et MS augmentent (41%, 43% des cas) et stagnent (25%, 23% des cas) principalement donc la réduction permet d'augmenter les scores atteints en convergence ainsi que les scores maximum. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 75% et 76% des tests statistiques rejettent l'hypothèse nulle.

ET diminue (87% des cas) principalement donc la réduction diminue les temps d'exécution. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 87% des tests statistiques rejettent l'hypothèse nulle.

Qualité

AVC augmente (68% des cas) et stagne (31% des cas). Une augmentation de l'AVC signifie que davantage de dimensions (*items*) sont utilisées dans le processus de recherche. Ce critère d'évaluation permet de tirer des informations fiables car 58% des tests statistiques rejettent l'hypothèse nulle.

CC stagne (35% des cas) et diminue (42% des cas) donc la réduction ne contribue pas à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est ainsi diminuée. De plus, ces résultats améliorent l'explicabilité de l'E&E car ils soulignent l'incapacité du processus de recherche à remplir les volumes de contraintes et par conséquent à atteindre des régions spécifiques et des optima de l'espace de recherche. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 72% des tests statistiques rejettent l'hypothèse nulle.

EEP diminue (86% des cas) principalement ce qui signifie que la réduction diminue la part que représente l'exploration par rapport à l'exploitation et donc qu'elle renforce potentiellement l'exploitation. Ces résultats contribuent à expliquer l'E&E en montrant les grandes orientations du processus de recherche en matière d'E&E. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 85% des tests statistiques rejettent l'hypothèse nulle.

Temporalité E&E

DMMG augmente (46% des cas) et diminue (35% des cas). AG diminue (35% des cas) et stagne (64% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus compactée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment la réduction impacte la régularité de l'E&E. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 74% et 86% des tests statistiques rejettent l'hypothèse nulle.

Conclusion

Les résultats montrent des valeurs similaires par rapport aux résultats de l'algorithme génétique et donc un impact positif du processus de réduction. Des conclusions similaires au cas de l'algorithme génétique peuvent en être tirées.

Par ailleurs, les résultats de [Tableau 4.13](#) montrent des variations notables pour *CI*, *MSI*, *CS*, *MS*, *ET*, *AVC*, *CC*, *EEP* et *DMMG*. Ces variations viennent souligner l'impact de l'approche.

4.6.3 Dataset Riid - Réduction et GA

Les résultats sont présentés dans [Tableau 4.14](#).

TABLE 4.14 – Riid - Réduction sur GA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	32	17	43	37	1	47	12	0	44	15
% →	3	2	16	19	2	53	39	4	2	9
% ↘	65	82	41	44	97	0	49	96	54	76
%Pass	48	64	83	84	98	40	77	96	87	90

TABLE 4.15 – Riid - Réduction sur GA - Variations / Baseline

	Performance					Qualité			
	CI	MSI	CS	MS	ET	AVC	CC	EEP	
$\overline{\%Var}$	-17	-30	39	36	-60	3	-3	-91	
$ \overline{\%Var} $	37	32	39	36	60	3	6	91	
$\overline{\%VarUP}$	36	3	39	36	No Data	7	9	No Data	

Performance

CI et MSI diminuent (65%, 82% des cas) donc la réduction diminue le nombre d'itérations requises pour atteindre la convergence ainsi que pour atteindre les scores maximum. Cela est positif pour le temps d'exécution mais potentiellement négatif si cela mène à des convergences prématurées.

CS et MS augmentent (43%, 37% des cas) et diminuent (41%, 44% des cas) donc la réduction permet dans de nombreux cas d'augmenter les scores atteints en convergence ainsi que les scores maximum.

ET diminue (97% des cas) principalement donc la réduction diminue les temps d'exécution.

Qualité

AVC augmente (47% des cas) et stagne (53% des cas). Une augmentation de l'AVC signifie que davantage de dimensions (*items*) sont utilisées dans le processus de recherche, ce qui est positif au regard du comportement d'E&E.

CC stagne (39% des cas) et diminue (49% des cas) donc la réduction ne contribue pas à trouver des *itemsets* qui remplissent mieux les contraintes.

EEP diminue (96% des cas) principalement ce qui signifie que la réduction diminue la part que représente l'exploration par rapport à l'exploitation et donc qu'elle renforce potentiellement l'exploitation.

Temporalité E&E

DMMG augmente (44% des cas) et diminue (54% des cas). AG diminue (76% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus compactée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment la réduction impacte la régularité de l'E&E.

Conclusion

Les résultats présentés ci-dessus montrent un impact positif du processus de réduction dans de nombreux cas. Cependant, la fonction de réduction optimale à utiliser dépend du cas d'application et donc de la structure de l'espace de recherche et des contraintes posées.

Concernant les critères de performance, la réduction permet dans de nombreux cas d'atteindre de meilleurs scores en moins d'itérations et dans des temps d'exécution moindres. En ce qui concerne les critères de qualité, la réduction permet l'évaluation de solutions plus diverses mais elle peine à conserver de bonnes complétions de contraintes. Par ailleurs, d'après EEP, la réduction favorise l'exploitation de l'espace de recherche. Concernant les critères temporels, la réduction impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

En résumé, la réduction a un impact sur le comportement de recherche. Notons qu'une analyse détaillée des résultats montre que certaines fonctions de réduction sont plus performantes que d'autres. Nous notons également que la réduction n'a pas d'impact positif dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes.

Par ailleurs, les résultats de [Tableau 4.15](#) montrent des variations notables pour *CI*, *MSI*, *CS*, *MS*, *ET*, *AVC*, *CC* et *EEP*. Ces variations viennent souligner l'impact de l'approche.

4.6.4 Dataset Riid - Réduction et RA

Les résultats sont présentés dans [Tableau 4.16](#)

TABLE 4.16 – Riid - Réduction sur RA

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	27	24	51	45	0	73	23	1	36	0
% →	2	3	13	20	2	27	30	3	1	4
% ↘	71	73	35	35	98	0	48	96	63	96
%Pass	33	26	81	82	97	65	73	96	90	96

TABLE 4.17 – Riid - Réduction sur RA - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	-17	-6	38	34	-71	14	0	-42	-20	-82
$ \overline{\%Var} $	23	7	38	34	71	14	4	42	114	82
$\overline{\%VarUP}$	11	1	38	34	No Data	18	7	No Data	150	No Data

Performance

CI et MSI diminuent (71%, 73% des cas) donc la réduction diminue le nombre d'itérations requises pour atteindre la convergence ainsi que pour atteindre les scores maximum. Cela est positif pour le temps d'exécution mais potentiellement négatif si cela mène à des convergences prématurées.

CS et MS augmentent (51%, 45% des cas) et diminuent (35%, 35% des cas) donc la réduction permet dans de nombreux cas d'augmenter les scores atteints en convergence ainsi que les scores maximum.

ET diminue (98% des cas) principalement donc la réduction diminue les temps d'exécution.

Qualité

AVC augmente (73% des cas) et stagne (27% des cas). Une augmentation de l'AVC signifie que davantage de dimensions (*items*) sont utilisées dans le processus de recherche, ce qui est positif au regard du comportement d'E&E.

CC augmente (23% des cas), stagne (30% des cas) et diminue (48% des cas) donc la réduction ne contribue pas à trouver des *itemsets* qui remplissent mieux les contraintes.

EEP diminue (96% des cas) principalement ce qui signifie que la réduction diminue la part que représente l'exploration par rapport à l'exploitation et donc qu'elle renforce potentiellement l'exploitation.

Temporalité E&E

DMMG augmente (36% des cas) et diminue (63% des cas). AG diminue (96% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus compactée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment la réduction impacte la régularité de l'E&E.

Conclusion

Les résultats présentés ci-dessus montrent un impact positif du processus de réduction dans de nombreux cas. Cependant, la fonction de réduction optimale à utiliser dépend du cas d'application et donc de la structure de l'espace de recherche et des contraintes posées.

Concernant les critères de performance, la réduction permet dans de nombreux cas d'atteindre de meilleurs scores en moins d'itérations et dans des temps d'exécution moindres. En ce qui concerne les critères de qualité, la réduction permet l'évaluation de solutions plus diverses mais elle peine à conserver de bonnes complétions de contraintes. Par ailleurs, d'après EEP, la réduction favorise l'exploitation de l'espace de recherche. Concernant les critères temporels, la réduction impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

En résumé, la réduction a un impact sur le comportement de recherche. Notons qu'une analyse détaillée des résultats montre que certaines fonctions de réduction sont plus performantes que d'autres. Nous notons également que la réduction n'a pas d'impact positif dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes.

Par ailleurs, les résultats de [Tableau 4.17](#) montrent des variations notables pour *CI*, *MSI*, *CS*, *MS*, *ET*, *AVC*, *CC*, *EEP*, *DMMG* et *AG*. Ces variations viennent souligner l'impact de l'approche.

4.6.5 Conclusion

Les critères de performance, de qualité et d'E&E étudiés montrent des impacts positifs du processus de réduction dans de nombreux cas d'exécution. Le processus de réduction impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

Le processus de réduction pourrait donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation, toujours au regard de la performance, de la qualité et de l'E&E.

Les résultats obtenus dépendent des cas d'exécution, ainsi le processus de réduction proposé n'a pas un impact positif dans tous les contextes.

Les résultats concernant les variations par rapport à la référence montrent des variations notables, ce qui vient souligner l'impact de l'approche.

Les résultats sont confirmés dans différents contextes (datasets, algorithmes, contraintes et autres), ce qui confirme la généralité de l'approche proposée.

Ces résultats sont prometteurs pour nos futurs travaux sur la réduction de données par échantillonnage statistiquement conscient ainsi que sur l'influence de l'E&E.

4.7 Réduction des données et influence de l'E&E

Le but est de constater si l'utilisation conjointe de la réduction et de l'influence (R_I) permet d'obtenir des résultats plus intéressants par rapport aux autres cas précédemment analysés (réduction seule et influence seule).

4.7.1 Dataset MovieLens - Réduction, Influence et GA

Pour chaque couple "objectif de recommandation / variation de contraintes" nous exécutons toutes les fonctions d'influence en utilisant le processus de réduction le plus performant d'après les résultats d'expérimentations de la section correspondante.

Considérant l'objectif de ces expérimentations, nous avons trois références possibles pour les comparaisons : l'algorithme génétique classique (G), l'algorithme génétique classique avec réduction (G_R) et l'algorithme génétique classique influencé (G_I). En analysant le détail des expérimentations précédentes sur la réduction et l'influence, nous constatons que les scores et temps d'exécution obtenus pour les G_R sont meilleurs que ceux des G et G_I. Nous présentons donc les résultats obtenus en analysant chaque critère d'évaluation de l'algorithme génétique classique à la fois réduit et influencé (G_R_I) par rapport aux critères d'évaluation de G_R. Ainsi, les tests statistiques ont aussi été réalisés par rapport à G_R.

Les résultats sont présentés dans [Tableau 4.18](#).

TABLE 4.18 – MovieLens - G_R_I

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	47	66	56	54	98	12	48	3	54	42
% →	2	2	43	46	2	88	48	53	13	50
% ↘	51	33	1	0	1	0	3	44	33	9
%Pass	16	24	56	67	99	11	54	54	46	72

Performance

CI augmente (47% des cas) et diminue (51% des cas) donc G_R_I augmente et diminue dans les mêmes proportions le nombre d'itérations requises pour atteindre la convergence, ce

TABLE 4.19 – MovieLens - G_R_I - Variations / Baseline

	Performance					Qualité		
	CI	MSI	CS	MS	ET	AVC	CC	EEP
$\overline{\%Var}$	-5	29	8	8	39	0	2	-17
$ \overline{\%Var} $	14	31	8	8	39	0	3	17
$\overline{\%VarUP}$	19	49	8	8	39	1	3	No Data

qui est respectivement négatif et positif pour le temps d'exécution et négatif si cela mène à des convergences prématurées. Cependant, ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 16% des tests statistiques rejettent l'hypothèse nulle.

MSI est principalement augmenté (66% des cas) donc G_R_I augmente le nombre d'itérations requises pour atteindre les scores maximum. Toutefois, ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 24% des tests statistiques rejettent l'hypothèse nulle.

CS et MS augmentent (56%, 54% des cas) et stagnent (43%, 46% des cas) donc G_R_I augmente les scores atteints en convergence ainsi que les scores maximum. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 56% et 67% des tests statistiques rejettent l'hypothèse nulle.

ET augmente (98% des cas) donc G_R_I augmente les temps d'exécution, ce qui était attendu car les traitements d'influence ont été ajoutés. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 99% des tests statistiques rejettent l'hypothèse nulle.

Qualité

AVC stagne (88% des cas) et augmente (12% des cas). Il faut préciser que AVC est totale ou presque dans l'ensemble des cas G_R, elle ne varie donc pas ou très peu pour les cas G_R_I. Ce critère d'évaluation ne permet pas de tirer des informations fiables car seulement 11% des tests statistiques rejettent l'hypothèse nulle.

CC stagne (48% des cas) et augmente (48% des cas) donc G_R_I contribue à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est ainsi augmentée. De plus, ces résultats améliorent l'explicabilité de l'E&E car ils soulignent la capacité du processus de recherche à remplir les volumes de contraintes et par conséquent à atteindre des régions spécifiques et des optima de l'espace de recherche. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 54% des tests statistiques rejettent l'hypothèse nulle.

EEP stagne (53% des cas) et diminue (44% des cas). G_R_I diminue donc la part que représente l'exploration par rapport à l'exploitation et donc renforce l'exploitation. Ces résultats contribuent à expliquer l'E&E en montrant les grandes orientations du processus de recherche en matière d'E&E. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 54% des tests statistiques rejettent l'hypothèse nulle.

Temporalité E&E

DMMG augmente (54% des cas) et diminue (33% des cas). AG stagne (50% des cas) et augmente (42% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus clairesmée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment G_R_I impacte la régularité de l'E&E. DMMG ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 46% des tests statistiques rejettent l'hypothèse nulle. AG permet de tirer des informations fiables dans le cadre de ces expérimentations car 72% des tests statistiques rejettent l'hypothèse nulle.

Conclusion

Ces expérimentations montrent que l'utilisation conjointe de méthodes de réduction des données et d'influence indépendante de l'E&E (R_I) a un impact positif sur les critères d'évaluation de la performance, de la qualité et du comportement d'E&E d'un algorithme de recherche. Comme attendu, au regard des expérimentations déjà menées, la jonction optimale de méthodes à utiliser dépend du cas d'application, de la structure de l'espace de recherche, des contraintes posées et des paramètres.

Concernant les critères de performance, la R_I permet d'atteindre de meilleurs scores en plus ou moins d'itérations et dans des temps d'exécution supérieurs. En ce qui concerne les critères de qualité, rappelons que les préférences des utilisateurs sont encodées dans les contraintes, par conséquent CC porte des informations sur la qualité de la recommandation. Comme CC augmente, nous pouvons conclure que la R_I permet de trouver de meilleures recommandations par rapport à la référence. Par ailleurs, d'après EEP, la R_I favorise l'exploitation de l'espace de recherche. Concernant les critères temporels, la R_I impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

En résumé, la R_I a un impact positif sur la performance, la qualité et les comportements d'E&E. Notons qu'une analyse détaillée (non présentée ici) des résultats montre que certaines fonctions d'influence sont plus performantes que d'autres. Nous notons également que la R_I n'a pas d'impact positif dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes.

Par ailleurs, les résultats de [Tableau 4.19](#) montrent des variations notables pour *CI*, *MSI*, *CS*, *MS*, *ET* et *EEP*. Ces variations viennent souligner l'impact de l'approche.

Ces résultats sont prometteurs pour nos futurs travaux touchant à la R_I.

4.7.2 Dataset MovieLens - Réduction, Influence et RA

Pour montrer l'adaptabilité de la R_I, nous choisissons de l'appliquer à un algorithme de renforcement traditionnel.

Les algorithmes par renforcement classique (RF), classique réduit (RF_R) et classique influencé (RF_I) nous servent de références. Pour rester conforme aux expérimentations précédentes, nous présentons les résultats obtenus en analysant chaque critère d'évaluation de l'algorithme par renforcement classique à la fois réduit et influencé (RF_R_I) par rapport aux critères d'évaluation de RF_R. Le but est de constater si l'utilisation conjointe de la réduction et de l'influence permet d'obtenir des résultats plus intéressants par rapport aux autres cas précédemment analysés (RF, RF_R, RF_I) et dont RF_R constitue donc la référence. Ainsi, les tests statistiques ont aussi été réalisés par rapport à RF_R.

Les résultats (Tableau 4.20) montrent des valeurs similaires par rapport aux résultats de l'algorithme génétique Tableau 4.18. Cependant, CI et MSI diminuent principalement, il y a plus de cas de diminution pour CS, MS et CC, DMMG augmente davantage, AG stagne davantage. Des conclusions similaires au cas de l'algorithme génétique peuvent être tirées.

Par ailleurs, les résultats de Tableau 4.21 montrent des variations notables pour *CI*, *MSI*, *CS*, *MS*, *ET* et *DMMG*. Ces variations viennent souligner l'impact de l'approche.

TABLE 4.20 – MovieLens - RF_R_I

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	19	28	31	26	98	31	16	0	63	13
% →	1	2	44	53	1	68	57	20	15	86
% ↘	80	70	25	21	1	0	27	80	21	1
%Pass	39	38	51	54	98	36	42	96	70	92

TABLE 4.21 – MovieLens - RF_R_I - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	6	1	9	6	45	0	0	-3	69	0
$ \%Var $	13	4	9	6	45	0	0	3	69	0
$\%VarUP$	19	5	9	6	45	No Data	No Data	No Data	69	No Data

4.7.3 Dataset Riid - Réduction, Influence et GA

Pour chaque couple “objectif de recommandation / variation de contraintes” nous exécutons toutes les fonctions d'influence en utilisant le processus de réduction le plus performant d'après les résultats d'expérimentations de la section correspondante (Section 4.6.3).

Les algorithmes génétique classique (G), génétique classique réduit (G_R) et génétique classique influencé (G_I) nous servent de références de comparaison.

Les résultats sont présentés dans Tableau 4.22.

Pour rester conforme aux expérimentations précédentes, nous présentons les résultats obtenus en analysant chaque critère d'évaluation de l'algorithme génétique classique à la fois réduit et influencé (G_R_I) par rapport aux critères d'évaluation de G_R. Le but est de constater si l'utilisation conjointe de la réduction et de l'influence permet d'obtenir des résultats plus intéressants par rapport aux autres cas précédemment analysés (G, G_R, G_I) et dont G_R constitue donc la référence. Ainsi, les tests statistiques ont aussi été réalisés par rapport à G_R. En somme, le protocole est le même, seul le jeu de données change.

Performance

CI augmente (41% des cas) et diminue (58% des cas) donc G_R_I augmente et diminue le nombre d'itérations requises pour atteindre la convergence, ce qui est respectivement négatif et positif pour le temps d'exécution et potentiellement négatif si cela mène à des convergences

TABLE 4.22 – Riid - G_R_I

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	41	72	59	59	100	2	35	2	37	32
% →	1	0	38	40	0	97	64	73	35	58
% ↘	58	27	3	1	0	1	0	25	28	10
%Pass	23	38	61	71	100	2	58	39	31	53

TABLE 4.23 – Riid - G_R_I - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	18	9	11	12	59	0	4	-16	-28	10
$ \%Var $	32	12	11	12	59	0	4	16	37	19
$\%VarUP$	44	24	11	12	59	No Data	6	No Data	29	100

prématurées. Ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 23% des tests statistiques rejettent l'hypothèse nulle.

MSI est principalement augmenté (72% des cas) donc G_R_I augmente le nombre d'itérations requises pour atteindre les scores maximum. Ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 38% des tests statistiques rejettent l'hypothèse nulle.

CS et MS augmentent (59%, 59% des cas) et stagnent (38%, 40% des cas) donc G_R_I augmente les scores atteints en convergence ainsi que les scores maximum. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 61% et 71% des tests statistiques rejettent l'hypothèse nulle.

ET augmente (100% des cas) donc G_R_I augmente les temps d'exécution, ce qui est normal car les traitements d'influence ont été ajoutés. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 100% des tests statistiques rejettent l'hypothèse nulle.

Qualité

AVC stagne (97% des cas). Il faut préciser que AVC est totale ou presque dans l'ensemble des cas G_R, elle ne varie donc pas ou très peu pour les cas G_R_I. Ce critère d'évaluation ne permet pas de tirer des informations fiables car seulement 2% des tests statistiques rejettent l'hypothèse nulle.

CC stagne (64% des cas) et augmente (35% des cas) donc G_R_I contribue à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est ainsi augmentée. De plus, ces résultats améliorent l'explicabilité de l'E&E car ils soulignent la capacité du processus de recherche à remplir les volumes de contraintes et par conséquent à atteindre des régions spécifiques et des optima de l'espace de recherche. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 58% des tests statistiques rejettent l'hypothèse nulle.

EEP stagne (73% des cas) et diminue (25% des cas). G_R_I diminue donc la part que représente l'exploration par rapport à l'exploitation et donc renforce l'exploitation. Ces résultats contribuent à expliquer l'E&E en montrant les grandes orientations du processus de recherche en matière d'E&E. Ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car 39% des tests statistiques rejettent l'hypothèse nulle.

Temporalité E&E

DMMG augmente (37% des cas) et diminue (28% des cas). AG stagne (58% des cas) et augmente (32% des cas). Cela signifie que la distribution temporelle des itérations d'exploration devient plus clairsemée et que les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment G_R_I impacte la régularité de l'E&E. DMMG ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 31% des tests statistiques rejettent l'hypothèse nulle. AG permet de tirer des informations fiables dans le cadre de ces expérimentations car 53% des tests statistiques rejettent l'hypothèse nulle.

Conclusion

Les résultats montrent que l'utilisation conjointe de méthodes statistiquement conscientes de réduction des données et d'influence indépendante de l'E&E (R_I) a un impact positif sur les critères d'évaluation de la performance, de la qualité et du comportement d'E&E d'un algorithme de recherche. Comme attendu, au regard des expérimentations déjà menées, la jonction optimale de méthodes à utiliser dépend du cas d'application, de la structure de l'espace de recherche, des contraintes posées et des paramètres.

Concernant les critères de performance, la R_I permet d'atteindre de meilleurs scores en plus ou moins d'itérations et dans des temps d'exécution supérieurs. En ce qui concerne les critères de qualité, rappelons que les préférences des utilisateurs sont encodées dans les contraintes, par conséquent CC porte des informations sur la qualité de la recommandation. Comme CC augmente dans de nombreux cas, nous pouvons conclure que la R_I permet de trouver de meilleures recommandations. Par ailleurs, d'après EEP, la R_I favorise l'exploitation de l'espace de recherche. Concernant les critères temporels, la R_I impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

En résumé, la R_I a un bon impact sur la performance, la qualité et les comportements d'E&E. Notons qu'une analyse détaillée des résultats montre que certaines fonctions d'influence sont plus performantes que d'autres. Nous notons également que la R_I n'a pas d'impact positif dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes.

Par ailleurs, les résultats de [Tableau 4.23](#) montrent des variations notables pour *CI*, *MSI*, *CS*, *MS*, *ET*, *CC*, *EEP*, *DMMG* et *AG*. Ces variations viennent souligner l'impact de l'approche.

4.7.4 Dataset Riiid - Réduction, Influence et RA

Pour chaque couple "objectif de recommandation / variation de contraintes" nous exécutons toutes les fonctions d'influence en utilisant le processus de réduction le plus performant d'après les résultats d'expérimentations de la section correspondante.

Les algorithmes par renforcement classique (RF), classique réduit (RF_R) et classique influencé (RF_I) nous servent de références de comparaison.

Les résultats sont présentés dans [Tableau 4.24](#).

Pour rester conforme aux expérimentations précédentes, nous présentons les résultats obtenus en analysant chaque critère d'évaluation de l'algorithme par renforcement classique à la fois réduit et influencé (RF_R_I) par rapport aux critères d'évaluation de RF_R. Le but est de constater si l'utilisation conjointe de la réduction et de l'influence permet d'obtenir des résultats plus intéressants par rapport aux autres cas précédemment analysés (RF, RF_R, RF_I) et dont RF_R constitue donc la référence. Ainsi, les tests statistiques ont aussi été réalisés par rapport à RF_R.

TABLE 4.24 – Riid - RF_R_I

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	18	27	29	31	89	12	18	0	25	12
% →	1	0	41	48	3	88	63	18	23	62
% ↘	81	73	30	22	8	1	19	82	52	27
%Pass	40	37	50	57	95	8	42	98	37	62

TABLE 4.25 – Riid - RF_R_I - Variations / Baseline

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
$\overline{\%Var}$	0	-1	6	5	6	0	2	-3	-12	4
$ \overline{\%Var} $	5	1	6	5	6	0	2	3	12	4
$\overline{\%VarUP}$	4	No Data	6	5	6	No Data	3	No Data	1	13

Performance

CI diminue (81% des cas), nous pouvons en déduire que la R_I diminue le nombre d'itérations requises pour atteindre la convergence, ce qui est positif pour le temps d'exécution et potentiellement négatif si cela mène à des convergences prématurées. Cependant, Ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 40% des tests statistiques rejettent l'hypothèse nulle.

MSI est principalement diminué (73% des cas) donc la R_I diminue le nombre d'itérations requises pour atteindre les scores maximum. Comme pour le précédent, ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 37% des tests statistiques rejettent l'hypothèse nulle.

CS et MS augmentent dans de nombreux cas (29%, 31% des cas) et stagnent (41%, 48% des cas) donc la R_I augmente les scores atteints en convergence ainsi que les scores maximum, dans de nombreux cas. Ces critères d'évaluation permettent de tirer des informations fiables dans le cadre de ces expérimentations car 50% et 57% des tests statistiques rejettent l'hypothèse nulle.

ET augmente (89% des cas) donc la R_I augmente les temps d'exécution, ce qui est normal car les traitements d'influence ont été ajoutés. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 95% des tests statistiques rejettent l'hypothèse nulle.

Qualité

AVC stagne (88% des cas). Il faut préciser que AVC est totale ou presque dans l'ensemble des cas RF_R, elle ne varie donc pas ou très peu pour les cas RF_R_I. Ce critère d'évaluation ne permet pas de tirer des informations fiables car seulement 8% des tests statistiques rejettent l'hypothèse nulle.

CC stagne (63% des cas), augmente (18% des cas) et diminue (19% des cas) donc RF_R_I contribue dans certains cas à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est ainsi augmentée. De plus, ces résultats soulignent la capacité du processus de recherche à remplir les volumes de contraintes et par conséquent à atteindre des régions spécifiques et des optima de l'espace de recherche. Ce critère d'évaluation ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 42% des tests statistiques rejettent l'hypothèse nulle.

EEP diminue (82% des cas). RF_R_I diminue donc la part que représente l'exploration par rapport à l'exploitation et donc renforce l'exploitation. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 98% des tests statistiques rejettent l'hypothèse nulle.

Temporalité E&E

DMMG augmente (25% des cas) et diminue (52% des cas). AG stagne (62% des cas) et diminue (27% des cas). Cela signifie que la distribution temporelle des itérations d'exploration est inchangée dans la plupart des cas, plus clairsemée dans certains cas, plus concentrée dans d'autres. Les différences temporelles extrêmes entre ces itérations ont tendance à être plus ou moins prononcées. Ces critères temporels expliquent comment RF_R_I impacte la régularité de l'E&E. DMMG ne permet pas de tirer des informations fiables dans le cadre de ces expérimentations car seulement 37% des tests statistiques rejettent l'hypothèse nulle. AG permet de tirer des informations fiables dans le cadre de ces expérimentations car 62% des tests statistiques rejettent l'hypothèse nulle.

Conclusion

Les résultats montrent que l'utilisation conjointe de méthodes statistiquement conscientes de réduction des données et d'influence indépendante de l'E&E (R_I) a un impact positif sur les critères d'évaluation de la performance, de la qualité et du comportement d'E&E d'un algorithme de recherche. Comme attendu, au regard des expérimentations déjà menées, la jonction optimale de méthodes à utiliser dépend du cas d'application, de la structure de l'espace de recherche, des contraintes posées et des paramètres.

Concernant les critères de performance, la R_I permet d'atteindre de meilleurs scores en plus ou moins d'itérations et dans des temps d'exécution supérieurs. En ce qui concerne les critères de qualité, rappelons que les préférences des utilisateurs sont encodées dans les contraintes, par conséquent CC porte des informations sur la qualité de la recommandation. Comme CC augmente dans certains cas, nous pouvons conclure que la R_I permet de trouver de meilleures recommandations dans ces cas. Par ailleurs, d'après EEP, la R_I favorise l'exploitation de l'espace de recherche. Concernant les critères temporels, la R_I impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

En résumé, la R_I peut avoir un bon impact sur la performance, la qualité et les comportements d'E&E. Notons qu'une analyse détaillée des résultats montre que certaines fonctions d'influence sont plus performantes que d'autres. Nous notons également que la R_I n'a pas

d'impact positif dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes.

Par ailleurs, les résultats de [Tableau 4.25](#) montrent des variations notables pour *CS*, *MS*, *ET*, *DMMG* et *AG*. Ces variations viennent souligner l'impact de l'approche.

4.7.5 Conclusion

Les critères de performance, de qualité et d'E&E étudiés montrent des impacts positifs de la R_I dans de nombreux cas d'exécution. La R_I impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques.

La R_I pourrait donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation, toujours au regard de la performance, de la qualité et de l'E&E.

Les résultats obtenus dépendent des cas d'exécution, ainsi la R_I n'a pas un impact positif dans tous les contextes.

Les résultats concernant les variations par rapport à la référence montrent des variations notables, ce qui vient souligner l'impact de l'approche.

Les résultats sont confirmés dans différents contextes (datasets, algorithmes, contraintes et autres), ce qui confirme la généralité de l'approche proposée.

Ces résultats sont prometteurs pour nos futurs travaux sur la R_I.

4.8 Réductions combinées

Nous présentons dans cette section les résultats des expérimentations concernant la combinaison de fonctions de réduction des données.

4.8.1 Dataset MovieLens - Réductions combinées - Référence GA

L'approche proposée dans cette section est utilisée seule pour trouver des solutions dans le cadre d'un problème d'optimisation et de recommandation. L'approche proposée est directe, elle n'opère pas de processus évolutionnaire, n'est donc pas une métaheuristique et n'est donc en théorie pas aussi performante qu'une approche métaheuristique classique. C'est la raison pour laquelle les critères d'évaluation considérés doivent prendre en compte le fait que l'approche proposée n'explore pas et n'exploite pas l'espace de recherche. Nous avons donc été contraints de considérer un nombre restreint de critères d'évaluation à savoir : le score maximum atteint, le temps d'exécution, la couverture et la complétion des contraintes. Seuls ces critères sont en commun entre l'approche proposée et les deux approches métaheuristicques classiques qui nous servent de référence (algorithme génétique et algorithme par renforcement).

Ce premier jeu d'expérimentations considère un algorithme génétique traditionnel comme référence.

Les résultats sont présentés dans [Tableau 4.26](#).

Performance

MS augmente (36% des cas) et diminue (58% des cas) donc le processus proposé mène à des scores maximum supérieurs et inférieurs à la référence. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 97% des tests statistiques rejettent l'hypothèse nulle.

TABLE 4.26 – MovieLens - Combinaisons de réductions sur GA

	Performance		Qualité	
	MS	ET	AVC	CC
% ↗	36	19	58	14
% →	6	0	19	14
% ↘	58	81	22	72
%Pass	97	97	72	100

TABLE 4.27 – MovieLens - Combinaisons de réductions sur GA - Variations / Baseline

	Performance		Qualité	
	MS	ET	AVC	CC
$\overline{\%Var}$	34	-37	14	-10
$ \overline{\%Var} $	34	49	14	10
%VarUP	34	27	25	No Data

ET diminue (81% des cas) principalement donc le processus proposé diminue les temps d'exécution. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 97% des tests statistiques rejettent l'hypothèse nulle.

Qualité

AVC augmente dans 58% des cas. Ce critère d'évaluation permet de tirer des informations fiables car 72% des tests statistiques rejettent l'hypothèse nulle.

CC diminue (72% des cas) donc le processus proposé ne contribue pas à trouver des *itemsets* qui remplissent mieux les contraintes. La qualité des recommandations est ainsi diminuée. Ce critère d'évaluation permet de tirer des informations fiables dans le cadre de ces expérimentations car 100% des tests statistiques rejettent l'hypothèse nulle.

Conclusion

Les résultats présentés ci-dessus montrent que le processus proposé peut avoir un impact positif sur le temps d'exécution, la couverture et le score. Cependant, il mène à de moins bonnes complétions de contraintes que la référence.

Notons qu'une analyse détaillée montre que le processus proposé obtient de meilleurs résultats dans certains contextes mais qu'il n'a pas d'impact positif dans tous les contextes, cela était attendu. Il est aussi intéressant de constater qu'il surpasse les scores de la référence dans plus d'un tiers des cas.

Par ailleurs, les résultats de [Tableau 4.27](#) montrent des variations notables pour *MS*, *ET*, *AVC* et *CC*. Ces variations viennent souligner l'impact de l'approche.

4.8.2 Dataset MovieLens - Réductions combinées - Référence RA

Ce deuxième jeu d'expérimentations considère un algorithme par renforcement traditionnel en référence. Les résultats ([Tableau 4.28](#)) montrent des valeurs similaires par rapport aux résultats

de l'algorithme génétique pour MS, AVC et CC. En revanche, concernant ET, le processus proposé présente dans presque tous les cas des temps d'exécution supérieurs à l'algorithme par renforcement.

Le processus proposé peut donc avoir un impact positif sur la couverture et le score. Cependant, il mène à de moins bonnes complétions de contraintes.

Notons qu'une analyse détaillée montre que le processus proposé obtient de meilleurs résultats dans certains contextes mais qu'il n'a pas d'impact positif dans tous les contextes, cela était attendu. Il est aussi intéressant de constater qu'il surpasse les scores de la référence dans plus d'un tiers des cas.

Par ailleurs, les résultats de [Tableau 4.29](#) montrent des variations notables pour MS, ET, AVC et CC. Ces variations viennent souligner l'impact de l'approche.

TABLE 4.28 – MovieLens - Combinaisons de réductions sur RA

	Performance		Qualité	
	MS	ET	AVC	CC
% ↗	36	94	67	19
% →	11	0	19	14
% ↘	53	6	14	67
%Pass	94	100	69	97

TABLE 4.29 – MovieLens - Combinaisons de réductions sur RA - Variations / Baseline

	Performance		Qualité	
	MS	ET	AVC	CC
$\overline{\%Var}$	32	177	53	-7
$ \%Var $	32	180	53	7
$\%VarUP$	32	204	85	No Data

4.8.3 Conclusion

Les critères d'évaluation étudiés montrent des impacts positifs de la réduction combinée dans de nombreux cas d'exécution.

La réduction combinée pourrait donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation.

Les résultats obtenus dépendent des cas d'exécution, ainsi la réduction combinée n'a pas un impact positif dans tous les contextes.

Les résultats concernant les variations par rapport à la référence montrent des variations notables, ce qui vient souligner l'impact de l'approche.

Les résultats sont confirmés dans différents contextes (algorithmes, contraintes et autres), ce qui confirme la généricité de l'approche proposée.

Ces résultats sont prometteurs pour nos futurs travaux sur la réduction combinée.

4.9 Explicabilité de l'E&E

Cette section est dédiée aux résultats d'expérimentations concernant nos contributions sur l'explicabilité de l'E&E c'est-à-dire concernant les indicateurs hyper-volumiques, les transitions entre E&E ainsi que la classification incrémentielle des vecteurs transitionnels.

Les expériences sont menées sur le jeu de données "MovieLens 25M" [167].

Nous menons ces expériences pour évaluer la pertinence des indicateurs introduits précédemment, c'est-à-dire leur capacité à fournir des éléments d'information concernant la manière avec laquelle les algorithmes explorent et exploitent l'espace de recherche, leur capacité à permettre de comparer les processus E&E des algorithmes mais aussi leur capacité à expliciter les transitions qui s'opèrent entre E&E.

Pour étudier l'évolution de ces indicateurs tout au long des exécutions, nous les représentons graphiquement. Ces expériences comprennent des centaines de cas d'exécution, nous présentons donc ici un sous-ensemble des graphes obtenus, qui représentent des comportements typiques et/ou marginaux de l'E&E. Un graphique représente un cas d'exécution c'est-à-dire une combinaison d'un profil, d'un algorithme, d'une variation de contraintes et d'un indicateur. Nous sélectionnons des graphiques notables et volontairement différents parmi ceux obtenus, puis nous présentons les variations des indicateurs au cours des itérations.

Par souci de représentation graphique aux côtés des autres indicateurs (qui sont des pourcentages), les valeurs des indicateurs TV Score, HVCV et HVCO sont normalisées entre 0 et 1.

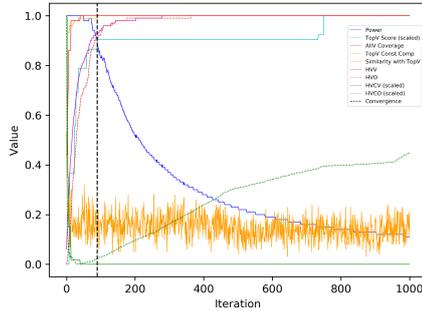
Veillez noter que ce que nous appelons "convergence" dans les sections suivantes représente le moment de convergence de l'algorithme vers les meilleures solutions. L'itération où la convergence est atteinte est mise en évidence dans certains graphiques.

4.9.1 Hyper-volumes

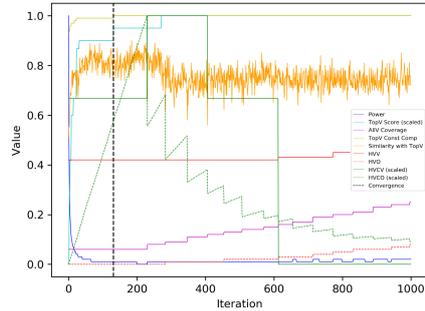
Nous nous intéressons ici aux résultats d'expérimentations de nos indicateurs hyper-volumiques (Section 3.3.1).

Nous avons exécuté un algorithme génétique classique ainsi que les algorithmes évolutionnaires proposés en contributions (Spycle, Struggle, Ramify, Spasm, Extossom, U3S, EEP).

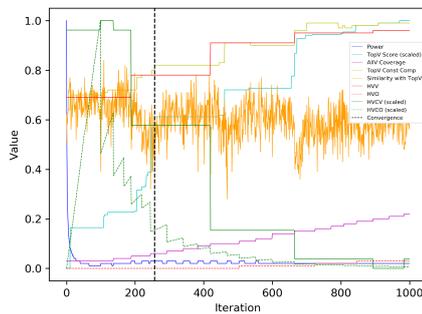
Indicateur HVV



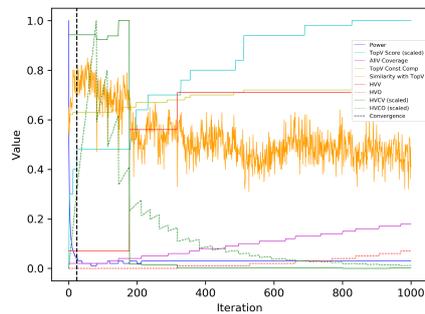
HVV Graphe 1



HVV Graphe 2



HVV Graphe 3



HVV Graphe 4

Le **Graphe 1** montre une croissance rapide de bas à haut niveau avant convergence puis une stagnation à haut niveau. Cela signifie que l'algorithme explore des valeurs dimensionnelles qui deviennent rapidement maximales.

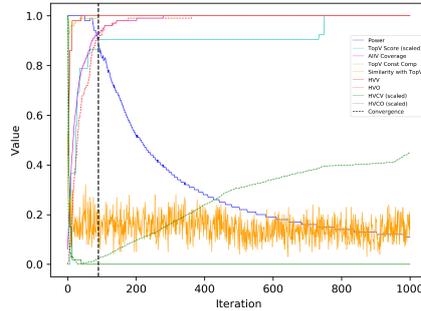
Le **Graphe 2** montre une stagnation à moyen niveau. L'algorithme reste à 50% du potentiel d'exploration des valeurs dimensionnelles.

Le **Graphe 3** montre une croissance par paliers depuis haut niveau jusqu'à la fin. L'algorithme explore des valeurs dimensionnelles élevées dans un premier temps et de plus en plus élevées jusqu'à atteindre des valeurs maximales ou proches.

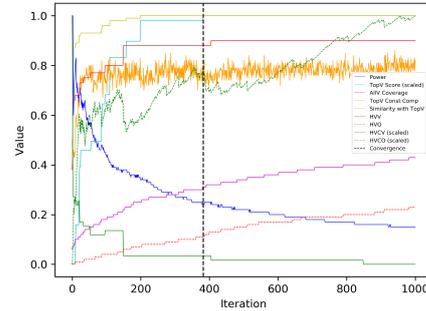
Le **Graphe 4** montre une croissance par paliers depuis bas niveau sans atteindre les valeurs maximales. L'algorithme explore des valeurs dimensionnelles basses dans un premier temps et de plus en plus élevées jusqu'à atteindre des valeurs élevées mais jamais maximales.

En résumé, ces graphiques mettent en évidence l'existence de différents comportements d'exploration des valeurs dimensionnelles et confirment que *HVV* aide à mettre en évidence leurs différences.

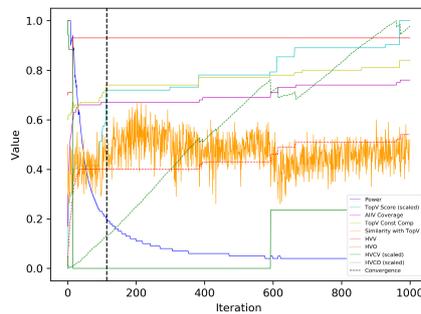
Indicateur HVO



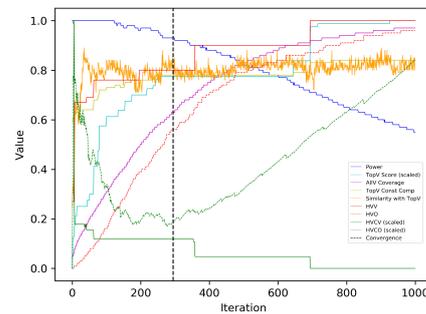
HVO Graphe 5



HVO Graphe 6



HVO Graphe 7



HVO Graphe 8

Le [Graphe 5](#) montre une croissance rapide de bas à haut niveau avant convergence puis une stagnation à haut niveau. L'algorithme explore rapidement le maximum des occurrences dimensionnelles possibles.

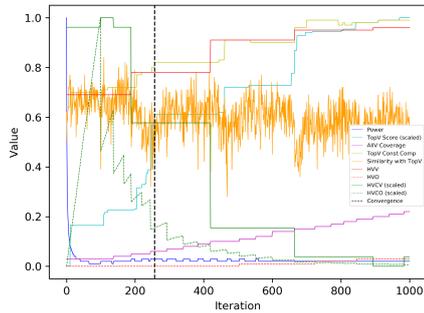
Le [Graphe 6](#) montre une croissance lente depuis bas niveau et restant à bas niveau. L'algorithme explore peu d'occurrences dimensionnelles.

Le [Graphe 7](#) montre une croissance rapide de bas à moyen niveau avant convergence puis une croissance lente par paliers très espacés et en restant à moyen niveau. L'algorithme explore rapidement 50% des occurrences dimensionnelles puis en explore un peu plus par paliers espacés.

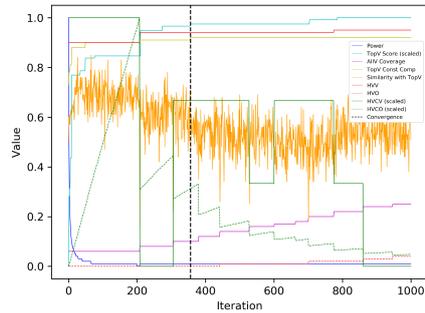
Le [Graphe 8](#) montre une croissance de bas à haut niveau avec moyen niveau atteint en convergence. L'algorithme explore de plus en plus d'occurrences dimensionnelles en atteignant 50% des occurrences en convergence et le maximum des occurrences sur la fin.

En résumé, ces graphiques mettent en évidence l'existence de différents comportements d'exploration des occurrences dimensionnelles et confirment que *HVO* aide à mettre en évidence leurs différences.

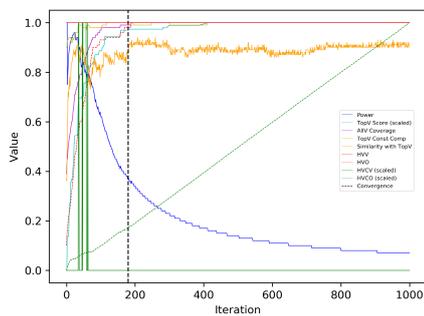
Indicateur HVCV



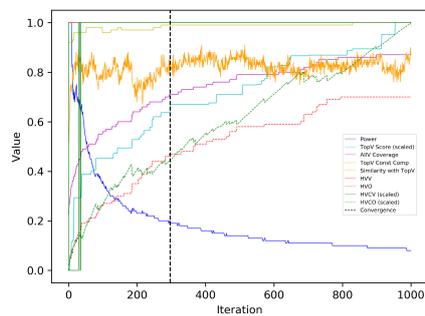
HVCV Graphe 9



HVCV Graphe 10



HVCV Graphe 11



HVCV Graphe 12

Le [Graphe 9](#) montre une décroissance de haut à bas niveau par paliers de stagnation espacés avec niveau moyen atteint en convergence. Lors des phases de stagnation, HVV stagne de même ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales découvertes ne changent pas. De plus cette situation de stagnation correspond à un seul cas présenté dans les combinaisons de variation ([Tableau 3.4](#)) ce qui signifie que la moyenne des valeurs moyennes des *digits* utilisés ne change pas et que le *digit* utilisé ayant la meilleure valeur moyenne ne change pas. Lors des phases de décroissance, HVV augmente ce qui signifie que HVVd augmente et donc que un ou plusieurs *digits* sont découverts, ayant une ou plusieurs valeurs dimensionnelles supérieures aux autres. De plus, cette situation correspond à trois cas présentés dans les combinaisons de variation ([Tableau 3.4](#)) mais seulement un d'entre eux correspond à la croissance simultanée observée de TV Score, ce qui signifie qu'un ou plusieurs *digits* sont découverts, augmentant la moyenne des valeurs moyennes des *digits* utilisés et/ou qu'un *digit* est découvert, ayant une valeur moyenne supérieure aux autres.

Le [Graphe 10](#) montre une succession de croissances et décroissances avec paliers de stagnation. Lors des phases de stagnation, HVV stagne de même ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales découvertes ne changent pas. De plus cette situation de stagnation correspond à un seul cas présenté dans les combinaisons de variation ([Tableau 3.4](#)) ce qui signifie que la moyenne des valeurs moyennes des *digits* utilisés ne change pas et que le *digit* utilisé ayant la meilleure valeur moyenne ne change pas. Lors des phases de décroissance, HVV stagne ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales ne changent pas. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation ([Tableau 3.4](#)) ce qui signifie que un ou plusieurs *digits* sont découverts, diminuant la moyenne des valeurs moyennes des *digits* utilisés. Lors des phases de croissance, HVV stagne ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales découvertes ne changent pas. De plus, cette situation correspond à un seul cas présenté dans les combinaisons

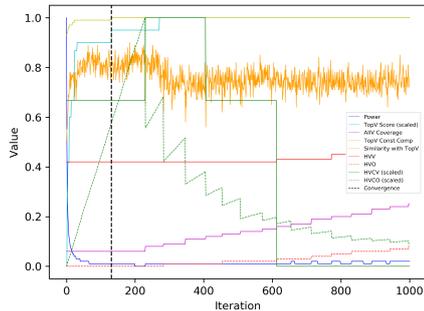
de variation (Tableau 3.4) ce qui signifie que un ou plusieurs *digits* sont découverts, augmentant la moyenne des valeurs moyennes des *digits* utilisés et/ou que un *digit* est découvert, ayant une valeur moyenne supérieure aux autres.

Le Graphe 11 montre une stagnation à bas niveau avec pics de croissance au début. En phase de stagnation, HVV stagne ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales découvertes ne changent pas. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.4) ce qui signifie que la moyenne des valeurs moyennes des *digits* utilisés ne change pas et/ou que le *digit* utilisé ayant la meilleure valeur moyenne ne change pas. Sur les pics de croissance, HVV stagne ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales découvertes ne changent pas. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.4) ce qui signifie que un ou plusieurs *digits* sont découverts, augmentant la moyenne des valeurs moyennes des *digits* utilisés et/ou que un *digit* est découvert, ayant une valeur moyenne supérieure aux autres.

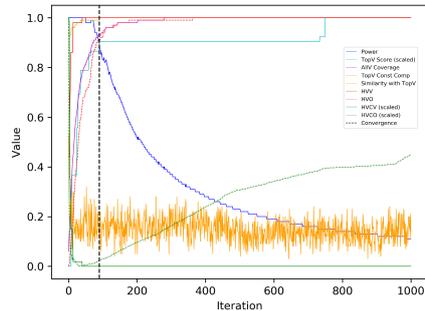
Le Graphe 12 montre une stagnation à haut niveau avec pics de décroissance au début. En phase de stagnation, HVV stagne ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales découvertes ne changent pas. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.4) ce qui signifie que la moyenne des valeurs moyennes des *digits* utilisés ne change pas et/ou que le *digit* utilisé ayant la meilleure valeur moyenne ne change pas. Sur les pics de décroissance, HVV stagne ce qui signifie que HVVd stagne et donc que les valeurs dimensionnelles maximales découvertes ne changent pas. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.4) ce qui signifie que un ou plusieurs *digits* sont découverts, diminuant la moyenne des valeurs moyennes des *digits* utilisés.

En résumé, ces graphiques mettent en évidence l'existence de différents comportements d'évolution de la concentration hyper-volumique en valeur et confirment que *HVCV* aide à mettre en évidence leurs différences.

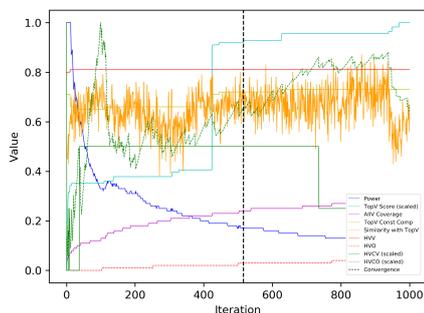
Indicateur HVCO



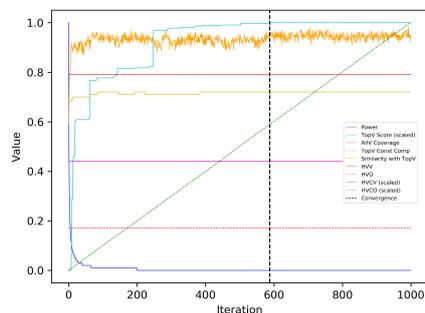
HVCO Graphe 13



HVCO Graphe 14



HVCO Graphe 15



HVCO Graphe 16

Le [Graphe 13](#) montre une croissance rapide jusqu'à haut niveau et jusqu'à bien après la convergence puis des pics de diminutions brutales pour servir une décroissance globale jusqu'à l'amorçage d'une stagnation à bas niveau. Lors des phases de croissance, HVO stagne ce qui signifie que HVOd stagne et donc que aucune occurrence n'est découverte, quelle que soit la dimension. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation ([Tableau 3.3](#)) ce qui signifie que les *digits* utilisés sont en moyenne plus utilisés et/ou que le *digit* le plus utilisé est encore plus utilisé ou un autre *digit* devient plus utilisé que lui. Lors des phases de stagnation, HVO stagne ce qui signifie que HVOd stagne et donc que aucune occurrence n'est découverte, quelle que soit la dimension. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation ([Tableau 3.3](#)) ce qui signifie que le nombre d'utilisations moyen des *digits* utilisés ne change pas et/ou que le nombre d'utilisations du *digit* le plus utilisé ne change pas. Lors des phases de décroissance, HVO stagne ce qui signifie que HVOd stagne et donc que aucune occurrence n'est découverte, quelle que soit la dimension. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation ([Tableau 3.3](#)) ce qui signifie que les *digits* utilisés sont en moyenne moins utilisés ce qui correspond à l'utilisation de nouveaux *digits* venant baisser cette moyenne.

Le [Graphe 14](#) montre une décroissance brutale de haut à bas niveau au début puis une croissance lente. En décroissance, HVO croît tout aussi brutalement ce qui signifie que HVOd croît et donc que une ou plusieurs occurrences sont découvertes dans une ou plusieurs dimensions. De plus, cette situation correspond à trois cas présentés dans les combinaisons de variation ([Tableau 3.3](#)) mais un seul correspond à l'augmentation simultanée observée de TV Score ce qui signifie que les *digits* utilisés sont en moyenne plus utilisés et/ou que le *digit* le plus utilisé est encore plus utilisé ou un autre *digit* devient plus utilisé que lui. En croissance, HVO continue de croître puis stagne ce qui signifie que HVOd (1) croît puis (2) stagne. Dans le cas (1) cela signifie que une ou plusieurs occurrences sont découvertes dans une ou plusieurs dimensions. Cette

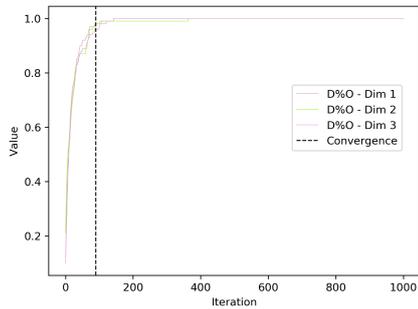
situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.3) ce qui signifie que les *digits* utilisés sont en moyenne plus utilisés et/ou que le *digit* le plus utilisé est encore plus utilisé ou un autre *digit* devient plus utilisé que lui. Dans le cas (2) cela signifie que aucune occurrence n'est découverte quelle que soit la dimension. Cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.3) ce qui signifie que les *digits* utilisés sont en moyenne plus utilisés et/ou que le *digit* le plus utilisé est encore plus utilisé ou un autre *digit* devient plus utilisé que lui.

Le Graphe 15 montre une croissance rapide de bas à haut niveau au début puis une décroissance jusqu'à moyen niveau puis une croissance globale par succession de petites croissances et décroissances, jusqu'à haut niveau. En croissance, HVO stagne donc HVOd stagne ce qui signifie que aucune occurrence n'est découverte, quelle que soit la dimension. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.3) ce qui signifie que les *digits* utilisés sont en moyenne plus utilisés et/ou que le *digit* le plus utilisé est encore plus utilisé ou un autre *digit* devient plus utilisé que lui. En décroissance, HVO stagne donc HVOd stagne ce qui signifie que aucune occurrence n'est découverte, quelle que soit la dimension. De plus, cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.3) ce qui signifie que les *digits* utilisés sont en moyenne moins utilisés ce qui correspond à l'utilisation de nouveaux *digits* venant baisser cette moyenne.

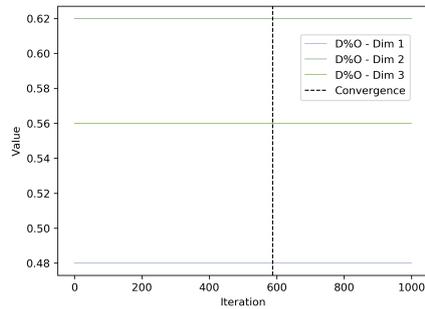
Le Graphe 16 montre une croissance continue de bas à haut niveau pour atteindre un niveau moyen en convergence. HVO stagne donc HVOd stagne ce qui signifie que aucune occurrence n'est découverte, quelle que soit la dimension. Cette situation correspond à un seul cas présenté dans les combinaisons de variation (Tableau 3.3) ce qui signifie que les *digits* utilisés sont en moyenne plus utilisés et/ou que le *digit* le plus utilisé est encore plus utilisé ou un autre *digit* devient plus utilisé que lui.

En résumé, ces graphiques mettent en évidence l'existence de différents comportements d'évolution de la concentration hyper-volumique en occurrences et confirment que *HVCO* aide à mettre en évidence leurs différences.

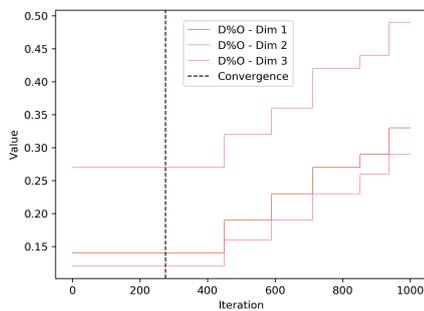
$D\%O_k$



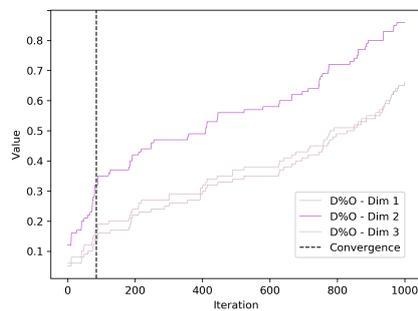
$D\%O_k$ Graphe 17



$D\%O_k$ Graphe 18



$D\%O_k$ Graphe 19



$D\%O_k$ Graphe 20

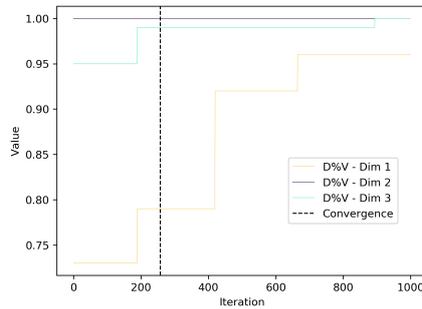
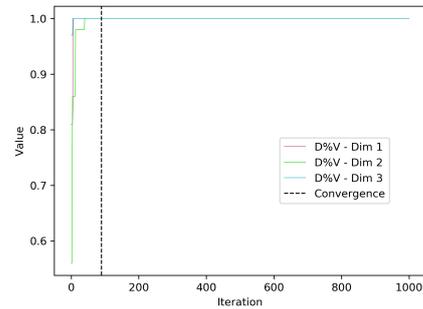
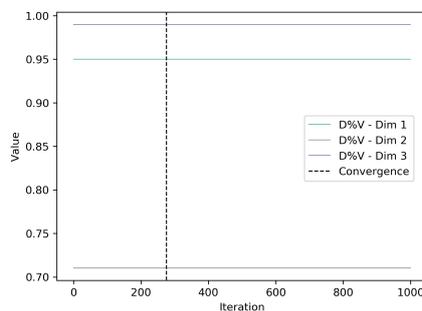
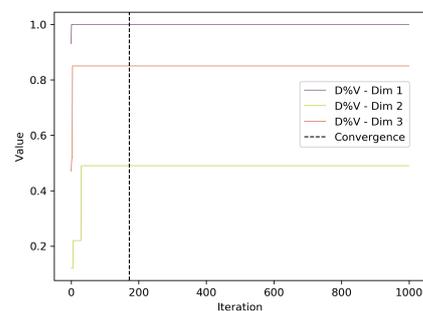
Le [Graphe 17](#) montre une croissance rapide de bas à haut niveau pour toutes les dimensions jusqu'à la convergence, puis l'amorçage d'une stagnation jusqu'à la fin. L'algorithme connaît donc peu d'occurrences dimensionnelles au départ et les découvre toutes très rapidement.

Le [Graphe 18](#) montre une stagnation pour l'ensemble des dimensions, chacune d'elles sur un niveau différent. L'algorithme connaît donc au départ des nombres d'occurrences dimensionnelles différents d'une dimension à l'autre et semble être dans l'incapacité de découvrir d'autres occurrences dimensionnelles, toutes dimensions confondues.

Le [Graphe 19](#) montre une stagnation pour l'ensemble des dimensions jusqu'après la convergence puis l'amorçage d'augmentations par paliers pour l'ensemble des dimensions. L'algorithme connaît donc au départ des nombres d'occurrences dimensionnelles différents d'une dimension à l'autre et semble être dans l'incapacité de découvrir d'autres occurrences dimensionnelles, toutes dimensions confondues. Cependant, après la convergence, l'algorithme est en capacité de découvrir lentement de nouvelles occurrences dimensionnelles dans toutes les dimensions.

Le [Graphe 20](#) montre des augmentations progressives de bas à haut niveau pour l'ensemble des dimensions et pour atteindre des niveaux bas en convergence. L'algorithme connaît donc peu d'occurrences dimensionnelles au départ et en découvre lentement de nouvelles dans toutes les dimensions et jusqu'à potentiellement toutes les découvrir.

En résumé, ces graphiques mettent en évidence l'existence de différents comportements d'évolution des amplitudes des explorations mono-dimensionnelles de l'espace de recherche en termes d'occurrences et confirment que $D\%O_k$ aide à mettre en évidence leurs différences.

$D\%V_k$  $D\%V_k$ Graphe 21 $D\%V_k$ Graphe 22 $D\%V_k$ Graphe 23 $D\%V_k$ Graphe 24

Le [Graphe 21](#) montre des évolutions différentes, une dimension voyant ses valeurs croître par paliers de stagnation de haut à haut niveau, une autre commençant à haut niveau et atteignant des valeurs quasi-maximales en convergence et la dernière stagnant au niveau maximum dès le début. L'algorithme connaît donc dès le début des valeurs dimensionnelles fortes et continue dans découvrir de plus fortes jusqu'à la convergence pour certaines dimensions et jusqu'à la fin pour d'autres.

Le [Graphe 22](#) montre une croissance rapide jusqu'à haut niveau pour toutes les dimensions jusqu'à la convergence, puis l'amorçage d'une stagnation jusqu'à la fin. L'algorithme connaît donc des valeurs dimensionnelles faibles au départ et découvre les valeurs dimensionnelles les plus fortes très rapidement.

Le [Graphe 23](#) montre une stagnation à haut niveau pour l'ensemble des dimensions mais à des niveaux différents. Deux dimensions stagnent à très haut niveau contrairement à la dernière. L'algorithme connaît donc dès le début des valeurs dimensionnelles fortes mais est dans l'incapacité de découvrir les valeurs dimensionnelles les plus fortes.

Le [Graphe 24](#) montre une croissance initiale rapide pour l'ensemble des dimensions, chacune d'elles démarrant à un niveau différent, suivie d'une stagnation jusqu'à la convergence et fin. Une dimension croît de bas à moyen niveau avant convergence avant de stagner ce qui signifie que l'algorithme connaît des valeurs dimensionnelles faibles au départ et en découvre des moyennes très rapidement sans jamais réussir à en trouver des hautes. Une autre dimension croît de moyen à haut niveau avant convergence avant de stagner ce qui signifie que l'algorithme connaît des valeurs dimensionnelles moyennes au départ et en découvre des hautes très rapidement sans jamais réussir à trouver les maximales. La dernière dimension croît de très haut niveau jusqu'au niveau maximal avant convergence avant de stagner ce qui signifie que l'algorithme connaît des valeurs dimensionnelles très fortes au départ et découvre les maximales très rapidement.

En résumé, ces graphiques mettent en évidence l'existence de différents comportements d'évo-

lution des amplitudes des explorations mono-dimensionnelles de l'espace de recherche en termes de valeurs et confirmer que $D\%V_k$ aide à mettre en évidence leurs différences.

Conclusion

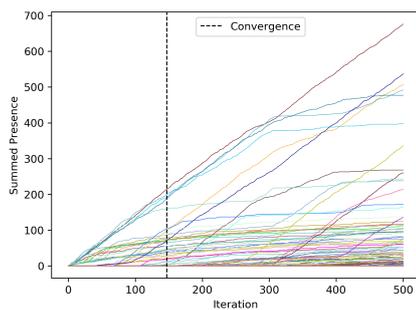
Pour conclure, les indicateurs proposés donnent une représentation de l'E&E sous de nouveaux angles car ils fournissent des éléments d'information concernant la manière avec laquelle les algorithmes explorent et exploitent l'espace de recherche tout au long d'une course itérative. Ils sont aussi des marqueurs de différences entre les algorithmes et entre les cas d'exécution. Ils permettent donc de comparer les comportements d'E&E, de souligner les comportements marginaux et de mettre à jour des comportements redondants.

4.9.2 Transitions entre E&E

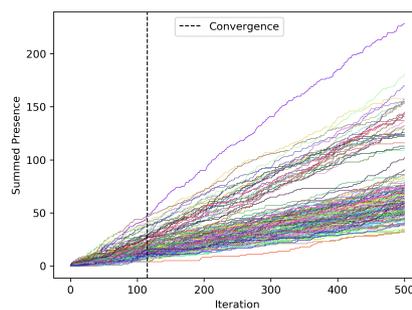
Nous nous intéressons ici aux résultats d'expérimentations concernant nos contributions sur les transitions entre E&E (Section 3.3.2).

Nous avons exécuté un algorithme génétique classique ainsi que les algorithmes évolutionnaires présentés en contribution (Spycle, Struggle, Ramify, Spasm, Extossom, U3S, EEP).

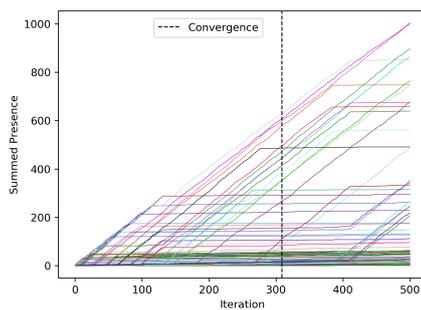
Évolution des présences sommées au long des itérations



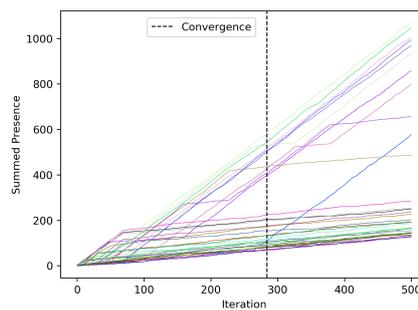
Graphe 1



Graphe 2



Graphe 3



Graphe 4

Nous étudions ici l'évolution de *vps*, l'indicateur des présences sommées, tout au long des itérations (Graphes 1 à 4). Pour rappel, *vps* représente, pour une itération, la somme du nombre de fois où chaque *digit* est présent dans les populations en considérant toutes les itérations précédentes. Chaque courbe correspond donc à un *digit*.

Les questions analytiques liées à ces graphiques sont les suivantes : Certains *digits* se distinguent-ils de la masse ? Y a-t-il des événements notables concernant la convergence ? Certains *digits*

commencent-ils à être utilisés au cours du processus de recherche? Comment les stagnations sont-elles organisées? Certains *digits* sont-ils utilisés en permanence?

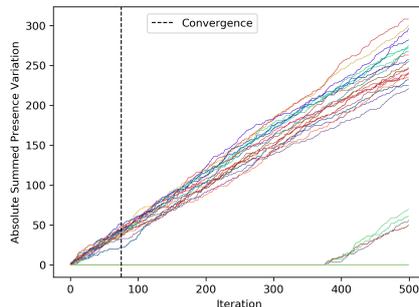
Nous constatons tout d'abord que tous les *digits* ou certains d'entre eux sont utilisés de manière continue tout au long du processus E&E (Graphe 2, Graphe 3).

En outre, certains *digits* commencent à être utilisés au cours et à différents moments en pré-convergence ou en post-convergence (Graphe 3) et certains ne sont même jamais utilisés (Graphe 1, Graphe 3). Dans (Graphe 1), la convergence coïncide avec l'arrêt ou le début de l'utilisation continue de certains *digits*, ce qui est respectivement représenté par la stagnation et la première utilisation de certains *digits*. La convergence est aussi souvent proche du point de rupture entre le bloc de *digits* en usage continu intensif et le reste des *digits* (Graphe 2), ce qui est représenté par la scission sur le graphique.

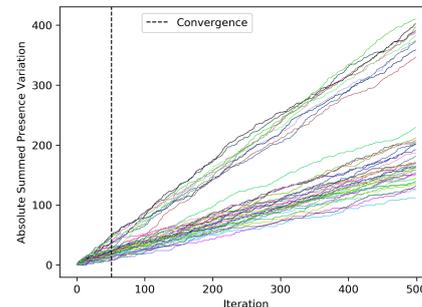
Une stagnation représente le fait qu'un *digit* n'est plus utilisé. La stagnation peut être observée à différents moments, être définitive (Graphe 3) ou peut constituer des niveaux (Graphe 3). Cela signifie que certains *digits* n'ont pas un rôle clairement déterminé tout au long du processus de recherche ou, en d'autres termes, qu'ils constituent des points d'indécision dans le processus de recherche.

En résumé, ces graphiques mettent en évidence l'existence de différents comportements d'E&E à travers l'utilisation de *digits* au cours du processus, et confirment que *vps* aide à mettre en évidence leurs différences.

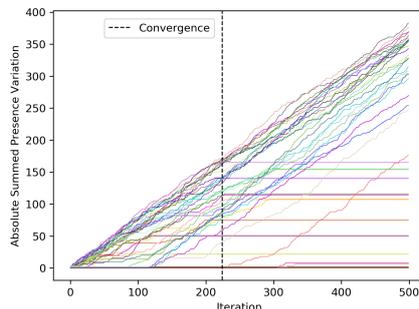
Évolution des variations de présences sommées au long des itérations



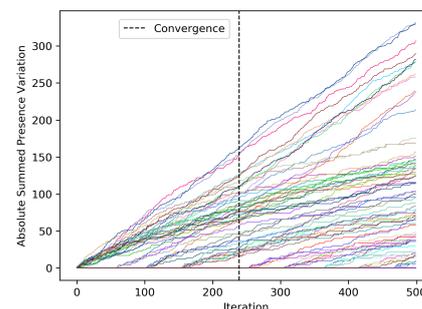
Graphe 5



Graphe 6



Graphe 7



Graphe 8

Nous étudions ici l'évolution de *vvps*, l'indicateur des variations de présences sommées, au long des itérations (Graphes 5 à 8). Pour rappel, *vvps* représente, pour une itération, la somme des *varp* en considérant toutes les itérations précédentes.

Les questions analytiques liées à ces graphiques sont les suivantes : Quand et comment les

digits varient-ils en présence ? Certains *digits* se distinguent-ils de la masse ? Y a-t-il des événements notables concernant la convergence ? Comment s'organisent les stagnations ?

Tous les *digits* ne varient pas en présence avant la convergence (Graphe 5). Néanmoins, dans tous les cas d'exécution, certains *digits* subissent des variations continues de présence et se distinguent des autres.

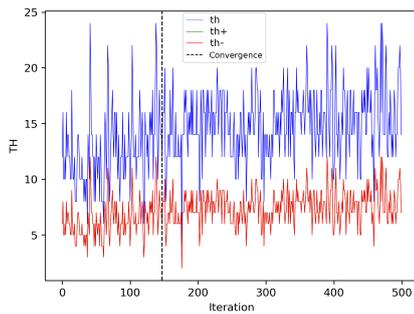
La convergence peut coïncider avec l'arrêt ou le début de la variation de la présence de certains *digits* (Graphe 7). La convergence est aussi souvent proche du point de rupture entre le bloc de *digits* en forte variation de présences et le reste des *digits* (Graphe 6).

Selon le cas d'exécution, soit tous les *digits* varient en présence dès le début du processus de recherche (Graphe 6), soit certains *digits* commencent à varier en plein cours et à différents moments en pré-convergence ou post-convergence (Graphe 8).

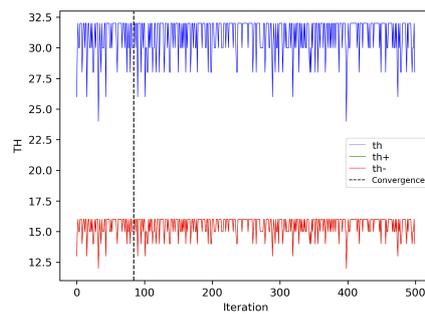
Les stagnations représentent le fait que les *digits* concernés ne varient plus en présence. Les stagnations peuvent être observées à différents moments, être définitives ou constituer des niveaux (Graphe 7).

Là encore, nous pouvons constater l'existence de différents comportements d'E&E, concernant les variations d'utilisation des *digits*. Ces graphiques confirment que *vvps* aide à comprendre les différences entre ces comportements d'E&E.

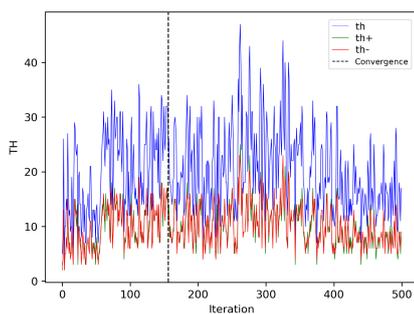
Évolution des chaleurs transitionnelles sommées au long des itérations



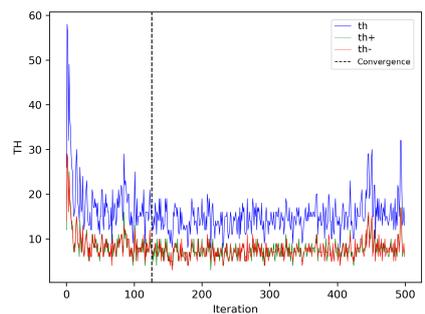
Graphe 9



Graphe 10



Graphe 11



Graphe 12

Nous étudions ici l'évolution des valeurs de th , $th+$ et $th-$, au long des itérations (Graphes 9 à 12). Pour rappel, $th+$ et $th-$ sont respectivement la somme des valeurs positives et la somme des valeurs absolues négatives d'un *varp*. Pour une itération i , th représente la somme des valeurs absolues de $varp_i$.

Les questions analytiques liées concernant ces graphes sont les suivantes : Comment les valeurs de th , $th+$ et $th-$ évoluent-elles ? Y a-t-il des événements notables concernant la convergence ?

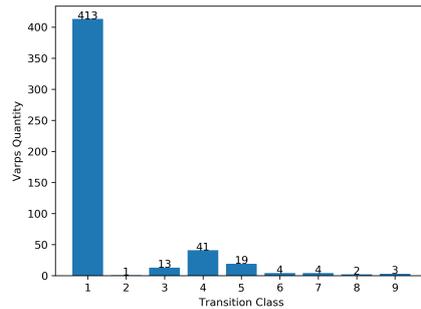
Les trois courbes oscillent avec une force qui dépend du cas d'exécution ([Graphes 9 et 10](#)). Les courbes $th+$ et $th-$ sont confondues dans tous les graphiques, ce qui représente le fait que lorsque certains *digits* ne sont plus utilisés, cela permet d'en utiliser de nouveaux, ce qui conduit à des augmentations et des diminutions synchronisées de $th+$ et $th-$. Rappelons ici que les valeurs de $th-$ sont des valeurs absolues.

Dans certains cas d'exécution, des valeurs élevées de $th-$ sont observées au début du processus de recherche ([Graphe 12](#)) et diminuent rapidement pour entrer dans des couloirs d'oscillation. Par ailleurs, il n'y a pas d'événements notables en pré-convergence ou en post-convergence.

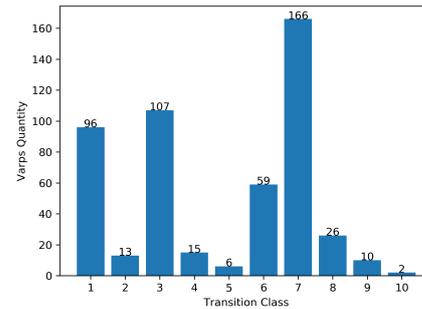
De plus, nous pouvons observer des pics th significatifs à différents moments du processus de recherche ([Graphe 9](#)), représentant des augmentations de puissance transitoire globale (pics orientés vers le haut) ou des diminutions (pics orientés vers le bas). Nous pouvons en déduire qu'à ces moments de pic, le comportement d'E&E est singulier car l'algorithme opère des changements notables dans la façon dont il attribue des valeurs aux *digits*. Nous devons également noter qu'il est possible de comparer la régularité et la force des pics entre différents cas d'exécution.

En résumé, ces graphiques confirment que les valeurs de th , $th+$ et $th-$ permettent de mieux comprendre les différences comportementales de l'E&E entre différents cas d'exécution.

Quantités de vecteurs de variations de présences dans chaque classe



Graphe 13



Graphe 14

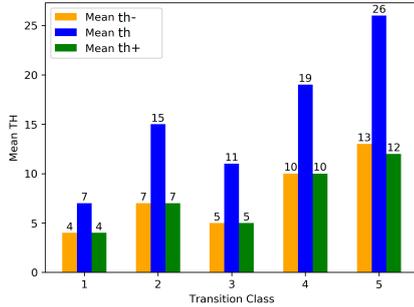
Nous nous concentrons maintenant sur les classes de transitions ([Graphes 13 et 14](#)) en représentant la quantité de *varp* dans chaque classe. Pour rappel, la classification des vecteurs de transition regroupe les vecteurs *varp* afin d'identifier les classes de transition.

La principale question analytique concernant ces graphes est : que pouvons-nous dire de la distribution des *varp* dans les classes ?

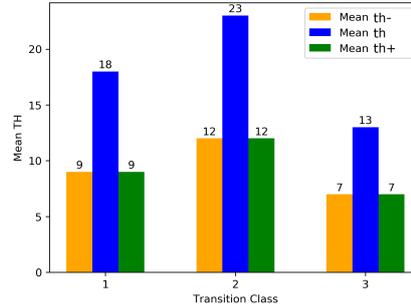
Dans tous les cas d'exécution, certaines classes écrasent les autres ou présentent des écarts importants par rapport aux autres et certaines classes sont très peu peuplées ([Graphe 13](#)). Des classes intermédiaires peuvent cependant exister ([Graphe 14](#)). Ce phénomène semble cohérent car il signifie que la plupart du temps le processus d'E&E est ponctué de vecteurs transitoires à faible impact, il est donc normal de trouver ces vecteurs dans quelques classes écrasant les autres ([Graphe 14](#)) ou même dans une seule classe ([Graphe 13](#)).

En résumé, ces graphiques valident l'existence de diverses classes de vecteurs transitionnels entre E&E et permettent de souligner leur importance dans le processus de recherche.

Chaleurs transitionnelles moyennes des classes



Graphe 15



Graphe 16

Nous étudions maintenant les chaleurs moyennes de transition ($th+$, $th-$, th) pour chaque classe de transition. Elles sont représentées dans [Graphes 15 et 16](#).

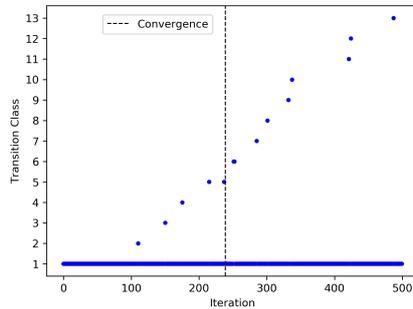
Les questions analytiques concernant ces graphes sont les suivantes : Que pouvons-nous dire de la distribution de th dans les classes ? Qu'en est-il de $th+$ par rapport à $th-$?

Dans tous les cas d'exécution, il n'y a pas d'écrasement de classes par d'autres. Cependant, il y a souvent de fortes différences entre les th moyennes des classes ([Graphe 15](#)).

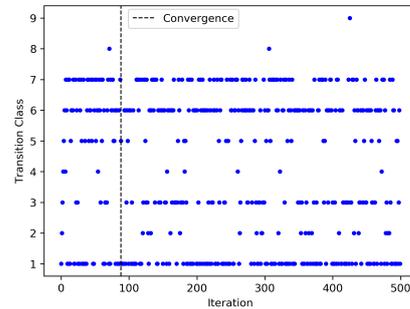
Notons que les valeurs de $th+$ et $th-$ sont toujours équilibrées presque parfaitement pour une classe ([Graphe 16](#)). Ceci s'explique une fois de plus par le fait qu'un *digit* ne peut être utilisé qu'un nombre maximal de fois correspondant à la taille de la population et qu'ensuite seule une moindre utilisation fera varier son th . Une oscillation a donc naturellement lieu, d'où l'équilibre visible sur ces graphiques.

En résumé, ces graphiques mettent en évidence les pouvoirs transitoires des classes et la façon dont ils sont répartis.

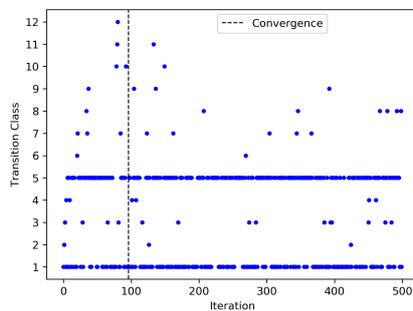
Répartition temporelle des classes des vecteurs de variations de présences au long des itérations



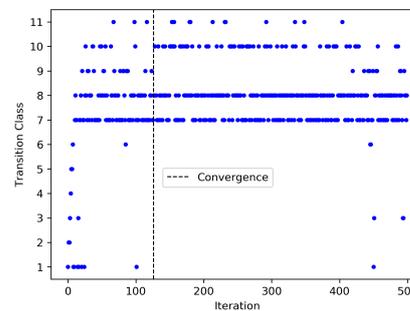
Graphe 17



Graphe 18



Graphe 19



Graphe 20

Nous étudions ici les moments (itérations) où chaque classe de transition a lieu (Graphes 17 à 20). Concrètement, ces graphes montrent, pour chaque itération, la classe de transition de son *varp*. Cela permet de visualiser l'évolution des interventions des classes de transition au fil des itérations.

Les questions concernant ces graphiques sont les suivantes : Y a-t-il des événements notables concernant la convergence ? Comment s'organisent les stagnations ? Quel est le schéma général d'apparition des classes ? Y a-t-il des classes qui commencent à être utilisées au cours du processus de recherche ?

Dans tous les cas d'exécution, nous constatons la présence d'une ou plusieurs classes très peuplées et apparaissant du début à la fin du processus de recherche, ce sont les lignes horizontales dans les graphiques. Le processus de recherche passe donc régulièrement par des vecteurs transitoires qui ont potentiellement peu d'impact mais qui constituent des repères. D'autres classes interviennent occasionnellement avec plus ou moins de régularité (Graphe 18).

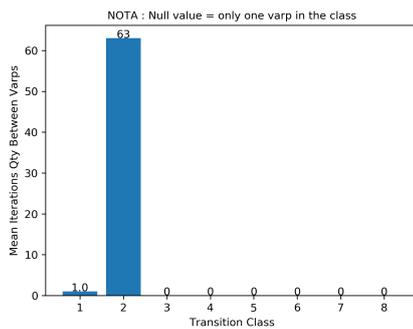
Toutes les classes ne sont pas nécessairement utilisées au moins une fois avant la convergence (Graphe 17), ceci montre que la convergence n'est pas pertinente dans certains cas car de meilleures solutions peuvent être trouvées après celle-ci. En d'autres termes, les classes intervenant en post-convergence et n'étant pas intervenues en pré-convergence peuvent avoir un impact au point de conduire à la découverte d'excellentes solutions.

Dans de nombreux cas d'exécution, on observe en pré-convergence une intense activité de passage inter-classes (Graphe 19) ainsi qu'une richesse de classes différentes (Graphe 20). L'enchaînement de ces grands ensembles transitoires semble donc conduire à la convergence. La convergence est aussi un moment particulier qui peut être le point de départ de l'apparition de certaines classes ou le point de rupture marquant la disparition de certaines classes (Graphe 17).

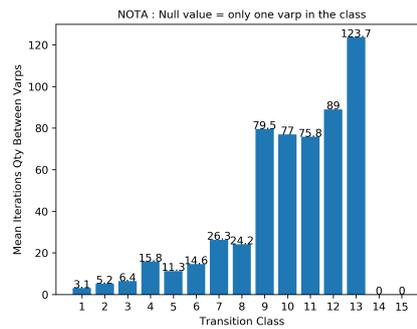
La stagnation peut prendre deux formes : mono-classe et multi-classe. La stagnation mono-classe correspond à une situation où une seule classe apparaît de manière continue (Graphe 17), tandis que la stagnation multi-classes correspond à une situation où un couloir est formé par deux ou plusieurs classes (Graphe 19). La stagnation peut représenter une situation qui n’apporte plus grand-chose en termes d’E&E ou même une situation de blocage dans un schéma d’E&E qui peut éventuellement être débloquée ultérieurement. Ces situations pourraient être croisées avec des informations concernant le blocage dans une zone d’optimum local de l’espace de recherche, par exemple. Notons qu’une stagnation peut être vertueuse si elle est ponctuée par l’intervention d’autres classes (Graphe 20).

Nous pouvons conclure que ces graphiques mettent en évidence les comportements transitionnels pré/post-convergence, les classes transitionnelles de référence, la capacité à générer des classes transitionnelles de référence, la capacité à générer des corridors transitionnels, la manière de se comporter par rapport à ces références et corridors ainsi que les classes avec des interventions spécifiques qui peuvent avoir l’impact le plus significatif. Ces éléments soulignent de multiples différences entre les processus d’E&E.

Nombres d’itérations moyens séparant les vecteurs de variations de présences des classes



Graphe 21



Graphe 22

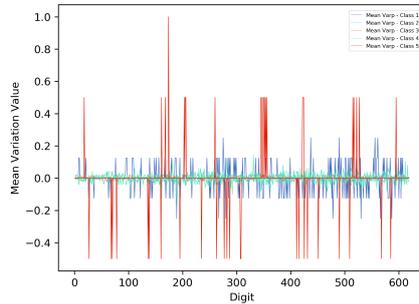
Nous étudions ici, pour chaque classe de transition, la quantité moyenne d’itérations séparant deux de ses interventions (Graphes 21 et 22).

Les questions concernant ces graphiques sont les suivantes : Existe-t-il des classes dont les distributions temporelles sont beaucoup plus espacées que les autres ? Y a-t-il des classes dont les distributions temporelles sont très rapprochées ? Qu’en est-il de la distribution globale du nombre d’itérations séparant les vecteurs *varp* des classes ?

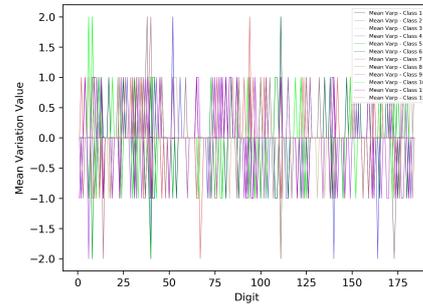
Certaines classes apparaissent avec beaucoup plus de régularité que d’autres (Graphe 22) et des cas d’écrasement apparaissent (Graphe 21). Ces graphes font directement écho aux graphes de la distribution temporelle en donnant une représentation quantitative des régularités des interventions des classes.

En résumé, ces graphes mettent en évidence la capacité à générer des classes transitionnelles de référence et leur régularité au sein du processus de recherche.

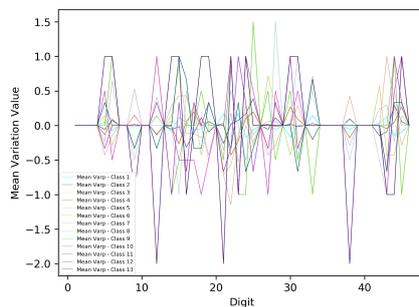
Vecteurs de variations de présences moyens des classes



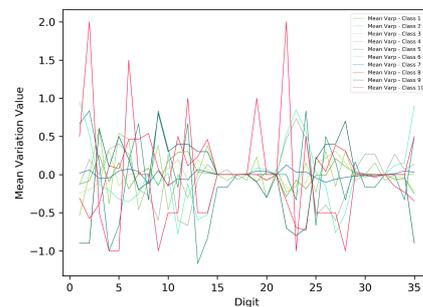
Graphe 23



Graphe 24



Graphe 25



Graphe 26

Nous nous concentrons maintenant sur les *varpmean*, en affichant, pour chaque classe de transition, la valeur de variation moyenne de chaque *digit* (Graphes 23 à 26). Pour rappel, le *varp* moyen d'une classe est $varpmean_j$ dont les valeurs représentent les moyennes par dimension (*digits*) des valeurs des vecteurs *varp* de cette classe.

Les questions concernant ces graphiques sont les suivantes : certains *varpmeans* se distinguent-ils de la masse ? Est-il possible de cibler les *digits* ayant des variations moyennes notables ? Existe-t-il des *digits* ayant des spectres de variation sur plusieurs *varpmean* ?

Les *varpmean* peuvent se démarquer des autres en présentant des *digits* ayant de fortes variations de présence moyenne (Graphe 23, Graphe 24). Il est donc possible de cibler les *digits* présentant des variations moyennes importantes dans certaines classes (Graphe 26). Il est également possible d'observer des spectres de variation sur plusieurs vecteurs *varpmean*, c'est-à-dire des *digits* ayant des variations moyennes différentes selon la classe (Graphe 25).

Ces graphiques mettent en évidence la capacité à générer des classes présentant des variations notables pour certains *digits*, les quantités de *digits* ayant des variations notables, les variations pour des *digits* spécifiques ainsi que les spectres de variations possibles. Ils confirment également l'existence de divers comportements d'E&E et les *varpmean* permettent de mieux comprendre les différences entre ces comportements.

Conclusion

En conclusion, ces expériences confirment que les indicateurs proposés et les visualisations associées contribuent à représenter l'E&E sous de nouveaux angles, car ils donnent un aperçu de la façon dont les algorithmes explorent et exploitent l'espace de recherche au cours d'une exécution itérative. De plus, les indicateurs proposés permettent de comparer les comportements transitoires d'E&E entre les algorithmes et entre les cas d'exécution. Ainsi, ils permettent de

comparer les comportements d'E&E, de souligner les comportements marginaux, de révéler les comportements redondants et d'expliciter les transitions qui s'opèrent entre E&E. Notons que des corrélations, redondances et même incohérences entre indicateurs peuvent exister. Nous ne nous sommes pas intéressé à l'explicitation de ces phénomènes qui demanderaient des expérimentations supplémentaires. Notons aussi que ces indicateurs n'ont pas pour objectif d'apporter une explicabilité complète des comportements d'E&E car, par définition, un indicateur reste un point de vue et même si plusieurs indicateurs sont considérés en même temps, cela ne garantit pas d'obtenir une représentation plus complète de l'E&E. Un indicateur est comme une fenêtre, un regard sur une situation et il en existe potentiellement énormément d'autres à proposer en contributions.

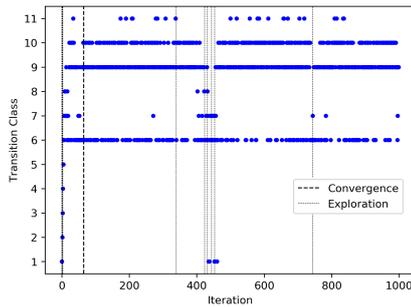
4.9.3 Classification incrémentielle

Nous nous intéressons ici aux résultats d'expérimentations concernant nos contributions sur la classification incrémentielle des *itemsets* (Section 3.3.3).

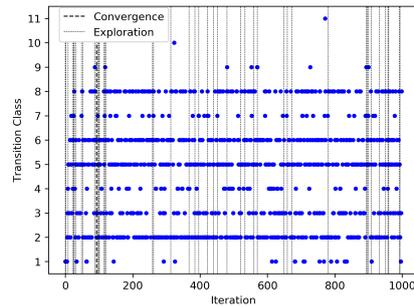
Deux algorithmes traditionnels de référence ont été exécutés, l'algorithme génétique et l'algorithme par renforcement.

Seuls les marqueurs des phases d'exploration sont représentés par des barres pointillées verticales, toutes les autres sont donc des phases d'exploitation.

Séquence des classes de transition



Graphe 1



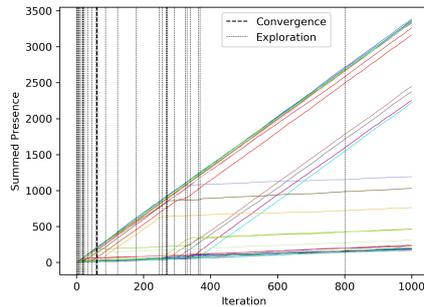
Graphe 2

Le [Graphe 1](#) montre plusieurs phases d'exploration dans les premières itérations qui sont corrélées à l'intervention des classes 1 à 5. Un bloc de phases d'exploration successives apparaît un peu après l'itération 400 et semble corrélé à l'intervention des classes 1, 7 et 8. Une phase d'exploration est présente avant l'itération 800 et semble corrélée à l'intervention de la classe 7.

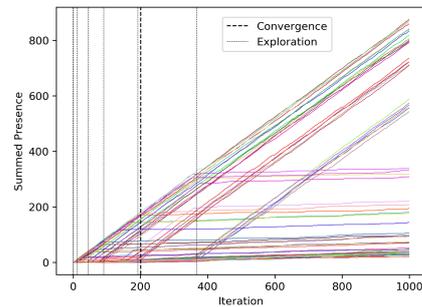
Le [Graphe 2](#) montre plusieurs phases d'exploration dans les premières itérations et jusqu'à la convergence qui sont corrélées à l'intervention des classes 1, 4, 7 et 9. Diverses autres phases d'exploration apparaissent ensuite dont une partie semble particulièrement corrélée à l'intervention des classes 7 et 9.

En résumé, ces graphiques mettent en évidence diverses corrélations entre les positionnements des phases d'exploration (issues de la classification incrémentielle) et l'intervention de classes transitionnelles à des moments spécifiques de la course itérative. La classification incrémentielle permet donc de mieux comprendre le rôle de certaines classes transitionnelles dans le processus de recherche, tout en mettant à jour de nouvelles transitions.

Présences des digits - sommées



Graph 3



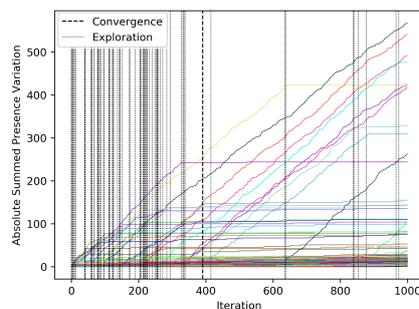
Graph 4

Le [Graph 3](#) montre plusieurs phases d'exploration dans les premières itérations et jusqu'à la convergence qui sont corrélées au début de l'utilisation continue de plusieurs *digits*. Un bloc de phases d'exploration apparaît entre les itérations 200 et 400, elles sont corrélées au début de l'utilisation continue de plusieurs *digits*.

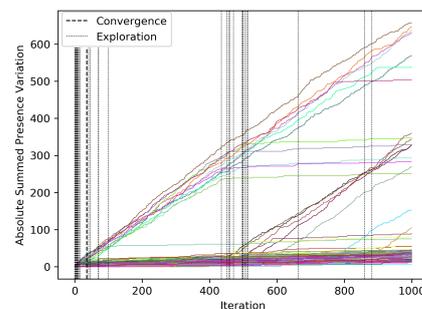
Le [Graph 4](#) montre plusieurs phases d'exploration dans les premières itérations et jusqu'à la convergence qui sont corrélées au début de l'utilisation continue de plusieurs *digits*. Une phase d'exploration apparaît entre les itérations 200 et 400, elle est corrélée au début de l'utilisation continue de plusieurs *digits*.

En résumé, ces graphiques mettent en évidence diverses corrélations entre les positionnements des phases d'exploration (issues de la classification incrémentielle) et le début de l'utilisation continue de plusieurs *digits*. La classification incrémentielle permet donc de mieux comprendre le rôle de certains *digits* dans le processus de recherche, tout en mettant à jour de nouvelles transitions.

Variations des présences des digits - sommées



Graph 5

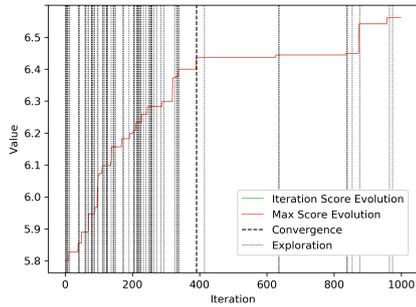


Graph 6

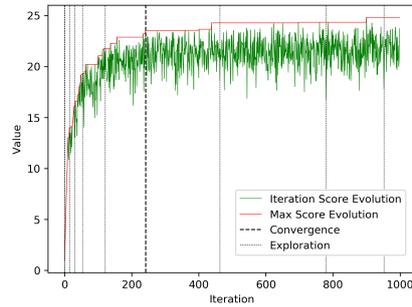
Le [Graph 5](#) et le [Graph 6](#) montrent plusieurs phases d'exploration dans les premières itérations et jusqu'à la convergence qui sont corrélées au début de variations de présences continues de plusieurs *digits*. Plusieurs autres phases d'exploration apparaissent ensuite qui sont corrélées au début de variations de présences continues de plusieurs *digits*.

En résumé, ces graphiques mettent en évidence diverses corrélations entre les positionnements des phases d'exploration (issues de la classification incrémentielle) et le début de variations de présences continues de plusieurs *digits*. La classification incrémentielle permet donc de mieux comprendre le rôle de certains *digits* dans le processus de recherche, tout en mettant à jour de nouvelles transitions.

Scores



Graphe 7



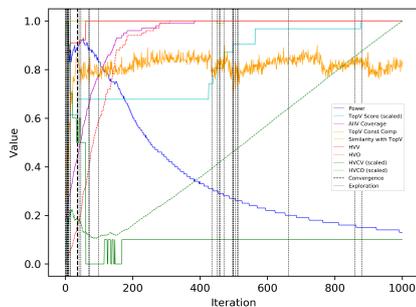
Graphe 8

Le **Graphe 7** montre plusieurs phases d'exploration dans les premières itérations et jusqu'à la convergence qui sont corrélées au franchissement de paliers de score de fitness. Plusieurs autres phases d'exploration apparaissent ensuite qui sont corrélées au franchissement de paliers de score de fitness.

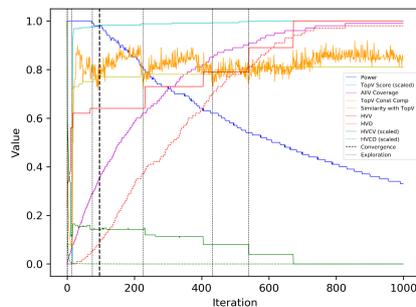
Le **Graphe 8** montre plusieurs phases d'exploration dans les premières itérations et jusqu'à la convergence qui sont corrélées au franchissement de paliers de score de fitness. Plusieurs autres phases d'exploration apparaissent ensuite qui sont corrélées au post-franchissement de paliers de score de fitness.

En résumé, ces graphiques mettent en évidence diverses corrélations entre les positionnements des phases d'exploration (issues de la classification incrémentielle) et le franchissement de paliers de score de fitness. La classification incrémentielle permet donc de mieux comprendre l'importance du franchissement de certains paliers de score de fitness dans le processus de recherche, tout en mettant à jour de nouvelles transitions.

Multi



Graphe 9



Graphe 10

Le **Graphe 9** montre plusieurs phases d'exploration dans les premières itérations et jusqu'à la convergence qui sont corrélées aux fortes variations de l'ensemble des indicateurs. Deux phases d'explorations suivent la convergence qui sont corrélées à l'atteinte d'un palier pour HVCV, TopV Score, Similarity with TopV, Power et HVV. Deux autres phases d'exploration les succèdent rapidement qui sont corrélées à l'atteinte d'un palier pour HVCV, HVCO et HVV, ainsi qu'à l'amorçage d'une variation pour HVCO et Power. Un bloc de phases d'exploration apparait entre les itérations 400 et 600, elles sont corrélées au franchissement de paliers pour TopV Score ainsi qu'à des variations significatives pour Similarity with TopV.

Le **Graphe 10** montre plusieurs phases d'exploration dans les premières itérations qui sont corrélées aux fortes variations de l'ensemble des indicateurs. Une phase d'exploration précède la

convergence et est corrélée à des changements significatifs pour Power, HVV, TopV Const Comp et Similarity with TopV. Une phase d'exploration est présente peu après l'itération 200 qui est corrélée à des variations significatives pour HVCV, HVV et Similarity with TopV. Une phase d'exploration est présente peu après l'itération 400 qui est corrélée à des variations significatives pour Similarity with TopV. Enfin, une phase d'exploration est présente avant l'itération 600 qui est corrélée avec le franchissement d'un palier pour HVCV, HVV et TopV Const Comp.

En résumé, ces graphiques mettent en évidence diverses corrélations entre les positionnements des phases d'exploration (issues de la classification incrémentielle) et certaines variations notables de divers indicateurs. La classification incrémentielle permet donc de mieux comprendre les corrélations qui existent entre divers indicateurs lors de la course itérative, tout en mettant à jour de nouvelles transitions.

Conclusion

Pour conclure, l'idée de distinguer les phases d'E&E en se basant sur la création de nouvelles classes de vecteurs au fil des itérations donne des marqueurs positionnés à des moments spécifiques de la course itérative. Ces moments se caractérisent par des variations notables de certains critères d'évaluation ou indicateurs. La contribution proposée donne donc une représentation de l'E&E sous un nouvel angle. Le positionnement des marqueurs, leurs espacements et répartitions constituent aussi des éléments permettant de comparer les algorithmes ainsi que les cas d'exécution. Ils permettent donc de comparer les comportements d'E&E, de souligner les comportements marginaux et de mettre à jour des comportements redondants. Enfin, la classification incrémentielle est capable de répertorier des transitions que les autres indicateurs ne parviennent pas à révéler.

4.9.4 Conclusion

Les indicateurs proposés donnent une représentation de l'E&E sous de nouveaux angles et sont aussi des marqueurs de différences entre les algorithmes et entre les cas d'exécution. Ils permettent donc de comparer les comportements d'E&E, de souligner les comportements marginaux, de mettre à jour des comportements redondants et d'explicitier les transitions qui s'opèrent entre E&E.

Notons que des corrélations, redondances et même incohérences entre indicateurs peuvent exister. Nous ne nous sommes pas intéressés à l'explicitation de ces phénomènes qui demanderaient des expérimentations supplémentaires. Notons aussi que ces indicateurs n'ont pas pour objectif d'apporter une explicabilité complète des comportements d'E&E. Il existe potentiellement énormément d'autres indicateurs à proposer en contributions.

La distinction des phases d'E&E par la classification incrémentielle donne des marqueurs positionnés à des moments spécifiques de la course itérative. Ces moments se caractérisent par des variations notables de certains critères d'évaluation ou indicateurs.

La classification incrémentielle est aussi capable de répertorier des transitions que les autres indicateurs ne parviennent pas à révéler.

Les résultats sont confirmés dans différents contextes (algorithmes, contraintes et autres), ce qui confirme la généralité des approches proposées.

Les approches proposées constituent donc des contributions participant à l'explicabilité de l'E&E.

Chapitre 5

Discussions et travaux futurs

5.1 Synthèse

Dans cette section nous rappelons nos questions de recherche, nos contributions et faisons une synthèse des conclusions de nos expérimentations.

5.1.1 Rappel des questions de recherche

Problématiques liées à l'influence de l'exploration et de l'exploitation

Il faut être en mesure d'influencer la dynamique de l'E&E au sein d'un important ensemble d'*itemsets* afin de garantir un bon comportement de recherche et ainsi trouver de bons *itemsets* au regard des critères d'évaluation. De notre point de vue, les processus d'influence proposés dans la littérature ne garantissent pas une expression très impactante de l'influence. Notre problématique est donc la suivante : Comment expliquer, mesurer, contrôler et influencer l'E&E ?

Problématiques liées à la réduction de données

Nous avons choisi de nous intéresser aux méthodes de réduction statistiquement conscientes car nous souhaitons conserver un maximum de caractéristiques des données, pour ne pas les dénaturer, tout en incluant des biais afin de faciliter l'atteinte de solutions difficilement accessibles. Nous avons donc besoin d'informations statistiques sur les données pour orienter le processus de réduction. La problématique que nous abordons dans ce travail est : Quels processus d'échantillonnage proposer afin de réduire un jeu de données en conservant un maximum de ses caractéristiques dans le but de servir une démarche d'E&E pour la recommandation ?

A partir des précédentes problématiques énoncées, nous proposons la problématique suivante : L'utilisation conjointe de processus de réduction des données statistiquement conscients et d'un processus pour influencer l'E&E, permet-elle d'avoir un impact significatif sur les critères d'évaluation de la performance, de la qualité et du comportement d'E&E d'un algorithme de recherche ?

Enfin, nous abordons aussi un problème de séquençage de fonctions de réduction de données en tant que processus seul permettant l'optimisation et la recommandation. Les problématiques associées sont : Un processus de séquençage de fonctions de réduction de données peut-il être utilisé seul pour la tâche d'optimisation et de recommandation ? Peut-il être révélateur de séquences compétitives de fonctions de réduction ?

Problématiques liées à l'explicabilité de l'E&E

Expliquer le comportement d'E&E permettrait indirectement de parvenir à de meilleures solutions et permettrait de renforcer la confiance dans les résultats des algorithmes. Nos problématiques sont donc les suivantes : Comment expliquer le comportement d'E&E ? Comment expliciter les transitions entre E&E ?

5.1.2 Rappel des contributions

Influence de l'E&E

Nos contributions relatives à l'influence de l'E&E portent sur trois points. Premièrement, nous proposons un processus d'influence indépendant de l'algorithme influencé afin de favoriser la pleine expression de l'influence et de ne pas opérer de modifications structurelles et/ou paramétriques à l'intérieur de l'algorithme influencé. Deuxièmement, nous proposons une influence systématique de l'E&E, à chaque itération de l'algorithme influencé et tant que les *itemsets* ne respectent pas un indicateur d'E&E. Enfin, nous introduisons des indicateurs pour mesurer et représenter l'E&E sous de nouveaux angles.

Nos travaux sur l'influence de l'E&E ont été l'occasion de proposer de nouveaux opérateurs de variations pour l'algorithmie évolutionnaire. Ces opérateurs peuvent être utilisés en tant qu'opérateurs d'influence de l'E&E mais aussi en tant qu'algorithmes évolutionnaires à part entière.

Réduction des données

Nos contributions consistent en un macro-processus et plusieurs fonctions pour la réduction d'un jeu de données par échantillonnage statistiquement conscient visant à réduire l'espace de recherche et à influencer indirectement l'E&E. Ces contributions discriminent les données en fonction d'informations représentatives. Cette approche apporte inévitablement des biais et nous faisons l'hypothèse que ces biais peuvent permettre d'atteindre des solutions difficilement accessibles.

Par ailleurs, nous proposons un nouveau traitement algorithmique opérant un séquençage de fonctions d'échantillonnage. Ce traitement est utilisé seul pour une tâche d'optimisation et de recommandation et permet de cibler les séquences de fonctions de réduction les plus intéressantes.

Explicabilité de l'E&E

Notre contribution relative à l'explicabilité de l'E&E consiste en de nouvelles approches permettant de donner des éléments d'information sur la manière avec laquelle une métaheuristique explore et exploite l'espace de recherche. Ces approches sont basées sur des indicateurs, issus d'hyper-volumes, de vecteurs de variation de présences ainsi que de résultats de classification, afin d'expliquer les transitions entre E&E.

Nous proposons plusieurs indicateurs hyper-volumiques permettant d'obtenir des informations sur le comportement d'E&E d'une MH.

Nous proposons aussi des indicateurs qui tentent de répondre à l'ouverture liée à la variété des transitions qui peuvent se produire entre E&E.

Par ailleurs, nous proposons un algorithme de classification d'*itemsets* intervenant à la fin de chaque itération d'une métaheuristique.

5.1.3 Synthèse des conclusions des expérimentations

Influence de l'E&E

Les critères de performance, de qualité et d'E&E étudiés montrent des impacts positifs du processus d'influence dans de nombreux cas d'exécution. L'approche proposée permet d'influencer le comportement d'E&E et les indicateurs d'influence proposés aident à fournir des explications concernant l'E&E. En effet, ils représentent les comportements d'E&E sous de nouveaux angles. Cependant, les résultats obtenus dépendent des cas d'exécution, nous concluons que l'influence proposée n'a pas un impact positif dans tous les contextes. Pour continuer, les résultats confirment que le processus d'influence n'induit pas de comportements aléatoires. Notons aussi que les résultats concernant les variations par rapport à la référence montrent des variations notables, ce qui vient souligner l'impact de l'approche. De plus, les résultats sont confirmés dans différents contextes (datasets, algorithmes, contraintes et autres), ce qui confirme la généralité de l'approche proposée.

Par ailleurs, les opérateurs proposés favorisent l'exploration et l'exploitation en fonction du cas d'exécution et impactent la régularité de l'E&E. Ils ont donc un impact sur le comportement de recherche. Notons qu'une analyse détaillée des résultats montre que certains opérateurs proposés sont plus impactant que d'autres. Nous notons également que les opérateurs proposés n'ont pas d'impact dans tous les contextes, cela était attendu car cela dépend de la structure de l'espace de recherche et de l'ensemble des contraintes. Soulignons que les opérateurs proposés provoquent donc des comportements d'E&E spécifiques, ce qui les rend potentiellement intéressants dans le cadre d'une démarche d'influence de l'E&E. Les opérateurs proposés pourraient donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation, toujours au regard de la performance, de la qualité et de l'E&E.

Réduction des données

Les critères de performance, de qualité et d'E&E étudiés montrent des impacts positifs du processus de réduction dans de nombreux cas d'exécution. Le processus de réduction impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques. Le processus de réduction pourrait donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation, toujours au regard de la performance, de la qualité et de l'E&E. Cependant, les résultats obtenus dépendent des cas d'exécution, ainsi le processus de réduction proposé n'a pas un impact positif dans tous les contextes. Soulignons que les résultats concernant les variations par rapport à la référence montrent des variations notables, ce qui vient souligner l'impact de l'approche. De plus, les résultats sont confirmés dans différents contextes (datasets, algorithmes, contraintes et autres), ce qui confirme la généralité de l'approche proposée.

Par ailleurs, les critères de performance, de qualité et d'E&E étudiés montrent des impacts positifs de la R_I dans de nombreux cas d'exécution. La R_I impacte la régularité de l'E&E et provoque donc des comportements d'E&E spécifiques. La R_I pourrait donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation, toujours au regard de la performance, de la qualité et de l'E&E. Ici aussi, la R_I n'a pas un impact positif dans tous les contextes, les variations par rapport à la référence viennent souligner l'impact de l'approche et les résultats sont confirmés dans différents contextes confirmant la généralité de l'approche proposée.

D'autre part, les critères d'évaluation étudiés montrent des impacts positifs de la réduction combinée dans de nombreux cas d'exécution. La réduction combinée pourrait donc, par exemple, permettre d'améliorer les solutions trouvées dans un contexte de recommandation. Une fois de

plus, la réduction combinée n'a pas un impact positif dans tous les contextes, les variations par rapport à la référence viennent souligner l'impact de l'approche et les résultats sont confirmés dans différents contextes confirmant la généralité de l'approche proposée.

Explicabilité de l'E&E

Les indicateurs proposés donnent une représentation de l'E&E sous de nouveaux angles et sont aussi des marqueurs de différences entre les algorithmes et entre les cas d'exécution. Ils permettent donc de comparer les comportements d'E&E, de souligner les comportements marginaux, de mettre à jour des comportements redondants et d'explicitier les transitions qui s'opèrent entre E&E. Notons que des corrélations, redondances et même incohérences entre indicateurs peuvent exister. Nous ne nous sommes pas intéressés à l'explicitation de ces phénomènes qui demanderaient des expérimentations supplémentaires. Notons aussi que ces indicateurs n'ont pas pour objectif d'apporter une explicabilité complète des comportements d'E&E. Il existe potentiellement énormément d'autres indicateurs à proposer en contributions.

Par ailleurs, la distinction des phases d'E&E par la classification incrémentielle donne des marqueurs positionnés à des moments spécifiques de la course itérative. Ces moments se caractérisent par des variations notables de certains critères d'évaluation ou indicateurs. De plus, la classification incrémentielle est aussi capable de répertorier des transitions que les autres indicateurs ne parviennent pas à révéler.

Enfin, les résultats sont confirmés dans différents contextes (algorithmes, contraintes et autres), ce qui confirme la généralité des approches proposées. Les approches proposées constituent donc des contributions participant à l'explicabilité de l'E&E.

5.2 Discussion

5.2.1 Généralités

Nos expérimentations ne pouvant être réalisées dans l'ensemble des contextes possibles, nous pouvons supposer que nos résultats peuvent être complétés et discutés au regard d'expérimentations dans des contextes d'exécution différents. En effet, les résultats dépendent des données, des objectifs de recommandation, des contraintes, des choix de paramètres et des algorithmes. Modifier ces éléments mènerait à des résultats différents. Dans le cadre des travaux présentés, nous avons choisi des éléments diversifiés mais rien ne nous permet de justifier nos choix par rapport à l'ensemble des combinaisons de choix possibles.

Il en va de même pour les volumes des expérimentations, ils sont discutables mais nous les jugeons suffisants au regard des volumes des expérimentations proposés dans l'état de l'art du domaine des MH.

Nos contributions peuvent aussi être analysées au regard d'autres critères d'évaluation qui pourraient mettre en lumière d'autres forces et faiblesses.

5.2.2 Influence de l'E&E

Concernant l'influence de l'E&E, nos résultats présentant des conclusions positives pour l'algorithme génétique ainsi que pour l'algorithme par renforcement, nous soulignons ici la généralité du processus d'influence proposé. Premièrement, l'influence indépendante proposée est adaptable aux MH et plus généralement à tout algorithme d'E&E. En effet, le principe d'une influence itérative indépendante est universel dès lors qu'il est possible de déterminer : (1) le moment auquel

appeler la fonction d'influence, (2) la nature de ce qui est passé en paramètre, (3) la manière dont les limites de l'indicateur d'E&E sont ajustées et (4) les variations à effectuer au sein d'une itération d'influence.

Pour continuer, les résultats dépendent des objectifs de recommandation, des contraintes et des choix de paramètres d'influence. Un ajustement systématique de ces paramètres de contexte permettrait aux fonctions d'influence d'améliorer plus souvent les performances des algorithmes non influencés et la qualité des recommandations.

Par ailleurs, le moment où la fonction d'influence est appelée impacte l'E&E et dépend de l'algorithme influencé ainsi que de la façon dont la fonction d'influence est conçue (indicateur d'E&E, opérateurs de variation, bornes, règles d'ajustement). Nous avons choisi de donner le dernier mot à la fonction d'influence en la positionnant avant la fin de chaque itération de l'algorithme influencé, ce choix peut être débattu.

De plus, les variations d'influence peuvent être effectuées de plusieurs façons, selon le contexte de mise en œuvre. Ajoutons que d'autres opérateurs de variations peuvent être utilisés, il serait donc intéressant de procéder à des expérimentations en ce sens.

Un autre point important dans l'approche proposée est que les variations d'influence opèrent directement sur les *itemsets* de l'itération actuelle de l'algorithme influencé. Ces *itemsets*, éventuellement mutés, sont renvoyés à l'algorithme influencé, qui continue avec eux tels quels. Cela risque potentiellement d'inclure trop d'aléatoire et donc de trop altérer les enrichissements de l'algorithme influencé. Cependant, cela dépend des opérateurs de variation utilisés, de la pertinence des indicateurs d'E&E et d'autres choix de paramètres. Pour déterminer ces éléments, des tests de fonctionnement peuvent être effectués, des valeurs arbitraires peuvent être utilisées, des informations statistiques de base concernant les données traitées peuvent également être exploitées.

Pour continuer, nous ne proposons pas de formalisation mathématique explicite des influences de nos processus d'influence sur le comportement d'E&E ainsi que sur l'atteinte de l'équilibre, ce qui nous ramène à cette importante frontière de l'état de l'art.

Nous ne nous sommes pas non plus intéressé aux synergies possibles entre opérateurs ni même aux propriétés de notre espace de recherche en terme d'influence sur l'E&E.

Enfin, concernant l'utilisation conjointe de processus de réduction des données et d'influence de l'E&E (R_I), les éléments de discussion sont identiques, portés à la dimension de la R_I .

5.2.3 Réduction de données

Concernant la réduction de données, les résultats dépendent des objectifs de recommandation, des contraintes et des choix de paramètres du processus de réduction. Un réglage fin de ces paramètres de contexte permettrait d'améliorer plus souvent l'impact de la réduction sur les performances, la qualité et le comportement d'E&E.

Ensuite, les processus de réduction proposés peuvent trop et/ou mal réduire les données. Il est donc ici possible de nous reprocher le fait que potentiellement notre approche risque de trop dénaturer l'espace de recherche en supprimant des optima intéressants. Cependant, cela dépend des indicateurs utilisés ainsi que de la structure initiale de l'espace de recherche. Des travaux annexes permettraient de clarifier la manière avec laquelle prendre en compte ces deux éléments afin d'obtenir de meilleures réductions des données.

Concernant l'utilisation de combinaisons de fonctions de réduction, les résultats dépendent des objectifs de recommandation et des contraintes. L'approche directe proposée, en tant que processus d'optimisation, n'est pas utile dans tous les contextes. Il est cependant intéressant de constater que dans certains contextes, elle surpasse des approches métaheuristiques classiques.

Nous expliquons cela par le fait qu'elle améliore la représentativité des données en prenant en compte différentes propriétés statistiques et donc différents biais de représentation. Mis en commun, ces biais peuvent faciliter l'accès à certaines solutions compétitives.

De plus, l'utilisation de l'approche par combinaisons de fonctions de réduction en tant que révélatrice d'hybridations de fonctions de réduction performantes semble pouvoir fournir des informations intéressantes quel que soit le contexte. En effet, il est possible de l'utiliser pour obtenir une séquence compétitive de fonctions de réduction pour ensuite utiliser complètement ou partiellement cette séquence au préalable de l'exécution d'une métaheuristique classique. Cependant, ces hybridations peuvent aussi faire exploser le temps de calcul en fonction de la nature des fonctions de réduction utilisées mais surtout en fonction du nombre d'occurrences de fonctions de réduction dans les ensembles de départ et d'arrivée en post-séquençage. L'inverse est aussi vrai.

5.2.4 Explicabilité de l'E&E

Concernant l'explicabilité de l'E&E, tout d'abord, nos indicateurs gagneraient à être analysés ensemble ainsi qu'en combinaison avec d'autres indicateurs (de l'état de l'art ou de nouveaux) ce qui permettrait sans doute d'obtenir une visualisation plus exhaustive du comportement d'E&E. Cependant, un compromis doit être fait car l'explicabilité, lorsqu'elle s'adresse à des intervenants humains, se doit d'être claire avant tout. D'autre part, lorsque l'explicabilité sert une démarche algorithmique, c'est-à-dire lorsqu'elle permet à un algorithme de prendre des décisions, elle se doit de reposer sur des principes statistiques solides. Le mélange des indicateurs présente donc des limites.

Ensuite, concernant l'échelle d'analyse, nous avons choisi d'opérer au niveau de l'itération, ce qui est un choix fort car nous pourrions aussi envisager d'opérer à plus petite échelle comme à l'échelle de la variation. Il est aussi possible d'envisager des contributions mélangeant des analyses à différentes échelles. Certains indicateurs pourraient donc être proposés, propres à des échelles spécifiques et/ou agissant sur des échelles intermédiaires c'est-à-dire des indicateurs de passage inter-échelles.

De plus, il serait intéressant de faire des analyses de corrélations entre les indicateurs. En effet, les graphiques issus de nos expérimentations semblent montrer diverses corrélations qu'il pourrait être utile d'explicitier mathématiquement pour servir une analyse plus fine du comportement d'E&E.

Enfin, nos travaux sur l'explicabilité de l'E&E ne font qu'effleurer la richesse et la complexité des processus d'influence de l'E&E ainsi que des transitions entre E&E.

5.3 Travaux futurs

En contributions générales, nous proposerons de nouveaux critères d'évaluation qui pourraient mettre en lumière d'autres forces et faiblesses de nos contributions et qui pourraient aussi permettre d'évaluer plus finement l'explicabilité de l'E&E.

5.3.1 Influence de l'E&E

Tout d'abord, nous ferons des expérimentations en effectuant les variations d'influence de diverses façons et aussi en utilisant d'autres opérateurs de variations.

De plus, nous nous intéresserons aux synergies possibles entre opérateurs de variations, aux propriétés des espaces de recherche en terme d'influence sur l'E&E ainsi qu'aux synergies possibles

entre indicateurs d'influence.

Pour continuer, nous proposerons de nouveaux processus d'influence de l'E&E ainsi que de nouveaux indicateurs d'influence de l'E&E.

Ensuite, il pourrait être pertinent de proposer une classification des indicateurs d'influence afin de mieux déterminer leurs conséquences et leurs coûts en fonction des contextes.

5.3.2 Réduction de données

Pour commencer, nous proposerons de nouvelles méthodes de réduction des données.

Ensuite, il serait intéressant de proposer un mode opératoire permettant de déterminer quelles fonctions de réduction utiliser pour quelles structures de l'espace de recherche.

Pour continuer, nous pourrions aussi nous intéresser à d'autres formes d'hybridations des méthodes de réduction en posant par exemple des conditions qui permettraient de changer de méthode de réduction lors du passage à une nouvelle itération de réduction. Ces autres formes d'hybridations devront s'appuyer sur des analyses plus fines de l'espace de recherche et des impacts respectifs des méthodes de réduction.

Par ailleurs, il serait aussi pertinent de proposer un indicateur de dégénération des données qui permettrait de stopper les itérations du macro-processus de réduction en cas de trop forte réduction ou en cas d'opération de réduction qui déforme trop l'espace de recherche. Pour juger de ces "trop", nous pourrions nous appuyer sur des indicateurs qui devront eux aussi être proposés en contributions. Il faut en effet être en capacité de juger de la morphologie de l'espace de recherche ainsi que de l'évolution de cette morphologie pour pouvoir prendre des décisions au regard des opérations de réduction.

De plus, et concernant en partie le point précédent, il pourrait être pertinent de proposer une classification des opérateur de réduction afin de mieux déterminer leurs conséquences et leurs coûts en fonction des contextes.

5.3.3 Explicabilité de l'E&E

Tout d'abord, nous analyserons nos indicateurs ensemble ainsi qu'en combinaison avec d'autres indicateurs (de l'état de l'art ou de nouveaux) ce qui permettra sans doute d'obtenir des visualisations plus exhaustives des comportements d'E&E. En ce sens, il serait intéressant de faire des analyses de corrélations entre les indicateurs. En effet, les graphiques issus de nos expérimentations semblent montrer diverses corrélations qu'il pourrait être utile d'explicitement pour servir une analyse plus fine du comportement d'E&E.

De plus, nous proposerons de nouveaux indicateurs et visualisations servant l'explicabilité de l'E&E.

Par ailleurs, nous travaillerons sur des méthodes exploitant les indicateurs proposés pour influencer l'E&E et améliorer la précision de la recherche. En effet, les informations émanant des processus d'explicabilité peuvent être utilisées en tant qu'indicateurs permettant de décider si l'E&E doivent être influencées.

Enfin, nous chercherons aussi à nous positionner sur l'explicabilité de l'impact des processus de réduction des données sur l'E&E.

5.4 Conclusion

Pour conclure, que ce soit en matière de réduction de données, d'influence de l'E&E ou d'explicabilité de l'E&E, le spectre des contributions futures possibles est important. Les éléments

de discussion concernant ce spectre le sont tout autant. A cela s'ajoute le fait que de nombreuses problématiques mentionnées dans l'état de l'art ne sont pas couvertes dans ce document. S'ajoute aussi l'ensemble des problématiques similaires hybrides aux domaines des MH et de l'apprentissage machine profond dont les jonctions sont de plus en plus populaires et débattues. Tout cela explique pourquoi ces domaines, malgré leur actuelle ancienneté, conserveront très probablement leur niveau d'intérêt dans les futures décennies.

Annexe A

Requêtes d'extraction MovieLens et Riiid

Algorithm 25 MovieLens - Requête d'extraction - 1

```
1: SELECT
2:   r1.movieId,
3:   r1.mean_rating,
4:   r1.rating_quantity,
5:   AVG(genome_scores.relevance) as mean_tag_relevance
6: FROM (
7:   SELECT
8:     movies.movieId as movieId,
9:     AVG(ratings.rating) as mean_rating,
10:    COUNT(*) as rating_quantity
11:  FROM movies
12:  INNER JOIN ratings
13:  ON movies.movieId = ratings.movieId
14:  WHERE movies.genres LIKE '%Action%'
15:  OR movies.genres LIKE '%Adventure%'
16:  OR movies.genres LIKE '%Romance%'
17:  GROUP BY movies.movieId
18:  HAVING mean_rating > 3
19:  AND rating_quantity >= 100
20: ) as r1
21: INNER JOIN genome_scores
22: ON r1.movieId = genome_scores.movieId
23: GROUP BY r1.movieId, r1.mean_rating, r1.rating_quantity
24: HAVING mean_tag_relevance > 0.08
```

Algorithm 26 MovieLens - Requête d'extraction - 2

```
1: SELECT
2:   r1.movieId,
3:   r1.mean_rating,
4:   r1.rating_quantity,
5:   AVG(genome_scores.relevance) as mean_tag_relevance
6: FROM (
7:   SELECT
8:     movies.movieId as movieId,
9:     AVG(ratings.rating) as mean_rating,
10:    COUNT(*) as rating_quantity
11:  FROM movies
12:  INNER JOIN ratings
13:  ON movies.movieId = ratings.movieId
14:  WHERE movies.genres LIKE '%Thriller%'
15:  GROUP BY movies.movieId
16:  HAVING mean_rating >= 2.5
17:  AND rating_quantity >= 1500
18: ) as r1
19: INNER JOIN genome_scores
20: ON r1.movieId = genome_scores.movieId
21: GROUP BY r1.movieId, r1.mean_rating, r1.rating_quantity
22: HAVING mean_tag_relevance > 0.1
```

Algorithm 27 MovieLens - Requête d'extraction - 3

```
1: SELECT
2:   r1.movieId,
3:   r1.mean_rating,
4:   r1.rating_quantity,
5:   AVG(genome_scores.relevance) as mean_tag_relevance
6: FROM (
7:   SELECT
8:     movies.movieId as movieId,
9:     AVG(ratings.rating) as mean_rating,
10:    COUNT(*) as rating_quantity
11:  FROM movies
12:  INNER JOIN ratings
13:  ON movies.movieId = ratings.movieId
14:  WHERE (
15:    movies.genres LIKE '%Documentary%'
16:    OR movies.genres LIKE '%Mystery%'
17:  )
18:  AND ratings.timestamp >= 1104537600
19:  GROUP BY movies.movieId
20:  HAVING mean_rating >= 3.5
21:  AND rating_quantity >= 1000
22: ) as r1
23: INNER JOIN genome_scores
24: ON r1.movieId = genome_scores.movieId
25: GROUP BY r1.movieId, r1.mean_rating, r1.rating_quantity
26: HAVING mean_tag_relevance > 0.05
```

Algorithm 28 Riiid - Requête d'extraction - 1

```
1: SELECT *
2: FROM Riiid_Reduced_Aggregated
3: WHERE avg_answered_correctly > 0.6
4: AND avg_prior_question_elapsed_time < 20000
5: AND avg_prior_question_had_explanation > 0.6
```

Algorithm 29 Riiid - Requête d'extraction - 2

```
1: SELECT *
2: FROM Riiid_Reduced_Aggregated
3: WHERE avg_answered_correctly > 0.3
4: AND avg_answered_correctly < 0.7
5: AND avg_prior_question_elapsed_time > 20000
6: AND avg_prior_question_elapsed_time < 30000
7: AND avg_prior_question_had_explanation > 0.3
8: AND avg_prior_question_had_explanation < 0.7
```

Algorithm 30 Riid - Requête d'extraction - 3

```
1: SELECT *
2: FROM Riid_Reduced_Aggregated
3: WHERE avg_answered_correctly > 0.5
4: AND avg_answered_correctly < 0.8
5: AND avg_prior_question_elapsed_time > 10000
6: AND avg_prior_question_elapsed_time < 40000
7: AND avg_prior_question_had_explanation > 0.5
8: AND avg_prior_question_had_explanation < 0.8
```

Bibliographie

- [1] 2021. URL : <https://www.kaggle.com/c/riiid-test-answer-prediction>.
- [2] en. In : *Machine Learning, chap. 13 Reinforcement Learning*, p. 367-390. URL : https://fr.wikipedia.org/wiki/Reference:Machine_Learning.
- [3] URL : <https://www.quora.com/How-do-you-measure-and-evaluate-the-quality-of-recommendation-engines>.
- [4] Monreale A. « Privacy-preserving distributed movement data aggregation ». en. In : *Geographic information science at the heart of Europe*. Springer, 2013, p. 225-245.
- [5] « A Framework for Dynamic Knowledge Modeling in Textbook-Based Learning ». en. In : *2016 Conference on User Modeling Adaptation and Personalization*. Halifax, Canada : ACM, 2016, p. 210.
- [6] « A hybrid evolutionary algorithm for tuning a cloth simulation model ». en. In : *Appl. Soft Comput* 12, 1 (2012), p. 266-273.
- [7] *A survey on data preprocessing for data stream mining : Current status and future directions*. fr. 2017.
- [8] Eiben A.E et Schippers C. « On evolutionary exploration and exploitation, 1998 ». it. In : *Fundamenta Informaticae* 35.1-4 (), p. 35-50.
- [9] Himan ABDOLLAHPOURI. *Incorporating System-Level Objectives into Recommender Systems*. en. 2019.
- [10] Himan ABDOLLAHPOURI et al. « Beyond personalization : Research directions in multi-stakeholder recommendation ». In : *arXiv preprint arXiv :1905.01986* (2019).
- [11] J. ACHARYA et al. « Fast and near-optimal algorithms for approximating distributions by histograms ». en. In : *Proceedings of the 34th ACM SIGMOD-SIGACTSIGAI Symposium on Principles of Database Systems (PODS'15)*. Melbourne, Australia, 2015, p. 249-263.
- [12] K. ACKERMANN et S.D. ANGUS. « A resource efficient big data analysis method for the social sciences : the case of global IP activity ». fr. In : *Procedia Comput Sci* 29 (2014), p. 2360-2369.
- [13] Gediminas ADOMAVICIUS et Young Ok KWON. pt. In : *Recommender systems handbook*. Springer, 2015, p. 847-880.
- [14] S.F. ADRA et P.J. FLEMING. « Diversity management in evolutionary many-objective optimization ». en. In : *IEEE Trans. Evol. Comput* 15, 2 (2011), p. 183-195.
- [15] C.C. AGGARWAL. « On biased reservoir sampling in the presence of stream evolution ». en. In : *Proceedings of the 32nd International Conference on Very Large Databases (VLDB'06)*, Seoul. Korea, 2006, p. 607-618.

- [16] E. ALBA et B. DORRONSORO. « The exploration/exploitation tradeoff in dynamic cellular genetic algorithms ». en. In : *IEEE Trans. Evol. Comput* 9, 3 (2005), p. 126-142.
- [17] Sura ALI et al. *A new model for large dataset dimensionality reduction based on teaching learning-based optimization and logistic regression*. en. 2020.
- [18] H.B. AMOR et A. RETTINGER. « Intelligent exploration for genetic algorithms : Using self-organizing maps in evolutionary computation ». en. In : *Proceedings of the 7th Genetic and Evolutionary Computation Conference*. 2005, p. 1531-1538.
- [19] C.J. ANDERSON. « The psychology of doing nothing : Forms of decision avoidance result from reason and emotion ». en. In : *Psychological Bulletin* 129 (2003), p. 139-167. DOI : [10.1037/0033-2909](https://doi.org/10.1037/0033-2909). URL : <http://dx.doi.org/10.1037/0033-2909>.
- [20] L. ARAUJO et J.J. MERELO. « Diversity through multiculturalism : Assessing migrant choice policies in an island model ». en. In : *IEEE Trans. Evol. Comput* 15, 4 (2011), p. 456-468.
- [21] H.R. ARKES et P. AYTON. « The sunk cost and Concorde effects : Are humans less rational than lower animals? ». en. In : *Psychological Bulletin* 125 (1999), p. 591-600. DOI : [10.1037/0033-2909.125.5](https://doi.org/10.1037/0033-2909.125.5). URL : <http://dx.doi.org/10.1037/0033-2909.125.5>.
- [22] Micheal O. AROWOLO et al. *A Hybrid Dimensionality Reduction Model for Classification of Microarray Dataset*. en. 2017.
- [23] G. ASTON-JONES et J.D. COHEN. « An integrative theory of locus coeruleus-norepinephrine function : Adaptive gain and optimal performance ». en. In : *Annual Review of Neuroscience* 28 (2005), p. 403-450. DOI : [10.1146/annurev](https://doi.org/10.1146/annurev). URL : <http://dx.doi.org/10.1146/annurev>.
- [24] Anne AUGER, Marc SCHOENAUER et Olivier TEYTAUD. « Local and global order 3/2 convergence of a surrogate evolutionary algorithm ». In : *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. 2005, p. 857-864.
- [25] J.M. AW, M. VASCONCELOS et A. KACELNIK. « How costs affect preferences : Experiments on state dependence, hedonic state and within-trial contrast in starlings ». en. In : *Animal Behaviour* 81 (2011), p. 1117-1128. DOI : [10.1016/j.anbehav.2011.02.015](https://doi.org/10.1016/j.anbehav.2011.02.015). URL : <http://dx.doi.org/10.1016/j.anbehav.2011.02.015>.
- [26] M Emin AYDIN et Terence C FOGARTY. « A distributed evolutionary simulated annealing algorithm for combinatorial optimisation problems ». In : *Journal of heuristics* 10.3 (2004), p. 269-292.
- [27] Di Martino B. « Big data (lost) in the cloud ». en. In : *Int J Big Data Intell* 1.1-2 (2014), p. 3-17.
- [28] T. BACK. « Selective pressure in evolutionary algorithms : A characterization of selection mechanisms ». en. In : *Proceedings of the 1st Conference on Evolutionary Computing*. 1994, p. 57-62.
- [29] T. BACK et al. « An empirical study on gas without parameters ». en. In : *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature*. Springer-Verlag, 2000, p. 315-324.
- [30] I.E. BACON, T.A. HURLY et S. HEALY. « Both the past and the present affect risk-sensitive decisions of foraging rufous hummingbirds ». en. In : *Behavioral Ecology* 21 (2010), p. 626-632. DOI : [10.1093/beheco/arq031](https://doi.org/10.1093/beheco/arq031). URL : <http://dx.doi.org/>.

- [31] M. BADER-EL-DEN, E. TEITEI et T. PERRY. « Biased random forest for dealing with the class imbalance problem ». en. In : *IEEE Trans Neural Netw Learn Syst* 2018.30(7) :2163-2172 ().
- [32] AO BAJEH et KO ABOLARINWA. « Optimization : A comparative study of genetic and tabu search algorithms ». In : *International Journal of Computer Applications* 31.5 (2011), p. 43-48.
- [33] K. BAKSHI. « Considerations for big data : Architecture and approach ». en. In : *Proc. of 2012 IEEE Aerospace Conference*. Big Sky, MT, USA, 2012, p. 1-7.
- [34] Daniel BALOUEK et AL. « Adding Virtualization Capabilities to the Grid'5000 Testbed ». In : *Cloud Computing and Services Science*. T. 367. Springer International Publishing, 2013, p. 3-20. ISBN : 978-3-319-04518-4. DOI : [10.1007/978-3-319-04519-1_1](https://doi.org/10.1007/978-3-319-04519-1_1).
- [35] T. BARTZ-BEIELSTEIN, C.W.G. LASARCZYK et M. PREUSS. « Sequential parameter optimization ». en. In : *Proceedings of the IEEE Congress on Evolutionary Computation*. 2005, p. 773-780.
- [36] Thomas BARTZ-BEIELSTEIN et al. *Experimental methods for the analysis of optimization algorithms*. Springer, 2010.
- [37] Joel AC BAUM, Stan Xiao LI et John M USHER. « Making the next move : How experiential and vicarious learning shape the locations of chains' acquisitions ». In : *Administrative Science Quarterly* 45.4 (2000), p. 766-801.
- [38] R.L. BECERRA, C.O.E.L.L.O. COELLO et C. A. « Cultured differential evolution for constrained optimization ». en. In : *Comput. Methods Appl. Mech. Eng* 195 (2006), p. 33-36, 4303-4322.
- [39] Samia BELDJOU DI, Hassina SERIDI et Nour El Islem KARABADJI. « Recommendation in collaborative e-learning by using linked open data and ant colony optimization ». In : *International Conference on Intelligent Tutoring Systems*. Springer. 2018, p. 23-32.
- [40] W.J. BELL. *Searching behaviour : The behavioural ecology of finding resources*. en. London, UK : Chapman et Hall, 1991.
- [41] T. BERSANO-BEGEY. « Controlling exploration, diversity and escaping local optima in GP : Adopting weights of training sets to model resource consumption ». en. In : *Proceedings of the Late Breaking Papers at the Genetic Programming Conference*. 1997, p. 7-10.
- [42] A. BETTINGER, A. BRUN et A. BOYER. « Independent Influence of Exploration and Exploitation for Metaheuristic-based Recommendations ». In : *Genetic and Evolutionary Computation Conference Companion (GECCO '22 Companion), July 9-13, 2022, Boston, MA, USA*. 2022. ISBN : 978-1-4503-9268-6/22/07. DOI : [10.1145/3520304.3528972](https://doi.org/10.1145/3520304.3528972).
- [43] A. BETTINGER, A. BRUN et A. BOYER. « Presence Variations for E&E Transitions Explainability ». In : *26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022)*. 2022.
- [44] Alexandre BETTINGER, Armelle BRUN et Anne BOYER. *Independent Influence of Exploration and Exploitation for Metaheuristic-based Recommendations*. fr. 2021. URL : <https://docs.google.com/document/d/18ctAiQlAYl6WdOdoaq95lNwsTDYFSCz70igdAgCS-zA/edit?usp=sharing>.
- [45] H.-G. BEYER et K. DEB. « On self-adaptive features in real-parameter evolutionary algorithms ». en. In : *IEEE Trans. Evol. Comput* 5, 3 (2001), p. 250-270.

- [46] M. BIRATTARI et al. « A racing algorithm for configuring metaheuristics ». en. In : *Proceedings of the Genetic and Evolutionary Computation Conference*. 2002, p. 11-18.
- [47] Mauro BIRATTARI et Janusz KACPRZYK. *Tuning metaheuristics : a machine learning perspective*. T. 197. Springer, 2009.
- [48] Darius BIRVINSKAS et al. « EEG Dataset Reduction and Feature Extraction using Discrete Cosine Transform ». en. In : (2012).
- [49] A.L. BLUM et P. LANGLEY. « Selection of relevant features and examples in machine learning ». en. In : *ARTIF INTELL* 97.1-2 (1997), p. 245-271. DOI : [10.1016/S0004-3702](https://doi.org/10.1016/S0004-3702). URL : <https://doi.org/10.1016/S0004-3702>.
- [50] C. BLUM et A. ROLI. *Metaheuristics in combinatorial optimization : Overview and conceptual comparison*. en. 2003.
- [51] C. BLUM et A. ROLI. « Metaheuristics in combinatorial optimization : overview and conceptual comparison ». en. In : *ACM Comput Surv* 35.3 (2003), p. 268-308.
- [52] C. BLUM et al. « Hybrid metaheuristics in combinatorial optimization : A survey ». en. In : *Appl. Soft Comput* 11, 6 (2011), p. 4135-4151.
- [53] R. BOGACZ et al. « The neural basis of the speed-accuracy tradeoff ». en. In : *Trends in Neurosciences* 33 (2010), p. 10-16. DOI : [10.1016/j.tins.2009](https://doi.org/10.1016/j.tins.2009). URL : <http://dx.doi.org/10.1016/j.tins.2009>.
- [54] T. BOGON et al. « Extraction of function features for an automatic configuration of particle swarm optimization ». en. In : *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence*. 2011, p. 51-60.
- [55] Y. BORENSTEIN et R. POLI. « Information landscapes ». en. In : *Proceedings of the Genetic and Evolutionary Computation Conference*. 2005, p. 1515-1522.
- [56] Y. BORENSTEIN et R. POLI. « Information landscapes and problem hardness ». en. In : *Proceedings of the Genetic and Evolutionary Computation Conference*. 2005, p. 1425-1431.
- [57] Y. BORENSTEIN et R. POLI. « Kolmogorov complexity, optimization and hardness ». en. In : *IEEE congress on evolutionary computation*. 2006, p. 112-119.
- [58] P. BOSMAN et D. THIERENS. « The balance between proximity and diversity in multiobjective evolutionary algorithms ». en. In : *IEEE Trans. Evol. Comput* 7, 2 (2003), p. 174-188.
- [59] David BOULANGER et Vivekanandan KUMAR. *An Overview of Recent Developments in Intelligent eTextbooks and Reading Analytics*. en. Edmonton, Canada : Athabasca University.
- [60] P. De BRA. « Teaching through adaptive hypertext on the www ». en. In : *Int. Journal of Educational Telecommunications* 3.2/3 (1997).
- [61] L. BREIMAN et al. *Classification and Regression Trees*. en. Boca Raton, FL, USA : CRC Press, 1984.
- [62] J. BREST et al. « Self-adapting control parameters in differential evolution : A comparative study on numerical benchmark problems ». pt. In : *IEEE Trans. Evol. Comput.* 10 6 (2006), p. 646-657.
- [63] C.T. BROWN. *BIGDATA : small : DA : DCM : low-memory streaming prefilters for biological sequencing data*. it. 2012.

- [64] P. BRUSILOVSKY et J. EKLUND. « A study of user-model based link annotation in educational hypermedia ». en. In : *J. of Universal Computer Science* 4.4 (1998).
- [65] E. BURKE, S. GUSTAFSON et G. KENDALL. « Diversity in genetic programming : An analysis of measures and correlation with fitness ». en. In : *IEEE Trans. Evol. Comput* 8, 1 (2004), p. 47-62.
- [66] E. BURKE et al. « Advanced population diversity measures in genetic programming ». en. In : *Proceedings of Parallel Problem Solving from Nature, Lecture Notes in Computer Science*. T. 2439. Springer, 2002, p. 341-350.
- [67] J.R. BUSEMEYER et J.T. TOWNSEND. « Decision field theory : A dynamic-cognitive approach to decision making in an uncertain environment ». en. In : *Psychological Review* 100 (1993), p. 432-459. URL : <http://dx.doi>.
- [68] Yang C. « A spatiotemporal compression based approach for efficient big data processing on Cloud ». it. In : *J Comput Syst Sci* 80.8 (2014), p. 1563-1583.
- [69] P.F. CARVALHO et al. « Analyzing the relative learning benefits of completing required activities and optional readings in online courses ». en. In : *The 11th Int. Conf. on Educational Data Mining*. T. 34, p. 68.
- [70] R. CATTELL. « Scalable SQL and NoSQL data stores ». en. In : *ACM SIGMOD Record* 39.4 (2011), p. 12-27.
- [71] H. CEN, K. KOEDINGER et B. JUNKER. « Learning factors analysis : A general method for cognitive model evaluation and improvement ». da. In : *Intelligent Tutoring Systems, Lecture Notes in Computer Science*. T. 4053. Berlin / Heidelberg : Springer, 2006, p. 164-175.
- [72] Alejandro CERVANTES, David QUINTANA et Gustavo RECIO. *Efficient dynamic resampling for dominance-based multiobjective evolutionary optimization*. en. 2016.
- [73] N. CHAIYARATANA, T. PIROONRATANA et N. SANGKAWELERT. « Effects of diversity control in single-objective and multi-objective genetic algorithms ». en. In : *J. Heuristics* 13 (2007), p. 1-34.
- [74] M. CHARIKAR, K. CHEN et M. FARACH-COLTON. « Finding frequent items in data streams ». en. In : *Proceedings of the 29th International Colloquium on Automata, Languages and Programming (ICALP'02)*. Malaga, Spain, 2002, p. 693-703.
- [75] E.L. CHARNOV. « Optimal foraging, the marginal value theorem ». en. In : *Theoretical Population Biology* 9.76 (1976), p. 129-136. DOI : [10.1016/0040-](https://doi.org/10.1016/0040-). URL : <http://dx.doi.org/10.1016/0040->.
- [76] S. CHAUDHURI, R. MOTWANI et V. NARASAYYA. « Random sampling for histogram construction : How much is enough ? ». en. In : *ACM SIGMOD Record* 27.2 (1998), p. 436-447.
- [77] G. CHEN, C.A. GONG et R. HUANG. « E-textbook : definition, function and key technical issues ». en. In : *Open Education Research* 18.2 (2012), p. 28-32.
- [78] G. CHEN, C.P. LOW et Z. YANG. « Preserving and exploiting genetic diversity in evolutionary programming algorithms ». en. In : *IEEE Trans. Evol. Comput* 13, 3 (2009), p. 661-673.
- [79] Jie CHEN et al. « Optimal contraction theorem for exploration–exploitation tradeoff in search and optimization ». In : *IEEE Transactions on Systems, Man, and Cybernetics-Part A : Systems and Humans* 39.3 (2009), p. 680-691.

- [80] Minmin CHEN. « Exploration in Recommender Systems ». In : *Fifteenth ACM Conference on Recommender Systems*. Amsterdam Netherlands : ACM, 2021, p. 551-553.
- [81] X. CHEN et M. XIE. « A split-and-conquer approach for analysis of extraordinarily large data ». it. In : *Statistica Sinica* 24.4 (2014), p. 1655-1684.
- [82] Min-Yuan CHENG et Doddy PRAYOGO. « Symbiotic organisms search : a new metaheuristic optimization algorithm ». In : *Computers & Structures* 139 (2014), p. 98-112.
- [83] Marco CHIARANDINI et al. « Experiments on metaheuristics : Methodological overview and open issues ». In : *Technical Report*. Citeseer, 2007.
- [84] L. CHITTKA, P. SKORUPSKI et N.E. RAINE. « Speed-accuracy tradeoffs in animal decision making ». en. In : *Trends in Ecology & Evolution* 24 (2009), p. 400-407. DOI : [10.1016/j.tree.2009.02.010](https://doi.org/10.1016/j.tree.2009.02.010). URL : <http://dx.doi.org/10.1016/j.tree.2009.02.010>.
- [85] Youngduck CHOI et al. « Ednet : A large-scale hierarchical dataset in education ». In : *International Conference on Artificial Intelligence in Education*. Springer. 2020, p. 69-73.
- [86] C.K. CHOW et S.Y. YUEN. « An evolutionary algorithm that makes decision based on the entire previous search history ». en. In : *IEEE Trans. Evol. Comput* 15, 6 (2011), p. 741-769.
- [87] H.G. COBB et J.J. GREFENSTETTE. « Genetic algorithms for tracking changing environments ». en. In : *Proceedings of the 5th International Conference on Genetic Algorithms*. 1993, p. 523-530.
- [88] J.D. COHEN, S.M. MCCLURE et A.J. YU. « Should I stay or should I go? How the human brain manages the trade-off between exploitation and exploration ». en. In : *Philosophical Transactions of the Royal Society of London Series B, Biological Sciences* 362 (2007), p. 933-942. URL : <http://dx.doi.org/10.1098/>.
- [89] Jonathan D COHEN, Samuel M MCCLURE et Angela J YU. « Should I stay or should I go? How the human brain manages the trade-off between exploitation and exploration ». In : *Philosophical Transactions of the Royal Society B : Biological Sciences* 362.1481 (2007), p. 933-942.
- [90] Z. COOK, D.W. FRANKS et E.J.H. ROBINSON. « Exploration versus exploitation in polydomous ant colonies ». en. In : *Journal of Theoretical Biology* 323 (2013), p. 49-56. DOI : [10.1016/j.jtbi.2013](https://doi.org/10.1016/j.jtbi.2013). URL : <http://dx.doi.org/10.1016/j.jtbi.2013>.
- [91] A.T. CORBETT et ANDERSON. « J.R. : Knowledge tracing : Modelling the acquisition of procedural knowledge ». en. In : *User Modeling and User-Adapted Interaction* 4.4 (1995), p. 253-278.
- [92] G. CORMODE et N. DUFFIELD. « Sampling for big data : A tutorial ». en. In : *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'14)*. New York, NY, USA, 2014, p. 1975-1975.
- [93] M. CREPINSEK, M. MERNIK et S.-H. LIU. « Analysis of exploration and exploitation in evolutionary algorithms by ancestry trees ». en. In : *Int. J. Innovative Comput. Appl* 3, 1 (2011), p. 11-19.
- [94] Matej CREPINSEK, Shih-hsi LIU et Marjan MERNIK. « Exploration and Exploitation in Evolutionary Algorithms : A Survey ». en. In : *ACM Computing Surveys* 45.35 (3 juin 2013), p. 1-33. DOI : [10.1145/2480741.2480752](https://doi.org/10.1145/2480741.2480752).

-
- [95] Matej ČREPINŠEK, Shih-Hsi LIU et Luka MERNIK. « A note on teaching–learning-based optimization algorithm ». In : *Information Sciences* 212 (2012), p. 79-93.
- [96] Erik CUEVAS et al. « A swarm optimization algorithm inspired in the behavior of the social-spider ». In : *Expert Systems with Applications* 40.16 (2013), p. 6374-6384.
- [97] D. CURRAN et C. O’RIORDAN. « Increasing population diversity through cultural learning ». en. In : *Adapt. Behav* 14, 4 (2006), p. 315-338.
- [98] A. CZARN et al. « Statistical exploratory analysis of genetic algorithms ». en. In : *IEEE Trans. Evol. Comput* 8, 4 (2004), p. 405-421.
- [99] P. D’HAESELEER et J. BLUMING. *Advances in Genetic Programming*. en. Cambridge, MA : MIT Press, 1994.
- [100] M. DASH et W. NG. « Efficient reservoir sampling for transactional data streams ». en. In : *Proc. of Sixth IEEE International Conference on Data Mining-Workshops (ICDMW’06), Hong Kong, China, 2006*, p. 662-666.
- [101] DE. DE JONG, R.A. WATSON et J.B. POLLACK. « Reducing bloat and promoting diversity using multi-objective methods ». en. In : *Proceedings of the 3rd Genetic and Evolutionary Computation Conference*. 2001, p. 11-18.
- [102] KA. DE JONG. « An analysis of the behavior of a class of genetic adaptive systems ». en. Ph.D. dissertation, Ann Arbor, MI : University of Michigan, 1975.
- [103] Kenneth A DE JONG. « Are genetic algorithms function optimizers ? » In : *PPSN*. T. 2. 1. 1992, p. 3-14.
- [104] Kenneth A DE JONG et William M SPEARS. « A formal analysis of the role of multi-point crossover in genetic algorithms ». In : *Annals of mathematics and Artificial intelligence* 5.1 (1992), p. 1-26.
- [105] K. DEB et D.E. GOLDBERG. « An investigation of niche and species formation in genetic function optimization ». en. In : *Proceedings of the 3rd International Conference on Genetic Algorithms*. 1989, p. 42-50.
- [106] J. DENRELL et G. LE MENS. « Seeking positive experiences can produce illusory correlations ». en. In : *Cognition* 119 (2011), p. 313-324. URL : <http://dx.doi.org/>.
- [107] J. DENRELL et J.G. MARCH. « Adaptation as information restriction : The hot stove effect ». en. In : *Organization Science* 12 (2001), p. 523-538. DOI : [10.1287/orsc.12.5.523.10092](https://doi.org/10.1287/orsc.12.5.523.10092). URL : <http://dx.doi.org/>.
- [108] Sachin DESALE et al. « Heuristic and meta-heuristic algorithms and their relevance to the real world : a survey ». In : *Int. J. Comput. Eng. Res. Trends* 351.5 (2015), p. 2349-7084.
- [109] R. DHAR. « The effect of decision strategy on deciding to defer choice ». en. In : *Journal of Behavioral Decision Making* 9.4265 (1996), p. 265-281. URL : <http://dx.doi.org/>.
- [110] Tansel DOKEROGLU et al. « A survey on new generation metaheuristic algorithms ». In : *C&IE* (2019).
- [111] *Domain Modeling for Personalized Guidance*. en. Peter Brusilovsky, University of Pittsburgh.
- [112] J. DREO. « Using performance fronts for parameter setting of stochastic metaheuristics ». en. In : *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference*. 2009, p. 2197-2200.

- [113] « Dynamic knowledge modeling with heterogeneous activities for adaptive textbooks ». en. In : *The 11th Int. Conf. on Educational Data Mining*. 2018, p. 230.
- [114] M. EAGLE et al. « Estimating individual differences for student modeling in intelligent tutors from reading and pretest data ». en. In : *Intelligent Tutoring Systems*. Sous la dir. d'A. MICARELLI, J. STAMPER et K. PANOURGIA. Cham : Springer International Publishing, 2016, p. 133-143.
- [115] W. EDWARDS. « Optimal strategies for seeking information : Models for statistics, choice reaction times, and human information processing ». en. In : *Journal of Mathematical Psychology* 2 (1965), p. 312-329. DOI : [10.1016/0022-2496\(65\)90007-6](https://doi.org/10.1016/0022-2496(65)90007-6). URL : [http://dx.doi.org/10.1016/0022-2496\(65\)90007-6](http://dx.doi.org/10.1016/0022-2496(65)90007-6).
- [116] B. EFRON et R.J. TIBSHIRANI. *An Introduction to the Bootstrap*. en. New York, NY, USA : Chapman & Hall, 1993.
- [117] A.E. EIBEN, R. HINTERDING et Z. MICHALEWICZ. « Parameter control in evolutionary algorithms ». pt. In : *IEEE Trans. Evol. Comput* 3, 2 (1999), p. 124-141.
- [118] A.E. EIBEN, E. MARCHIORI et V.A. VALKO. « Evolutionary algorithms with on-the-fly population size adjustment ». en. In : *Proceedings of Parallel Problem Solving from Nature, Lecture Notes in Computer Science*. T. 3242. Springer, 2004, p. 41-50.
- [119] A.E. EIBEN et S.K. SMIT. « Parameter tuning for configuring and analyzing evolutionary algorithms ». en. In : *Swarm Evol. Comput* 1, 1 (2011), p. 19-31.
- [120] Agoston E EIBEN et Cornelis A SCHIPPERS. « On evolutionary exploration and exploitation ». In : *Fundamenta Informaticae* 35.1-4 (1998), p. 35-50.
- [121] L.J. ESHELMAN. « The CHC adaptive search algorithm : How to have safe search when engaging in non-traditional genetic recombination ». en. In : *Found. Genetic Algorithms* 1 (1991), p. 265-283.
- [122] L.J. ESHELMAN et J. SCHAFFER. « Preventing premature convergence in genetic algorithms by preventing incest ». en. In : *Proceedings of the 4th International Conference on Genetic Algorithms*. 1991, p. 115-122.
- [123] J. FAN, F. HAN et H. LIU. « Challenges of big data analysis ». da. In : *Nat Sci Rev* 1.2 (2014), p. 293-314.
- [124] Rebecca FERGUSON. « Learning analytics : drivers, developments and challenges ». In : *International Journal of Technology Enhanced Learning* 4.5/6 (2012), p. 304-317.
- [125] T.S. FERGUSON. « Who solved the secretary problem ? » en. In : *Statistical Science* 4 (1989), p. 282-289. DOI : [10.1214/ss/1177012493](https://doi.org/10.1214/ss/1177012493).
- [126] K. FIEDLER et Y. KAREEV. « Clarifying the advantage of small samples : As it relates to statistical Wisdom and Cahan's ». en. In : *Journal of Experimental Psychology : Learning, Memory, and Cognition* 37 (2011), p. 1039-1043. DOI : [10.1037/a0023259](https://doi.org/10.1037/a0023259). URL : <http://dx.doi.org/10.1037/a0023259>.
- [127] K. FIEDLER et Y. KAREEV. « Does decision quality (always) increase with the size of information samples ? Some vicissitudes in applying the law of large numbers ». en. In : *Journal of Experimental Psychology : Learning, Memory, and Cognition* 32 (2006), p. 883-903. DOI : [10.1037/0278-7393](https://doi.org/10.1037/0278-7393). URL : <http://dx.doi.org/10.1037/0278-7393>.
- [128] Iztok FISTER, Marjan MERNIK et Bogdan FILIPIC. « A hybrid self-adaptive evolutionary algorithm for marker optimization in the clothing industry ». In : *Applied Soft Computing* 10.2 (2010), p. 409-422.

- [129] I.K. FODOR. *A Survey of Dimension Reduction Techniques*. fr. 2002.
- [130] T.C. FOGARTY. « Varying the probability of mutation in the genetic algorithm ». en. In : *Proceedings of the 3rd International Conference on Genetic Algorithms*. 1989, p. 104-109.
- [131] C.M. FONSECA et P.J. FLEMING. « Multiobjective genetic algorithms made easy : Selection, sharing and mating restriction ». en. In : *Proceedings of the 1st International Conference on Genetic Algorithms in Engineering Systems : Innovations and Applications*. 1995, p. 45-52.
- [132] Gerhard FRIEDRICH et Markus ZANKER. « A taxonomy for generating explanations in recommender systems ». In : *AI Magazine* 32.3 (2011), p. 90-98.
- [133] T. FRIEDRICH, N. HEBBINGHAUS et F. NEUMANN. « Rigorous analyses of simple diversity mechanisms ». en. In : *Proceedings of the Genetic and Evolutionary Computation Conference*. 2007, p. 1219-1225.
- [134] T. FRIEDRICH et al. « Theoretical analysis of diversity mechanisms for global exploration ». en. In : *Proceedings of the Genetic and Evolutionary Computation Conference*. 2008, p. 945-952.
- [135] W.-T. FU et W.D. GRAY. « Suboptimal tradeoffs in information seeking ». en. In : *Cognitive Psychology* 52 (2006), p. 195-242. DOI : [10.1016/j](https://doi.org/10.1016/j). URL : <http://dx.doi.org/10.1016/j>.
- [136] W.-T. FU et P. PIROLI. « SNIF-ACT : A cognitive model of user navigation on the World Wide Web ». en. In : *Human-Computer Interaction* 22 (2007), p. 355-412.
- [137] S. GARCIA, J. LUENGO et F. HERRERA. *Data Preprocessing in Data Mining*. da. Springer, 2015.
- [138] S. GARCIA, J. LUENGO et F. HERRERA. « Tutorial on practical tips of the most influential data preprocessing algorithms in data mining, Knowl ». en. In : *Based Syst* 98 (2016), p. 1-29.
- [139] M. GEN et R. CHENG. *Genetic Algorithms and Engineering Design*. en. Itoboken, NJ : John Wiley et Sons, 1997.
- [140] B. GENG et al. « Nnia-rs : A multi-objective optimization based recommender system ». en. In : *Physica A : Statistical Mechanics and its Applications* 424 (2015), p. 383-397.
- [141] A. GHOSH, S. TSUTSUI et H. TANAKA. « Individual aging in genetic algorithms ». en. In : *Proceedings of the Australian New Zealand Conference on Intelligent Information Systems*. 1996, p. 276-279.
- [142] G. GIGERENZER et W. GAISSMAIER. « Heuristic decision making ». en. In : *Annual Review of Psychology* 62 (2011), p. 451-482. DOI : [10.1146/annurevpsych](https://doi.org/10.1146/annurevpsych)-. URL : <http://dx.doi.org/10.1146/annurevpsych>-.
- [143] J.C. GITTINS. « Bandit processes and dynamic allocation indexes ». en. In : *Journal of the Royal Statistical Society Series B, Methodological* 41 (1979), p. 148-177.
- [144] Fred W GLOVER et Gary A KOCHENBERGER. *Handbook of metaheuristics*. T. 57. Dordrecht : Springer Science & Business Media, 2006.
- [145] K.S. GOH, A. LIM et B. RODRIGUES. « Sexual selection for genetic algorithms ». en. In : *Artif. Intell. Rev* 19 (2003), p. 123-152.

- [146] J.I. GOLD et M.N. SHADLEN. « The neural basis of decision making ». en. In : *Annual Review of Neuroscience* 30 (2007), p. 535-574. URL : <http://dx.doi.org/10.1146/>.
- [147] D.E. GOLDBERG et J. RICHARDSON. « Genetic algorithms with sharing for multimodal function optimization ». en. In : *Proceedings of the 2nd International Conference on Genetic Algorithms*. 1987, p. 41-49.
- [148] DE GOLDBERG. *Genetic Algorithms in Search, Optimization, and Machine Learning*. 1989.
- [149] R.L. GOLDSTONE, B.C. ASHPOLE et M.E. ROBERTS. « Knowledge of resources and competitors in human foraging ». en. In : *Psychonomic Bulletin & Review* 12 (2005), p. 81-87. URL : <http://dx.doi.org/10.3758/>.
- [150] W. GONG, Z. CAI et L. JIANG. « Enhancing the performance of differential evolution using orthogonal design method ». en. In : *Appl. Math. Comput* 206, 1 (2008), p. 56-69.
- [151] C. GONZALEZ et V. DUTT. « Instance-based learning : Integrating sampling and repeated decisions from experience ». en. In : *Psychological Review* 118 (2011), p. 523-551. DOI : [10.1037/a0024558](https://doi.org/10.1037/a0024558). URL : <http://dx.doi.org/10.1037/a0024558>.
- [152] C. GONZALEZ et V. DUTT. « Refuting data aggregation arguments and how the instance-based learning model stands criticism : A reply to Hills and Hertwig (2012) ». en. In : *Psychological Review* 119 (2012), p. 893-898. DOI : [10.1037/a0029445](https://doi.org/10.1037/a0029445). URL : <http://dx.doi.org/10.1037/a0029445>.
- [153] C. GONZALEZ, J.F. LERCH et C. LEBIERE. « Instance-based learning in dynamic decision making ». en. In : *Cognitive Science* 27 (2003), p. 591-635. URL : <http://dx.doi.org/10.1037/a0029445>.
- [154] L. GREEN, J. MYERSON et E. MCFADDEN. « Rate of temporal discounting decreases with amount of reward ». en. In : *Memory & Cognition* 25 (1997), p. 715-723. DOI : [10.3758/BF03211314](https://doi.org/10.3758/BF03211314). URL : [http://](http://dx.doi.org/10.3758/BF03211314).
- [155] G.W. GREENWOOD, G.B. FOGEL et M. CIOBANU. « Emphasizing extinction in evolutionary programming ». en. In : *Proceedings of the Congress of Evolutionary Computation*. 1999, p. 666-671.
- [156] J.J. GREFENSTETTE. « Genetic algorithms for changing environments ». en. In : *Proceedings of Parallel Problem Solving from Nature*. Elsevier, Amsterdam, 1992, p. 137-144.
- [157] J.J. GREFENSTETTE. « Optimization of control parameters for genetic algorithms ». en. In : *IEEE Trans. Syst., Man Cybernetics* 16, 1 (1986), p. 122-128.
- [158] S. GUHA et A. MCGREGOR. « Space-efficient sampling ». en. In : *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS 2007)*. San Juan, PUR, USA, 2007, p. 169-176.
- [159] A.K. GUPTA, K.G. SMITH et C.E. SHALLEY. « The interplay between exploration and exploitation ». en. In : *Academy of Management Journal* 49 (2006), p. 693-706. DOI : [10.5465/AMJ](https://doi.org/10.5465/AMJ). URL : <http://dx.doi.org/10.5465/AMJ>.
- [160] Shalini GUPTA et Veer Sain DIXIT. « A Meta-Heuristic Algorithm Approximating Optimized Recommendations for E-Commerce Business Promotions ». In : *IJITPM* 11.2 (2020), p. 23-49.
- [161] Zou H. « Flexanalytics : a flexible data analytics framework for big data applications with I/O performance improvement ». en. In : *Big Data Res* 1 (2014), p. 4-13.

- [162] M. HALL et al. « The weka data mining software : An update ». en. In : *ACM SIGKDD Explorations Newsletter* 11.1 (2009), p. 10-18.
- [163] Pierre HANSEN et Nenad MLADENović. « An introduction to variable neighborhood search ». In : *Meta-heuristics*. Springer, 1999, p. 433-458.
- [164] G.R. HARIK. « Finding multimodal solutions using restricted tournament selection ». en. In : *Proceedings of the 6th International Conference on Genetic Algorithms*. 1995, p. 24-31.
- [165] G.R. HARIK, F. LOBO et D.E. GOLDBERG. « The compact genetic algorithm ». en. In : *IEEE Trans. Evol. Comput* 3, 4 (1999), p. 287-297.
- [166] Georges R HARIK, Fernando G LOBO et al. « A parameter-less genetic algorithm. » In : *GECCO*. T. 99. 1999, p. 258-267.
- [167] F Maxwell HARPER et Joseph A KONSTAN. « The movielens datasets : History and context ». In : *ACM tiis* 5.4 (2015), p. 1-19.
- [168] J. HE, C. REEVES et C. WITT. « A note on problem difficulty measures in blackbox optimization : Classification, realizations and predictability ». en. In : *Evolutionary Computation* 15.4 (2007), p. 435-443.
- [169] Zi-Lin HE et Poh-Kam WONG. « Exploration vs. exploitation : An empirical test of the ambidexterity hypothesis ». In : *Organization science* 15.4 (2004), p. 481-494.
- [170] J.J. HECKMAN. « Sample selection bias as a specification error ». en. In : *ECONOMETRICA* 47 (1979), p. 153-161. DOI : [10.2307/1912352](https://doi.org/10.2307/1912352).. URL : <https://doi.org/10.2307/1912352>..
- [171] N. HENZE et al. « Adaptive hyperbooks for constructivist teaching ». de. In : *Kunstliche Intelligenz* 13.4 (1999).
- [172] F. HERRERA et M. LOZANO. « Adaptation of genetic algorithm parameters based on fuzzy logic controllers ». en. In : *Genetic Algorith. Soft Comput* (1996), p. 95-125.
- [173] F. HERRERA et M. LOZANO. « Adaptive genetic operators based on coevolution with fuzzy behaviors ». en. In : *IEEE Transactions on Evolutionary Computation* 5.2 (2001), p. 149-165.
- [174] R. HERTWIG et T.J. PLESKAC. « Decisions from experience : Why small samples? » fr. In : *Cognition* 115 (2010), p. 225-237. DOI : [10.1016/j.cognition](https://doi.org/10.1016/j.cognition). URL : <http://dx.doi.org/10.1016/j.cognition>.
- [175] R. HERTWIG et al. « Decisions from experience and the effect of rare events in risky choice ». en. In : *Psychological Science* 15 (2004), p. 534-539. DOI : [10.1111/j](https://doi.org/10.1111/j). URL : <http://dx.doi.org/10.1111/j>.
- [176] J. HESSER et R. MANNER. « Toward an optimal mutation probability for genetic algorithms ». en. In : *Proceedings of the 1st Conference of PPSN*. T. 496. Springer, 1991, p. 23-32.
- [177] P.S.M. HILL, J. HOLLIS et H. WELLS. « Foraging decisions in nectarivores : Unexpected interactions between flower constancy and energetic rewards ». en. In : *Animal Behaviour* 62 (2001), p. 729-737. DOI : [10.1006/anbe.2001.1775](https://doi.org/10.1006/anbe.2001.1775).
- [178] P.S.M. HILL, P.H. WELLS et H. WELLS. « Spontaneous flower constancy and learning in honey bees as a function of colour ». en. In : *Animal Behaviour* 54 (1997), p. 615-627. DOI : [10.1006/anbe](https://doi.org/10.1006/anbe). URL : <http://dx.doi.org/10.1006/anbe>.

- [179] T.T. HILLS. « Animal foraging and the evolution of goal-directed cognition ». en. In : *Cognitive Science* 30 (2006), p. 3-41. URL : <http://dx.doi.org/10.1207/>.
- [180] T.T. HILLS et R. HERTWIG. « Information search in decisions from experience. Do our patterns of sampling foreshadow our decisions ? » en. In : *Psychological Science* 21 (2010), p. 1787-1792. URL : <http://dx.doi.org/>.
- [181] T.T. HILLS et R. HERTWIG. « Two distinct exploratory behaviors in decisions from experience : Comment on Gonzalez and Dutt (2011) ». en. In : *Psychological Review* 119 (2012), p. 888-892. DOI : [10.1037/a0028004](https://doi.org/10.1037/a0028004). URL : <http://dx.doi.org/>.
- [182] T.T. HILLS, M.N. JONES et P.M. TODD. « Optimal foraging in semantic memory ». en. In : *Psychological Review* 119 (2012), p. 431-440. DOI : [10.1037/a0027373](https://doi.org/10.1037/a0027373). URL : <http://dx.doi.org/>.
- [183] T.T. HILLS, T. NOGUCHI et M. GIBBERT. « Information overload or search-amplified risk? Set size and order effects on decisions from experience ». en. In : *Psychonomic Bulletin & Review* 20 (2013), p. 1023-1031. DOI : [10.3758/s13423-013-](https://doi.org/10.3758/s13423-013-013-1). URL : [http://dx.doi.org/10.3758/s13423-013-](http://dx.doi.org/10.3758/s13423-013-1).
- [184] T.T. HILLS et T. PACHUR. « Dynamic search and working memory in social recall ». en. In : *Journal of Experimental Psychology : Learning, Memory, and Cognition* 38 (2012), p. 218-228. URL : <http://dx.doi.org/10.1037/a0027373>.
- [185] T.T. HILLS, P.M. TODD et R.L. GOLDSTONE. « The central executive as a search process : Priming exploration and exploitation across domains ». en. In : *Journal of Experimental Psychology : General* 139 (2010), p. 590-609. DOI : [10.1037/a0020666](https://doi.org/10.1037/a0020666). URL : <http://dx.doi.org/10.1037/a0020666>.
- [186] Thomas T HILLS et al. « Exploration versus exploitation in space, mind, and society ». In : *Trends in cognitive sciences* 19.1 (2015), p. 46-54.
- [187] Yu-Chi HO et David L PEPYNE. « Simple explanation of the no-free-lunch theorem and its implications ». In : *Journal of optimization theory and applications* 115.3 (2002), p. 549-570.
- [188] J.H. HOLLAND. *Adaptation in Natural and Artificial Systems*. en. Ann Arbor, MI : University of Michigan Press, 1975.
- [189] Y. HUANG et al. « A framework for multifaceted evaluation of student models ». en. In : *Proc. the 8th Int. Conf. on Educational Data Mining*. 2015, p. 203-210.
- [190] Yun HUANG et al. « A Framework for Dynamic Knowledge Modeling in Textbook-Based Learning ». en. In : *24th Conference on User Modeling Adaptation and Personalization. ACM*. 2016, p. 141-150.
- [191] J.M.C. HUTCHINSON, A. WILKE et P.M. TODD. « Patch leaving in humans : Can a generalist adapt its rules to dispersal of items across patches ? » en. In : *Animal Behaviour* 75 (2008), p. 1331-1349. URL : <http://dx.doi.org/>.
- [192] M. HUTTER et S. LEGG. « Fitness uniform optimization ». fr. In : *IEEE Trans. Evol. Comput* 10, 5 (2006), p. 568-589.
- [193] H. ISHIBUCHI, N. TSUKAMOTO et Y. NOJIMA. « Diversity improvement by non-geometric binary crossover in evolutionary multiobjective optimization ». en. In : *IEEE Trans. Evol. Comput* 14, 6 (2010), p. 985-998.

- [194] Folasade Olubusola ISINKAYE, Yetunde O FOLAJIMI et Bolande Adefowoke OJOKOH. « Recommendation systems : Principles, methods and evaluation ». In : *Egyptian informatics journal* 16.3 (2015), p. 261-273.
- [195] Pari JAHANKHANI, Kenneth REVETT et Vassilis KODOGIANNIS. *Data Mining an EEG Dataset With an Emphasis on Dimensionality Reduction*. en. 2007.
- [196] B. JALALI et M.H. ASGHARI. « The anamorphic stretch transform : putting the squeeze on big data ». en. In : *Opt Photonics News* 25.2 (2014), p. 24-31.
- [197] T. JANSEN. « On classifications of fitness functions ». en. In : *Theoretical Aspects of Evolutionary Computing*. Sous la dir. de L. KALLEL, B. NAUDTS et A. ROGERS. Berlin : Springer-Verlag, 2001, p. 371-385.
- [198] D. JIA, G. ZHENG et M.K. KHAN. « An effective memetic differential evolution algorithm based on chaotic local search ». it. In : *Inform. Sci* 181, 15 (2011), p. 3175-3187.
- [199] X. JIN et R. REYNOLDS. « Using knowledge-based evolutionary computation to solve nonlinear constraint optimization problems : A cultural algorithm approach ». en. In : *Proceedings of the Congress on Evolutionary Computation*. 1999, p. 1672-1678.
- [200] R. JOAN-ARINYO, M.V. LUZON et E. YEGUAS. « Parameter tuning of pbil and chc evolutionary algorithms applied to solve the root identification problem ». en. In : *Appl. Soft Comput* 11, 1 (2011), p. 754-767.
- [201] E.J. JOHNSON et D. GOLDSTEIN. « Medicine. Do defaults save lives? » fr. In : *Science* 302 (2003), p. 1338-1339. DOI : [10.1126/science.1091721](https://doi.org/10.1126/science.1091721). URL : <http://dx.doi.org/10.1126/science.1091721>.
- [202] L.M.S. JOSE-REVUELTA. « A new adaptive genetic algorithm for fixed channel assignment ». it. In : *Inform. Sci* 177 (2007), p. 2655-2678.
- [203] Alexander JUNG et Pedro H.J. NARDELLI. *An Information-Theoretic Approach to Personalized Explainable Machine Learning*. en. 2020.
- [204] Sung-Moo JUNG et Kwang-Bin LIM. « Leading future education : Development of digital textbooks in Korea ». In : *proc. 12th UNESCO-APEID international conference quality innovations for teaching and learning*. 2009.
- [205] J.W. KABLE et P.W. GLIMCHER. « The neural correlates of subjective value during intertemporal choice ». en. In : *Nature Neuroscience* 10 (2007), p. 1625-1633. DOI : [10.1038/nn2007](https://doi.org/10.1038/nn2007). URL : <http://dx.doi.org/10.1038/nn2007>.
- [206] J.H. KAGEL, L. GREEN et T. CARACO. « When foragers discount the future : Constraint or adaptation ». en. In : *Animal Behaviour* 34 (1986), p. 271-283. URL : [http://dx.doi.org/10.1016/0003-0439\(86\)90001-0](http://dx.doi.org/10.1016/0003-0439(86)90001-0).
- [207] Marius KAMINSKAS et Derek BRIDGE. « Diversity, serendipity, novelty, and coverage : A survey and empirical analysis of beyond-accuracy objectives in recommender systems ». en. In : *ACM TiiS* 7, 1 (2017), p. 2.
- [208] Rahul KATARYA. « Movie recommender system with metaheuristic artificial bee ». In : *Neural Computing and Applications* 30.6 (2018), p. 1983-1990.
- [209] S. KATE et L. LOPES. « Measuring instance difficulty for combinatorial optimization problems ». en. In : *Computers & Operations Research* 39 (2012), p. 875-889.
- [210] M. AL-KATEB et B.S. LEE. « Stratified reservoir sampling over heterogeneous data streams ». en. In : *Proceedings of the 22nd International Conference on Scientific and Statistical Database Management (SSDBM'10)*. Heidelberg, Germany, 2010, p. 621-639.

- [211] A. KAVCIC. « Fuzzy User Modeling for Adaptation in Educational Hypermedia ». en. In : *IEEE Transactions on Systems, Man and Cybernetics* 34.4 (2004).
- [212] Peter Brusilovsky KHUSHBOO THAKER et Daqing HE. *Student Modeling with Automatic Knowledge Component Extraction for Adaptive Textbooks*. en. USA : University of Pittsburgh.
- [213] Yun Huang KHUSHBOO THAKER. *Dynamic Knowledge Modeling with Heterogeneous Activities for Adaptive Textbooks*. en. USA : Peter Brusilovsky, Daqing He, School of Computing et Information, University of Pittsburgh.
- [214] W. KINTSCH et D.M. WELSCH. *The construction-integration model : A framework for studying memory for text*. en. 1991.
- [215] A. KIRSCH et al. « A rigorous statistical approach for identifying significant itemsets ». en. In : *Proceedings of the IEEE International Conference on Data Mining (ICDM'08)*. Pisa, Italy, 2008, p. 1-10.
- [216] T. KOHONEN. *Self-Organizing Maps*. nl. 3rd. New York, NY : Springer-Verlag, 2001.
- [217] Natallia KOKASH. « An introduction to heuristic algorithms ». In : *Department of Informatics and Telecommunications* (2005), p. 1-8.
- [218] Pigi KOUKI et al. « User preferences for hybrid explanations ». In : *Proceedings of the Eleventh ACM Conference on Recommender Systems*. 2017, p. 84-88.
- [219] V. KOUMOUSIS et C.P. KATSARAS. « A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance ». en. In : *IEEE Trans. Evol. Comput* 10, 1 (2006), p. 19-28.
- [220] J.R. KOZA. *Genetic Programming : On the Programming of Computers by Means of Natural Selection*. en. Cambridge, MA : MIT Press, 1992.
- [221] D.L. KRAMER et D.M. WEARY. « Exploration versus exploitation : A field study of time allocation to environmental tracking by foraging chipmunks ». fr. In : *Animal Behaviour* 41 (1991), p. 443-449. DOI : [10.1016/S0003-3472\(05\)80846-2](https://doi.org/10.1016/S0003-3472(05)80846-2). URL : <http://dx>.
- [222] N. KRASNOGOR et J. SMITH. « A tutorial for competent memetic algorithms : Model, taxonomy, and design issues ». it. In : *IEEE Trans. Evol. Comput* 9, 5 (2005), p. 474-488.
- [223] J.R. KREBS, A. KACELNIK et P. TAYLOR. « Test of optimal sampling by foraging great tits ». en. In : *Nature* 275 (1978), p. 27-31. DOI : [10.1038/275027a0](https://doi.org/10.1038/275027a0). URL : <http://dx.doi.org/10.1038/275027a0>.
- [224] T. KRINK, P. RICKERS et R. THOMSEN. « Applying self-organised criticality to evolutionary algorithms ». en. In : *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature*. New York, NY : Springer-Verlag, 2000, p. 375-384.
- [225] Igor LABUTOV. « Semi-Supervised Techniques for Mining Learning Outcomes and Prerequisites ». en. In : Carnegie Mellon University.
- [226] W.B. LANGDON. *Data Structures and Genetic Programming*. nl. Alphen aan den Rijn, Netherlands : Kluwer, 1998.
- [227] J.-Y. LEE, M.-S. KIM et J.-J. LEE. « Compact genetic algorithm using belief vectors ». en. In : *Appl. Soft Comput* 11, 4 (2011), p. 3385-3401.
- [228] M.D. LEE et E.Y. CORLETT. « Sequential sampling models of human text classification ». en. In : *Cognitive Science* 27 (2003), p. 159-193. DOI : [10.1207/s15516709cog2702_2](https://doi.org/10.1207/s15516709cog2702_2). URL : <http://dx.doi.org/>.

- [229] M.D. LEE et al. « Psychological models of human and optimal performance in bandit problems ». en. In : *Cognitive Systems Research* 12 (2011), p. 164-174. DOI : [10.1016/j.cogsys.2010.07.007](https://doi.org/10.1016/j.cogsys.2010.07.007). URL : <http://dx.doi.org/>.
- [230] Y. LEUNG et Y. WANG. « An orthogonal genetic algorithm with quantization for global numerical optimization ». en. In : *IEEE Trans. Evol. Comput* 5, 1 (2001), p. 41-53.
- [231] B.B. LI, L. WANG et B. LIU. « An effective PSO-based hybrid algorithm for multiobjective permutation flow shop scheduling ». en. In : *IEEE Trans. Syst., Man, Cybern. Part A, Syst., Humans* 38.4 (2008), p. 818-831.
- [232] J.-P. LI et al. « A species conserving genetic algorithm for multimodal function optimization ». ro. In : *Evol. Comput* 10 3 (2002), p. 207-234.
- [233] M. LI, Z. CAI et G. SUN. « An adaptive genetic algorithm with diversity-guided mutation and its global convergence property ». en. In : *J. Central South Univ. Technol* 11, 3 (2004), p. 323-327.
- [234] Z. LI et X. WANG. « Chaotic differential evolution algorithm for solving constrained optimization problems ». pt. In : *Inform. Technol. J* 10, 12 (2011), p. 2378-2384.
- [235] Zhe LI. *Towards the Next Generation of Multi-Criteria Recommender Systems*. en. 2018.
- [236] Y. LIANG et K.-S. LEUNG. « Genetic algorithm with adaptive elitist-population strategies for multimodal function optimization ». en. In : *Appl. Soft Comput* 11, 2 (2011), p. 2017-2034.
- [237] T.W. LIAO. « Two hybrid differential evolution algorithms for engineering design optimization ». en. In : *Appl. Soft Comput* 10, 4 (2010), p. 1188-1199.
- [238] Marcelo A.A. LIMA et al. *A Method of Dynamic Resampling for DFT-Based Harmonic Analysis under Time-Varying Frequency Conditions*. en. 2010.
- [239] J. LIU, H.A. ABBASS et D.G. GREEN. « Motif Difficulty (MD) : A Predictive Measure of Problem Difficulty for Evolutionary Algorithms Using Network Motifs ». en. In : *Evolutionary Computation* 20.3 (2012), p. 321-347.
- [240] S.-H. LIU, M. MERNIK et B.R. BRYANT. « A clustering entropy-driven approach for exploring and exploiting noisy functions ». en. In : *Proceedings of the 22nd ACM Symposium on Applied Computing*. 2007, p. 738-742.
- [241] S.-H. LIU, M. MERNIK et B.R. BRYANT. « To explore or to exploit : An entropy-driven approach for evolutionary algorithms ». en. In : *Int. J. Knowl. Intell. Eng. Syst* 13, 3 (2009), p. 185-206.
- [242] X.-Y. LIU, J. WU et Z.-H. ZHOU. « Exploratory undersampling for class-imbalance learning ». en. In : *IEEE Trans Syst Man Cybern B Cybern* 39.2 (2009), p. 539-550.
- [243] Yang LIU, Yan LIU et Keith C.C. CHAN. *Dimensionality reduction for heterogeneous dataset in rushes editing*. en. 2009.
- [244] F.J. LOBO, C.F. LIMA et Z. MICHALEWICZ. *Parameter Setting in Evolutionary Algorithms*. sv. Berlin : Springer, 2007.
- [245] M. LOCATELLI et G. WOOD. « Objective function features providing barriers to rapid global optimization ». en. In : *Journal of Global Optimization* 31.4 (2005), p. 549-565.
- [246] R. LONGADGE et S. DONGRE. *Class imbalance problem in data mining review*. es. arXiv preprint arXiv :1305.1707. 2013.

- [247] M. LOZANO, F. HERRERA et J.R. CANO. « Replacement strategies to preserve useful diversity in steady-state genetic algorithms ». it. In : *Inform. Sci* 178, 23 (2008), p. 4421-4433.
- [248] Manuel LOZANO et Carlos GARCIA-MARTINEZ. « Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification : Overview and progress report ». In : *CBOR* 37.3 (2010), p. 481-497.
- [249] M.F. LUCE. « Choosing to avoid : Coping with negatively emotion-laden consumer decisions ». en. In : *Journal of Consumer Research* 24 (1998), p. 409-433. DOI : [10.1086/209518](https://doi.org/10.1086/209518). URL : <http://dx.doi.org/10.1086/209518>.
- [250] M.H. LUERSSSEN. « Phenotype diversity objectives for graph grammar evolution ». en. In : *Recent Advances in Artificial Life*. Singapore : World Scientific Publishing, 2005, p. 159-170.
- [251] Pin Lv et al. « Utilizing knowledge graph and student testing behavior data for personalized exercise recommendation ». In : *Proceedings of ACM Turing Celebration Conference-China*. 2018, p. 53-59.
- [252] Lin M-S. « Malicious URL filtering - a big data application ». en. In : *2013 IEEE international conference on big data*. 2013.
- [253] S.W. MAHFOUD. *Niching methods for genetic algorithms*. en. Tech. rep., Urbana Champaign, IL : University of Illinois at, 1995.
- [254] Mohammad Sultan MAHMUD et al. *A Survey of Data Partitioning and Sampling Methods to Support Big Data Analysis*. en. 2020.
- [255] M. MAJIG et M. FUKUSHIMA. « Adaptive fitness function for evolutionary algorithm and its applications ». en. In : *Proceedings of the International Conference on Informatics Education and Research for Knowledge-Circulating Society*. 2008, p. 119-124.
- [256] V. MALBASA et S. VUCETIC. « A reservoir sampling algorithm with adaptive estimation of conditional expectation ». en. In : *Proc. of International Joint Conference on Neural Networks*. Orlando, FL, USA, 2007, p. 2200-2204.
- [257] R. MALLIPEDDI et al. « Differential evolution algorithm with ensemble of parameters and mutation strategies ». en. In : *Appl. Soft Comput* 11, 2 (2011), p. 1679-1696.
- [258] G.S. MANKU, S. RAJAGOPALAN et B.G. LINDSAY. « Random sampling techniques for space efficient online computation of order statistics of large datasets ». en. In : *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data (SIGMOD'99)*. Philadelphia, PA, USA, 1999, p. 251-262.
- [259] J.G. MARCH. « Exploration and exploitation in organizational learning ». fr. In : *Organization Science* 2 (1991), p. 71-87. DOI : [10.1287/orsc.2.1.71](https://doi.org/10.1287/orsc.2.1.71). URL : <http://dx.doi.org/10.1287/orsc.2.1.71>.
- [260] Silvano MARTELLO et Paolo TOTH. *Knapsack problems : algorithms and computer implementations*. Hoboken, New Jersey, US : John Wiley & Sons, Inc., 1990.
- [261] W.N. MARTIN, J. LIENIG et J.P. COHOON. « Island (Migration) Models ». en. In : *Evolutionary Algorithms based on Punctuated Equilibria (In Handbook of Evolutionary Computation)*. Oxford University Press, 1999.
- [262] Ignas MARTISIUS et al. *EEG Dataset Reduction and Classification Using Wave Atom Transform*. en. 2013.

- [263] Shinichiro MARUYAMA, Robert J.M. EVELEIGH et John M. ARCHIBALD. *Treertrimmer : a method for phylogenetic dataset size reduction*. en. 2013.
- [264] M.H. MASHINCHI, M.A. ORGUN et W. PEDRYCZ. « Hybrid optimization with improved tabu search ». en. In : *Appl. Soft Comput* 11, 2 (2011), p. 1993-2006.
- [265] L. MASISI, V. NELWAMONDO et T. MARWALA. « The use of entropy to measure structural diversity ». en. In : *Proc. of IEEE on Comput. Cyber.* 2008, p. 41-45.
- [266] K. MATSUI. « New selection method to improve the population diversity in genetic algorithms ». en. In : *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*. 1999, p. 625-630.
- [267] M.L. MAULDIN. « Maintaining diversity in genetic search ». en. In : *Proceedings of the National Conference on Artificial Intelligence*. 1984, p. 247-250.
- [268] S.J. MAYOR et al. « Habitat selection at multiple scales ». fr. In : *Ecoscience* 16 (2009), p. 238-247. URL : <http://dx.doi>.
- [269] Catherine C MCGEOCH. « Experimental analysis of algorithms ». In : *Notices of the AMS* 48.3 (2001), p. 304-311.
- [270] B. MCGINLEY et al. « Maintaining healthy population diversity using adaptive crossover, mutation, and selection ». en. In : *IEEE Trans. Evol. Computat* 15, 5 (2011), p. 692-714.
- [271] Amy MCGOVERN et al. « Making the black box more transparent : Understanding the physical implications of machine learning ». In : *Bulletin of the American Meteorological Society* 100.11 (2019), p. 2175-2199.
- [272] James MCINERNEY et al. « Explore, exploit, and explain : personalizing explainable recommendations with bandits ». In : *Proceedings of the 12th ACM conference on recommender systems*. 2018, p. 31-39.
- [273] N.F. MCPHEE et N.J. HOPPER. « Analysis of genetic diversity through population history ». en. In : *Proc. of the 1st GEC Conference*. 1999.
- [274] K. MEHLHORN et al. « Observed variability and values matter : Toward a better understanding of information search and decisions from experience ». en. In : *Journal of Behavioral Decision Making* 27 (2014), p. 328-339. DOI : [10.1002/bdm.1809](https://doi.org/10.1002/bdm.1809).
- [275] K. MEHLHORN et al. « Unpacking the Exploration-Exploitation Tradeoff : A Synthesis of Human and Animal Literatures ». en. In : *Decision* 2.3 (2015), p. 191-215. DOI : [10.1037/dec0000033](https://doi.org/10.1037/dec0000033).
- [276] Jaroslav MELESKO et Eugenijus KURILOVAS. « Semantic Technologies in e-Learning ». en. In : *Learning Analytics and Artificial Neural Networks in Personalised Learning Systems*. 2018.
- [277] O.J. MENGSHOEL et D.E. GOLDBERG. « Probabilistic crowding : Deterministic crowding with probabilistic replacement ». en. In : *Proceedings of the Genetic and Evolutionary Computation Conference*. 1999, p. 409-416.
- [278] *MENJ Official Website*. URL : <https://www.education.gouv.fr>.
- [279] P. MERZ et B. FREISLEBEN. « Fitness landscape analysis and memetic algorithms for the quadratic assignment problem ». en. In : *IEEE Trans. Evol. Comput* 4, 4 (2000), p. 337-352.

- [280] Yuan MIAO. *An Intelligent Tutoring System Using Interest Based Negotiation*. da. 2008.
- [281] Zbigniew MICHAŁEWICZ et Zbigniew MICHAŁEWICZ. *Genetic algorithms+ data structures= evolution programs*. Springer Science & Business Media, 1996.
- [282] A. MISEVICIUS. « Generation of grey patterns using an improved genetic-evolutionary algorithm : Some new results ». en. In : *Inform. Technol. Control* 40, 4 (2011), p. 330-343.
- [283] A. MISEVICIUS et D. RUBLIAUSKAS. « Enhanced improvement of individuals in genetic algorithms ». pt. In : *Inform. Technol. Control* 37, 3 (2008), p. 179-186.
- [284] E. MONTERO et M.-C. RIFF. « On-the-fly strategies for evolutionary algorithms ». en. In : *Inform. Sci* 181 (2011), p. 552-566.
- [285] Ajinkya MORE. *Survey of resampling techniques for improving classification performance in unbalanced datasets*. fr. 2016.
- [286] N. MORI et al. « A thermodynamical selection rule for the genetic algorithm ». en. In : *Proceedings of the 2nd IEEE International Conference on Evolutionary Computation*. 1995, p. 188-192.
- [287] P. MOSCATO. « Memetic algorithms : A short introduction ». en. In : *New Ideas in Optimization, McGraw Hill Ltd*. Maidenhead, U.K, 1999, p. 219-234.
- [288] A. MOZDGIR, I. MAHDAVI et I. BADELEH. « Using the Taguchi method to optimize the differential evolution algorithm parameters for minimizing the workload smoothness index in simple assembly line balancing ». pt. In : *Mathematical and Computer Modelling* 57 (2013), p. 137-151.
- [289] H. MUHLENBEIN et G. PAASS. « From recombination of genes to the estimation of distributions I. Binary parameters ». en. In : *Proceedings of Parallel Problem Solving from Nature, Lecture Notes in Computer Science*. T. 1141. Springer, 1996, p. 178-187.
- [290] V. NANNEN et A.E. EIBEN. « A method for parameter calibration and relevance estimation in evolutionary algorithms ». en. In : *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*. 2006, p. 183-190.
- [291] B. NAUDTS et L. KALLEL. « A comparison of predictive measures of problem difficulty in evolutionary algorithms ». en. In : *IEEE Transactions on Evolutionary Computation* 4.1 (2000), p. 1, 15.
- [292] Amos H.C. NG, Florian SIEGMUND et Kalyanmoy DEB. *Reference point based evolutionary multi-objective optimization with dynamic resampling for production systems improvement*. en. 2018.
- [293] P. NONACS. « Patch exploitation ». en. In : *Encyclopedia of animal behavior*. Sous la dir. de M.D. BREED et J. MOORE. Oxford, UK : Academic Press, 2010, p. 683-690. DOI : [10.1016/B978-0-08-045337-8](https://doi.org/10.1016/B978-0-08-045337-8). URL : <http://dx.doi.org/10.1016/B978-0-08-045337-8>.
- [294] Jean-Pierre NZIGA. *Minimal Dataset for Network Intrusion Detection Systems via Dimensionality Reduction*. en. 2011.
- [295] G. OCHOA, B. BILGIN et E.E. KORKMAZ. « A comprehensive analysis of hyper-heuristics ». en. In : *Intell. Data Anal* 12, 1 (2008), p. 3-23.
- [296] G. OCHOA, M. TOMASSINI et S. VEREL. « A study of NK landscapes basins and local optima networks ». en. In : *Proceedings of the 2008 Genetic and Evolutionary Computation Conference*. 2008, p. 555-562.

- [297] T.E. OLIPHANT. « Scipy : Open source scientific tools for python ». en. In : *Computing in Science and Engineering* 9.3 (2007), p. 10-20.
- [298] F. OLKEN. « Random sampling from databases ». en. PhD dissertation, Berkeley, CA, USA : University of California at Berkeley, 1993.
- [299] F. OLKEN et D. ROTEM. « Random sampling from database files : A survey ». en. In : *Proceedings of the 5th International Conference on Statistical and Scientific Database Management (SSDBM'1990)*. Charlotte, NC, USA, 1990, p. 92-111.
- [300] Y.-S. ONG et al. « Classification of adaptive memetic algorithms : A comparative study ». en. In : *IEEE Trans. Syst. Man Cybernetics Part B* (2006), p. 141-152.
- [301] F. OPPACHER et M. WINEBERG. « The shifting balance GA : Improving the GA in dynamic environment ». en. In : *Proceedings of the 1st Genetic and Evolutionary Computation Conference*. 1999, p. 504-510.
- [302] Ibrahim H OSMAN et Gilbert LAPORTE. *Metaheuristics : A bibliography*. 1996.
- [303] Stefanos OUGIAROGLOU et Georgios EVANGELIDIS. *Efficient Dataset Size Reduction by Finding Homogeneous Clusters*. en. 2012.
- [304] I. PAENKE, Y. JIN et J. BRANKE. « Balancing population- and individual-level adaptation in changing environments ». en. In : *Adapt. Behav* 17, 2 (2009), p. 153-174.
- [305] Q.-K. PAN et al. « A differential evolution algorithm with self-adapting strategy and control parameters ». pt. In : *Comput. Oper. Res* 38, 1 (2011), p. 394-408.
- [306] Shailesh Singh PANWAR et YP RAIWANI. « Data reduction techniques to analyze NSL-KDD Dataset ». In : *Int. J. Comput. Eng. Technol* 5.10 (2014), p. 21-31.
- [307] Apostolos PAPAGEORGIOU, Bin CHENG et Erno KOVACS. « Real-time data reduction at the network edge of Internet-of-Things systems ». In : *2015 11th international conference on network and service management (CNSM)*. IEEE. 2015, p. 284-291.
- [308] Hina PARVEEN, Mohd ASHRAF et Rubina PARVEEN. « Improving the performance of multi-criteria recommendation system using fuzzy integrated meta heuristic ». In : *Int. Conf. on Comp., Com. & Aut.* IEEE. 2015, p. 304-308.
- [309] Samuele PASSALACQUA et al. *Visualisation Analysis for Exploring Prerequisite Relations in Textbooks*. et.
- [310] P. PAVLIK, H. CEN et K. KOEDINGER. « Performance Factors Analysis - A New Alternative to Knowledge Tracing ». en. In : *Proc. the 2009 Conf. on Artificial Intelligence in Education : Building Learning Systems that Care : From Knowledge Representation to Affective Modelling*. IOS Press, 2009, p. 531-538.
- [311] J.W. PAYNE, J.R. BETTMAN et E.J. JOHNSON. « Behavioral decision research : A constructive processing perspective ». en. In : *Annual Review of Psychology* 43 (1992), p. 87-131. DOI : [10.1146/annurev.ps](https://doi.org/10.1146/annurev.ps). URL : <http://dx.doi.org/10.1146/annurev.ps>.
- [312] Arthur PCHELKIN. « Efficient exploration in reinforcement learning based on utile suffix memory ». In : *Informatica* 14.2 (2003), p. 237-250.
- [313] F. PEDREGOSA et al. « Scikit-learn : Machine learning in python ». en. In : *The Journal of Machine Learning Research* 12.10 (2011), p. 2825-2830.

- [314] C.J. PERRY et A.B. BARRON. « Honey bees selectively avoid difficult choices ». en. In : *PNAS Proceedings of the National Academy of Sciences of the United States of America*. T. 110. 2013, p. 19155-19159. DOI : [10.1073/pnas.1314571110](https://doi.org/10.1073/pnas.1314571110). URL : <http://dx.doi.org/10.1073/pnas.1314571110>.
- [315] A. PETROWSKI. « A clearing procedure as a niching method for genetic algorithms ». en. In : *Proceedings of the IEEE International Conference on Evolutionary Computation*. 1996, p. 798-803.
- [316] B. PFAHRINGER, H. BENSUSAN et C. GIRAUD-CARRIER. « Metalearning by landmarking various learning algorithms ». en. In : *Proceedings of the 17th international conference on machine learning*. SanFrancisco, 2000, p. 743-750.
- [317] P. PIROLI et S.K. CARD. « Information foraging ». en. In : *Psychological Review* 106 (1999), p. 643-675. DOI : [10.1037/0033-295X.106.4.643](https://doi.org/10.1037/0033-295X.106.4.643). URL : <http://>.
- [318] J.M. PLEASANTS. « Optimal foraging by nectarivores : A test of the marginal-value theorem ». en. In : *American Naturalist* 134 (1989), p. 51-71. URL : <http://dx.doi>.
- [319] S.C. PRATT et D.J.T. SUMPTER. « A tunable algorithm for collective decision-making ». en. In : *PNAS Proceedings of the National Academy of Sciences of the United States of America*. T. 103. 2006, p. 15906-15910. DOI : [10.1073/pnas](https://doi.org/10.1073/pnas). URL : <http://dx.doi.org/10.1073/pnas>.
- [320] A. PRUDIUS et S. ANDRADTTIR. « Simulation optimization using balanced explorative and exploitative search ». en. In : *Proc. Winter Simul. Conf.* 2004, p. 545-549.
- [321] A.K. QIN, V.L. HUANG et P.N. SUGANTHAN. « Differential evolution algorithm with strategy adaptation for global numerical optimization ». it. In : *IEEE Trans. Evol. Comput* 13, 2 (2009), p. 398-417.
- [322] S. RAHNAMAYAN, H.R. TIZHOOSH et M.M.A. SALAMA. « Opposition-based differential evolution ». fr. In : *IEEE Trans. Evol. Comput* 12, 1 (2008), p. 64-79.
- [323] T. RAKOW, K.A. DEMES et B.R. NEWELL. « Biased samples not mode of presentation : Reexamining the apparent underweighting of rare events in experience-based choice ». en. In : *Organizational Behavior and Human Decision Processes* 106 (2008), p. 168-179. DOI : [10.1016/j.obhdp.2008](https://doi.org/10.1016/j.obhdp.2008). URL : <http://dx.doi.org/10.1016/j.obhdp.2008>.
- [324] T. RAKOW et B.R. NEWELL. « Degrees of uncertainty : An overview and framework for future research on experience-based choice ». en. In : *Journal of Behavioral Decision Making* 23 (2010), p. 1-14. DOI : [10.1002/bdm.681](https://doi.org/10.1002/bdm.681). URL : <http://>.
- [325] C.L. RAMSEY et J.J. GREFENSTETTE. « Case-based initialization of genetic algorithms ». en. In : *Proceedings of the 5th International Conference on Genetic Algorithms*. 1993, p. 84-91.
- [326] R Venkata RAO, Vimal J SAVSANI et DP VAKHARIA. « Teaching-learning-based optimization : a novel method for constrained mechanical design optimization problems ». In : *Computer-Aided Design* 43.3 (2011), p. 303-315.
- [327] R. RATCLIFF et L.A. REAL. « A theory of memory retrieval ». en. In : *Psychological Review* 85 (1978), p. 59-108. DOI : [10.1086/285399](https://doi.org/10.1086/285399). URL : <http://dx.doi>.
- [328] Siddharth REDDY. *Learning Student and Content Embeddings for Personalized Lesson Sequence Recommendation*. en. Thorsten Joachims. Ithaca, USA.
- [329] Muhammad Habib REHMAN et al. *Big Data Reduction Methods : A Survey*. en. 2016.

- [330] M.T. RIBEIRO et al. « Pareto-efficient hybridization for multi-objective recommender systems ». en. In : *Proceedings of the 6th ACM conference on Recommender systems*. 2012, p. 19-26.
- [331] I. RITOV et J. BARON. « Status-quo and omission biases ». en. In : *Journal of Risk and Uncertainty* 5 (1992), p. 49-61. DOI : [10.1007/BF00208786](https://doi.org/10.1007/BF00208786). URL : <http://dx.doi.org/10.1007/BF00208786>.
- [332] J.A.R. ROJAS et al. « Sampling techniques to improve big data exploration ». fr. In : *Proc. of 2017 IEEE 7th Symposium on Large Data Analysis and Visualization (LDAV)*. Phoenix, AZ, USA, 2017, p. 26-35.
- [333] C. ROMERO et S. VENTURA. *Educational data mining A survey from 1995 to 2005*. en. Spain : Department of Computer Sciences, University of Cordoba, 2005.
- [334] E. RONALD. « When selection meets seduction ». en. In : *Proceedings of the 6th International Conference on Genetic Algorithms*. 1995, p. 167-173.
- [335] J. ROSCA. « Entropy-driven adaptive representation ». en. In : *Proc. Workshop on Gen. Prog.* 1995, p. 23-32.
- [336] H. ROSE, W. EBELING et T. ASSELMAYER. « The density of states - a measure of the difficulty of optimisation problems ». en. In : *Parallel problem solving from nature-PPSN IV : international conference on evolutionary computation*. Berlin : Springer Verlag, 1996, p. 22-26.
- [337] Alan SAID. *Evaluating the Accuracy and Utility of Recommender Systems*. en. 2013.
- [338] W. SAMUELSON et R. ZECKHAUSER. « Status quo bias in decision making ». en. In : *Journal of Risk and Uncertainty* 1 (1988), p. 7-59. URL : <http://dx.doi.org/10.1007/>.
- [339] Ke SANG, Peter TODD et Robert GOLDSTONE. « Learning near-optimal search in a minimal explore/exploit task ». In : *Proceedings of the Annual Meeting of the Cognitive Science Society*. T. 33. 33. 2011.
- [340] Bruno SARENI et Laurent KRAHENBUHL. « Fitness sharing and niching methods revisited ». In : *IEEE transactions on Evolutionary Computation* 2.3 (1998), p. 97-106.
- [341] M.J. SASENA, P. PAPALAMBROS et P. GOOVAERTS. « Exploration of metamodeling sampling criteria for constrained global optimization ». en. In : *Optim* 34.3 (2002), p. 263-278.
- [342] Harshit SAXENA et Vineet RICHAARIYA. *Intrusion Detection in KDD99 Dataset using SVM-PSO and Feature Reduction with Information Gain*. en. 2014.
- [343] J.D. SCHAFFER et al. « A study of control parameters affecting online performance of genetic algorithms for function optimization ». en. In : *Proceedings of the 3rd International Conference on Genetic Algorithms*. 1989, p. 51-60.
- [344] B. SCHWARTZ. *The paradox of choice : Why more is less*. en. New York, NY : HarperCollins, 2004.
- [345] D.A. SEALE et A. RAPOPORT. « Optimal stopping behavior with relative ranks : The secretary problem with unknown population size ». en. In : *Journal of Behavioral Decision Making* 13.1002/1099-0771(200010/12)13 (2000), p. 391-411. URL : <http://>.
- [346] Nitika SHARMA et Kriti SAROHA. *A Novel Dimensionality Reduction Method for Cancer Dataset using PCA and Feature Ranking*. en. 2015.
- [347] Doaa SHAWKY et Ashraf BADAWI. *Towards a Personalized Learning Experience Using Reinforcement Learning*. en. 2018.

- [348] Mohammad Amin SHAYEGAN et Saeed AGHABOZORGI. *A New Dataset Size Reduction Approach for PCA-Based Classification in OCR Application*. en. 2014.
- [349] H. SHIMODAIRA. « DCGA : A diversity control oriented genetic algorithm ». en. In : *Proceedings of the 2nd International Conference on Genetic Algorithms in Engineering Systems : Innovations and Applications*. 1997, p. 444-449.
- [350] Florian SIEGMUND, Amos H.C. NG et Kalyanmoy DEB. *A Comparative Study of Dynamic Resampling Strategies for Guided Evolutionary Multi-Objective Optimization*. en. 2013.
- [351] Florian SIEGMUND, Amos H.C. NG et Kalyanmoy DEB. *Hybrid Dynamic Resampling for Guided Evolutionary Multi-Objective Optimization*. en. 2015.
- [352] Florian SIEGMUNDA, Amos H.C. NGA et Kalyanmoy DEBB. *Standard Error Dynamic Resampling for Preference-based Evolutionary Multi-objective Optimization*. en. 2015.
- [353] H.A. SIMON. « Invariants of human behavior ». en. In : *Annual Review of Psychology* 41 (1990), p. 1-20. DOI : [10.1146/annurev.ps.41.020190.000245](https://doi.org/10.1146/annurev.ps.41.020190.000245). URL : <http://dx>.
- [354] G. SINGH et K. DEB. « Comparison of multi-modal optimization algorithms based on evolutionary algorithms ». en. In : *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*. 2006, p. 1305-1312.
- [355] S.K. SMIT et A.E. EIBEN. « Comparing parameter tuning methods for evolutionary algorithms ». en. In : *Proceedings of the IEEE Congress on Evolutionary Computation*. 2009, p. 399-406.
- [356] J.E. SMITH et T.C. FOGARTY. « Operator and parameter adaptation in genetic algorithms ». en. In : *Soft Comput.* 1 2 (1997), p. 81-87.
- [357] R.E. SMITH, S. FORREST et A.S. PERELSON. « Searching for diverse, cooperative subpopulations with genetic algorithms ». en. In : *Evol. Comput* 1, 2 (1993), p. 127-149.
- [358] R.E. SMITH et E. SMUDA. « Adaptively resizing populations : Algorithm, analysis, and first results ». en. In : *Complex Syst* 9 (1995), p. 47-72.
- [359] R.T. SOKOLOV et J.C. ROGERS. *Removing harmonic signal non-stationarity by dynamic resampling*. 1995.
- [360] Maxime SOLER. « Large Data Reduction and Structure Comparison with Topological Data Analysis ». Theses. Sorbonne Universite, juin 2019. URL : <https://hal.archives-ouvertes.fr/tel-02171190>.
- [361] Ngo Tung SON et al. « Meta-Heuristic Algorithms for Learning Path Recommender at MOOC ». In : *IEEE Access* 9 (2021), p. 59093-59107.
- [362] Kenneth SORENSEN, Marc SEVAUX et Fred GLOVER. « A history of metaheuristics ». In : *arXiv preprint arXiv :1704.00853* (2017).
- [363] Kenneth SÖRENSEN. « Metaheuristics—the metaphor exposed ». In : *International Transactions in Operational Research* 22.1 (2015), p. 3-18.
- [364] C.O.S. SORZANO, J. VARGAS et A. PASCUAL-MONTANO. *A survey of dimensionality reduction techniques*. fr. 2014.
- [365] C. SOZA et al. « Solving timetabling problems using a cultural algorithm ». en. In : *Appl. Soft Comput* 11, 1 (2011), p. 337-344.
- [366] M. SRINIVAS et L.M. PATNAIK. « Adaptive probabilities of crossover and mutation in genetic algorithms ». en. In : *IEEE Trans. Syst. Man Cybernetics* 24 (1994), p. 656-667.

- [367] M. SRINIVAS et M. PATNAIK. « Adaptive probabilities of crossover and mutation in genetic algorithms ». en. In : *IEEE Transactions on Systems, Man, and Cybernetics* 24.4 (1994), p. 656-667.
- [368] Christian STADLER, Tazeeb RAJWANI et Florence KARABA. « Solutions to the exploration/exploitation dilemma : Networks as a new level of analysis ». In : *International Journal of Management Reviews* 16.2 (2014), p. 172-193.
- [369] D.W. STEPHENS et J.R. KREBS. *Foraging theory*. en. Princeton, NJ : Princeton University Press, 1987.
- [370] T. STORCH. « On the choice of the population size ». en. In : *Proceedings of the Genetic and Evolutionary Computation Conference*. 2004, p. 748-760.
- [371] Özge SÜRER, Robin BURKE et Edward C MALTHOUSE. « Multistakeholder recommendation with provider constraints ». In : *Proceedings of the 12th ACM Conference on Recommender Systems*. Vancouver British Columbia Canada : ACM, 2018, p. 54-62.
- [372] Seba SUSAN et Amitesh KUMAR. *The balancing trick : Optimized sampling of imbalanced datasets - A brief survey of the recent State of the Art*. en. 2019.
- [373] R.S. SUTTON et A.G. BARTO. *Reinforcement Learning : An Introduction*. en. T. 1. Cambridge : MIT Press, 1998, p. 295.
- [374] R.S. SUTTON et A.G. BARTO. *Reinforcement learning : An introduction*. en. Cambridge, MA : MIT Press, 1998.
- [375] Anna SYBERFELDT et al. *Evolutionary optimisation of noisy multi-objective problems using confidence-based dynamic resampling*. en. 2010.
- [376] Hills T.T, Todd P.M et Goldstone R.L. *The central executive as a search process : Priming exploration and exploitation across domains*. en. 2010.
- [377] E.G. TALBI. *A taxonomy of hybrid metaheuristics*. cs. 2002.
- [378] El-Ghazali TALBI. *Metaheuristics : from design to implementation*. T. 74. John Wiley & Sons, 2009.
- [379] Vincent TAM et al. *Enhancing Educational Data Mining Techniques on Online Educational Resources with A Semi-Supervised Learning Approach*. en. The University of Hong Kong, Allan H.K. Yuen.
- [380] Maritzol TENEMAZA et al. « Improving itinerary recommendations for tourists through metaheuristic algorithms : an optimization proposal ». In : *IEEE Access* 8 (2020), p. 79003-79023.
- [381] K. TEODORESCU et al. « On the decision to explore new alternatives : The coexistence of under- and over-exploration ». en. In : *Journal of Behavioral Decision Making* 27 (2014). 0956797614543022, p. 109-123. DOI : [10.1002/bdm.1785](https://doi.org/10.1002/bdm.1785). URL : <http://dx.doi.org/>.
- [382] K. THAKER, P. CARVALHO et K. KOEDINGER. *Comprehension factor analysis : Modeling student's reading behaviour : Accounting for reading practice in predicting students' learning in moocs*. en. LAK19 : ACM, 2019, p. 111-115. DOI : [10.1145/3303772.3303817](https://doi.org/10.1145/3303772.3303817). URL : <https://doi.org/10.1145/3303772.3303817>.
- [383] Nava TINTAREV et Judith MASTHOFF. « A survey of explanations in recommender systems ». In : *2007 IEEE 23rd international conference on data engineering workshop*. IEEE. 2007, p. 801-810.

- [384] P.M. TODD. « Coevolved cognitive mechanisms in mate search : Making decisions in a decision-shaped world ». en. In : *Evolution and the social mind : Evolutionary psychology and social cognition*. Sous la dir. de J.P. FORGAS, M.G. HASELTON et W.Von HIPPE. New York, NY : Psychology Press, 2007, p. 145-159.
- [385] P.M. TODD, T.T. HILLS et T. ROBBINS. *Cognitive search : Evolution, algorithms, and the brain*. en. Cambridge, MA : MIT Press, 2012.
- [386] P.M. TODD et G.F. MILLER. « From pride and prejudice to persuasion : Satisficing in mate search ». en. In : sous la dir. de G. GIGERENZER, P.M. TODD et EDS. New York, NY : Oxford University Press & the ABC Research Group, 1999, p. 287-308.
- [387] A. TOFFOLO et E. BENINI. « Genetic diversity as an objective in multi-objective evolutionary algorithms ». en. In : *Evol. Comput* 11, 2 (2003), p. 151-167.
- [388] I. TOMEK. « Two modifications of CNN ». fr. In : *IEEE Trans Syst Man Commun SMC-6* 6.11 (1976), p. 769-772.
- [389] Y. TSUJIMURA et M. GEN. « Entropy-based genetic algorithm for solving tsp ». en. In : *Proceedings of the 2nd International Conference on Knowledge-Based Intelligent Electronic Systems*. 1998, p. 285-290.
- [390] S. TSUTSUI, Y. FUJIMOTO et A. GHOSH. « Forking genetic algorithms : GAs with search space division schemes ». de. In : *Evol. Comput* 5, 1 (1997), p. 61-80.
- [391] S. TSUTSUI et al. « A real coded genetic algorithm with an explorer and an exploiter populations ». en. In : *Proceedings of the 7th International Conference on Genetic Algorithms*. 1997, p. 238-245.
- [392] E.M. TUTTLE, L. WULFSON et T. CARACO. « Risk-aversion, relative abundance of resources and foraging preference ». en. In : *Behavioral Ecology and Sociobiology* 26 (1990), p. 165-171. DOI : [10.1007/BF00172083](https://doi.org/10.1007/BF00172083). URL : <http://dx.doi.org/>.
- [393] O.E. TYKOCINSKI et A. ORTMANN. « The lingering effects of our past experiences : The sunkcost fallacy and the inaction-inertia effect ». en. In : *Social and Personality Psychology Compass* 5 (2011), p. 653-664. DOI : [10.1111/j.1751-9004.2011](https://doi.org/10.1111/j.1751-9004.2011). URL : <http://dx.doi.org/10.1111/j.1751-9004.2011>.
- [394] R. URSEM. « Diversity-guided evolutionary algorithms ». en. In : *Proceedings of Parallel Problem Solving from Nature, Lecture Notes in Computer Science*. T. 2439. Springer, 2002, p. 462-471.
- [395] R. URSEM. « Multinational GAs : Multimodal optimization techniques in dynamic environments ». en. In : *Proceedings of the 2nd Genetic and Evolutionary Computation Conference*. 2000, p. 19-26.
- [396] T.J. VALONE. « From eavesdropping on performance to copying the behavior of others : A review of public information use ». en. In : *Behavioral Ecology and Sociobiology* 62 (2007), p. 1-14. URL : <http://dx.doi.org/10.1007/>.
- [397] Roberto S VASSOLO, Jaideep ANAND et Timothy B FOLTA. « Non-additivity in portfolios of exploration activities : A real options-based analysis of equity alliances in biotechnology ». In : *Strategic Management Journal* 25.11 (2004), p. 1045-1061.
- [398] J.L. VERDOLIN. « Meta-analysis of foraging and predation risk trade-offs in terrestrial systems ». en. In : *Behavioral Ecology and Sociobiology* 60 (2006), p. 457-464. DOI : [10.1007/s00265-006-0172-6](https://doi.org/10.1007/s00265-006-0172-6). URL : <http://dx.doi.org/10.1007/s00265-006-0172-6>.

-
- [399] Freek VERMEULEN et Harry BARKEMA. « Learning through acquisitions ». In : *Academy of Management journal* 44.3 (2001), p. 457-476.
- [400] D. VICKERS. *Decision processes in visual perception*. en. New York, NY : Academic Press, 1979.
- [401] J.S. VITTER. « Random sampling with a reservoir ». en. In : *ACM Transactions on Mathematical Software* 11.1 (1985), p. 37-57.
- [402] E. VUL et al. « One and done? Optimal decisions from very few samples ». en. In : *Proceedings of the Thirty-First Annual Conference of the Cognitive Science Society*. Sous la dir. de N.A. TAATGEN et H. RIJN. 2009, p. 148-153.
- [403] Shang WANG. *Designing Interactive and Personalized Concept Mapping Learning Environments*. en. Tempe AZ, USA : Arizona State University.
- [404] Y. WANG, Z. CAI et Q. ZHANG. « Enhancing the search ability of differential evolution through orthogonal crossover ». en. In : *Inform. Sci* 185, 1 (2012), p. 153-177.
- [405] G. WEBER et P. BRUSILOVSKY. « ELM-ART : An adaptive versatile system for web-based instruction ». en. In : *Int. Journal of Artificial Intelligence in Education* 12.4 (2001).
- [406] E. WEINBERGER. « Local properties of Kauffman's NK model : a tunably rugged energy landscape ». en. In : *Physical Review A* 44.10 (1991), p. 6399-6413.
- [407] D. WHITLEY, K. MATHIAS et P. FITZHORN. « Delta coding : An iterative search strategy for genetic algorithms ». en. In : *Proceedings of the 4th International Conference on Genetic Algorithms*. 1991, p. 77-84.
- [408] D. WHITLEY et D. STARKWEATHER. « Genitor-ii : A distributed genetic algorithm ». it. In : *J. Exp. Theor. Artif. Intell* 2, 3 (1990), p. 189-214.
- [409] J.M. WOLFE. « When is it time to move to the next raspberry bush? Foraging rules in human visual search ». en. In : *Journal of Vision* 13 (2013), p. 10. URL : <http://dx.doi>.
- [410] David H WOLPERT et William G MACREADY. « No free lunch theorems for optimization ». In : *IEEE transactions on evolutionary computation* 1.1 (1997), p. 67-82.
- [411] Y.-Y. WONG et al. « A novel approach in parameter adaptation and diversity maintenance for genetic algorithms ». en. In : *Soft Comput* 7 (2003), p. 506-515.
- [412] G. XIANG. « From traditional textbooks to electronic textbooks ». en. In : *Information Technology Education* 5 (2005). in Chinese, p. 8-10.
- [413] J. XU et J. ZHANG. « Differential Evolution Algorithm Based on Uniform Design for Multiobjective Optimization Problems ». en. In : *ICIC Express Letters* 5.9B (2011), p. 3381-3388.
- [414] Junqin XU et Jihui ZHANG. « Exploration-exploitation tradeoffs in metaheuristics : Survey and analysis ». In : *Proceedings of the 33rd Chinese control conference*. IEEE. Nanjing, China : IEEE, 2014, p. 8633-8638.
- [415] X. XU et J. LIU. « Achievement evaluation in e-Textbooks ». en. In : *Journal of Xi'an Aeronautical College* 9 (2003), p. 24-25.
- [416] Xinzheng XU et al. *Review of Classical Dimensionality Reduction and Sample Selection Methods for Large-scale Data Processing*. en. 2018.
- [417] A. YANG, H.A. ABBASS et R. SARKER. « Characterizing warfare in red teaming ». en. In : *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 36.2 (2006), p. 268-285.

- [418] Xin-She YANG. *Nature-inspired metaheuristic algorithms*. Luniver press, 2010.
- [419] X. YAO, Y. LIU et G. LIN. « Evolutionary programming made faster ». en. In : *IEEE Trans. Evol. Comput* 3, 2 (1999), p. 82-102.
- [420] Gary YEN, Fengming YANG et Travis HICKEY. « Coordination of exploration and exploitation in a dynamic environment ». In : *International Journal of Smart Engineering System Design* 4.3 (2002), p. 177-182.
- [421] X. YIN et N. GERMAY. « A fast genetic algorithm with sharing scheme using cluster analysis method in multi-modal function optimization ». en. In : *Proceedings of the International Conference on Artificial Neural Nets and Genetic Algorithms*. 1993, p. 450-457.
- [422] E.L. YU et P.N. SUGANTHAN. « Ensemble of niching algorithms ». it. In : *Inform. Sci* 180, 15 (2010), p. 2815-2833.
- [423] S.Y. YUEN et C.K. CHOW. « A genetic algorithm that adaptively mutates and never revisits ». en. In : *IEEE Trans. Evol. Comput* 13, 2 (2009), p. 454-472.
- [424] B. ZADROZNY. « Learning and Evaluating Classifiers under Sample Selection Bias ». en. In : *Twenty-first International Conference on Machine Learning*. T. 114. 2004. DOI : [10.1145/1015330.1015425](https://doi.org/10.1145/1015330.1015425).. URL : <https://doi.org/10.1145/1015330.1015425>..
- [425] Gustavo R ZAVALA et al. « A survey of multi-objective metaheuristics applied to structural optimization ». In : *Structural and Multidisciplinary Optimization* 49.4 (2014), p. 537-558.
- [426] J. ZHANG et A.C. SANDERSON. « Jade : Adaptive differential evolution with optional external archive ». en. In : *IEEE Trans. Evol. Comput* 13, 5 (2009), p. 945-957.
- [427] Jinbao ZHANG et Liangang ZHANG. *Semantic Enhanced e-Textbooks - Information Model and Its Application*. en. Beijing Normal University, China : School of Educational Technology.
- [428] Jun ZHANG et al. « A survey on algorithm adaptation in evolutionary computation ». In : *Frontiers of Electrical and Electronic Engineering* 7.1 (2012), p. 16-31.
- [429] Y. ZHENG et al. « Fairness in reciprocal recommendations : A speed-dating study ». en. In : *Adjunct Proceedings of the ACM conference on User Modelling, Adaptation and Personalization*. ACM, 2018.
- [430] Yong ZHENG. « Preference corrections : capturing student and instructor perceptions in educational recommendations ». In : *Smart Learning Environments* 6.1 (2019), p. 1-15.
- [431] Yong ZHENG, Nastaran GHANE et Milad SABOURI. « Personalized educational learning with multi-stakeholder optimizations ». In : *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*. 2019, p. 283-289.
- [432] Yong ZHENG et Aviana PU. « Utility-based multi-stakeholder recommendations by multi-objective optimization ». In : *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE. Santiago, Chile : IEEE, 2018, p. 128-135.
- [433] R. ZHOU, M. LIU et T. LI. « Characterizing the efficiency of data deduplication for big data storage management ». en. In : *2013 IEEE international symposium on workload characterization (IISWC)*. 2013.
- [434] Z. ZHU et X. YU. « E-schoolbag system and their functional modeling ». en. In : *E-education Research* 4 (2011). in Chinese, p. 24-27.

-
- [435] Cai-Nicolas ZIEGLER et al. « Improving recommendation lists through topic diversification ». en. In : *Proceedings of the 14th international conference on World Wide Web. ACM*. 2005, p. 22-32.
- [436] K. ZIELINSKI et al. « Optimization of power allocation for interference cancellation with particle swarm optimization ». en. In : *IEEE Trans. Evol. Comput* 8, 2 (2009), p. 128-150.
- [437] E. ZITZLER, K. DEB et L. THIELE. « Comparison of multiobjective evolutionary algorithms : Empirical results ». en. In : *Evol. Comput* 8, 2 (2000), p. 173-195.
- [438] E. ZITZLER et L. THIELE. « Multiobjective evolutionary algorithms : A comparative case study and the strength pareto approach ». en. In : *IEEE Trans. Evol. Comput* 3, 4 (1999), p. 257-271.
- [439] E. ZITZLER et L. THIELE. *Multiobjective evolutionary algorithms : A comparative case study and the strength pareto approach*. en. 1999.
- [440] Junyi ZUO. *Dynamic resampling for alleviating sample impoverishment of particle filter*. en. 2013.
- [441] Y. ZUO et al. « Personalized recommendation based on evolutionary multi-objective optimization ». fr. In : *IEEE Computational Intelligence Magazine* 10.1 (2015), p. 52-62.