
Rapport de Stage

Débat basé sur la théorie de la décision*

Auteur :
Tony SEGUIN

Encadrants :
Olivier CAILLOUX
Meltem OZTÜRK



Résumé

Ce rapport introduit un système de recommandation par le biais de deux approches de système de recommandation, que nous avons sélectionné, qui débattent sur un problème de décision donné. L'objectif est que le débat entre ces deux approches forme un graphe d'argument qui, en fonction de la perspective du décideur, permettrait d'aboutir à une décision. Dans ce rapport nous établissons (actuellement) un état de l'art sur les systèmes de recommandation, et nous exposons notre recherche d'approches dans lequel nous en sélectionneront deux pour effectuer un débat entre leurs deux systèmes de recommandation, ensuite nous présentons, manière détaillées, les approches retenues.

Table des matières

1	Introduction	2
2	État de l'art	3
2.1	Deux grandes familles de système de recommandation	3
2.2	Argumentation dans des systèmes de recommandation	6
2.3	Objectifs des argumentations des systèmes de recommandation	9
2.4	Élargissement de l'état de l'art : choix des approches pour le débat	10
2.4.1	Critères de sélection	10
2.4.2	Recherche	10
2.4.3	Présentations des approches sélectionnées	12
3	Les 2 Approches retenues	15
3.1	Notations	15
3.1.1	Connaissances	15
3.1.2	Arguments et relations	16
3.2	Approche Labreuche	17
3.2.1	Modèle	17
3.2.2	Arguments	18
3.3	Approche Nunes et al	21
3.3.1	Modèle	21
3.3.2	Sélection des paramètres pour compléter une explication	22
4	Travaux à venir	28
5	Annexe A : Développement de l'approche de Labreuche	29

1 Introduction

Une forme de problème d'aide à la décision consiste à formaliser et déterminer les préférences d'un décideur dans le but de fournir une alternative recommandable par rapport à un ensemble d'alternatives possibles. En fonction du contexte du problème, fournir juste une recommandation à l'aide d'un modèle de décision n'est pas forcément suffisante, les décideurs peuvent avoir besoin d'explication, sur comment la recommandation a été faite et pourquoi elle est la meilleure pour eux. En effet, l'argumentation, la justification, d'une recommandation est une part importante de la décision. Construire une explication compréhensible et convaincante est nécessaire pour un décideur. Dans le cas des systèmes de recommandation (SR) en ligne, il a été montré que les explications augmentent l'acceptabilité d'une alternative recommandée [Pu and Chen, 2007]. Dans ce contexte-ci, une explication doit être simple et complète pour installer une confiance au décideur qu'une alternative recommandée est celle qui lui convient le mieux.

La théorie de la décision propose des modèles capturant les préférences du décideur en vue de l'aider à éclaircir son problème de décision. Nous nous basons sur un article qui considère un problème de décision comme un débat [Cailloux and Meinard, 2018]), où deux *SR* argumenteraient chacun leur tour à propos de la décision sur le problème. Cette argumentation entre les systèmes permettrait de générer un graphe d'arguments et en fonction de la préférence du décideur, après avoir pris en compte tous ces arguments [Dung, 1995], la décision sera prise. Nous nous plaçons dans le cadre des systèmes d'Aide Multi-Critère à la Décision (AMCD). Le premier objectif du stage est donc de trouver des *SR* qui génèrent une *argumentation* sur la base de modèle de préférence. Une recherche sera entreprise, avec comme point de départ une *review* [Nunes and Jannach, 2017]. Après avoir mis en évidence un certain nombre d'approches générant des arguments, nous devrons en sélectionner deux qui pourront être adaptées avec notre système afin de générer des arguments et des contre-arguments basés sur leurs modèles d'argumentation. Le but est de faire argumenter ces deux approches entre elles pour générer une explication. Ce qui implique la nécessité d'établir un *langage commun* entre les *SR* et compréhensible pour le décideur. L'approche par le biais d'un débat entre deux *SR* nous semble être un bon compromis entre une explication simple et complète. En effet quoi de mieux qu'un débat quasi-naturel pour convaincre une personne que ce que l'on énonce est vrai ou non ? C'est pour cela que notre approche se penche sur cette problématique d'argumentation pour la recommandation avec cette nuance qu'est le débat.

Ce rapport est divisé (actuellement) en 2 parties. Dans un premier temps, Section 2, un état de l'art sur les systèmes de recommandation et d'aide à la décision, basé sur la review de Nunes and Jannach [2017], est effectué afin d'avoir une idée assez large des approches que l'on pourrait utiliser dans le cadre du projet. Une recherche d'approche compatible supplémentaire a été effectuée sur la base de critères de sélection que nous avons défini, permettant de lister des approches compatibles. Et finalement, Section 3 (en cours de rédaction), nous présentons en détail les deux approches sélectionnées pour notre débat.

2 État de l’art

Dans cette section nous allons décrire ce qui a été entrepris dans le domaine des systèmes d’aide à la décision et de l’argumentation d’après la *review* de [Nunes and Jannach, 2017]. Les graphiques et données dans cette section ont pour source cet article uniquement. Nous établissons d’abord une vision large du domaine, sur les quatre décennies précédentes, ensuite nous discuterons autour des caractéristiques d’une explication en général, les méthodes de génération, le contenu, la manière dont elle est fournie au décideur et les objectifs des systèmes de recommandation. Finalement nous exposerons notre recherche d’approches compatibles pour le débat que nous voulons entreprendre et finirons par une courte présentation de chaque approche.

2.1 Deux grandes familles de système de recommandation

Dans la *review*, les articles étudiés sont classés en quatre catégories : (i) *Technique*, pour les articles générant une nouvelle forme d’explication (ii) *Tool*, les articles décrivant un outil incluant un processus d’explication (iii) *Evaluation*, les articles évaluant ou comparant des formes d’explication et (iv) *Foundational*, les articles discutant autour des aspects des explications.



FIGURE 1: Nombre d’articles par catégorie par décennie. La dernière décennie correspond à 2010 - 12 Août 2016.

D’après la FIGURE 1, le nombre de publications sur le sujet par décennie augmente constamment. Les articles d’outils étaient plus communs dans le passé, selon les auteurs ce genre d’article était peut-être d’avantage considéré comme une contribution dans la recherche durant cette période. En revanche il y a une augmentation du nombre d’articles qui introduisent des nouvelles techniques d’explication tout au long des décennies. Les articles d’évaluations d’explications sont plus nombreux ces dernières années, d’après les auteurs cela serait dû à une amélioration de la maturité par rapport à la communauté en matière de méthodologie de recherche. Et nous remarquons très peu d’articles fondamentaux sur le sujet.

Entre les années 90 et 2000 une stagnation du nombre de publications est remarquée dû au rôle déclinant des systèmes basés sur les connaissances (principalement ceux basés sur des règles)

et à l'engouement pour le *Machine Learning* (ML) dans le domaine des systèmes de recommandation, durant les années 2000, qui ne cherchait qu'à déterminer les bonnes recommandations au détriment de fournir une explication. Cela s'explique car la *review* ne s'intéresse qu'aux articles où la notion d'explication est précisée.

Ces remarques nous montrent les principales catégories dans les systèmes de recommandation, à savoir :

Les systèmes basés sur le Machine Learning qui sont essentiellement appliqués au commerce en ligne (ex : Amazon, Google) et aux multimédias (ex : Netflix, Steam, Facebook). Ils se décomposent en 3 sous-catégories :

Basé sur le *contenu* («Content-based»)

Ces approches sont basées sur le contenu d'un profil d'un décideur, construit à l'aide des données sur celui-ci, soit par interaction avec le décideur soit par apprentissage des informations sur ses dernières actions (ex : ses derniers achats en ligne). Le mécanisme de cette approche est de recommander une alternative *similaire* à une des alternatives contenue dans son profil, c'est-à-dire que le système cherche à trouver une paire d'alternatives (a, b) , où a est dans le profil du décideur et b dans un voisinage proche. Cela se fait à l'aide d'une *fonction de similarité*, il en existe plusieurs, par exemple *Normalised Google Distance* qui détermine la similarité entre des termes textuels en utilisant leurs co-occurrences sur des sites internet :

$$d(a, b) = \frac{\max\{\log f(a), \log f(b)\} - \log f(a, b)}{\log M - \min\{\log f(a), \log f(b)\}} \quad (1)$$

Où M est le nombre total de page Google cherchées, $f(a)$ et $f(b)$ sont respectivement le nombre d'occurrence de a et de b , et $f(a, b)$ est le nombre de co-occurrence de a et b .

Filtrage Collaboratif («Collaborative filtering»)

On note D l'ensemble des décideurs, X l'ensemble des alternatives, $V_d \subseteq X$ pour l'ensemble des alternatives que le décideur d a déjà noté, et R l'ensemble des notes r_{dx} des décideurs $d \in D$ sur les alternatives $x \in V_d$. Dans la base de données de *Netflix* par exemple, les notes sont des entiers compris entre 1 et 5 pour signifier respectivement avoir aimé ou détesté le film.

Le but des approches par filtrage collaboratif est d'être capable de *prédire* la note p_{dx} qu'un décideur d donnera à une alternative x . Le décideur d est supposé actif, c'est-à-dire qu'il a déjà noté un certain nombre d'alternatives, donc $V_d \neq \emptyset$, et que l'alternative x prédite n'est pas encore connue du décideur, $x \notin V_d$.

Ce genre d'approche a plusieurs méthodes de prédiction possible, l'une des plus utilisées est celle basée sur les profils des décideurs à l'aide d'une fonction de *similarité* $\text{sim}(d, d')$ déterminant la similarité entre deux décideurs d et d' . Ceci permet de créer un ensemble de taille K de voisins de d , noté T_d , qui sont les K décideurs qui maximisent leurs similarités avec le décideur d . Une prédiction possible pour la note du décideur d sur l'alternative x est d'utiliser une somme pondérée des notes des plus proches voisins $d' \in T_d$ qui ont déjà noté l'alternative x :

$$p_{dx} = \frac{\sum_{\{d' \in T_d | x \in V_{d'}\}} \text{sim}(d, d') \times r_{d'x}}{\sum_{\{d' \in T_d | x \in V_{d'}\}} |\text{sim}(d, d')|} \quad (2)$$

Hybride («Hybrid»)

Ces approches combinent au moins deux techniques de recommandation dans le but d'obtenir de meilleure performance avec moins de retour. La plus commune est une approche de filtrage collaboratif combinée avec d'autres techniques dans le but d'éviter le problème

de démarrage à froid. Ce problème se rencontre lorsque le système n'a pas assez d'information sur le profil du décideur. Un exemple d'approche *hybride* est la combinaison d'une approche basée sur le contenu qui force les alternatives à être, en même temps, proche du profil du décideur, et notées par les décideurs du voisinage du décideur [Balabanović and Shoham, 1997].

Les systèmes basés sur les connaissances («Knowledge-based») qui utilisent des connaissances explicites sur l'ensemble des alternatives X et les préférences du décideur, par exemple l'importance des critères ou des contraintes fixées. Ces approches s'appliquent sur des problèmes de décision, par exemple un décideur possède une liste de choix et il ne sait pas quel choix prendre. On dénombre un certain nombre de sous-catégories.

Basé sur des règles («Rule-based»)

Ces approches s'appuient sur les connaissances d'un expert d'un domaine, ses connaissances sont exprimées sous la forme de règles. En appliquant les données d'entrée du problème, des règles seront activées et chaque règle fournit une argumentation. Une règle est de la forme if-then, c'est-à-dire qu'un ou plusieurs prérequis sont nécessaires pour activer la règle pour donner un argument. Par exemple dans Wick and Slagle [1989] une règle est :

IF high water in the reservoir
THEN cracks in the interface
BECAUSE high water in the reservoir causes cracks in the interface.

Où la clause **BECAUSE** est la justification de la recommandation **THEN** par rapport à la condition **IF**.

Basé sur la logique («Logic-based»)

Ce genre d'approche est principalement liée à la *Logique Floue (Fuzzy Logic)* [Zadeh, 1965]. Elle est basée sur des *degrés de vérité* plutôt que sur la logique booléenne «vrai ou faux» (0 ou 1).

La *Logique Floue* inclut 0 et 1 comme des cas extrêmes de vérité mais inclut une variété de cas intermédiaires de vérité entre eux. Par exemple la vitesse normale est de 90km/h sur les routes nationales françaises. La vitesse est considérée comme élevée au-dessus de 100km/h et réglementaire en dessous de 80km/h. Si nous nous plaçons dans le contexte où on veut savoir si un véhicule a une vitesse élevée, on obtient qu'un véhicule n'ait pas de vitesse élevée en dessous de 80km/h (0), qu'il ait une vitesse élevée à partir de 100km/h (1), mais s'il est par exemple à 90km/h on dit que la véhicule à une vitesse élevée à 50% (0.5).

Basé sur un raisonnement par cas («Case-based reasoning»)

Le raisonnement par cas, [Riesbeck and Schank, 1989], est le processus de résolution de nouveaux problèmes basés sur les solutions d'anciens problèmes similaires. Autrement dit, ces approches s'appuient sur d'anciennes recommandations possédant un contexte (préférence du décideur) *similaire* au nouveau problème. Par exemple en médecine, si un patient p possède les mêmes symptômes qu'un ancien patient p' , le médecin va prescrire le traitement x qu'il avait prescrit au patient p' . Dans [Mcsherry, 2005], la fonction de similarité entre deux cas $c \in C$, C étant l'ensemble des cas, et q un sous-ensemble de critères N_Q est de la forme :

$$sim(c, q) = \sum_{n \in N_q} w_n sim_n(c, q) \quad (3)$$

où $sim_n(c, q)$ est la similarité locale entre c et q sur le critère n . Cette fonction permet d'établir le voisinage de q et ainsi de déterminer les cas similaires à q .

Prise de décision multi-critère («Multi Criteria Decision Making»)

Selon Triantaphyllou [2000] *Multi-Criteria Decision Making (MCDM)* et *Multi-Attribute Decision Making (MADM)* représentent la même catégorie, se regroupant sous la bannière *MCDM*. Les approches se basant sur des modèles *MCDM*, s'appuient essentiellement sur deux informations : l'ensemble des alternatives X décrites sur un ensemble de critères N et le vecteur w représentant l'importance de chaque critère. Il existe plusieurs méthodes pour déterminer l'alternative qui sera la plus à même de satisfaire le décideur, comme *analytic hierarchy process* (AHP) [Saaty, 2008] ou *weighted product model*, mais la plus connue étant la méthode *Multi-Attribute Value Theory* (MAVT) [Keeney and Raiffa, 1993] :

$$X^* = \max_{x \in X} \sum_{i=1}^n w_i \times v_i x_i. \quad (4)$$

où X^* est le score de la meilleure alternative et $n = |N|$.

2.2 Argumentation dans des systèmes de recommandation

Les systèmes d'aide à la décision étaient plus focalisés sur le fait de déterminer la meilleure recommandation plutôt que l'apport d'explications. C'est à partir de l'engouement pour le *ML*, voir Section 3.3, et les débuts du domaine des *Multi-Criteria Decision Analyses* que l'argumentation a pris plus de considération.

Méthode de génération

Les articles cités dans la *review* donnent peu de détails à propos du processus de génération d'explications. Ceci s'explique car le processus est étroitement lié avec la méthode d'inférence de décision et les données utilisées pour déterminer la meilleure alternative. Si la méthode d'inférence est basée sur des règles, l'explication fournie au décideur consistera en un ensemble de représentations de langage naturel des règles qui ont été activées.

Avec l'engouement historique des explications des systèmes experts à base de règles, il est normal que la majorité des études adoptent une approche basée sur les connaissances pour la décision par inférence et par extension pour la génération des explications. On remarque à nouveau la déclinaison des systèmes à base de règles au fil du temps et un engouement croissant pour les approches basées sur le *ML*. Les explications par les systèmes de filtrage collaboratif sont principalement étudiées depuis le début des années 2000, dû à la forte croissance du nombre et des formes de données, ces approches sont plus adaptées.

La catégorie *Autres* représente les formes d'approches alternatives par exemple l'utilisation d'heuristique spéciale ou d'ontologies (Ontology Web Language) et les approches qui utilisent un raisonnement basé sur les connaissances sans fournir plus de détails sur leur fonctionnement.

Contenu

Une explication doit fournir une information dépendant de divers facteurs, incluant les expertises ou les intérêts du décideur voire leur situation contextuelle courante. On distingue quatre grandes parties de contenu, répertoriées dans la Figure 3 :

Basé sur les connaissances



Machine Learning



FIGURE 2: Occurrence des types de méthodes utilisées par décennie



FIGURE 3: Occurrence du type de contenu utilisé dans une argumentation. Une explication d'un article peut fournir plusieurs types de contenus.

Préférence du décideur : Une manière pour expliquer la suggestion faite par le système est d'utiliser les entrées fournies par le décideur comme explication, c'est-à-dire fournir : (1) quelles entrées sont les plus décisives par rapport à la recommandation. (2) quelles contraintes sont respectées et lesquelles ne le sont pas (3) les critères importants (par une analyse de sensibilité par exemple) (4) et dans quelle mesure l'alternative recommandée est appropriée. Ce qu'on voit dans la Figure 3, c'est que ce sont les entrées décisives du décideur qui sont le plus souvent utilisées dans cette partie, ainsi que le respect de ses préférences.

Processus de décision par inférence : Fournir une information à propos du processus itératif d'un problème de décision spécifique (une trace) est l'approche la plus commune. Ceci s'explique par rapport aux systèmes à base de règles qui fournissaient la liste des règles activées comme argumentation. Quelques explications fournissent seulement la logique générale du processus d'inférence interne du système, c'est-à-dire le fonctionnement interne de l'algorithme. D'autres fournissent la confiance du système sur sa suggestion ou taux de succès sur les situations de prise de décision passées.

Contexte et informations complémentaires : Peu d'explications fournissent des informations contextuelles supplémentaires qui sont spécifiques à la prise de décision courante. Divers types de contexte et informations complémentaires sont relevés, par exemple les tendances populaires. Ces types de contenu sont utilisés principalement par les approches en *ML*.

Alternatives et critères : Les approches *MCDM* en revanche expliquent la suggestion du système en analysant les caractéristiques des alternatives (chaque critère est étudié pour chaque proposition possible). L'argumentation derrière consiste en une liste de caractéristiques, le compromis (le pour et le contre), pour chaque alternative. D'autres réfèrent les relations de dominance basées sur les caractéristiques, mais la majorité des explications montrent quels critères sont décisifs durant le processus de recommandation.

Présentation

Majoritairement, les explications sont présentées sous forme de langage naturel, comprenant celles basées sur des canevas, ou «texte à trous», qui sont par exemple instanciées avec des listes d'arguments avant d'être soumis au décideur. Un exemple de canevas serait «*Il n'y a aucune raison de choisir <alternative 1> , alors que <alternative 2> est meilleure sur tous les critères.*», ceci résumant la domination de l'alternative 2 sur l'alternative 1. Il existe aussi la présence assez forte d'utilisation de formes de visualisation (graphe, arbre, etc.), pour imager et ainsi améliorer l'argumentation fournie au décideur [Grando et al., 2011, Narayanan and McGuinness, 2008].

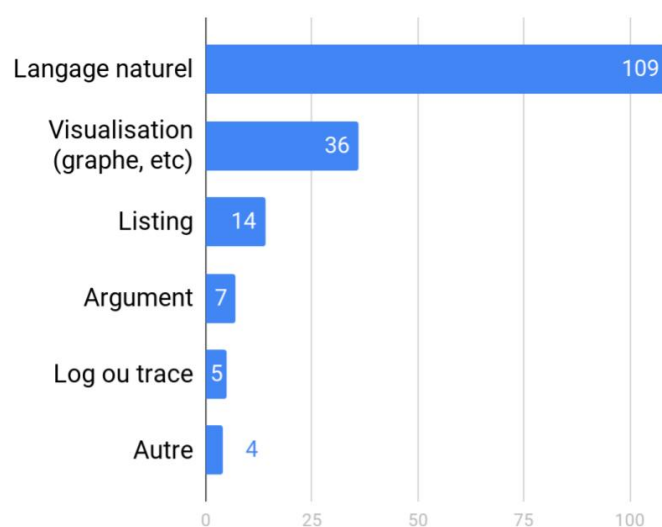


FIGURE 4: Occurrence des manières de présenter une explication

D'une tout autre manière, un certain nombre d'approches donnent un listing de différents items pour l'argumentation comme par exemple une liste des contraintes satisfaites (Deep et al, 1988) ou une liste de relations de préférence sur certains critères entre deux alternatives [Labreuche et al., 2015]. Les 7 articles, catégorisés dans Argument sont des approches qui utilisent une structure des arguments de Toulmin [2003].

2.3 Objectifs des argumentations des systèmes de recommandation

Il est nécessaire de relever l'importance du but d'une explication [Tintarev and Masthoff, 2007]. D'après les auteurs il faut différencier les buts annoncés des vrais buts. La plupart des études de la *review* ne précisent pas forcément les buts des explications fournies. La Figure 5 se base seulement sur les buts annoncés dans ces articles.

Le premier but recherché le plus commun est la *transparence*, c'est-à-dire expliquer comment le système est parvenu à sa suggestion. Les explications fournies dans ces études se focalisent sur l'exposition du processus d'inférence dans le but de rendre la décision recommandée compréhensible. Il existe d'après eux des liens entre les buts, par exemple, la transparence impliquerait la *confiance* du décideur envers le système de recommandation, ainsi la confiance serait un effet indirect attendu de la transparence. Le deuxième plus fréquent est l'*efficacité*, c'est-à-dire d'aider les décideurs à évaluer si l'alternative recommandée est bien adéquate pour eux. La force de *persuasion*, c'est-à-dire la capacité d'un système à pousser le décideur dans une certaine direction (en conflit avec l'efficacité [Chen and Wang, 2014]) est dans un nombre important d'études de la *review*, principalement les articles de *ML*.

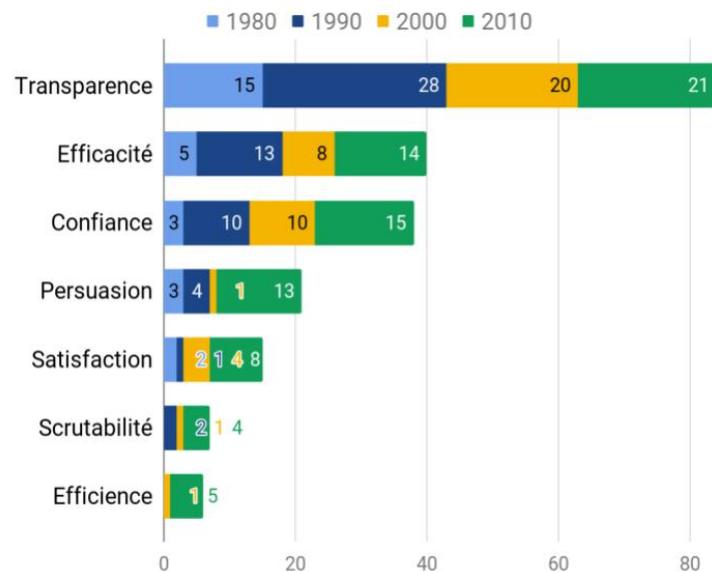


FIGURE 5: Nombre d'articles Technique et Outils par but

Ces dernières années, la *satisfaction* du décideur, la *scrutabilité* (la possibilité de dire au système que c'est faux) et le fait d'aider le décideur à faire un choix *plus rapide*, sont au goût du jour. En effet, réduire la charge cognitive et essayer d'améliorer la satisfaction des décideurs par un système est l'aspect essentiel des applications *e-commerce*, qui est un sujet très recherché ces dernières années. Dans ce contexte, le potentiel persuasif naturel des explications est de grand intérêt dans la recherche.

2.4 Élargissement de l'état de l'art : choix des approches pour le débat

Nous allons maintenant discuter des approches que nous avons sélectionnées. Tout d'abord nous allons préciser quels critères de sélection nous établissons pour les approches compatibles avec notre projet. Nous allons exposer notre recherche d'une part sur la *review* et d'autre part sur élargissement de recherche et nous finissons par un léger résumé de chacune des approches retenues.

2.4.1 Critères de sélection

Dans le cadre du projet, les articles doivent être cohérents entre eux, c'est-à-dire qu'ils adoptent, dans les grandes lignes, des modèles de recommandation et donc d'argumentation compatible entre eux. Il est donc nécessaire que les articles que nous allons sélectionner, pour la recommandation basé sur un débat, soient spécifiques à aucun domaine et par conséquent non contextualisés. Ce qui est notre premier critère de sélection.

Le second critère est sur quelles informations nos approches vont s'appuyer pour argumenter ; comme vu Section 3, les approches basées sur les connaissances permettent une argumentation riche et détaillée. En adoptant un Modèle d'Aide Multi-Critère à la Décision (MAMCD), le compromis, les critères décisifs (pertinents ou non) sont des types d'arguments qui rentrent dans notre schéma de débat. C'est pourquoi nous nous pencherons seulement sur les approches basées sur les connaissances et utilisant un MAMCD.

Et comme dernier critère, seules les approches fournissant une argumentation en faveur d'une seule alternative seront retenues. Ces argumentations peuvent être de type : «A meilleur que B» ou «A meilleur que tous les autres».

2.4.2 Recherche

La recherche des approches a été faite en deux parties, dans un premier temps une sélection des articles étudiés dans une *review* [Nunes and Jannach, 2017] a été réalisée et dans un second temps une recherche pour élargir notre vision d'ensemble et obtenir un nombre assez conséquent d'approches a été effectuée.

Review

Nous avons expliqué que la *review* catégorise les articles en quatre catégories (*Tool*, *Technique*, *Fundamental* et *Evaluation*). Dans un premier temps, seuls les articles *Technique* sont conservés, car ce sont des articles avec des approches non contextualisées (contrairement aux *Tools*) pour la majorité, soit 101 articles. Ensuite nous avons conservé seulement les articles étiquetés par les auteurs comme étant des approches basées sur les connaissances, plus précisément celles utilisant un MAMCD, ce qui nous donne 8 articles. Les résumés et conclusions ont été inspectés, ainsi qu'une lecture transversale de l'ensemble des articles nous a permis de réduire à 5 articles. En effet un article [Bielza et al., 2000] est basé sur un modèle de type Multi Attribute Utility Theory (MAUT), le second article rejeté est contextualisé dans le domaine de la médecine [Bohanec et al., 2000]. Les articles conservés sont listés en Table 1.

Table 1 Articles retenus de la review

Auteurs	Titres
Klein et Shortliffe	A framework for explaining decision-theoretic advice
Carenini et Moore	Generating and evaluating evaluative arguments
Labreuche	A general framework for explaining the results of a multi-attribute preference model
Nunes et al	Pattern-based Explanation for Automated Decisions
Belahcene et al	Explaining robust additive utility models by sequences of preference swaps

Élargissement

Ensuite, une recherche supplémentaire a été exécutée pour élargir le nombre d’approches possibles, nous avons d’abord établi les mots-clés pour la recherche ainsi que leurs synonymes. En effet dans la littérature tout le monde ne s’accorde pas à utiliser les mêmes termes pour désigner la même chose. Le premier terme est *argumentation*, qui est l’ingrédient essentiel de notre recherche, ensuite le deuxième terme est *decision support system* incluant ainsi les systèmes de recommandation et le troisième est *multi-criteria* désignant toutes les variantes telles que *Multi-Criteria Decision Making* par exemple.

Table 2 Termes et synonymes

Termes	Synonymes
argumentation	explanation, justification
decision support system	decision making, recommandation knowledge-based system, knowledge based system
multi-criteria	multi-attribute, multi attribute, multi criteria

La recherche à l’aide de ces mots-clés sur les plus larges bases de données de librairies d’articles en ligne, à savoir : *ACM Digital Library*, *IEEE Xplore Digital Library*, *ScienceDirect* et *Springer Link*, prend forme de la manière suivante :

$$(\text{argumentation} \vee \text{justification} \vee \text{explanation}) \wedge (\text{decision support system} \vee \text{decision making} \vee \text{recommendation} \vee \text{knowledge-based system} \vee \text{knowledge based system}) \wedge (\text{multi-criteria} \vee \text{multi-attribute} \vee \text{multi attribute} \vee \text{multi criteria})$$

Ces termes ont été recherchés dans les titres et les résumés de chaque article des librairies, nous avons obtenu 98 articles, sur l’ensemble des librairies, voir Table 3. Dans un premier temps, les titres et résumés ont été étudiés, pour réaliser un premier filtre, ensuite une lecture plus en détail des articles restants a été entreprise.

Table 3 Résultats de recherche par source

Sources	Nombre d'articles
ACM Digital Library	19
IEEE Xplore Digital Library	8
ScienceDirect	19
Springer Link	52
Total	98

Étonnamment, pour la majorité des cas, les articles ont une approche de type *ML* ou sont des outils pour un problème précis. Toutes ces approches ont été rejetées. Après ce premier filtre nous sommes parvenus à 6 articles restants et après lecture des textes de ces articles, certains ont été rejetés. Par exemple, [Delle Site and Filippi \[2009\]](#), ne fournit pas d'argumentation, [Yevseyeva et al. \[2016\]](#) utilisent des données externes, [Kadziński et al. \[2017\]](#) est une analyse expérimentale d'une fonction additive dans plusieurs méthodes de désagrégation des préférences.

Table 4 Articles retenus de la recherche d'élargissement

Auteurs	Titres
Labreuche and al	A Dialogue Game for Recommendation with Adaptive Preference methods : An experimental analysis
Geldermann	Explanation Systems
Papamichail et French	Explaining and justifying the advice of a decision support system : a natural language generation approach

2.4.3 Présentations des approches sélectionnées

En s'appuyant sur la review de [Nunes and Jannach \[2017\]](#) et l'élargissement présenté précédemment, 8 approches respectent les critères imposés. Une présentation par ordre chronologique de chaque article est établie ci-après.

Klein et Shortliffe

L'approche de [Klein and Shortliffe \[1994\]](#) est l'une des premières approches à fournir une argumentation dans un modèle ADMC, elle pose les bases de cette branche de l'aide à la décision. Les auteurs présentent plusieurs stratégies basées sur la MAVT pour expliquer automatiquement les décisions parmi plusieurs objectifs en conflit. Ils décrivent ces stratégies dans un framework prénommé IVA (Interpretive Value Analysis), dans un cadre large d'explications et d'acquisition de pointe dans des systèmes experts qui modélisent des décisions à forte intensité de compromis, c'est-à-dire l'exposition du pour et du contre d'une alternative. Les concepts d'interprétation jouent le rôle de primitives d'explication dans les stratégies d'IVA et sont également utilisés comme fonctions d'évaluation qui guident la composition des explications. Les stratégies génèrent des comparaisons sommaires de paires particulières d'alternatives en limitant la profondeur et la largeur d'un arbre de valeurs. Les stratégies produisent des comparaisons plus détaillées d'alternatives, fournissant des traces pas à pas de calculs des différences de valeurs multi-attributs.

Papamichail et French

Cette approche décrit une méthode pour générer une explication dans une décision de contexte analytique. D'après [Papamichail and French \[2003\]](#) le point fort de leur approche est le développement d'une librairie de textes planifiés, c'est-à-dire un ensemble de canevas instanciés par les arguments trouvés par le système de recommandation, pour structurer le message soumis au décideur. L'approche est en revanche générique. Le système fournit deux types de rapport : (i) un rapport de comparaison expliquant le raisonnement derrière le classement des alternatives et (ii) un rapport d'analyse de sensibilité fournissant une évaluation globale du modèle de décision et décrit l'effet de la variation d'un paramètre de décision.

L'analyse de variation des paramètres de décision nous révèle un intérêt pour cette approche.

Carenini et Moore

L'article de [Carenini and Moore \[2006\]](#) se veut interdisciplinaire (théorie de l'argumentation, théorie de la décision, linguistique informatique, psychologie sociale et interaction homme-machine). L'approche se focalise principalement sur la génération de l'argumentation en langage naturel. La sélection et l'organisation du contenu de l'argumentation est basée sur les principes de la théorie de l'argumentation.

En se concentrant uniquement sur les parties argumentation et théorie de la décision, l'approche se détache des autres car elle forme un graphe de relations entre les arguments. Ce graphe est utilisé pour générer ensuite une explication en langage naturel, compréhensible humainement, et adaptée aux préférences du décideur.

Geldermann

L'article de [Geldermann \[2010\]](#) fournit une application pour un système d'explication pour des systèmes d'aide à la décision basés sur MAUT et plus spécifiquement sur MAVT. L'argumentation se fait par le biais d'un rapport comparatif, c'est-à-dire qu'elle fournit une interprétation des résultats d'évaluations du modèle en comparant deux alternatives. Le rapport discute de la façon dont une alternative évalue l'autre sur chaque critère d'évaluation, en soulignant les arguments pour et contre de chaque alternative, sur la base des scores des critères réels. Ainsi, il examine à quel point une alternative est meilleure qu'une autre et souligne les facteurs qui les différencient. Un autre type de rapport est fourni, l'analyse de sensibilité, qui explique les graphiques d'analyse de sensibilité et illustre l'effet du changement du poids d'un attribut dans le classement des alternatives et discute de la robustesse de la meilleure alternative.

Cette approche est trop similaire à l'approche de [Papamichail and French \[2003\]](#), si l'une est retenue la seconde ne le sera pas.

Labreuche

Cette approche veut fournir une explication plus simple à comprendre par rapport à l'approche de [Klein and Shortliffe \[1994\]](#). En effet [Labreuche \[2011\]](#) propose une approche pour sélectionner les arguments utilisables dans une explication faite pour un problème de décision multi-critères pondérés par des poids assignés à ces critères. Il se base sur l'analyse des valeurs de ces poids ainsi que le score des alternatives pour les comparer. Un seul modèle utilisé sur les trois de l'article nous intéresse, le modèle Expected Utility (EU), qui est un modèle MAVT. Le but de l'approche est de rechercher certains changements dans le vecteur de poids w qui permet une inversion de la décision fournie par le modèle entre deux alternatives. L'explication se focalise alors sur le ou les critères qui ont été modifiés dans w . Les autres critères ne sont pas mentionnés pour une explication se voulant la plus courte possible. Il y a deux stratégies de modification de

w : (i) le remplacement de w par un autre vecteur de poids de référence w^F et (ii) la permutation des poids de w . La première stratégie permet de mettre évidence les critères importants et non importants, et la seconde permet de déterminer les critères décisifs.

Cet article a pour objectif d’argumenter avec le minimum d’arguments possibles. Sa vision d’argumenter en utilisant les mauvais cotés comme étant compensés par une meilleure partie d’une alternative par rapport à une autre alternative, en fait un bon candidat.

Nunes et al

L’approche de Nunes et al. [2014] est l’une des plus complètes, elle s’appuie sur une génération d’explications basées sur un groupe d’algorithmes pour identifier les paramètres permettant de remplir le canevas d’explications qu’ils fournissent dans l’article. En effet une explication possède 7 formes possibles, s’adaptant ainsi aux différents cas particuliers de décision. Si plusieurs formes d’explications sont possibles pour une alternative recommandée, une ordre pré-établi des formes d’explication est installé pour en choisir qu’une seule. Les principales formes sont : (i) *attribut critique*, l’alternative est recommandée car elle possède la meilleure performance sur un critère donné (ii) *domination*, l’alternative domine sur tous les critères (iii) *critère décisif*, l’alternative est sélectionnée en raison d’un ensemble de critères (iv) *compromis*, l’alternative a un côté avantageux sur un ensemble de critères qui compense ses défauts. Une étude utilisateur a été réalisée incluant une trentaine de participants. Leur approche a été comparée à deux autres approches, [Klein and Shortliffe, 1994, Labreuche, 2011], les résultats indiquant que leurs approches à de meilleures performances.

Cette approche est plus générale que l’approche de Labreuche [2011] par rapport à son éventail de types d’arguments possibles supérieurs et son argumentation de compromis. C’est ce qui nous fait retenir cette approche.

Labreuche and al

L’interaction via un protocole de dialogue, entre un décideur (1) et leur système de recommandation (2), en fait sa particularité, c’est-à-dire que (1) et (2) s’expriment à tour de rôle. Ceci en fait sa particularité. En effet, Labreuche et al. [2015] proposent une méthode où le système s’adapte en fonction des retours du décideur et passe d’un modèle de décision à l’autre en fonction des informations disponibles. Dans l’article seul un modèle nous intéresse, car c’est le seul qui est de MAVT. L’argumentation par contre est sommaire, elle fournit un listing des arguments en faveur d’une alternative par rapport à une autre.

L’approche est intéressante par rapport à son interaction avec le décideur et son protocole de dialogue, mais l’argumentation ainsi que la représentation des connaissances devraient être modifiées si l’approche est retenue.

Belahcene et al

L’approche de Belahcene et al. [2017] fournit une explication dite complète, contrairement à d’autres approches qui ne fournissent que les points-clés décisifs, dans un contexte de décision multi-critères. Les informations initiales prennent une forme de comparaison par paire d’alternatives. La génération d’explications s’inspire d’une méthode «Even Swap», une procédure d’élection utilisant un modèle de valeur additif des valeurs des préférences du décideur et basée sur le *compromis* entre des paires de critères. La version des auteurs est une généralisation d’«Even Swap» appliqué sur l’échange des préférences dans le but de simplement montrer une comparaison d’alternatives. L’explication construit un graphe d’arguments positifs et négatifs entre deux alternatives, pour argumenter pourquoi l’une est préférée à l’autre. La visualisation se fait par un graphe biparties entre les deux types d’arguments.

La génération d'un graphe biparties (argument et contre-argument), prenant en compte tous les critères incluant ceux non décisifs, rend l'approche pertinente. Cette approche se rapproche de celle de [Carenini and Moore \[2006\]](#) en matière de visualisation des graphes.

Après avoir considéré les approches listées ci-dessus, nous avons décidé de sélectionner les approches de [Labreuche \[2011\]](#) et de [Nunes et al. \[2014\]](#).

3 Les 2 Approches retenues

Dans cette section nous allons présenter les deux approches que nous allons utiliser pour notre débat. Dans un premier temps nous présenterons l'approche de [Labreuche \[2011\]](#) et dans un second temps celle de [Nunes et al. \[2014\]](#). Nous verrons en détail la mécanique de ces deux approches.

3.1 Notations

Avant d'entrer dans les détails des approches, il est nécessaire de poser les définitions et les notations que nous allons utiliser tout au long du projet. Nous établissons d'abord les connaissances pour les systèmes de recommandation et ensuite les arguments et les relations entre eux pour le jugement de la décision lors d'un débat.

3.1.1 Connaissances

Nous prenons comme notations des connaissances celles tirées de [Labreuche \[2011\]](#) qui est aussi l'une des approches retenues :

$X = X_1 \times \dots \times X_n$: l'ensemble des alternatives décrit sur N critères, $x \in X$;

N : l'ensemble des critères ;

X_i : l'échelle des performances des alternatives sur le critère i ;

x_i : la performance de l'alternative x sur le critère i ;

$w = (w_1, \dots, w_n)$: le vecteur de poids, $\sum_{i=1}^n w_i = 1$;

w_i : le poids du critère i , $w_i \in [0, 1]$.

En combinant les poids des critères et les performances des alternatives, notre approche s'appuie sur les modèles de décision de type (MAVT) [\[Keeney and Raiffa, 1993\]](#) afin d'attribuer un score global à chaque alternative permettant ainsi de déterminer quelle alternative est la meilleure en fonction des préférences du décideur. L'approche de [Labreuche \[2011\]](#) utilise une fonction de valeurs, en revanche l'approche de [Nunes et al. \[2014\]](#) utilise une fonction d'utilité. En général, la fonction de décision associée au modèle est de la forme, pour $x \in X$,

$$v(x) = \sum_{i=1}^n w_i \times v_i(x_i),$$

où $v_i(x_i)$ représente l'utilité de x sur le critère i . On note $x_i \succsim_i y_i \Leftrightarrow v_i(x_i) \geq v_i(y_i)$ et $v(x) \rightarrow [0, 1]$. Cette fonction v quantifie la relation de préférence \succsim , c'est-à-dire $\forall x, y \in X$, $x \succsim y \Leftrightarrow v(x) \geq v(y)$.

La sélection d'une alternative par rapport aux autres s'établit par la relation binaire \succsim_i sur chaque ensemble X_i , représentant ainsi les préférences du décideur sur les éléments de X_i . On note \succ_i et \sim_i comme les parties asymétrique et symétrique de \succsim_i , et nous notons :

$x_i \succsim_i y_i$: x est au moins aussi bon que y sur le critère i .

Nous disons qu'une alternative x *domine* une alternative y si et seulement si, $\forall i \ x_i \succ y_i$. Nous disons aussi qu'une alternative est *dominante*, si et seulement si, aucune autre alternative ne la domine.

3.1.2 Arguments et relations

Les relations entre arguments suivent la notation et les définitions suivantes [Cailloux and Meinard, 2018] :

- S : l'ensemble des arguments ;
- $S^+(x, y)$: ensemble des arguments en faveur de x par rapport à y ;
- $S^-(x, y)$: ensemble des contre-arguments ;
- $S^=(x, y)$: ensemble des arguments neutres ;

Ces arguments permettent de mettre en évidence les phases de compromis pour une alternative qui n'est pas dominante. Les trois derniers ensembles d'arguments sont inclus dans l'ensemble S .

- T : ensemble des propositions possibles ;
- P : ensemble des perspectives.

Dans notre cas, les propositions possibles sont $\{\forall x, y \in X, t_{x \geq y}\}$, représentant ainsi chaque paire possible de comparaison d'une alternative sur une autre. Une perspective $p \in P$ représente le point de vue du décideur, qui peut changer au cours du débat s'il a été convaincu d'un argument nouveau. C'est-à-dire qu'un argument s peut être valide selon une perspective p mais non valide sur une perspective p' . Les ensembles P , T et S déterminent la position argumentative du décideur. Elle est représentée via les relations suivantes :

- $\rightsquigarrow \subseteq S \times T$: on note $s \rightsquigarrow t$, l'argument s *soutient* la proposition t ;
- $\triangleright_{\exists} \subseteq S \times S$: on note $s_2 \triangleright_{\exists} s_1$, l'argument s_2 *attaque* l'argument s_1 , s'il l'attaque dans au moins une perspective, s_1 devient un argument invalide ;
- $\not\triangleright_{\exists} \subseteq S \times S$: on note $s_2 \not\triangleright_{\exists} s_1$, l'argument s_2 n'attaque pas l'argument s_1 , s_1 reste un argument valide.

Un argument $s \in S$ peut soutenir plusieurs propositions comme aucune. On admet $\neg(a_2 \triangleright_{\exists} a_1) \Rightarrow a_2 \not\triangleright_{\exists} a_1$. Si le décideur change d'avis, les relations $(\triangleright_{\exists}, \not\triangleright_{\exists})$ suffisent à capturer ce changement.

Définition 1 Une situation de décision est définie par le tuple $(T, S, \rightsquigarrow, \triangleright_{\exists}, \not\triangleright_{\exists})$.

- \triangleright_{\forall} : défini comme $a_2 \triangleright_{\forall} a_1 \Leftrightarrow \neg(a_2 \not\triangleright_{\exists} a_1)$;
- $\not\triangleright_{\forall}$: défini comme $a_2 \not\triangleright_{\forall} a_1 \Leftrightarrow \neg(a_2 \triangleright_{\exists} a_1)$.

La relation $a' \triangleright_V a$ signifie que l'argument a' attaque l'argument a sur toutes les perspectives. En revanche la relation $a' \not\triangleright_V a$ signifie que a' n'attaque jamais a .

Définition 2 Étant donné une situation de décision $(T, S, \rightsquigarrow, \triangleright_\exists, \not\triangleright_\exists)$, un argument $s \in S$ est décisif, si et seulement si, $\forall s' \in S : s' \not\triangleright_V s$.

Définition 3 Étant donné une situation de décision $(T, S, \rightsquigarrow, \triangleright_\exists, \not\triangleright_\exists)$, une proposition t est :

- acceptable si et seulement si, $\exists s \in S \mid ss \rightsquigarrow t, \forall s' : s' \not\triangleright_V s$;
- rejetable si et seulement si, $\forall s \in S \mid s \rightsquigarrow t, \exists s_c \mid s_c \triangleright_V s$ et $\forall s_{cc} : s_{cc} \not\triangleright_V s_c$.

Définition 4 Une situation de décision $(T, S, \rightsquigarrow, \triangleright_\exists, \not\triangleright_\exists)$ est claire, si et seulement si, chaque proposition dans T est acceptable ou rejetable.

Définition 5 Le jugement délibéré du décideur correspondant à la situation de décision $(T, S, \rightsquigarrow, \triangleright_\exists, \not\triangleright_\exists)$ est :

$$T_{decideur} = \{ t \in T \mid t \text{ est acceptable} \}.$$

3.2 Approche Labreuche

L'approche se veut de fournir une explication la plus courte possible de la décision $x \succ y$ en fonction du vecteur de poids w , on note cette décision aussi $x \succ_w y$. Labreuche [2011] distingue quatre formes d'explications à fournir au décideur pour justifier la décision. Chaque forme d'explication cherche la coalition $C \in N$ minimale permettant par extension une explication minimale. Nous présenterons dans un premier temps le modèle utilisé, dans un deuxième temps, les différents arguments possibles dans les explications fournies et leurs générations et finalement l'étude expérimentale faite par Labreuche [2011].

3.2.1 Modèle

En fonction d'un ensemble d'alternatives X , d'un ensemble de critères N et d'un vecteur de poids w , Labreuche [2011] s'appuie sur l'importance des critères $i \in N$ et par extension leurs poids w_i dans la décision. Son approche s'applique sur 3 modèles, un modèle *AMCD*, un modèle pour la *Décision dans l'Incertain* et finalement un modèle dans le domaine du *Choix Social*. Nous avons décidé de nous placer dans un modèle de *MAVT* donc nous aborderons uniquement le modèle *AMCD*. La représentation des connaissances et des préférences suit ce qui a été défini en Section 3.1.2. A l'aide de la relation de préférence \succsim trois types d'éléments d'argumentation se définissent de la manière suivante :

$$\begin{aligned} N^+(x, y) &= \{i \in N, x_i \succ_i y_i\} \\ N^-(x, y) &= \{i \in N, x_i \prec_i y_i\} \\ N^=(x, y) &= \{i \in N, x_i \sim_i y_i\} \end{aligned}$$

Ces ensembles d'éléments seront ensuite utilisés pour générer une argumentation $a \in S^+(x, y)$. Du point de vue de l'approche de Labreuche [2011] les arguments sont les critères. De notre point de vue, ces critères sont des éléments d'argumentation qui seront utilisés pour former un argument. Pour des soucis de compréhension on parlera d'argument dans cette sous-section.

Le but est donc de fournir, pour une décision, parmi l'ensemble des explications possibles, l'explication minimale qui permet au décideur de comprendre la décision. L'explication est une coalition C qui permet de remplir un canevas en fonction de la forme d'explication appliquée.

On note aussi, $(w_C, w'_{N \setminus C})$: le vecteur de poids composé tel que $(w_C, w'_{N \setminus C})_i = w_i$ si $i \in C$ et w'_i sinon. L'indice de w' est omis par moment.

3.2.2 Arguments

On dénombre quatre types d'explications en fonction des quatre points d'ancrages $\psi \in \Psi$, définis par Labreuche [2011] : $\psi_{ALL}, \psi_{NOA}, \psi_{IVT}, \psi_{RMG}$, dans l'ordre de simplicité. Chaque ancrage est applicable dans certaines conditions. On adopte l'argument correspondant à l'ancrage applicable le plus simple.

On note

$$\Delta \in \mathbb{R}^n : \Delta_i = x_i - y_i, \forall i \in N$$

représentant la différence de performance de x par rapport à y sur l'ensemble des critères.

Dans la suite nous considérons $x \succ_w y$.

Ancrage ψ_{ALL} : "all"

Lorsque l'alternative x est préférée à l'alternative y sur l'ensemble des critères.

$$\psi_{ALL}(x, y) \text{ est vrai, si et seulement si, } \forall i \in N, \Delta_i > 0.$$

Le fait que $x \succ_w y$ est trivial. La seule explication possible est l'ensemble des arguments N et donc $x \succ y$ peu importe le modèle de décision. La phrase suivante est générée pour expliquer la décision.

x est préférée à y car x est meilleure que y sur tous les critères.

Ancrage ψ_{NOA} "not on average"

Avant d'appliquer cet ancrage, si le fait d'utiliser le vecteur poids de référence w^{ref} au lieu de w , inverse la décision ($y \succeq_{w^{ref}} x$). On définit le vecteur de référence w^{ref} comme suit : $\forall i \in N$ $w_i^{ref} = \frac{1}{n}$ avec $n = |N|$.

$$\psi_{NOA}(x, y) \text{ est vrai, si et seulement si, } y \succeq_{w^{ref}} x.$$

Dans cette section nous considérons que $y \succeq_{w^{ref}} x$.

La génération de la coalition C se fait de la manière suivante.

Définissons

$$\phi(i) = (w_i - \frac{1}{n})\Delta_i, \text{ pour tout } i \in N,$$

représentant les contributions marginales de chaque critère i .

Réordonnons les critères en les numérotant $1, 2, 3, \dots$ pour obtenir $\phi(1) \geq \phi(2) \geq \dots$. On veut maintenant obtenir une coalition C aussi petite que possible telle que $x \succ_{(w_C, w^{ref})} y$.

On itère en considérant $C = \emptyset$ puis $C = \{1\}$, $C = \{1, 2\}$, ... Notons que $y \succeq_{(w_\emptyset, w^{ref})} x$ tandis que $x \succ_{(w_N, w^{ref})} y$.

Notons que $i \in C \cap N^+(x, y)$ implique $w_i > \frac{1}{n}$. En effet, $y \succeq_{(w_{C \setminus i}, w^{ref})} x$ et $x \succ_{(w_C, w^{ref})} y$. De même $i \in C \cap N^-(x, y)$ implique $w_i < \frac{1}{n}$.

L'argumentation se fait en deux temps, d'abord fournir $C \cap N^+(x, y)$ qui sont les critères i tel que $w_i > \frac{1}{n}$ donc importants, sur lesquels x est meilleure que y et ensuite $C \cap N^-(x, y)$ qui sont les critères j où $w_j < \frac{1}{n}$ donc moins importants, sur lesquels y est meilleure que x . La phrase suivante est générée pour expliquer la décision.

Même si y est en moyenne meilleure que x , x est préférée à y car x est meilleure que y sur les critères $C \cap N^+(x, y)$ qui sont importants alors que x est pire que y sur les critères $C \cap N^-(x, y)$ qui sont non importants.

Si $\sum_{i \in N \setminus C} w_i \Delta_i > 0$, la phrase suivante peut être ajoutée.

De plus, x est meilleure en moyenne que y sur les autres critères.

Ancrage ψ_{IVT} : "invert"

L'idée est de préciser que la décision ne serait pas la même si une permutation π des poids de w est appliquée sur certains critères. On note $\Pi(C)$ l'ensemble de toutes les permutations sur la coalition $C \subseteq N$.

$\psi_{IVT}(x, y)$ est vrai si et seulement si $\exists \pi \in \Pi(N) \ y \succeq_{w \circ \pi} x$.

Étant donnée une coalition $C \subseteq N$ définissons la permutation $\pi_C \in \Pi_C$ comme la permutation qui affecte les poids de la manière la plus défavorable possible pour x . Ainsi, π_C associe au critère $i \in C$ maximisant la valeur de Δ_i le critère de plus petit poids w_i .

Pour cet ancrage on cherche une explication sous la forme d'un ensemble \mathcal{C} de coalitions disjointes. Étant donné une explication \mathcal{C} on définit la permutation $\pi_{\mathcal{C}}$ comme suit. Sur chaque coalition $C \in \mathcal{C}$ la permutation égale π_C . La permutation $\pi_{\mathcal{C}}$ laisse les critères non présents dans \mathcal{C} inchangés.

Les explications candidates sont toutes les explications \mathcal{C} telles que $\pi_{\mathcal{C}}$ inverse la décision : \mathcal{C} est une explication candidate si et seulement si $y \succeq_{w \circ \pi_{\mathcal{C}}} x$.

Nous cherchons en outre une explication minimale dans le sens suivant. Étant donnée une explication \mathcal{C} , notons D_C la perte endurée par x sur la coalition C en passant de w à $w \circ \pi_C$:

$$D_C = \sum_{i \in C} w_i \Delta_i - \sum_{i \in C} w_{\pi_C(i)} \Delta_i.$$

Étant données deux coalitions $C, C' \subseteq N$ la coalition C est dite meilleure que C' , $C \triangleright C'$, si elle est plus petite ($|C| < |C'|$), ou, si ex-aequo, si $D_C > D_{C'}$. On dit qu'une explication \mathcal{C} est meilleure qu'une explication \mathcal{C}' , $\mathcal{C} \sqsupset \mathcal{C}'$, si la meilleure coalition $C \in \mathcal{C}$ est plus petite que la meilleure coalition $C' \in \mathcal{C}'$, ou, si ex-aequo, la seconde meilleure coalition est plus petite que la seconde meilleure coalition de \mathcal{C}' , ..., ou, après épuisement de ces comparaisons, si \mathcal{C} a un plus petit nombre de coalitions que \mathcal{C}' , ou, si encore ex-aequo, la comparaison procède de la même façon en utilisant les valeurs D_C (une valeur plus grande étant meilleure). Une explication est minimale s'il n'y a pas de meilleure explication candidate.

La détermination d'une explication \mathcal{C} minimale est déterminée par l'algorithme branch-and-bound ALGO présenté ci-après. Nous sommes intéressés, pour construire une explication minimale, par les coalitions C telles que $D_C > 0$ et π_C forme un cycle complet sur C , ces coalitions candidates étant considérées en suivant l'ordre \triangleright et notées T_i ci-dessous, à partir de T_1 .

Dans cet algorithme \mathcal{B} désigne la meilleure explication trouvée pour le moment, et \mathcal{C} désigne la solution en cours de construction. De plus nous définissons une valeur particulière pour \mathcal{B} , **Invalidé**, telle que pour toute explication candidate \mathcal{C} , $\mathcal{C} \sqsubset \text{Invalidé}$.

L'appel de l'algorithme est initialisé avec $\mathcal{C} = \emptyset$, $\mathcal{B} = \text{Invalidé}$ et $k = 0$.

Observons que $\sum_{C \in \mathcal{C}'} D_C < v(x) - v(y)$ si et seulement si $x \succ_{w \circ \pi_{\mathcal{C}'}} y$.

Algorithm 1 ALGO

```
1: procedure ALGO( $\mathcal{C}, \mathcal{B}, k$ )
2:   for  $i = k+1, \dots, p$  do
3:     if  $[T_i \cap C = \emptyset, \forall C \in \mathcal{C}]$  then
4:        $\mathcal{C}' \leftarrow \mathcal{C} \cup \{T_i\}$ 
5:       if  $[\sum_{C \in \mathcal{C}'} D_C < v(x) - v(y)]$  then //Branch
6:          $\mathcal{C}' \leftarrow \text{ALGO}(\mathcal{C}', \mathcal{B}, i)$ 
7:       if  $[\mathcal{C}' \sqsubset \mathcal{B}]$  then //Mise à jour meilleure explication
8:          $\mathcal{B} \leftarrow \mathcal{C}'$ 
9:       if  $[\mathcal{C} \cup \{T_i\} \not\sqsubset \mathcal{B}]$  then //Bound
10:      return  $\mathcal{B}$ 
11: return Invalide
```

Après avoir déterminé la meilleure explication \mathcal{B} , on définit les paires de critères qui vont constituer notre argument.

$$R = \cup_{C \in \mathcal{B}} \{(i, j) \in C^2 \mid \Delta_i < \Delta_j \text{ et } w_i < w_j\}.$$

$R^* \subseteq R$ est la réduction transitive de R .

Les critères dans chaque couple $(i, j) \in R^*$ sont répartis parmi les cinq catégories suivantes : K_{PS} (positive and strong), K_{PRS} (positive and relatively strong), K_{NW} (negative and weak), K_{NRW} (negative and relatively weak) et K_{PN} (positive and negative argument).

$\forall (i, j) \in R^*$:

- si $\Delta_j > 0, \Delta_i < 0$:
 - si $w_i \leq \frac{1}{n} \leq w_j$, alors $j \in K_{PS}$ et $i \in K_{NW}$;
 - si $w_i \ll w_j < \frac{1}{n}$, alors $(i, j) \in K_{PN}$;
 - si $\frac{1}{n} < w_i \ll w_j$, alors $(i, j) \in K_{PN}$;
 - si $w_i < w_j < \frac{1}{n}$, alors $i \in K_{NW}$;
 - si $\frac{1}{n} < w_i \leq w_j$, alors $j \in K_{PS}$;
- si $\Delta_j > 0, \Delta_i > 0$:
 - si $\frac{1}{n} \leq w_j$, alors $j \in K_{PS}$;
 - si $w_j < \frac{1}{n}$, alors $j \in K_{PRS}$;
- si $\Delta_j < 0, \Delta_i < 0$:
 - si $w_i \leq \frac{1}{n}$, alors $i \in K_{NW}$;
 - si $\frac{1}{n} < w_i$, alors $i \in K_{NRW}$. (Labreuche [2011] écrit w_j au lieu de w_i pour ce point et le précédent. Nous avons considéré que c'est une faute de frappe.)

Ces différentes possibilités peuvent se résumer comme suit.

Algorithm 2 SORT

```
procedure SORT( $i, j$ )
if  $[w_i \ll w_j < \frac{1}{n}]$  or  $[\frac{1}{n} < w_i \ll w_j]$  then
   $K_{PN} \leftarrow K_{PN} \cup \{(i, j)\}$ 
else
  if  $[\Delta_j > 0]$  and  $[w_j \geq \frac{1}{n}]$  then
     $K_{PS} \leftarrow K_{PS} \cup \{j\}$ 
  if  $[\Delta_j > 0]$  and  $[\Delta_i > 0]$  and  $[w_j < \frac{1}{n}]$  then
     $K_{PRS} \leftarrow K_{PRS} \cup \{j\}$ 
  if  $[\Delta_i < 0]$  and  $[w_i \leq \frac{1}{n}]$  then
     $K_{NW} \leftarrow K_{NW} \cup \{i\}$ 
  if  $[\Delta_i < 0]$  and  $[\Delta_j < 0]$  and  $[w_i > \frac{1}{n}]$  then
     $K_{NRW} \leftarrow K_{NRW} \cup \{i\}$ 
```

La phrase suivante est générée pour expliquer la décision.

x est préférée à y car x est meilleure que y sur les critères K_{PS} qui sont importants et sur les critères K_{PRS} qui sont relativement importants, y est meilleur que x sur les critères K_{NW} qui ne sont pas importants et sur les critères K_{NRW} qui ne sont pas réellement importants.

$\forall(i, j) \in K_{PN}$ on ajoute la phrase suivante.

Le critère j pour lequel x est meilleure que y est plus important que le critère i pour lequel x est pire que y .

Ancrage ψ_{RMG} : "remaining"

On applique cet ancrage lorsque les trois ancrages précédents ne sont pas applicables. C'est-à-dire,

$\psi_{RMG}(x, y)$ est vrai si et seulement si, $\exists i \Delta_i < 0, x \succ_{w^{ref}} y$ et $\forall \pi \in \Pi(N) x \succ_{w \circ \pi} y$.

La génération de l'argumentation se fait en fonction de $\mathcal{W} := \max_{i \in N} |w_i - \frac{1}{n}|$.

Nous avons deux cas :

- si $\mathcal{W} \leq \epsilon_w$, alors les poids de w ont quasiment la même valeur et sont donc neutres, on a donc $x \succ_w y$ qui suit $x \succ_w^{ref} y$. Nous avons x préféré à y car x est en moyenne meilleure sur l'ensemble des critères. La phrase générée pour l'explication de la décision est de la forme " x est préférée à y car x est en moyenne meilleure que y et les critères sont presque de mêmes poids."
- si $\mathcal{W} > \epsilon_w$, alors les poids de w sont significativement différents. On peut dire que les valeurs de Δ de $N^+(x, y)$ sont plus grands que les valeurs de Δ de $N^-(x, y)$ et donc que les performances de x par rapport à y sur $N^+(x, y)$ sont supérieures et qu'elles compensent les performances de x sur $N^-(x, y)$.

Deux formes de phrases sont générées selon \mathcal{W} , si $\epsilon_w < \mathcal{W} < \epsilon'_w$, alors la phrase générée est la suivante " x est préférée à y car l'intensité de la préférence de x par rapport à y sur $N^+(x, y)$ est significativement plus grande que l'intensité de la préférence de y par rapport à x sur $N^-(x, y)$, et tous les critères ont plus ou moins les mêmes poids." et si $\mathcal{W} \geq \epsilon'_w$ alors la phrase générée est de la forme " x est préférée à y car l'intensité de la préférence de x par rapport à y sur $N^+(x, y)$ est plus grande que l'intensité de la préférence de y par rapport à x sur $N^-(x, y)$ ".

3.3 Approche Nunes et al

Le but de l'approche de Nunes et al. [2014] n'est pas à proprement parlé le même que celui de Labreuche [2011], en effet ils cherchent à déterminer l'explication la plus adéquate parmi leur catalogue de 7 modèles d'explications possibles. Cette approche, dans un premier temps, cherche à déterminer s'il existe une alternative dominante, puis dans un deuxième temps, applique des filtres pour réduire le nombre d'alternatives éligibles, et finalement dans un troisième temps s'il reste des alternatives, donner une explication multi-critères. Nous présenterons dans un premier temps le modèle de l'approche et dans un second temps les différents arguments possibles dans les explications fournies ainsi que leur génération.

3.3.1 Modèle

En fonction d'un ensemble d'alternatives X , où chaque alternative est décrite par un ensemble fini d'attributs Att , d'un ensemble de contraintes C et d'un vecteur de poids w , Nunes et al. [2014] déterminent les paramètres qui seront utilisés lors de l'explication. Nunes et al. [2014] définissent un poids sur un critère i différent en fonction d'une l'alternative, c'est à dire que le poids w_i de

la performance d'une alternative x sur le critère i n'est pas le même pour une alternative y . Nous n'appliquerons pas cette répartition des poids en fonction des alternatives mais en fonction de l'importance des critères.

Notons que la fonction de décision $v(x)$ dans ce modèle, qu'on notera $d(x, y)$ par la suite, diffère de ce qu'on a noté en Section 3.1.1. Dans ce modèle, $d(x, y)$ est une somme pondérée de 3 fonctions.

- $Cost(x, y) = \sum_i w_i \times AttCost(x, y, i)$
avec $AttCost(x, y, i)$ qui est égale à $v_i(y_i) - v_i(x_i)$ si $v_i(y_i) > v_i(x_i)$ ou 0 sinon, représentant le coût de x pour égaliser y sur le critère i et donc $Cost(x, y)$ représente le perte que le décideur aura si x est recommandée au lieu de y .
- $ExtAversion(x, y)$ qui est égale à $ext(y) - ext(x)$ si $ext(x) < ext(y)$ ou 0 sinon. La fonction $ext(x)$ retourne la déviation standard de l'ensemble $DV = \{1 - x_i | i \in N\}$. Cette fonction permet de pénaliser les alternatives qui ont sont plus extrême qu'une autre. Une alternative est dite extrême si elle possède des performances élevées (proche de 1) sur certains critères qui compensent des performances basses (proche de 0) sur d'autres.
- - Notons $TO(x, y) = \frac{Cost(y, x)}{Cost(x, y)}$, le ratio coût-bénéfice entre x et y , ce ratio existe, si et seulement si, $Cost(x, y) > Cost(y, x)$. Et notons avg_{TO} la moyenne de tous les TO existant par rapports aux alternative dans X .
 $ToContrast(x, y)$, la troisième partie de la fonction de décision, est égale à $avg_{TO} - TO(x, y)$ si $TO(x, y)$ existe et $TO(x, y) \leq avg_{TO}$, à $TO(y, x)$ si $TO(y, x)$ existe et $TO(y, x) > avg_{TO}$.

On note, $d(x, y) = A \times Cost(x, y) + B \times ExtAversion(x, y) + C \times ToContrast(x, y)$ et si $d(x, y) < d(y, x)$ alors, x est meilleure que y . x est appelée ensuite alternative recommandée et y alternative rejetée.

Une contrainte $c \in C$ est une contrainte sur les valeurs d'attributs d'un attribut $att(c)$. Par exemple c est de la forme $prix < 100$ et $att(c) = prix$. La contrainte c est associée à une valeur d'utilité entre $[-1, 1]$, ce qui signifie que les valeurs d'attributs qui satisfont c promeuvent la valeur d'utilité associée à $v(c)$. Les valeurs extrêmes (c'est-à-dire, $v(c) = -1$ et $v(c) = 1$) indiquent des contraintes négatives et positives fortes, ce qui signifie que les alternatives $x \in X$ dont les valeurs d'attributs x_i satisfont c , tel que $v(c) = -1$, ou ne satisfont pas c , tel que $v(c) = 1$, devraient être rejetées.

3.3.2 Sélection des paramètres pour compléter une explication

Il y a 7 modèles d'explication, chacun associé à un canevas qui est paramétré par un seul ou plusieurs arguments. Dans cette section, nous expliquons comment Nunes et al. [2014] sélectionnent les arguments.

Sélection mono-critère

Attribut critique

Soit x^* la meilleure alternative dans X , un attribut a^* est critique pour la décision si, pour toute autre alternative $y \in X \setminus \{x^*\}$, on a $AttCost(y, x^*, a^*) > AttCost(x^*, y, a^*)$ et pour tout autre attribut b différent de a^* , $AttCost(x^*, y, b) = AttCost(y, x^*, b) = 0$. La phrase suivante est générée pour expliquer la décision.

x^* à été recommandée parce qu'elle possède la meilleure valeur sur a^* .

Ceci est un cas particulier de la domination qui suit.

Domination

Pour ce pattern, aucun paramètre est utilisé dans l'explication fournie. Soit x l'alternative recommandée et y l'alternative rejetée, la décision $x \succ y$ est triviale et l'explication de la décision est de cette forme :

Il n'y a aucune raison de choisir y , alors que x est meilleure sur tous les critères.

Rejet

Les paramètres de ce modèle sont sélectionnés en fonction de deux cas : (i) les alternatives avec des contraintes fortes insatisfaites (égale à 1), spécifiant les valeurs d'attribut inacceptables, quels que soient les attributs restants; et (ii) les valeurs d'attribut moindre qui, bien qu'elles puissent être compensées par d'autres valeurs d'attributs, sont utilisés comme une raison de rejeter l'alternative.

Le premier cas est trivial, pour chaque contrainte c on vérifie si $v(c) = 1$ pour chaque attribut de chaque alternative x . Pour le second cas, nous définissons d'abord la fonction LPV qui est évaluée à vrai quand une alternative y a une valeur moins préférée selon une contrainte c . Notons que $\text{sat}(x, c)$ signifie : x satisfait la contrainte c .

$$LPV(y, c) = [\text{sat}(y, c) \wedge v(c) < 0] \vee [\neg \text{sat}(y, c) \wedge v(c) > 0]$$

Définissons $\text{stg}_{cst}(y, C) = \{c \mid c \in C \wedge LPV(y, c) \wedge [\forall c' \in C \setminus \{c\}, LPV(y, c') \wedge |v(c)| \geq |v(c')|]\}$, la contrainte positive (ou négative) la plus forte insatisfaite (satisfaite) par une alternative sur un ensemble de contrainte C .

Sur la base de cette contrainte la plus forte, nous détectons les attributs de rejet. Le critère de rejet, associé à ces attributs, est formellement défini comme suit

Soit $x, y \in X$, où x est l'alternative recommandée et y une alternative rejetée, C un ensemble de contraintes, et $c \in C$. Un attribut $\text{att}_{co} \in \text{Att}$ est dit être un rejet, ou $\text{CutOff}(y, x)$, si $[(\text{sat}(y, c) \wedge v(c) = -1) \vee (\neg \text{sat}(y, c) \vee v(c) = 1) \wedge (LPV(y, c) \wedge |v(c)| > \text{stg}_{cst}(x, C))] \wedge \text{att}(c) = \text{att}_{co}$,

si plusieurs critères satisfont cette propriété, celui qui possède le plus haut poids est sélectionné. La phrase suivante est ainsi générée pour expliquer la décision.

y a été rejetée car elle ne satisfait pas la contrainte associée au critère c .

Exigences minimales +/−

$\text{Exigencesminimales}^+$ justifie le choix, tandis que $\text{Exigencesminimale}^-$ explique les rejets, lorsqu'il existe un sous-ensemble d'alternatives respectant C et qu'il existe un attribut déterminant pour choisir parmi les alternatives restantes.

Si nous sommes dans ce modèle, et qu'il existe un attribut, que nous appelons *tie-breaker*, qui est déterminant dans le choix d'une alternative parmi l'ensemble, nous adoptons ces modèles pour expliquer l'alternative recommandée et les alternatives rejetées, en excluant celles rejetées en raison d'attributs de domination ou de rejet.

L'attribut a_{tb} est dit un attribut tie-breaker, ou $\text{TieBreaker}(x)$, s'il existe une alternative $y \in \text{Opt} \setminus \{x\}$ rejetée car elle ne satisfait pas une contrainte, c'est-à-dire $\exists a$ tel que $\text{CutOff}(y, c) = a$, et $\forall y_r \in \text{Opt} \setminus \{x, y\} \nexists a$ tel que $\text{CutOff}(y_r, x) = a$, nous avons $\text{AttCost}(y_r, x, a_{tb}) > \text{AttCost}(x, y_r, a_{tb})$. En outre, a_{tb} est unique car $\nexists a'$ tel que $a' \neq a_{tb}$ et $\text{AttCost}(y_r, x, a') > \text{AttCost}(x, y_r, a)$.

Lorsque qu'on le souhaite expliquer pour quoi x a été recommandée, la phrase générée est comme suit, "En plus de satisfaire toutes vos exigences, x a la meilleure valeur sur a_{tb} .", sinon la phrase générée pour expliquer pour quoi y est rejetée est de la forme suivant, "Même si y satisfait toutes vos exigences, elle est moins bonne sur a_{tb} que x ."

Sélection multi-critères

Avant d'introduire la détermination des critères décisifs, nous allons réutiliser ce qui a été défini dans la Section 3.2.1 à savoir les ensembles $N^+(x, y)$ et $N^-(x, y)$ qui pour l'approche de Nunes et al. [2014] sont définis par

- $N^+(x, y) = \{ a \mid a \in Att \wedge w(y, a) \times AttCost(y, x, a) > 0 \},$
- $N^-(x, y) = \{ a \mid a \in Att \wedge w(x, a) \times AttCost(x, y, a) > 0 \}.$

Chacun de ces ensembles est respectivement lié au *pour* et au *contre* lors de la comparaison entre x et y . On définit

- $Pros(x, y) = \sum_{a_+ \in N^+(x, y)} w(y, a_+) \times AttCost(y, x, a_+)$
- $Cons(x, y) = \sum_{a_- \in N^-(x, y)} w(x, a_-) \times AttCost(x, y, a_-)$

représentant le *pour* et le *contre* de x par rapport à y de manière quantitative. Les critères décisifs sont différents pour une alternative rejetée et une alternative recommandée.

Critères décisifs

Nous allons d'abord déterminer pourquoi une alternative est rejetée. Pour une alternative rejetée y et l'alternative x recommandée, l'ensemble des critères décisifs $D \subset Att$ rejetant y est l'union de tous sous-ensembles minimaux $S \subset Att^-(y, x)$ tels que $\sum_{a \in D} w(y, a) \times AttCost(y, x, a) < Pros(y, x)$.

Comme nous devons identifier différents sous-ensembles d'attributs, il est important de fournir un moyen efficace de les identifier. C'est le rôle de l'algorithme 3. La première partie de l'algorithme détermine la cardinalité minimale d'un sous-ensemble possible qui satisfait la propriété des critères décisifs. Dans le but de déterminer l'ensemble D , un classement des attributs est effectué en fonction de leur *contre* (du plus grand au plus petit coût) et la construction d'un ensemble d'attribut est effectué pas à pas en accumulant leurs *contre* (Cons). Lorsque la somme de l'ensemble qui accumule les *contre* est plus grande que le *pour*, l'algorithme a déterminé l'ensemble minimal de critères décisifs. La cardinalité étant trouvée, l'appel à l'algorithme 4 est effectué pour construire l'ensemble D .

Algorithm 3

```

procedure DecisiveCriteria-(x,y)
    SortedN-  $\leftarrow$  Sort( $N^-(y, x)$ ,  $a_i \succ a_j \leftrightarrow w(y, a_i) \times AttCost(y, x, a_i) < w(y, a_j) \times AttCost(y, x, a_j)$ );
    ACons  $\leftarrow$  0;
    Card  $\leftarrow$  0;
    while ACons  $\leq$  Pros(y, x)  $\wedge$   $i < |N^-(y, x)|$  do
         $a \leftarrow SortedN^-[Card]$ ;
        ACons  $\leftarrow$  ACons +  $w(y, a) \times AttCost(y, x, a)$ ;
        Card  $\leftarrow$  Card + 1;
    if [Card  $< |N^-(y, x)|$ ] then
         $D, Stop \leftarrow DC(\emptyset, 0, \emptyset, 0, Card, y, x, SortedN^-)$ ;
        if [ $|D| < |N^-(y, x)|$ ] then
            return  $D$ ;
    return  $\emptyset$ ;

```

Algorithm 4

```
procedure DC( $D, Idx, CAtt, ACons, Card, y, x, SortedN^-$ )  
  if [  $|CAtt| = Card$  ] then  
    if [  $ACons > Pros(or, oc)$  ] then  
       $D \leftarrow D \cup CAtt$ ;  
      return  $D, false$ ;  
    else  
      return  $D, true$ ;  
  else  
    for  $i \leftarrow Idx$  to  $SortedN^-$  do  
       $a \leftarrow SortedN^-[i]$ ;  
       $Cons \leftarrow ACons + w(y, a) \times AttCost(y, x, a)$ ;  
       $D, Stop \leftarrow DC(D, i + 1, CAtt \cup a, ACons, Card, y, x, SortedN^-)$ ;  
      if [  $Stop$  ] then  
        return  $D, true$ ;  
    return  $D, false$ ;
```

Nous allons maintenant expliquer pourquoi une alternative a été recommandée. Les critères décisifs d'une alternative recommandée peuvent être, soit l'ensemble des attributs tel que l'alternative x recommandée possède de meilleures valeurs que la majorité des autres alternatives, ou, si le premier ensemble n'existe pas, les critères décisifs pour rejeter l'alternative étant la deuxième meilleure alternative après celle recommandée. Dans les deux cas, les alternatives rejetées par *domination* ($Expl(y, x) = \Psi_{dom}$) ou *rejet* ($Expl(y, x) = \Psi_{cutOff}$) ne sont pas considérées.

Pour identifier l'ensemble d'attributs du premier cas, nous définissons pour une alternative recommandée $x \in X$, les meilleurs attributs $B \subset Att$ l'ensemble des attributs tels que $\forall a \in B$ et $\forall y \in X_{r^*}$ ($X_{r^*} = X \setminus \{y \mid Expl(y, x) = \Psi_{cutOff} \vee Expl(y, x) = \Psi_{dom}\} \cup \{x\}$), on a $AttCost(y, x, a) > AttCost(x, y, a)$, pour au moins $\frac{|X_{r^*}|}{2}$ alternatives, et $AttCost(y, x, a) = AttCost(x, y, a) = 0$ pour les autres. B est maximal au sens de \subset .

Pour le second cas, soit $x \in X$ l'alternative recommandée. L'ensemble des critères décisifs $D \subset Att$ est le meilleur ensemble d'attributs B pour x . Si $B = \emptyset$ alors D est l'ensemble de critères décisifs par rapport à la seconde meilleure alternative après x , c'est à dire $DecisiveCriteria^-(y_{2ndB}, x)$, tel que $Pros(x, y_{2ndB}) - Cons(x, y_{2ndB})$ est le minimum, pour toute alternative $y \in X \setminus \{x\}$.

Algorithm 5

```
procedure DecisiveCriteria+( $x$ )  
   $Opt_{r^*} \leftarrow Opt - \{o \in X \mid o = x \vee Expl(o, x) = \Psi_{cutOff} \vee Expl(o, oc) = \Psi_{dom}\};$   
   $D \leftarrow \emptyset;$   
  for each  $a \in Att$  do  
     $in \leftarrow true;$   
     $counter \leftarrow 0;$   
    for each  $y \in Opt_{r^*}$  do  
      if  $[ AttCost(y, x, a) = AttCost(x, y, a) = 0 ]$  then  
         $counter \leftarrow counter + 1;$   
      else  
        if  $AttCost(y, x, a) < AttCost(x, y, a)$  then  
           $in \leftarrow false;$   
      if  $[ in \wedge counter < \frac{|Opt_{r^*}|}{2} ]$  then  
         $D \leftarrow D \cup \{a\};$   
  if  $[ D = \emptyset ]$  then  
     $y_{2ndB} \leftarrow y \mid y \in X \wedge \min(Pros(oc, y) - Cons(oc, y));$   
     $D \leftarrow DecisiveCriteria^-(y_{2ndB}, x)$   
return  $D;$ 
```

Les critères décisifs pour une alternative recommandée sont déterminés par l'algorithme 5, où la première partie tente d'identifier les meilleurs attributs. S'ils n'existent pas, la seconde partie tente de déterminer les critères décisifs en comparaison avec la seconde meilleure alternative. L'explication générée pour ce modèle est de la forme,

$[x \mid y]$ à été $[recommandée \mid rejetée]$ à cause de ses D .

Compromis

Ce modèle s'applique lorsque aucun autre n'est applicable. Pour expliquer le choix de l'alternative *recommandée* qui ne possède pas d'ensemble d'attributs décisifs pour la décision du modèle précédent nous avons trois cas à envisager. Ils représentent les trois raisons distinctes pourquoi il n'y pas de critère décisif

- Lorsque une alternative x recommandée ne possède pas un ou plusieurs attributs qui sont meilleurs que les attributs des autres alternatives, et que la différence entre le *pour* et le *contre* par rapport à la seconde meilleure alternative est négative ($Pros(x, y) < Cons(x, y)$) cela signifie que le *trade-off contrast* et/ou *extremeness aversion*, définit dans $d(x, y)$ sont responsables du choix de x au lieu de y selon Nunes et al. [2014]. Nous avons deux alternatives qui dépendent de l'existence d'un ensemble $D \subset Att$ tel que $D = DecisiveCriteria^-(x, y)$
- Quand D existe, l'explication fournie surligne que y a D comme *pour*, et annonce que x a un meilleur coût bénéfique (selon les principes centrés sur le décideur).
- Quand D n'existe pas, 6 détermine l'ensemble maximal de *pour* qui doivent être pris en compte pour permettre l'existence des critères décisifs pour rejeter x . Donc pour l'alternative rejetée y où son *pour* est meilleur que le *contre* de l'alternative recommandée x , $DecisiveProsCons(x, y)$ identifie le *contre* qui devrait être fournit dans l'explication et aussi le *pour* qui doit être mentionné qui compense le *contre*. De plus, les principes centrés sur l'utilisateur jouant un rôle important dans la décision, la relation coûts-bénéfices est également mise en évidence.
- Lorsque l'alternative recommandée x a le meilleur équilibre entre le *pour* et le *contre*, mais aucun attribut de x n'a pas la meilleure valeur en comparaison avec le reste des

alternatives acceptable et qu'aucun critère décisif n'existe en comparaison avec la seconde meilleur alternative.

L'explication fournie dans ce cas est de la même manière que le précédent cas mais dans le sens inverse ($DecisiveProsCons(y, x)$).

Algorithm 6

```

procedure DECISIVEPROSCONS( $x, y$ )
   $SortedN^+ \leftarrow Sort(N^+(x, y), a_i \succ a_j \leftrightarrow$ 
     $w(y, a_i) \times AttCost(y, x, a_i) > w(y, a_j) \times AttCost(y, x, a_j));$ 
   $ProsLeft \leftarrow Pros(x, y);$ 
   $P \leftarrow \emptyset;$ 
   $C \leftarrow \emptyset;$ 
  while  $C = \emptyset \wedge SortedN^+ \neq \emptyset$  do
     $a \leftarrow Last(SortedN^+);$ 
     $SortedN^+ \leftarrow SortedN^+ \setminus \{a\};$ 
     $ProsLeft = ProsLeft - w(y, a) \times AttCost(y, x, a);$ 
     $P \leftarrow P \cup \{a\};$ 
     $C \leftarrow DecisiveCriteria^-(x, y);$  // en considérant seulement les ProsLeft
  if  $[C = \emptyset]$  then
     $C \leftarrow N^-(x, y);$ 
  return  $P, C;$ 

```

Le raisonnement pour justifier une alternative rejetée est similaire à ce qui est présenté précédemment. Dans un premier temps on analyse si l'alternative rejetée y a un meilleur équilibre entre le *pour* et le *contre* que l'alternative recommandée x ($Pros(y, x) > Cons(y, x)$) : si c'est le cas l'approche précédente est adoptée. Si $DecisiveCriteria^-(x, y)$ est non vide, on expose ces aspects positifs de y mais nous déclarons aussi que y a une relation coût-bénéfice moins bonne par rapport à x ; s'il n'y pas de critères décisifs, on sélectionne le P et C retournés par $DecisiveProsCons(x, y)$. Cette procédure est appliquée aussi lorsque $Pros(y, x) \leq Cons(x, y)$ mais qu'aucun critères décisifs justifie la décision.

L'explication générée est de la forme suivant (en fonction des cas mentionnés ci-avant);

Même si y fournit un meilleur *pour* que x , elle a un pire *contre*.

Les explications sont testées dans l'ordre suivant pour une alternative recommandée : Ψ_{crit} , Ψ_{dom+} , $\Psi_{minReq+}$, $\Psi_{decisif}$ et $\Psi_{tradeOff}$ et pour une alternative rejetée l'ordre est le suivant : Ψ_{crit} , Ψ_{cutOff} , Ψ_{dom-} , $\Psi_{minReq-}$, $\Psi_{decisif}$ et $\Psi_{tradeOff}$.

4 Travaux à venir

Dans un premier temps nous allons implémenter les deux approches retenues et les tester pour obtenir des résultats cohérents avec leur article respectif. Ensuite la prochaine étape est d'adapter indépendamment chacune de ces approches à l'article de [Cailloux and Meinard \[2018\]](#) dans le but de générer des arguments et contre arguments et dans le même temps établir le langage commun entre les deux approches et le décideur. Parallèlement nous réfléchirons à l'établissement d'un protocole de dialogue lors de la résolution du problème de décision. La finalité est de faire la convergence des deux approches dans un système unique, de tester sa robustesse, et en fonction du temps faire une étude utilisateur du résultat final.

Calendrier prévisionnel

Juin

- Implémentation et tests de l'approche Labreuche (2semaines) ✓
- Implémentation et tests de l'approche Nunes (2semaines)

Juillet

- établir le langage commun entre les deux approches et le décideur (1/2 semaine)
- établir un protocole de dialogue lors de la résolution du problème de décision (1/2 semaine)
- adapter indépendamment les deux approches pour le débat dans le but de générer des arguments et contre arguments (1 semaines)
- Convergence des deux approches dans un système unique(2 semaines)

Août

- Convergence des deux approches dans un système unique (suite) (1semaine),
- Tester de robustesse (1semaine)
- Étude utilisateur (2semaine)

1er septembre - 15 septembre

- Fin de rapport (1 semaine)
- Slides de présentation et entraînement (1 semaine)

5 Annexe A : Développement de l'approche de Labreuche

Le développement de l'approche de Labreuche [2011] s'appuie sur les travaux de *Elie Abi Hanna Daher*, nous utilisons actuellement sa représentation des alternatives et des critères.

Une alternative x possède les variables

id	l'identifiant de x , sous la forme d'un entier naturel.
name	le nom de x sous la forme d'une chaîne de caractère.
evaluations	les performances de x sous la forme d'une table associant pour chaque critère une valeur.

Un critère i possède les variables

id	l'identifiant de i , sous la forme d'un entier naturel.
name	le nom de i sous la forme d'une chaîne de caractère.
scale	l'échelle des performances possibles des alternative sur le critère i . Variable actuellement initialisée avec une liste vide car non exploitée.

De notre coté pour le développement de l'approche nous avons créer des variables globales supplémentaires.

Nom	Type (code)	
criteria	List<Criterion>	L'ensemble des critères.
alternatives	List<Alternative>	L'ensemble des alternatives.
weights	Map<Criterion,Double>	Le vecteur de poids w .
weightsReference	Map<Criterion,Double>	Le vecteur de poids de référence w^{ref} .
deltas	Map<Criterion,Double>	Le vecteur Δ , cf description ancrage ψ_{NOA} .
scoreboard	MAp<Double,Alternative>	Le tableau de classement des alternatives en fonction de leur score attribué par la fonction de valeurs $v(x)$.
positiveArguments	List<Criterion>	l'ensemble N^+ .
negativeArguments	List<Criterion>	l'ensemble N^- .
nullArguments	List<Criterion>	l'ensemble $N^=$.
epsilon	Double	le seuil ϵ_w fixé.

Le développement des ancrage ont été fait de la manière suivante, on suppose que nous avons nos deux alternatives x et y déterminées au préalable par le début du modèle à savoir les deux meilleurs dans *scoreboard*. Une vérification de l'implémentation à été faite en comparant les résultats obtenus avec les résultats fournis dans l'approche de Labreuche [2011].

ψ_{ALL}

Une simple vérification du signe des Δ_i permet de vérifier si cet ancrage est applicable. Si $\forall i \Delta_i > 0$ alors x domine y sur tout les critères. La figure 6 illustre le fonctionnement de l'ancrage.

```

Criteria    <-   Weight :

c1 <- w_1 = 0.41
c2 <- w_2 = 0.06
c3 <- w_3 = 0.24
c4 <- w_4 = 0.29

Alternatives :
X  : ( 0.42  0.66  0.66  0.57 )
Y  : ( 1.0   1.0   1.0   1.0 )

Top of alternatives :
Y : 1.0
X : 0.5355

Delta Y > X :
      ( 0.58 0.34 0.34 0.43 )

Explanation why Y is better than X :

Y is preferred to X since Y is better then X on ALL criteria.

Problem solved in : 3 milliseconds

```

FIGURE 6: Exemple de ψ_{ALL}

ψ_{NOA}

On vérifie dans un premier temps si la fonction $v(x)$ en appliquant *weightReference* comme poids inverse la décision entre x et y .

Si c'est le cas, on détermine pour chaque critère i , $(w_i - \frac{1}{n}) \times \Delta_i$ qu'on stock comme clé dans une table et la valeurs associé est le critère i . Ensuite nous trions l'ensemble des clefs et nous le parcourons de la plus grand à la plus petite valeurs pour déterminer p précisé dans la génération de cette ancrage précédemment. Avec l'ensemble obtenu qu'on note C , on détermine les arguments positifs et négatifs par simple intersection entre C et S^+ d'un coté et C et S^- de l'autre. Les critères dans ces intersections sont les arguments dans l'explication. Les figures 7, 8, 9, 10 illustrent les résultats de cet ancrage.

```

Criteria    <-   Weight :

c1 <- w_1 = 0.41
c2 <- w_2 = 0.06
c3 <- w_3 = 0.24
c4 <- w_4 = 0.29

Alternatives :
X  : ( 0.42  0.66  0.66  0.57 )
Y  : ( 0.54  0.04  0.89  0.76 )

Top of alternatives :
Y : 0.6577999999999999
X : 0.5355

Delta Y > X :
      ( 0.12 -0.62 0.23 0.19 )

Explanation why Y is better than X :

Even though X is better than Y on average, Y is preferred to X since
Y is better than X on the criteria { } that are important whereas
Y is worse than X on the criteria { c2 } that are not important.
Moreover, Y is on average better than X on the other criteria.

Problem solved in : 3 milliseconds

```

FIGURE 7: Résultat similaire à l'exemple 5

```

Criteria    <-   Weight :

c1 <- w_1 = 0.18
c2 <- w_2 = 0.11
c3 <- w_3 = 0.12
c4 <- w_4 = 0.24
c5 <- w_5 = 0.35

Alternatives :
X  : ( 0.95  0.67  0.64  0.27  0.39 )
Y  : ( 0.3   0.37  0.41  0.94  0.49 )

Top of alternatives :
Y : 0.5409999999999999
X : 0.5227999999999999

Delta Y > X :
      ( -0.65 -0.3 -0.23 0.67 0.1 )

Explanation why Y is better than X :

Even though X is better than Y on average, Y is preferred to X since
Y is better than X on the criteria { c4, c5 } that are important whereas
Y is worse than X on the criteria { c2, c3 } that are not important.

Problem solved in : 3 milliseconds

```

FIGURE 8: Résultat similaire à l'exemple 6


```

Criteria    <-   Weight :

c1 <- w_1 = 0.06
c2 <- w_2 = 0.11
c3 <- w_3 = 0.19
c4 <- w_4 = 0.11
c5 <- w_5 = 0.31
c6 <- w_6 = 0.08
c7 <- w_7 = 0.14

Alternatives :
X : ( 0.5 0.06 0.03 0.95 0.87 0.2 0.95 )
Y : ( 0.99 0.35 0.31 0.51 0.62 0.57 0.52 )

Top of alternatives :
X : 0.5655
Y : 0.5235

Delta X > Y :
( -0.49 -0.29 -0.28 0.44 0.25 -0.37 0.43 )

Explanation why X is better than Y :

Even though Y is better than X on average, X is preferred to Y since
X is better than Y on the criteria { c5 } that are important whereas
X is worse than Y on the criteria { c1 } that are not important.

Problem solved in : 3 milliseconds

```

FIGURE 9: Résultat similaire à l'exemple 13

```

Criteria    <-   Weight :

c1 <- w_1 = 0.14
c2 <- w_2 = 0.05
c3 <- w_3 = 0.17
c4 <- w_4 = 0.23
c5 <- w_5 = 0.17
c6 <- w_6 = 0.11
c7 <- w_7 = 0.13

Alternatives :
X : ( 0.5 0.06 0.03 0.95 0.87 0.2 0.95 )
Y : ( 0.99 0.35 0.31 0.51 0.62 0.57 0.52 )

Top of alternatives :
X : 0.5900000000000001
Y : 0.5618

Delta X > Y :
( -0.49 -0.29 -0.28 0.44 0.25 -0.37 0.43 )

Explanation why X is better than Y :

Even though Y is better than X on average, X is preferred to Y since
X is better than Y on the criteria { c4 } that are important whereas
X is worse than Y on the criteria { c2 } that are not important.

Problem solved in : 3 milliseconds

```

FIGURE 10: Résultat similaire à l'exemple 14

ψ_{IVT}

On détermine tous les sous ensembles possible de taille supérieur à 2 dans l'ensemble N à l'aide de la fonction $allSubset(<Un\ ensemble\ C>)$ pour chaque sous-ensemble c on vérifie deux conditions : d'abord que la fonction $d_{eu}(<c\ un\ sous-ensemble\ C>)$ renvoie un double supérieur à zéro, ensuite on détermine toute les permutations possible dans c à l'aide de la fonction $allPi(c)$ et on vérifie qu'il existe une permutation π qui donne le score minimal à $\sum_{i \in c} w_{\pi(i)} \Delta_i$ tel que $\pi = c$ alors on ajoute cet ensemble c dans une liste qui comporte tout les T_i . Cette liste est trié lexicographiquement par la fonction $sortLexi(<Une\ liste\ L>)$ et à l'aide de cette liste nous

appliquons *Algo-Eu* défini par Labreuche [2011]. Ensuite on construit K à l'aide du \mathcal{B} retourné par *Algo-Eu* et l'argumentation est fournie. Les figures 11, 12, 13 et 14 illustrent cet ancrage.

```
Criteria    <-   Weight :

c1 <- w_1 = 0.06
c2 <- w_2 = 0.11
c3 <- w_3 = 0.21
c4 <- w_4 = 0.29
c5 <- w_5 = 0.33

Alternatives :
X : ( 0.89  0.03  0.07  0.32  0.38 )
Y : ( 0.36  0.76  0.6  0.25  0.75 )

Top of alternatives :
Y : 0.5512
X : 0.289599999999999997

Delta Y > X :          ( -0.53 0.73 0.53 -0.07 0.37 )

Minimal permutation : { { c1, c3, c5 } }

R* : { (c1,c3) (c1,c5) }

K_ps = { c3, c5 } K_nw = { c1 }

Explanation why Y is better than X :

Y is preferred to X since Y is better than X on the criteria { c3, c5 } that are important
and on the criteria { } that are relatively important, X is better than Y on the criteria { c1 }
that are not important and on the criteria { } that are not really important.

Problem solved in : 15 milliseconds
```

FIGURE 11: Résultat similaire à l'exemple 9

```
Criteria    <-   Weight :

c1 <- w_1 = 0.13
c2 <- w_2 = 0.04
c3 <- w_3 = 0.12
c4 <- w_4 = 0.1
c5 <- w_5 = 0.07
c6 <- w_6 = 0.19
c7 <- w_7 = 0.15
c8 <- w_8 = 0.03
c9 <- w_9 = 0.01
c10 <- w_10 = 0.16

Alternatives :
X : ( 0.61  0.28  0.08  0.02  0.81  0.15  0.16  0.38  0.24  0.75 )
Y : ( 0.45  0.64  0.86  0.76  0.87  0.54  0.17  0.04  0.55  0.05 )

Top of alternatives :
Y : 0.467000000000000001
X : 0.345099999999999996

Delta Y > X :          ( -0.16 0.36 0.78 0.74 0.06 0.39 0.01 -0.34 0.31 -0.7 )

Minimal permutation : { { c6, c8 } { c3, c9 } }

R* : { (c8,c6) (c9,c3) }

K_ps = { c6, c3 } K_nw = { c8 }

Explanation why Y is better than X :

Y is preferred to X since Y is better than X on the criteria { c6, c3 } that are important
and on the criteria { } that are relatively important, X is better than Y on the criteria { c8 }
that are not important and on the criteria { } that are not really important.

Problem solved in : 13 milliseconds
```

FIGURE 12: Résultat similaire à l'exemple 10

```

Criteria    <-   Weight :

c1 <- w_1 = 0.11
c2 <- w_2 = 0.14
c3 <- w_3 = 0.13
c4 <- w_4 = 0.02
c5 <- w_5 = 0.27
c6 <- w_6 = 0.25
c7 <- w_7 = 0.08

Alternatives :
  X : ( 0.5  0.06  0.03  0.95  0.87  0.2  0.95 )
  Y : ( 0.99  0.35  0.31  0.51  0.62  0.57  0.52 )

Top of alternatives :
  Y : 0.5599
  X : 0.44720000000000004

Delta Y > X :          ( 0.49 0.29 0.28 -0.44 -0.25 0.37 -0.43 )

Minimal permutation : { { c4, c6 } }

R* : { (c4,c6) }

K_ps = { c6 }  K_nw = { c4 }

Explanation why Y is better than X :

Y is preferred to X since Y is better than X on the criteria { c6 } that are important
and on the criteria { } that are relatively important, X is better than Y on the criteria { c4 }
that are not important and on the criteria { } that are not really important.

Problem solved in : 10 milliseconds

```

FIGURE 13: Résultat similaire à l'exemple 15

```

Criteria    <-   Weight :

c1 <- w_1 = 0.24
c2 <- w_2 = 0.2
c3 <- w_3 = 0.25
c4 <- w_4 = 0.06
c5 <- w_5 = 0.02
c6 <- w_6 = 0.19
c7 <- w_7 = 0.04

Alternatives :
  X : ( 0.5  0.06  0.03  0.95  0.87  0.2  0.95 )
  Y : ( 0.99  0.35  0.31  0.51  0.62  0.57  0.52 )

Top of alternatives :
  Y : 0.5572
  X : 0.2899

Delta Y > X :          ( 0.49 0.29 0.28 -0.44 -0.25 0.37 -0.43 )

Minimal permutation : { { c1, c7 } { c3, c4 } }

R* : { (c7,c1) (c4,c3) }

K_ps = { c1, c3 }  K_nw = { c7, c4 }

Explanation why Y is better than X :

Y is preferred to X since Y is better than X on the criteria { c1, c3 } that are important
and on the criteria { } that are relatively important, X is better than Y on the criteria { c7, c4 }
that are not important and on the criteria { } that are not really important.

Problem solved in : 8 milliseconds

```

FIGURE 14: Résultat similaire à l'exemple 16

ψ_{RMG}

On détermine d'abord \mathcal{W} et on vérifie s'il est plus petit que ϵ_w en fonction du résultat de ce test on applique la l'explication adéquate. Les figures 15 et 16 illustrent cet ancrage.

```

Criteria    <-   Weight :

c1 <- w_1 = 0.16
c2 <- w_2 = 0.14
c3 <- w_3 = 0.15
c4 <- w_4 = 0.1
c5 <- w_5 = 0.16
c6 <- w_6 = 0.15
c7 <- w_7 = 0.14

Alternatives :
X : ( 0.5  0.06  0.03  0.95  0.87  0.2  0.95 )
Y : ( 0.99  0.35  0.31  0.51  0.62  0.57  0.52 )

Top of alternatives :
Y : 0.5624
X : 0.4901

Delta Y > X :
( 0.49 0.29 0.28 -0.44 -0.25 0.37 -0.43 )

Explanation why Y is better than X :

Y is preferred to X since the intensity of the preference Y over Y on
{ c1, c2, c3, c6 } is significantly larger than the intensity of X over Y on
{ c4, c5, c7 }, and all the criteria have more or less the same weights.

Problem solved in : 3 milliseconds

```

FIGURE 15: Résultat similaire à l'exemple 17

```

Criteria    <-   Weight :

c1 <- w_1 = 0.12
c2 <- w_2 = 0.16
c3 <- w_3 = 0.15
c4 <- w_4 = 0.16
c5 <- w_5 = 0.15
c6 <- w_6 = 0.14
c7 <- w_7 = 0.12

Alternatives :
X : ( 0.5  0.06  0.03  0.95  0.87  0.2  0.95 )
Y : ( 0.99  0.35  0.31  0.51  0.62  0.57  0.52 )

Top of alternatives :
Y : 0.5381
X : 0.498600000000000004

Delta Y > X :
( 0.49 0.29 0.28 -0.44 -0.25 0.37 -0.43 )

Explanation why Y is better than X :

Y is preferred to X since Y is on average better than X
and all the criteria have almost the same weights.

Problem solved in : 3 milliseconds

```

FIGURE 16: Résultat similaire à l'exemple 18

Résultats expérimentaux de Labreuche

Labreuche [2011] fournit des résultats expérimentaux à propos du modèle *EU* et l'élaboration des explications les plus courtes. L'expérience se fait sur une taille de N compris entre 4 et 20, qui est l'intervalle regroupant la majorité des pratiques dans AMCD selon lui.

D'après la FIGURE 6, on remarque que pour des valeurs de n petite ce sont principalement ψ_{IVT} et ψ_{RMG} qui sont appliqués. La probabilité de l'application de ψ_{ALL} décroît fortement en fonction de n . Pour les plus grandes valeurs de n ψ_{IVT} est appliqué majoritairement.

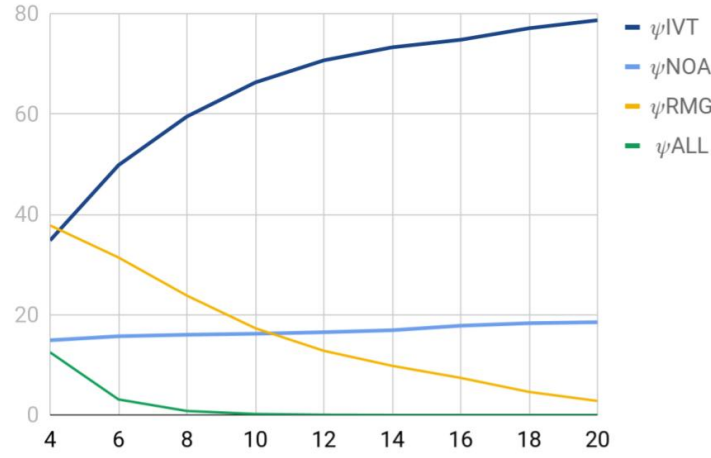


FIGURE 17: Pourcentage d'utilisation de chaque ancrage en fonction du nombre de critères

Seul le temps d'exécution pour ψ_{IVT} est fourni, FIGURE 7, car ψ_{ALL} , ψ_{NOA} et ψ_{RMG} n'ont au maximum qu'un vecteur de n composant à trier (ψ_{NOA}). Alors que ψ_{IVT} a besoin de l'algorithme *Algo* qui cherche dans un arbre assez large pour déterminer l'explication minimale. D'après la FIGURE 8, on remarque que l'algorithme *Algo* termine une fois sur deux à la première itération. Ce qui arrive d'après Labreuche [2011] lorsque T_1 forme à lui seul l'explication et donc le classement de C en fonction de \leq_{lexi} est efficace. D'un autre côté le pourcentage d'exploration de l'arbre décroît fortement en fonction de n .

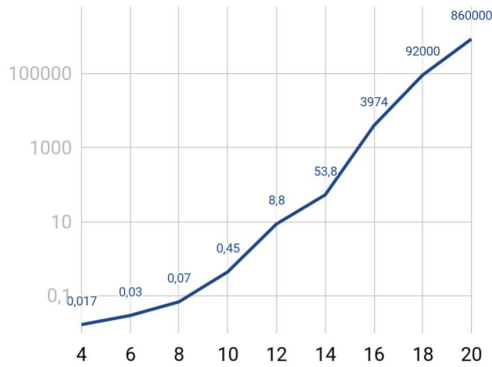


FIGURE 18: Temps de calcul moyen (en ms) en fonction du nombre de critères.

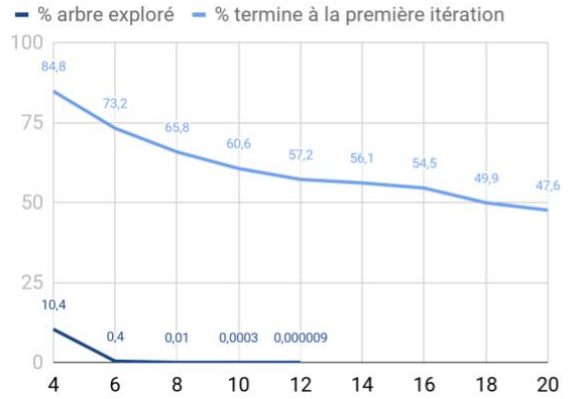


FIGURE 19: Pourcentage d'exploration de l'arbre et de l'arrêt à la première itération de *Algo* en fonction du nombre de critères

Références

- M. Balabanović and Y. Shoham. Fab : Content-based, collaborative recommendation. *Commun. ACM*, 40(3) :66–72, Mar. 1997. ISSN 0001-0782. doi : 10.1145/245108.245124. URL <http://doi.acm.org/10.1145/245108.245124>.
- K. Belahcene, C. Labreuche, N. Maudet, V. Mousseau, and W. Ouerdane. Explaining robust additive utility models by sequences of preference swaps. *Theory and Decision*, 82(2) :151–183, Feb 2017. ISSN 1573-7187. doi : 10.1007/s11238-016-9560-1. URL <https://doi.org/10.1007/s11238-016-9560-1>.
- C. Bielza, G. M. R.-I. S, and F. d. P. J, A. Structural, elicitation and computational issues faced when solving complex decision making problems with influence diagrams. *Computers & Operations Research*, 27(7) :725 – 740, 2000. ISSN 0305-0548. doi : [https://doi.org/10.1016/S0305-0548\(99\)00113-6](https://doi.org/10.1016/S0305-0548(99)00113-6). URL <http://www.sciencedirect.com/science/article/pii/S0305054899001136>.
- M. Bohanec, B. Zupan, and V. Rajkovič. Applications of qualitative multi-attribute decision models in health care. *International Journal of Medical Informatics*, 58-59 :191 – 205, 2000. ISSN 1386-5056. doi : [https://doi.org/10.1016/S1386-5056\(00\)00087-3](https://doi.org/10.1016/S1386-5056(00)00087-3). URL <http://www.sciencedirect.com/science/article/pii/S1386505600000873>.
- O. Cailloux and Y. Meinard. A formal framework for deliberated judgment. *CoRR*, abs/1801.05644, 2018. URL <http://arxiv.org/abs/1801.05644>.
- G. Carenini and J. D. Moore. Generating and evaluating evaluative arguments. *Artificial Intelligence*, 170(11) :925 – 952, 2006. ISSN 0004-3702. doi : <https://doi.org/10.1016/j.artint.2006.05.003>. URL <http://www.sciencedirect.com/science/article/pii/S000437020600066X>.
- L. Chen and F. Wang. Sentiment-enhanced explanation of product recommendations. In *Proceedings of the 23rd International Conference on World Wide Web, WWW '14 Companion*, pages 239–240, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2745-9. doi : 10.1145/2567948.2577276. URL <http://doi.acm.org/10.1145/2567948.2577276>.
- P. Delle Site and F. Filippi. Weighting methods in multi-attribute assessment of transport projects. *European Transport Research Review*, 1(4) :199–206, Dec 2009. ISSN 1866-8887. doi : 10.1007/s12544-009-0018-1. URL <https://doi.org/10.1007/s12544-009-0018-1>.
- P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2) :321 – 357, 1995. ISSN 0004-3702. doi : [https://doi.org/10.1016/0004-3702\(94\)00041-X](https://doi.org/10.1016/0004-3702(94)00041-X). URL <http://www.sciencedirect.com/science/article/pii/000437029400041X>.
- J. Geldermann. Explanation systems. 5 :241–259, 08 2010. doi : 10.1007/978-90-481-9045-4_14.
- M. A. Grando, L. Moss, D. Glasspool, D. Sleeman, M. Sim, C. Gilhooly, and J. Kinsella. Argumentation-logic for explaining anomalous patient responses to treatments. In M. Peleg, N. Lavrač, and C. Combi, editors, *Artificial Intelligence in Medicine*, pages 35–44, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-22218-4.
- M. Kadziński, M. Ghaderi, J. Wąsikowski, and N. Agell. Expressiveness and robustness measures for the evaluation of an additive value function in multiple criteria preference disaggregation methods : An experimental analysis. *Computers & Operations Research*, 87 : 146 – 164, 2017. ISSN 0305-0548. doi : <https://doi.org/10.1016/j.cor.2017.05.011>. URL <http://www.sciencedirect.com/science/article/pii/S0305054817301235>.

- R. Keeney and H. Raiffa. *Decisions with Multiple Objectives : Preferences and Value Tradeoffs*. Cambridge University Press, 1993.
- D. A. Klein and E. H. Shortliffe. A framework for explaining decision-theoretic advice. *Artificial Intelligence*, 67(2) :201 – 243, 1994. ISSN 0004-3702. doi : [https://doi.org/10.1016/0004-3702\(94\)90053-1](https://doi.org/10.1016/0004-3702(94)90053-1). URL <http://www.sciencedirect.com/science/article/pii/S0004370294900531>.
- C. Labreuche. A general framework for explaining the results of a multi-attribute preference model. *Artificial Intelligence*, 175(7) :1410 – 1448, 2011. ISSN 0004-3702. doi : <https://doi.org/10.1016/j.artint.2010.11.008>. URL <http://www.sciencedirect.com/science/article/pii/S0004370210001979>. Representing, Processing, and Learning Preferences : Theoretical and Practical Challenges.
- C. Labreuche, N. Maudet, W. Ouerdane, and S. Parsons. A dialogue game for recommendation with adaptive preference models. pages 959–967, 2015. URL <http://dl.acm.org/citation.cfm?id=2772879.2773275>.
- D. Mcsherry. Explanation in recommender systems. *Artificial Intelligence Review*, 24(2) :179–197, Oct 2005. ISSN 1573-7462. doi : [10.1007/s10462-005-4612-x](https://doi.org/10.1007/s10462-005-4612-x). URL <https://doi.org/10.1007/s10462-005-4612-x>.
- T. Narayanan and D. L. McGuinness. Towards leveraging inference web to support intuitive explanations in recommender systems for automated career counseling. In *First International Conference on Advances in Computer-Human Interaction, ACHI 2008, February 10-15, 2008, Sainte Luce, Martinique, France*, pages 164–169, 2008. doi : [10.1109/ACHI.2008.36](https://doi.org/10.1109/ACHI.2008.36). URL <https://doi.org/10.1109/ACHI.2008.36>.
- I. Nunes and D. Jannach. A systematic review and taxonomy of explanations in decision support and recommender systems. *User-Modeling and User-Adapted Interaction*, 27(3-5) :393–444, 2017.
- I. Nunes, S. Miles, M. Luck, S. Barbosa, and C. Lucena. Pattern-based explanation for automated decisions. *Artificial Intelligence*, 170(11) :669 – 674, 2014. ISSN 0004-3702. doi : <https://doi.org/10.1016/j.artint.2006.05.003>. URL <http://www.sciencedirect.com/science/article/pii/S000437020600066X>.
- K. Papamichail and S. French. Explaining and justifying the advice of a decision support system : a natural language generation approach. *Expert Systems with Applications*, 24(1) :35 – 48, 2003. ISSN 0957-4174. doi : [https://doi.org/10.1016/S0957-4174\(02\)00081-7](https://doi.org/10.1016/S0957-4174(02)00081-7). URL <http://www.sciencedirect.com/science/article/pii/S0957417402000817>.
- P. Pu and L. Chen. Trust-inspiring explanation interfaces for recommender systems. *Knowledge-Based Systems*, 20(6) :542 – 556, 2007. ISSN 0950-7051. doi : <https://doi.org/10.1016/j.knosys.2007.04.004>. URL <http://www.sciencedirect.com/science/article/pii/S0950705107000445>. Special Issue On Intelligent User Interfaces.
- C. K. Riesbeck and R. C. Schank. *Inside Case-Based Reasoning*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1989. ISBN 0898597676.
- T. L. Saaty. Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, 1(1) :83–98, 2008. doi : [10.1504/IJSSci.2008.01759](https://doi.org/10.1504/IJSSci.2008.01759). URL <https://www.inderscienceonline.com/doi/abs/10.1504/IJSSci.2008.01759>.

- N. Tintarev and J. Masthoff. A survey of explanations in recommender systems. In *Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering Workshop, ICDEW '07*, pages 801–810, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 978-1-4244-0831-3. doi : 10.1109/ICDEW.2007.4401070. URL <http://dx.doi.org/10.1109/ICDEW.2007.4401070>.
- S. E. Toulmin. *The uses of argument*. Cambridge University Press, 2003.
- E. Triantaphyllou. *Multi-Criteria Decision Making Methods*, pages 5–21. Springer US, Boston, MA, 2000. ISBN 978-1-4757-3157-6. doi : 10.1007/978-1-4757-3157-6_2. URL https://doi.org/10.1007/978-1-4757-3157-6_2.
- M. R. Wick and J. R. Slagle. An explanation facility for today’s expert systems. *IEEE Expert : Intelligent Systems and Their Applications*, 4(1) :26–36, Mar. 1989. ISSN 0885-9000. doi : 10.1109/64.21897. URL <https://doi.org/10.1109/64.21897>.
- I. Yevseyeva, C. Morisset, and A. van Moorsel. Modeling and analysis of influence power for information security decisions. *Performance Evaluation*, 98 :36 – 51, 2016. ISSN 0166-5316. doi : <https://doi.org/10.1016/j.peva.2016.01.003>. URL <http://www.sciencedirect.com/science/article/pii/S0166531616000043>.
- L. A. Zadeh. Fuzzy sets. *Information and Control*, 8 :338–353, 1965.