



HAL
open science

Influence indépendante de l'exploration et de l'exploitation dans les métaheuristiques : application aux systèmes de recommandation

Alexandre Bettinger, Armelle Brun, Anne Boyer

► To cite this version:

Alexandre Bettinger, Armelle Brun, Anne Boyer. Influence indépendante de l'exploration et de l'exploitation dans les métaheuristiques : application aux systèmes de recommandation. Conférence Nationale en Intelligence Artificielle 2022 (CNIA 2022), Jun 2022, Saint-Etienne, France. hal-03866081

HAL Id: hal-03866081

<https://hal.science/hal-03866081>

Submitted on 22 Nov 2022

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

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

Influence indépendante de l'exploration et de l'exploitation dans les métaheuristiques : application aux systèmes de recommandation

Alexandre Bettinger, Armelle Brun, Anne Boyer

Université de Lorraine, CNRS LORIA

prénom.nom@loria.fr

Résumé

L'exploration et l'exploitation (E&E) d'un espace de recherche sont deux processus fondamentaux en intelligence artificielle. L'influence de l'E&E vise à améliorer ces processus. L'explicabilité de l'E&E est aussi un enjeu important. Cet article introduit un processus indépendant d'influence de l'E&E et des indicateurs pour l'explicabilité de l'E&E. Nos expérimentations sur les algorithmes génétique et par renforcement confirment que (1) le processus d'influence proposé a un impact positif sur l'E&E, (2) les indicateurs proposés représentent l'E&E avec plus de clarté.

Mots-clés

Recommandation, métaheuristique, algorithme évolutionnaire, exploration, exploitation, influence, renforcement

Abstract

The exploration and exploitation (E&E) of a search space are two fundamental processes in artificial intelligence. The influence of E&E aims to improve these processes. The explainability of E&E is also an important issue. This article introduces an E&E independent influence process and indicators for the explainability of E&E. Our experiments on genetic and reinforcement algorithms confirm that (1) the proposed influence process has a positive impact on E&E, (2) the proposed indicators represent E&E more clearly.

Keywords

Recommendation, metaheuristic, evolutionary algorithm, exploration, exploitation, influence, reinforcement

1 Introduction

La recommandation est le fait de filtrer des informations afin de cibler des éléments (*items*, ressources) susceptibles d'intéresser un ou plusieurs utilisateurs. Cette tâche peut être vue comme le traitement d'un vaste espace de recherche qui représente l'ensemble des recommandations possibles. Selon le contexte de la recommandation, une recommandation peut prendre différentes formes telles que des *items* uniques, des ensembles d'*items* (*itemsets*) ou des séquences d'*items*. Dans ce qui suit, nous nous intéressons au cadre de recommandation d'*itemsets* (également appelés vecteurs ou solutions).

L'espace de recherche peut être exploré par un processus d'exploration et d'exploitation (E&E) pour trouver de bonnes recommandations [4]. Explorer signifie s'assurer que de nombreuses régions de l'espace de recherche sont examinées [6], 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 [6] pour voir si elle contient de bons optima locaux. Dans le cadre de la recommandation, un optimum local est un *itemset* 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 que le domaine des métaheuristiques (MH) aborde la problématique de la proposition de recommandations dans de vastes espaces de recherche [6].

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 [6]. Influencer l'E&E implique de savoir quand, pourquoi et comment influencer. 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. De plus, ils augmentent les capacités d'E&E à trouver de meilleures recommandations.

Dans un cadre de recommandation, un utilisateur peut avoir des préférences explicites ou implicites sur les recommandations attendues, qui peuvent être représentées par des contraintes [27] associées à des attributs d'*items*. Nous représentons les contraintes sous la forme de besoins de maximisation et de minimisation ainsi que de plages de valeurs totales. Par exemple, dans le cas de la recommandation d'un ensemble d'articles 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 \$.

La problématique que nous visons à aborder dans ce travail est donc : Comment contrôler, influencer et expliquer

l'E&E dans le domaine des métaheuristiques ? Nos expérimentations sont réalisées dans le cadre de la recommandation par MH.

L'apport de ce travail est triple. 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 structurales 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, ce qui vient servir l'explicabilité de l'E&E.

2 Travaux connexes

Dans de vastes espaces de recherche, filtrer les informations est fastidieux et effectuer une recherche exhaustive est inconcevable. Par conséquent, le ciblage des *items* à recommander nécessite de parcourir partiellement l'espace de recherche à l'aide de mécanismes algorithmiques capables d'explorer et d'exploiter [4]. L'E&E sont les deux comportements clés pour parcourir un grand espace de recherche [6]. L'E&E sont aussi appelées diversification et intensification [19], ou recherche globale et recherche locale [6]. L'E&E sont donc intrinsèquement en conflit [6] car si beaucoup de temps est passé sur l'une, moins de temps est disponible pour l'autre. Eiben et Schippers [11] ont abordé la nécessité de trouver un équilibre entre E&E, considérant ces deux processus comme antagonistes.

Nous proposons une définition des MH, expliquons pourquoi elles sont utilisées pour la recommandation et proposons une synthèse de trois revues [6][29][22] qui traitent des questions liées à l'influence de l'E&E, par compromis ou guidage ou recherche d'équilibre. Bon nombre des points abordés dans ces revues peuvent être généralisés à d'autres domaines de recherche et d'application.

2.1 Métaheuristiques

Les méthodes heuristiques désignent des techniques de résolution de problèmes, d'apprentissage et de découverte où une recherche exhaustive n'est pas possible en raison de problèmes liés à la puissance de calcul, au temps d'exécution mais aussi à la consommation d'énergie. Les MH sont des heuristiques intégrées dans des processus itératifs afin d'augmenter leurs capacités d'E&E [12]. 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 [7]. Les MH se sont avérées bien adaptées aux problèmes d'optimisation multi-objectifs et mono-objectifs car elles sont capables de capturer plusieurs solutions simultanément, tout en tenant compte des contraintes. Notons que traiter un problème d'optimisation mono-objectif consiste à trouver une ou plusieurs bonnes solutions pour un ou plusieurs attributs, en respectant des contraintes et en essayant de satisfaire des métriques de performance et de qualité. Il est

possible de combiner les objectifs d'un problème d'optimisation multi-objectifs pour le convertir en un problème d'optimisation à objectif unique en définissant la fonction de *fitness* comme la somme pondérée des valeurs normalisées des attributs.

Notre contribution relève du domaine des MH car (1) elles sont transparentes et explicables [21] 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 (3) elles sont particulièrement efficaces pour optimiser les décisions [32].

2.2 Métaheuristiques et recommandation

Les MH peuvent être utilisées pour résoudre le problème de la recommandation dans d'immenses espaces de recherche. Dans la littérature, l'utilisation des MH n'est pas systématiquement référée à la recommandation mais souvent à l'optimisation qui est, à bien des égards, très proche de la recommandation. En effet, en considérant le problème du sac à dos [20], un des problèmes d'optimisation les plus anciens et les plus connus, nous supposons que les solutions trouvées par les MH pour résoudre ce problème peuvent constituer des recommandations simples et qu'il est possible d'ajouter des contraintes représentant les préférences d'un utilisateur pour fournir des recommandations personnalisées, sans changer l'essence du problème d'optimisation et en apportant le fait que les solutions trouvées constituent des recommandations personnalisées pour l'utilisateur. Par conséquent, un état de l'art sur l'utilisation des MH pour la tâche de recommandation doit prendre en considération toutes les contributions du domaine des MH. Deux revues particulièrement complètes relatives aux MH [9] [10] et des travaux spécifiques référant l'utilisation des MH à la recommandation [18] [26] [28] constituent la base de notre état de l'art sur ce sujet. Dans [26], un outil de recommandation multi-objectifs est proposé. Le système de recommandation génère des parcours d'apprentissage appropriés pour les apprenants à l'aide d'algorithmes MH, notamment l'algorithme génétique (Genetic Algorithm - GA) et l'algorithme d'optimisation des colonies de fourmis (Ant Colony Optimization - ACO). Concrètement, les *items* à recommander (des cours) sont des nœuds dans un graphe, la MH est utilisée pour générer des chemins à travers le graphe et les meilleurs chemins trouvés sont recommandés. Dans [18], un système de recommandation hybride appliqué au jeu de données Movielens est proposé. Concrètement, un algorithme de *clustering* k-means est utilisé pour créer des *clusters* d'utilisateurs, chaque *cluster* ayant un centroïde. Ensuite, un algorithme de colonie d'abeilles artificielles (Artificial Bee Colony - ABC) est utilisé pour optimiser les distances des utilisateurs aux centroïdes et pour les reclasser si nécessaire. Les recommandations données à un utilisateur sont directement liées aux distances aux centroïdes de l'utilisateur, en effet, des contenus d'utilisateurs similaires (films), au regard des distances aux centroïdes, sont recommandés. Dans [28], les auteurs proposent d'améliorer les recommandations d'itinéraires pour les touristes

grâce à des MH. Concrètement, un algorithme de *clustering* 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 montrent que, comme dit plus haut, la tâche de recommandation peut être considérée comme une sélection optimisée d'*items* dans un espace de recherche.

2.3 Influence de l'exploration et de l'exploitation dans les métaheuristiques

2.3.1 Définitions

L'E&E peuvent être caractérisées comme une recherche globale et locale [1], l'acquisition et l'utilisation de connaissances [30], des processus qui se distinguent par l'intensité du caractère aléatoire [24], ou comme deux types de comportement dans l'acquisition d'informations [3].

Les MH se retrouvent souvent piégées dans un optimum local, la principale raison étant la difficulté à bien arbitrer entre E&E. Une bonne MH doit donc faire de bons compromis entre E&E, mais aucune réponse complète et/ou générale à ce problème n'a été apportée à ce jour.

2.3.2 Contrôle de l'exploration et de l'exploitation

Chercher à contrôler l'E&E revient à poser les questions suivantes : quand, quoi et comment contrôler ? La mesure de la diversité, génotypique ou phénotypique est devenue largement acceptée comme approche de contrôle [23]. La mesure de la diversité génotypique peut être basée sur la différence, la distance, l'entropie, la probabilité ou même les ancêtres/l'histoire. La mesure de la diversité phénotypique peut être basée sur la différence, la distance, l'entropie, la probabilité.

2.3.3 Influence de l'exploration et de l'exploitation

De nombreux facteurs ont un impact sur l'E&E. Une discussion initiale sur l'E&E est menée dans [11] et d'autres travaux ont ensuite poursuivi cette discussion, traitant par exemple de la puissance d'exploration [8] et du redimensionnement de la population [13].

Dans la littérature, l'influence de l'E&E est qualifiée de compromis, d'orientation ou d'équilibre. L'influence est cruciale dans les méthodes d'optimisation car elle améliore l'efficacité de la recherche. La littérature a proposé de nombreuses approches adaptatives [31] et hybrides [19] pour influencer l'E&E. De plus, la plupart des approches reposent sur la diversité par le maintien, le contrôle ou l'apprentissage. Il est important de noter que bien que de nombreux travaux effectués sur la diversité s'attendent à un meilleur équilibre entre E&E, toutes les diversifications ne sont pas utiles. De plus, il faut garder à l'esprit que les phases d'E&E sont toujours imbriquées, ce qui rend leur identification plus complexe, il est également quasiment impossible d'observer clairement leurs contributions respectives dans le processus d'E&E. Finalement, il existe peu de travaux théoriques sur la diversité [8], la plupart étant applicatifs.

2.3.4 Directions de recherche

À partir de cette revue de la littérature, nous pouvons voir qu'il existe encore de nombreuses questions ouvertes concernant l'E&E, telles que définir formellement/mathématiquement les phases d'E&E et l'équation d'équilibre, proposer des métriques (directes) pour mesurer l'E&E, définir quand et comment contrôler l'équilibre entre E&E, mener des analyses sur les différentes approches de maintien, de contrôle et d'apprentissage de la diversité.

2.4 Influence de l'exploration et de l'exploitation en psycho-sociologie

De nombreuses décisions dans la vie des êtres vivants nécessitent un équilibre entre l'exploration de différentes options et l'exploitation de leurs récompenses [22]. Cette dernière ressource explore comment le compromis entre E&E 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 eux, des objectifs du décideur. En effet, 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 plusieurs échelles dans l'espace et dans le temps. Les décisions sur la manière de se comporter à ces différents niveaux peuvent être basées sur une heuristique connue ou nouvelle. La manière de choisir les heuristiques est donc également importante. De nombreuses approches abordent les problèmes de compromis entre E&E [15] [22].

Trois éléments sont mis en évidence par [22] pour une unification de la recherche en E&E : un continuum E&E [5], différents types de transitions entre E&E [25] et le rôle des objectifs des agents dans ce processus [5]. De nombreuses ouvertures s'ensuivent : (1) proposer une théorie globale pour couvrir les dimensions conceptuelles de l'E&E, (2) saisir les interactions entre les différents niveaux d'abstraction sur lesquels se situe l'E&E, (3) proposer une utilisation hiérarchisée des mécanismes de compromis à différentes échelles et à différents niveaux d'abstraction, (4) 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, (5) proposer une théorie complète sur la variété des transitions qui peuvent se produire entre E&E, et (6) explorer dans quelles mesures les mécanismes de compromis sont sensibles à divers facteurs.

2.5 Conclusion

Pour résumer, de nombreux travaux se sont intéressés à l'arbitrage entre E&E, sans aboutir à une réponse complète ou générale. Le cadre de la recommandation est fortement concerné par ces problèmes. Des contributions proposant une nouvelle façon d'explorer et d'exploiter l'espace de recherche sont attendues. Il en va de même pour les contributions visant à expliquer, mesurer, contrôler et influencer l'E&E.

3 Contributions

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. 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. Nous présentons nos travaux dans le cadre des algorithmes évolutionnaires (Evolutionary Algorithm - EA) et des algorithmes par renforcement, en raison de leur popularité. Notons que l'indépendance fait que l'influence peut être utilisée sur la plupart des MH, sans adaptation majeure.

Les tableaux de *digits* binaires sont une manière traditionnelle de représenter des *items* ou *itemsets* recommandés [20]. 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 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.

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

3.1 Processus d'influence de l'exploration et de l'exploitation, indépendant de l'algorithme influencé

Soulignons 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 1, Figure 1), (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. 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

Algorithm 1 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
-

définies en amont, ainsi que la valeur du paramètre utilisé pour ajuster ces limites (voir 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.

Le principe de fonctionnement du processus d'influence repose sur une fonction indépendante appelée systématiquement lors de l'exécution de l'algorithme influencé (Algorithme 2). Dans la littérature, l'influence n'est actionnée qu'une seule fois par itération. En ce qui nous concerne, 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 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). 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

Algorithm 2 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()
```

(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.

3.2 Nouveaux indicateurs pour l'influence et l'explicitabilité de l'exploration et de l'exploitation

Comme le suggère la littérature, l'E&E mériteraient d'être représentées sous de nouveaux angles [6][29][22]. C'est ce que nous entendons faire en proposant trois types d'indicateurs qui mesurent et représentent les comportements d'E&E et qui peuvent être utilisés au sein de la fonction d'influence introduite précédemment.

3.2.1 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, 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. Cela donne la complétion des contraintes d'un $item$ (CC_{item}). La somme des CC_{item} d'un $itemset$ donnent sa complétion des contraintes ($CC_{itemset}$). L'indicateur CC est calculé en faisant la moyenne de l'ensemble des $CC_{itemset}$ (Equation (1)). L'ensemble des $CC_{itemset}$ est noté CC_{itsets} .

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.

$$CC = \overline{CC_{itsets}} \quad (1)$$

3.2.2 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.

Nous évaluons le nombre d'itérations d'exploration divisé par le nombre d'itérations d'exploitation ($\#eit$), 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 (2)), il représente le ratio d'exploration comme la synthèse de ses valeurs.

$$EEP = \overline{(\#eit, \#evec, \#dnd)} \quad (2)$$

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 nnd 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. $nnd(ie)$ correspond donc à nnd au marqueur de fin d'itération ie . Nous pouvons donc définir p comme dans Equation (3). 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.

$$p(ie + 1) = \begin{cases} explore, & \text{if } \overline{nnd(ie)} > \overline{nnd(ie + 1)} \\ exploit, & \text{if } \overline{nnd(ie)} = \overline{nnd(ie + 1)} \end{cases} \quad (3)$$

3.2.3 Temporalité

Nous proposons de représenter l'alternance des itérations d'E&E comme des écarts entre des valeurs temporelles. Les écarts temporels entre les instants de démarrage des itérations d'exploration sont évalués et font ressortir les écarts minimum (min) et maximum (max), la différence entre l'écart minimum et l'écart maximum ($DMMG$) (Equation (4)), l'écart moyen (AG) (Equation (5)). Idem pour les itérations d'exploitation. $DMMG$ 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.

$$DMMG = maxg - ming \quad (4)$$

$$AG = \overline{gaps} \quad (5)$$

4 Expérimentations

4.1 Données et mesures d'évaluation

Nos expérimentations ont été réalisées sur le jeu de données très utilisé dans la littérature de la recommandation « MovieLens 25M » [14]. 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*.

Les objectifs et les contraintes des recommandations ont été déterminés arbitrairement en veillant à apporter suffisamment de diversité, de même que les limites des indicateurs d'E&E et les valeurs utilisées pour les mettre à jour.

Une exécution concerne une variation de contrainte, un objectif de recommandation, un cas d'exécution (influencé ou non) et une macro-exécution. Les expérimentations comprennent six variations de contraintes, six objectifs de recommandation, huit cas d'exécution : un non influencé (référence) et sept influencés (qui varient selon leur indicateur d'E&E), et trente boucles de macro-exécution, pour un total de plusieurs milliers d'exécutions, chacune comprenant mille itérations.

Les fonctions d'influence sont appliquées sur un algorithme génétique traditionnel qui sert de référence pour les comparaisons. Les fonctions d'influence sont aussi appliquées sur un algorithme par renforcement traditionnel servant à son tour de référence. L'objectif est de montrer les différences comportementales de l'algorithme de référence lorsqu'il est influencé par le processus proposé et avec différents indicateurs d'E&E.

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.

Des tests statistiques traditionnels de type *ranksum* sont effectués afin de s'assurer que les valeurs des critères d'évaluation, qui sont des moyennes sur toutes les macro-itérations, proviennent bien de distributions différentes par rapport à celles de l'algorithme de référence. Il s'agit d'une évaluation traditionnelle en MH.

L'infrastructure informatique Grid'5000 [2] est utilisée pour ces expérimentations.

4.2 Macro-processus

Le macro-processus adopté dans ces expérimentations s'articule autour de différentes étapes. 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 sont sous la forme de tuples (*item* id, attribut 1, ..., attribut n). Nous choisissons d'utiliser les attributs numériques suivants de MovieLens : la note moyenne, la quantité de notes et la pertinence moyenne des *tags*. 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*. Cette combinaison de choix présentée ci-dessus n'a pas été observée dans la littérature.

L'algorithme de référence (évolutionnaire ou renforcement) 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 de la performance et de la qualité sont renvoyées pour toutes les exécutions. Divers éléments de visualisation sont enregistrés en conséquence.

4.3 Critères d'évaluation

Nous décidons d'évaluer notre processus d'influence en considérant plusieurs critères liés à la performance, à la qualité et au temps qui sont soit ceux introduits dans la section 3.2, soit proposés dans la littérature [17][16].

4.3.1 Critères de performance

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 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, mieux c'est,

car cela signifie que le processus de recherche a trouvé de nombreux *items*.

Constraints Completion (CC) : Pourcentage moyen de complétion des contraintes. Il représente dans quelle mesure les *items* recommandés remplissent les contraintes.

E&E Power (EEP) : Part de l'exploration par rapport à l'exploitation. Voir la section 3 pour plus de détails.

4.3.3 Critères temporels

Min-Max Gap (DMMG) : Différence des écarts temporels minimum et maximum entre deux itérations d'exploration.

Average Gap (AG) : Écart temporel moyen entre deux itérations d'exploration.

4.4 Résultats - Algorithme génétique

L'algorithme génétique de référence utilise un opérateur de croisement de type "one point" et un opérateur de mutation de type "bit string", ce dernier est effectué tant que l'*itemset* muté respecte les contraintes maximales. Les pourcentages agrégés sont indiqués dans Table 1. Les lignes % ↗, % → et % ↘ font respectivement référence aux pourcentages de résultats des critères d'évaluation augmentés, inchangés et diminués par rapport à l'algorithme de référence, en tenant compte de tous les cas d'exécution. Pour compléter ces informations, nous effectuons des tests statistiques représentés par la ligne %Pass indiquant le pourcentage de tests réussis en tenant compte de 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 non influencé et de l'une de ses versions influencées, 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 qu'un critère d'évaluation permet de tirer des conclusions fiables si son %Pass est supérieur à 50%.

TABLE 1 – Résultats - Impact de l'Influence sur l'E&E de l'Algorithme Génétique

	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

4.4.1 Critères de performance

CI et *MSI* augmentent majoritairement par l'influence avec respectivement 62% et 65%. 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. Dans l'ensemble des cas d'exécution, aucun problème de convergence prématurée n'a été observé. Cette conclusion doit être vue à la lumière de tests statistiques 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 avec 56% mais obtient toujours un important ratio d'augmentation avec 40%. *MS* est augmenté avec 63% et obtient un taux de stagnation significatif de 36%. Globalement, 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 raisonnablement augmenté par l'influence, ce qui est validé par les tests statistiques.

4.4.2 Critères de qualité

AVC est inchangé (50%) ou augmente (50%) 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 donc ce critère d'évaluation permet de tirer des conclusions fiables dans le cadre de ces expérimentations.

CC reste inchangé (48%) ou augmente (44%) 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 car 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, en effet la plupart d'entre eux rejettent l'hypothèse nulle donc ce critère d'évaluation permet de tirer des conclusions fiables dans le cadre de ces expérimentations.

EEP diminue globalement (85%) 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, en effet la plupart d'entre eux rejettent l'hypothèse nulle donc ce critère d'évaluation permet de tirer des conclusions fiables dans le cadre de ces expérimentations.

4.4.3 Critères temporels

Les valeurs des critères temporels sont augmentées par l'influence, avec des ratios de 82% pour *DMMG* et de 77% 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%), 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

rejetent l’hypothèse nulle, donc ces critères d’évaluation permettent de tirer des conclusions fiables dans le cadre de ces expérimentations. Ces expérimentations montrent que les indicateurs temporels expliquent comment l’influence affecte la régularité de l’E&E.

4.4.4 Conclusion

Les critères étudiés ci-dessus 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. 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. 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. Comme 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 proposé 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, 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*.

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. Ces résultats sont prometteurs pour nos travaux futurs.

4.5 Résultats - Algorithme par renforcement

Pour montrer l’adaptabilité du processus d’influence proposé, nous choisissons de l’appliquer à un algorithme de renforcement traditionnel (Algorithme (3)) dont la structure globale est similaire à l’algorithme génétique. *Q* et *N* représentent respectivement le tableau des scores de renforcement associés à chaque digit et le tableau du nombre de

Algorithm 3 RA Calling The Influence Function

```

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()

```

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 un dans *N* (Equation (6)) et par la formule traditionnelle dans *Q* (Equation (7)). Les résultats sont présentés dans Tableau (2).

$$N[d]_{+} = 1 \quad (6)$$

$$Q[d]_{+} = 1/N[d] * (reward - Q[d]) \quad (7)$$

TABLE 2 – Résultats - Impact de l’Influence sur l’E&E de l’Algorithme par Renforcement

	Performance					Qualité			Temporel	
	CI	MSI	CS	MS	ET	AVC	CC	EEP	DMMG	AG
% ↗	37	30	34	28	100	71	31	19	89	71
% →	0	1	37	47	0	27	40	6	1	28
% ↘	63	69	29	25	0	2	28	76	10	0
%Pass	45	38	54	58	100	67	39	79	93	95

4.5.1 Critères de performance

CI et *MSI* diminuent majoritairement par l’influence avec respectivement 63% et 69%. Ainsi, 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. Dans l’ensemble des cas d’exécution, aucun problème de convergence prématurée n’a été observé.

CS et *MS* sont inchangés (37%, 47%), améliorés (34%, 28%) et diminués (29%, 25%) par l’influence. Dans l’ensemble, un nombre important de cas d’influence conduisent à des scores au moins égaux à ceux de l’algorithme de référence et dans de nombreux cas sont capables de les améliorer.

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

4.5.2 Critères de qualité

AVC est inchangé (27%) ou augmente (71%) par l’influence. Une augmentation de *AVC* signifie que davantage de dimensions (*items*) sont utilisés dans le processus de recherche.

CC reste inchangé (40%), augmente (31%) et diminue (28%) par l’influence, ce qui montre que le processus d’influence proposé contribue dans de nombreux cas à trouver

des *items* qui remplissent mieux les contraintes. La qualité des recommandations est améliorée dans ces cas.

EEP diminue globalement (76%) 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.

4.5.3 Critères temporels

Les valeurs de *DMMG* et *AG* sont augmentées par l'influence (89%, 71%). 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.

4.5.4 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.

5 Discussions et travaux futurs

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. Modifier ces éléments mènerait à des résultats différents en permettant ou non aux fonctions d'influence d'améliorer plus souvent les performances des algorithmes non influencés et la qualité des recommandations. Dans le cadre des travaux présentés, nous avons utilisé des valeurs diversifiées pour ces éléments 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.

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.

Un autre point important dans l'approche proposée est que les variations d'influence opèrent directement sur les

items de l'itération actuelle de l'algorithme influencé. Ces *items*, 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.

Enfin, nos contributions pourraient être analysées par rapport à d'autres critères d'évaluation.

Nos futurs travaux seront marqués par des développements autour de nouveaux processus et indicateurs d'influence de l'E&E tout en continuant à améliorer ceux déjà mis en place. Nous effectuerons également des expérimentations en utilisant d'autres MH pour davantage soutenir les résultats obtenus. De plus, nous avons l'intention de mener des expérimentations plus approfondies et diversifiées dans le cadre de la recommandation pour étudier en profondeur les caractéristiques des recommandations fournies. Enfin, nous poursuivrons nos travaux sur l'explicabilité de l'E&E.

Remerciements

Ce travail est réalisé dans le cadre de la convention PEACE avec le Rectorat français de l'Académie de Nancy-Metz ainsi qu'avec le Ministère de l'Éducation Nationale et de la Jeunesse.

Références

- [1] Anne Auger, Marc Schoenauer, and 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*, pages 857–864, 2005.
- [2] Daniel Balouek and al. Adding virtualization capabilities to the Grid'5000 testbed. In *Cloud Computing and Services Science*, volume 367, pages 3–20. Springer International Publishing, 2013.
- [3] Jie Chen, Bin Xin, Zhihong Peng, Lihua Dou, and Juan Zhang. Optimal contraction theorem for exploration–exploitation tradeoff in search and optimization. *IEEE Transactions on Systems, Man, and Cybernetics-Part A : Systems and Humans*, 39(3) :680–691, 2009.
- [4] Minmin Chen. Exploration in recommender systems. In *Fifteenth ACM Conference on Recommender Systems*, pages 551–553, Amsterdam Netherlands, 2021. ACM.
- [5] Jonathan D Cohen, Samuel M McClure, and Angela J Yu. Should i stay or should i go? how the human brain manages the trade-off between exploitation and exploration. *Philosophical Transactions of the Royal*

- Society B : Biological Sciences*, 362(1481) :933–942, 2007.
- [6] Matej Crepinsek, Shih-hsi Liu, and Marjan Merinik. Exploration and exploitation in evolutionary algorithms : A survey. *ACM Computing Surveys*, 45(35) :1–33, 06 2013.
- [7] Kenneth A De Jong. Are genetic algorithms function optimizers? In *PPSN*, volume 2, pages 3–14, 1992.
- [8] Kenneth A De Jong and William M Spears. A formal analysis of the role of multi-point crossover in genetic algorithms. *Annals of mathematics and Artificial intelligence*, 5(1) :1–26, 1992.
- [9] Sachin Desale, Akhtar Rasool, Sushil Andhale, and Priti Rane. Heuristic and meta-heuristic algorithms and their relevance to the real world : a survey. *Int. J. Comput. Eng. Res. Trends*, 351(5) :2349–7084, 2015.
- [10] Tansel Dokeroglu, Ender Sevinc, Tayfun Kucukyilmaz, and Ahmet Cosar. A survey on new generation metaheuristic algorithms. *C&IE*, 2019.
- [11] Agoston E Eiben and Cornelis A Schippers. On evolutionary exploration and exploitation. *Fundamenta Informaticae*, 35(1-4) :35–50, 1998.
- [12] Fred W Glover and Gary A Kochenberger. *Handbook of metaheuristics*, volume 57. Springer Science & Business Media, Dordrecht, 2006.
- [13] Georges R Harik, Fernando G Lobo, et al. A parameter-less genetic algorithm. In *GECCO*, volume 99, pages 258–267, 1999.
- [14] F Maxwell Harper and Joseph A Konstan. The movielens datasets : History and context. *ACM tiis*, 5(4) :1–19, 2015.
- [15] Thomas T Hills, Peter M Todd, David Lazer, A David Redish, Iain D Couzin, Cognitive Search Research Group, et al. Exploration versus exploitation in space, mind, and society. *Trends in cognitive sciences*, 19(1) :46–54, 2015.
- [16] Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Adefowoke Ojokoh. Recommendation systems : Principles, methods and evaluation. *Egyptian informatics journal*, 16(3) :261–273, 2015.
- [17] Marius Kaminskas and Derek Bridge. Diversity, serendipity, novelty, and coverage : A survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM TiiS*, 7, 1 :2, 2017.
- [18] Rahul Katarya. Movie recommender system with metaheuristic artificial bee. *Neural Computing and Applications*, 30(6) :1983–1990, 2018.
- [19] Manuel Lozano and Carlos García-Martínez. Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification : Overview and progress report. *C&OR*, 37(3) :481–497, 2010.
- [20] Silvano Martello and Paolo Toth. *Knapsack problems : algorithms and computer implementations*. John Wiley & Sons, Inc., Hoboken, New Jersey, US, 1990.
- [21] Amy McGovern, Ryan Lagerquist, David John Gagne, G Eli Jergensen, Kimberly L Elmore, Cameron R Homeyer, and Travis Smith. Making the black box more transparent : Understanding the physical implications of machine learning. *Bulletin of the American Meteorological Society*, 100(11) :2175–2199, 2019.
- [22] K. Mehlhorn, B. R. Newell, P. M. Todd, M. D. Lee, K. Morgan, V. A. Braithwaite, D. Hausmann, K. Fiedler, and C. Gonzalez. Unpacking the exploration-exploitation tradeoff : A synthesis of human and animal literatures. *Decision*, 2(3) :191–215, 2015.
- [23] Zbigniew Michalewicz and Zbigniew Michalewicz. *Genetic algorithms+ data structures= evolution programs*. Springer Science & Business Media, 1996.
- [24] Arthur Pchelkin. Efficient exploration in reinforcement learning based on utile suffix memory. *Informatica*, 14(2) :237–250, 2003.
- [25] Ke Sang, Peter Todd, and Robert Goldstone. Learning near-optimal search in a minimal explore/exploit task. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33, 2011.
- [26] Ngo Tung Son, Jafreezal Jaafar, Izzatdin Abdul Aziz, and Bui Ngoc Anh. Meta-heuristic algorithms for learning path recommender at mooc. *IEEE Access*, 9 :59093–59107, 2021.
- [27] Özge Sürer, Robin Burke, and Edward C Malthouse. Multistakeholder recommendation with provider constraints. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 54–62, Vancouver British Columbia Canada, 2018. ACM.
- [28] Maritzol Tenemaza, Sergio Lujan-Mora, Angelica De Antonio, and Jaime Ramirez. Improving itinerary recommendations for tourists through metaheuristic algorithms : an optimization proposal. *IEEE Access*, 8 :79003–79023, 2020.
- [29] Junqin Xu and Jihui Zhang. Exploration-exploitation tradeoffs in metaheuristics : Survey and analysis. In *Proceedings of the 33rd Chinese control conference*, pages 8633–8638, Nanjing, China, 2014. IEEE, IEEE.
- [30] Gary Yen, Fengming Yang, and Travis Hickey. Coordination of exploration and exploitation in a dynamic environment. *International Journal of Smart Engineering System Design*, 4(3) :177–182, 2002.
- [31] Jun Zhang, Wei-Neng Chen, Zhi-Hui Zhan, Wei-Jie Yu, Yuan-Long Li, Ni Chen, and Qi Zhou. A survey on algorithm adaptation in evolutionary computation. *Frontiers of Electrical and Electronic Engineering*, 7(1) :16–31, 2012.
- [32] Yong Zheng and Aviana Pu. Utility-based multi-stakeholder recommendations by multi-objective optimization. In *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 128–135, Santiago, Chile, 2018. IEEE, IEEE.