



HAL
open science

Analyse des Préférences et Tournois Pondérés

Alain Guénoche

► **To cite this version:**

| Alain Guénoche. Analyse des Préférences et Tournois Pondérés. 2016. hal-01324077v1

HAL Id: hal-01324077

<https://hal.science/hal-01324077v1>

Preprint submitted on 31 May 2016 (v1), last revised 8 Feb 2017 (v2)

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

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

Analyse des Préférences et Tournois Pondérés

Alain Guénoche

Institut de Mathématiques de Marseille (AMU - CNRS)

*Correspondance : alain.guenoche@univ-amu.fr

Résumé

Dans de nombreuses études expérimentales, on dispose de n éléments ordonnés suivant plusieurs classements (votes, notes ou critères). Nous traitons et comparons deux problèmes : (i) Etablir un classement unique (ordre total) des n items et (ii) sélectionner les k meilleurs éléments parmi n . Il s'agit, dans les deux cas, de minimiser le nombre de préférences qui vont à l'encontre de ces choix.

Mots-Clés

Préférences ; Tournoi ; Ordres consensus ; Sélection candidats

I INTRODUCTION

Les données de préférences sont très particulières ; ce sont fondamentalement des ordres :

- des ordres totaux, c'est à dire des classements sur un ensemble d'individus ou d'items, tous classés, qui n'admettent pas d'ex-aequo ;
- des préordres totaux, qui admettent les ex-aequo. Ce sont donc des ordres totaux sur des classes d'équivalence, constituées des items à égalité du point de vue de l'ordre.

Ces ordres peuvent venir de votes (opinions formulées par des classements sur des individus ou des jugements d'appréciation sur des produits). Ils peuvent aussi provenir de notes attribuées aux items par des critères quantitatifs ou des épreuves comparatives. On retrouve ainsi les problèmes liés aux procédures de dépouillement de scrutins, dont l'origine remonte à Condorcet (1785) (6), aux classements hédonistes basés sur des appréciations d'experts, à la sélection de candidats pour des concours de recrutements ou au classement des élèves selon plusieurs matières.

Soit un groupe E d'experts ($|E| = m$) et un ensemble X d'items ($|X| = n$). Les experts classent les items et définissent un *profil* Π d'ordres ou de préordres totaux.

$$\Pi = \{S_1, S_2, \dots, S_m\}.$$

Nous abordons deux problèmes :

- (i) Etablir un classement collectif, un ordre total sur tous les candidats. Cet ordre *consensus* minimise la somme des préférences en contradiction avec cet ordre. S'il n'est pas unique, on cherchera à énumérer tous les ordres optimaux.
- (ii) Sélectionner un certain nombre d'items sans tenir compte des préférences entre les sélectionnés d'une part ou les rejetés d'autre part. On construit donc un groupe de taille k fixée, sans chercher à classer ou départager les items à l'intérieur des deux classes.

Du point de vue méthodologique, nous allons voir que ces deux problèmes sont différents. Le premier est classique dans le cadre de la *Théorie du choix social* (vote) ou de l'*Agrégation des préférences* selon des comparaisons par paires (1). Pourtant, le problème de sélection est souvent traité via un ordre total, en retenant les k éléments qui sont les mieux classés. Nous verrons que, du point de vue de la minimisation du nombre de préférences contradictoires, c'est une erreur !

Les méthodes développées ci-dessous reposent sur des comparaisons par paires des éléments de X . De chaque ordre ou préordre du profil, on retient que x est meilleur que y , ou l'inverse ou encore qu'ils sont à égalité. Notons $T(x, y)$ le poids des préférences de x en faveur de y . Le plus souvent, $T(x, y)$ est le nombre de classements de x en faveur de y . Mais on peut tenir compte de l'écart entre x et y ou, dans le cas de mesures quantitatives, des différences entre les notes de x et de y . Dans ce cas, $T(x, y)$ est la somme des différences positives entre les valeurs de x et de y et une procédure de normalisation des notes est préconisée.

Ce qu'on veut minimiser,

- dans le problème (i), c'est le nombre de préférences en faveur d'un item y placé après x dans un ordre consensus,
- dans le problème (ii), c'est le nombre de préférences en faveur de y non retenu par rapport à un candidat x sélectionné.

Mais les préférences de x en faveur de y sont compensées par les préférences de y en faveur de x . On considère donc des préférences *majoritaires*, les préférences minoritaires compensant une partie des majoritaires.

Le profil des ordres se modélise à l'aide d'un *tournoi* (graphe complet orienté à n sommets) dont les arcs correspondent aux préférences *majoritaires*. Entre les sommets x et y on met un arc (x, y) de x vers y si et seulement si $T(x, y) > T(y, x)$. Cet arc a pour poids la différence $w(x, y) = T(x, y) - T(y, x)$ et, dans la table W des poids, on posera $w(y, x) = 0$. Si $T(x, y) = T(y, x)$, on peut mettre l'arc dans n'importe quel sens, puisque $w(x, y) = w(y, x) = 0$.

Tout ordre total sur les candidats $O = (x_1, x_2, \dots, x_n)$ distingue deux types d'arcs ; les *arcs directs* qui sont orientés selon l'ordre O et les *arcs-retour* qui sont orientés dans le sens opposé. Soit $W(O)$ la somme des poids de ces arcs-retour qui mesure l'*écart* de l'ordre O au tournoi. Cet ordre est d'autant plus compatible avec le tournoi que $W(O)$ est faible. Mais que faut-il compter dans W ?

- Dans le premier problème, on considère tous les arcs-retour et on cherche donc un ordre total sur X qui minimise

$$W_o(O) = \sum_{i < j} w(x_j, x_i).$$

- Dans le second cas, on cherche seulement deux classes, une bipartition orientée, avec d'un côté les sélectionnés S et de l'autre les rejetés R et on ne compte que les arcs-retour qui vont de la classe R vers S .

$$W_p(S|R) = \sum_{x \in S, y \in R} w(y, x)$$

Souvent, un ordre π à écart minimum d'un tournoi est appelé *ordre médian* pour le profil correspondant. C'est tout à fait justifié dans le cas d'un profil d'ordres totaux, pour la distance de Kendall δ (distance de la différence symétrique sur les paires ordonnées) et si $T(x, y)$ est

le nombre d'ordres du profil qui placent x avant y . Cet ordre consensus π est une *médiane* du profil et

$$\sum_{i=1, \dots, m} \delta(S_i, \pi)$$

est minimum dans l'ensemble des ordres totaux \mathcal{S} sur X (1). Mais si le profil contient des préordres, ou si les pondérations des arcs ne sont pas les nombres d'ordres, un ordre à écart minimum du tournoi n'est pas assuré d'être une médiane. Mais on continuera d'employer, de manière abusive, le terme *ordre médian* pour tout *ordre à écart minimum d'un tournoi*.

5

Pour construire des ordres médians ou une classe d'items sélectionnés, on a recourt à des méthodes d'énumération. Pour améliorer leur efficacité il est nécessaire de disposer de bornes supérieures aux optimima que l'on cherche à atteindre. Elles sont calculées à l'aide d'ordres dont l'écart au tournoi est optimisé de façon approchée. Dans la section 2, on présente deux heuristiques classiques plus une troisième basée sur la décomposition du tournoi. Elle les sur-
10 passe et, à ma connaissance, elle est originale. Elle permet de traiter de larges problèmes et, par exemple, de trouver des ordres alternatifs au classement des universités selon les critères de Shanghai.

Dans la section 3 nous reprenons la méthode *Branch & Bound* développée dans (11; 2; 8) et nous
15 l'adaptions au problème de l'énumération de tous les ordres médians. L'étude d'un petit tournoi, correspondant au choix de conférenciers invités lors d'un colloque et qui admet plusieurs ordres médians, nous amène à étudier la question du choix d'un de ces ordres. La section 4 est dédiée au problème de sélection de k éléments selon des ordres, que l'on résout aussi par énumération. Un dernier tableau des classements de villes universitaires permet de montrer que la détermination
20 des k villes préférées ne correspond pas toujours aux k villes les mieux placées dans un ordre total optimal.

II DES HEURISTIQUES POUR UN "BON" ORDRE

Nous verrons que la construction d'ordre optimal à partir d'un tournoi résulte d'une méthode *Branch & Bound* qui nécessite la connaissance de bornes supérieures de l'écart de cet ordre
25 au tournoi. Ces bornes sont indispensables pour établir un ordre médian, car elles permettent de limiter la taille de l'arborescence de recherche. Nous ne reprenons que deux heuristiques classiques, la méthode gloutonne et une méthode de score (Borda (5)) avant de présenter notre méthode de décomposition. D'autres approches, comme celle de Smith & Payne (20) (retournement de l'arc qui supprime le plus de 3-circuits) s'étant avérées sans efficacité pour les
30 problèmes pondérés, surtout quand leur taille augmente :

Il existe d'autres approches qui relèvent de l'optimisation stochastique, à commencer par le Recuit Simulé (Metropolis), l'optimisation en voisinage variable (12) ou la méthode de bruitage (10). Les paramètres à prendre en compte, les temps de calcul et les codes propres à leurs auteurs font que nous ne les considérons pas dans cette étude. Bien évidemment, toute solution
35 heuristique peut servir de point de départ à toute procédure d'optimisation.

2.1 La méthode gloutonne

Elle procède du même principe que la procédure *Branch & Bound*, sauf qu'elle ne développe pas d'arborescence des sections commençantes et choisit à chaque itération l'item qui promet

un ordre de moindre coût. On calcule donc les sommes colonnes de la table des poids du tournoi

$$Sum(y) = \sum_{x \in X} w(x, y).$$

$Sum(x)$ est la contribution de x à l'écart d'un ordre commençant par x . A chaque itération

- on choisit l'item x tel que $Sum(x)$ est minimum
- on met à jour les valeurs des sommes colonnes : $Sum(y) \leftarrow Sum(y) - w(x, y)$.

Cette heuristique est clairement en $O(n^2)$.

5 2.2 Les méthodes de score (Borda)

Ce sont des méthodes basées sur des fonction de score. Chaque ordre attribue des points à chaque item et le score d'un item est la somme éventuellement pondérée, des points acquis. Suivant la signification des points, l'ordre consensus est l'ordre croissant ou décroissant des scores. L'exemple type est la procédure de Borda, rival de Condorcet pour désigner les vain-
10 queurs d'une élection. Son score est le rang du candidat (1 point pour le premier, 2 pour le second et l'ordre final est l'ordre croissant de la somme des rangs. Dans le cas de préordres, il faut attribuer à chaque item le rang moyen des ex-aequo.

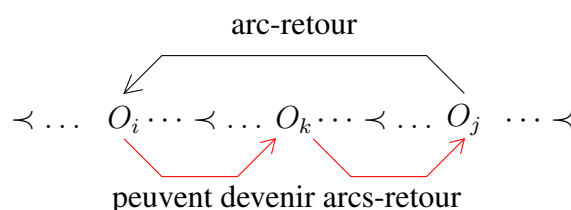
Un autre exemple classique est le championnat des pilotes de Formule 1 ; chaque course attribue des points (non proportionnels) aux dix premiers, et 0 pour les autres. Le classement est l'ordre
15 décroissant de la somme des points acquis par les pilotes.

2.3 Une optimisation locale

A ces deux heuristiques, qui construisent un ordre $O = (o_1, o_2, \dots, o_n)$ sur X , on applique une procédure d'optimisation locale. On recherche pour tout élément o_j , le dernier élément o_i placé avant lui que o_j domine ; on a donc $i < j$, $w(o_j, o_i) > 0$ et (o_j, o_i) est le plus petit arc-retour partant de o_j . Le cas où o_i et o_j sont consécutifs correspond à une procédure très classique : en transposant ces deux éléments, on diminue l'écart au tournoi, puisqu'on supprime un arc-retour. Sinon, on a intérêt à échanger o_j et o_i si les items placés entre o_i et o_j ne créent pas des arcs-retour de poids plus grand. C'est ce que l'on vérifie en sommant les valeurs

$$Som = \sum_{i > k > j} w(o_k, o_j) + \sum_{i > k > j-1} w(o_i, o_k).$$

La première somme porte sur les arcs entrant en o_j et la seconde sur les arcs sortant de o_i qui deviendraient des arcs-retour une fois o_i et o_j échangés. Le dernier poids $w(o_i, o_{j-1})$ n'est pas considéré dans la seconde somme car, s'il est positif, il suffit de transposer o_{j-1} et o_i qui est
20 venu prendre la place de o_j . Si $w(o_j, o_i) > Som$ l'intervalle $(o_i, o_{i+1}, \dots, o_{j-1}, o_j)$ devient soit $(o_j, o_{i+1}, \dots, o_i, o_{j-1})$, soit $(o_j, o_{i+1}, \dots, o_{j-1}, o_i)$ suivant que $w(o_i, o_{j-1})$ est positif ou non.



Cette procédure est appliquée itérativement tant qu'il existe des transpositions de ce type à réaliser. A ma connaissance, elle est originale et rapide puisque, pour tout item, il suffit de remonter vers le dernier item précédent dominé et d'effectuer des sommations dans l'intervalle ainsi défini. Chaque itération est en $O(n^2)$ dans le cas le pire (celui de l'ordre induit par un tournoi transitif pour lequel on cherche en vain un intervalle à retourner).

Dans la suite de ce texte, nous appelons *BestH* la meilleure des deux heuristiques après application de la procédure d'optimisation locale.

2.4 Décomposition d'un tournoi

Dans ce paragraphe, nous abordons la question de la décomposition d'un tournoi de grande taille en sous-tournois plus facilement traitables. Nous montrons, à l'aide de simulations sur des tournois aléatoires, que cette stratégie de décomposition est efficace.

Devant des problèmes de grande taille ($n \gg 30$), les heuristiques classiques sont peu efficaces. D'où l'idée de décomposer le tournoi en sous-tournois, c'est à dire partitionner les items de façon que chaque classe contiennent des éléments voisins dans un ordre optimal. Il suffit alors de calculer un ordre consensus pour chaque classe et de les concaténer pour obtenir un ordre, certes approché, mais plus proche du tournoi que les heuristiques globales. C'est ce que nous allons tester.

2.4.1 Une décomposition régulière

L'ordre donné par l'une des heuristiques précédentes induit une décomposition régulière. Etant donné un nombre de classes q , il suffit de construire les classes comme des intervalles de cet ordre. Les n/q premiers vont dans la première classe, les n/q suivants dans la seconde, etc. On obtient ainsi une partition en classes équilibrées, notée P_R .

2.4.2 Une partition des items basée sur une distance

En considérant la table W des poids des arcs du tournoi, on peut associer à chaque item x une bipartition : Soit x_+ , les éléments de X qui devraient être placés avant x parce qu'ils le dominent et x_- , qui devraient être placés après x parce qu'il les domine.

$$x_+ = \{z \in X | w(z, x) > 0\} \text{ et } x_- = \{z \in X | w(x, z) > 0\}.$$

A l'aide de ces bipartitions, on construit un indice de dissimilarité sur X ,

$$D(x, y) = \Delta(x_+, y_+) + \Delta(x_-, y_-),$$

dans lequel $\Delta(x_+, y_+)$ est la distance de la différence symétrique entre les ensembles x_+ et y_+ (resp. x_- et y_-).

Remarque : D n'est pas une distance, car il se peut que $D(x, y) = 0$ si $w(x, y) = w(y, x) = 0$.

Proposition

Si T est un tournoi transitif,

- deux éléments consécutifs dans l'ordre optimal ont une distance minimum égale à 2 ;
- $D(x, y)$ est proportionnelle à la différence des rangs de x et y dans l'ordre médian.

Preuve

Soit $x \prec y$ deux éléments consécutifs dans l'ordre correspondant à un tournoi transitif. Les classes x_+ et y_+ (resp. x_- et y_-) ne diffèrent que d'un seul élément, x (resp. y) et donc $D(x, y) = 2$. De même, si x et y sont séparés par k items dans l'ordre médian, $D(x, y) = 2(k + 1)$. Ainsi les valeurs sont croissantes quand on s'écarte de la diagonale, et D est une distance (car il n'y a pas d'indifférence dans les préférences).

Ainsi, des classes homogènes selon D sont des classes d'éléments proches dans l'ordre médian.

Pour construire, à partir de D une partition en q classes, notée P_M , on cherche à minimiser la somme des distances intra-classe. Le choix du nombre de classes, qui conditionne la taille moyenne des classes, donc des sous-tournois, reste à faire. Il sera conduit par une procédure de simulation.

Etant donnée une partition des n items en q classes, notée $P = \{P_1, \dots, P_q\}$, le critère tend à minimiser la somme des distances moyennes de chaque item aux éléments de sa classe.

$$\sum_{k=1}^q \left[\sum_{x \in P_k} \frac{1}{|P_k|} \sum_{y \in P_k} D(x, y) \right]$$

La partition P_M calculée est obtenue par une procédure itérative à la manière des k -means. On part de la partition triviale qui ne contient que des singletons. A chaque itération un item est transféré vers la classe pour laquelle sa moyenne de distance est minimum. La procédure s'arrête quand il n'y a plus d'item à déplacer.

Pour chaque classe de l'une ou l'autre de ces partitions, on calcule

- son rang moyen, égal à la moyenne des rangs de ses éléments dans BestH ;
- le sous-tournoi relatif à la classe, avec les valeurs extraites de W ;
- un ordre qui minimise l'écart au sous-tournoi (procédure gloutonne).

Ensuite les ordres relatifs à chaque classe sont concaténés suivant les rangs moyens croissants selon BestH. On obtient ainsi un *ordre composé* (des sous-ordres relatifs à chaque classe). L'ordre obtenu après la procédure d'optimisation locale est noté $Comp_R$ (resp. $Comp_M$), suivant la méthode de partitionnement choisie et son écart au tournoi est calculé.

2.4.3 Complexité, efficacité

Le calcul de la table des $O(n^2)$ valeurs de distances est en $O(n^3)$, puisque pour chaque paire d'items on compare les positions relatives de n éléments. L'algorithme de décomposition régulière est linéaire. La partition P_M est obtenue par un algorithme itératif, dont on ignore le nombre d'itérations (comme pour les k -means), mais qui est bien connu pour son efficacité. Après, on applique l'heuristique gloutonne à chaque classe et la procédure d'optimisation à l'ordre composé.

Notons que la méthode de l'*Ordre composé* est rapide. Il faut compter 1''20 pour P_R , 19''30 pour P_M sur un tournoi à 1000 items sur un MacPro de 2010.

2.4.4 Simulations et résultats

Sur des profils aléatoires

On génère des profils d'ordres totaux en tirant au hasard des permutations d'ordre n (17).
 5 Chaque profil permet de calculer le tournoi avec les poids des arcs (matrice W). Après application des optimisations locales, on dispose de deux ordres totaux. La meilleure des solutions, notée BestH, est celle qui minimise l'écart au tournoi.

On se fixe alors le nombre de classes q . Pour le partitionnement des items, on calcule D avec les bipartitions $P_x = (x_+|x_-)$. On a :

- 10 — la partition régulière (P_R) en suivant l'ordre donné par BestH qui aboutit à l'ordre $Comp_R$;
- une partition (P_M) qui cherche à minimiser les moyennes des distances intra classes et qui aboutit à l'ordre $Comp_M$;

Pour BestH et chacun de ces ordres, on va calculer leur écart au tournoi.

15 Les tests portent sur 100 profils de mêmes paramètres ; à nombre d'items fixé n , on peut faire varier le nombre d'ordres totaux dans le profil m et le nombre de classes de la décomposition q . Chaque ligne de la table ci-dessous donne l'écart moyen aux tournois ainsi générés.

n	m	q	BestH	$Comp_R$	$Comp_M$
50	10	2	153	152	151
50	20	3	259	251	249
50	30	4	330	319	318
100	10	3	755	731	719
100	20	3	1236	1203	1187
100	30	3	1569	1534	1522
200	30	4	6825	6678	6614
200	50	5	9047	8820	8784
200	100	6	13129	12762	12782
500	100	5	86600	85336	85107
500	100	10	86600	84670	85635
500	100	15	86600	84636	86078

TABLE 1 : Valeurs moyennes des écarts obtenus par les heuristiques sur des tournois générés par des permutations aléatoires.

20 Il est clair que les ordres composés sont largement meilleurs que la meilleure des heuristiques classiques. Ils sont systématiquement gagnants, sauf pour quelques problèmes avec $n = 100$. Soulignons que ce sont des écarts moyens et que, pour un profil ou pour un tournoi particulier, les deux décompositions peuvent être testées. Mais on ne sait pas quelle méthode est la plus efficace, ni combien de classes assurent un écart minimum. La partition P_M semble donner les
 25 meilleurs résultats, mais peut être que le nombre de classes est trop petit, comme le suggère le nombre croissant de classes pour $n = 500$.

Sur des tournois à écart borné

30 En tirant au hasard des ordres totaux indépendants, on génère des tournois qui sont loin d'être transitifs et donc les ordres obtenus sont d'écart élevé. Et il est impossible de savoir s'ils sont

proches de l'optimum. D'un certain point de vue, il est inutile de chercher un ordre consensus à ces profils, puisqu'aucune tendance commune ne peut émerger. Nous allons appliquer notre programme de simulation à des tournois générés à partir d'un ordre unique en transposant un certain nombre de préférences. Soit t le paramètre qui définit le nombre de transpositions.

- 5 On génère m ordres totaux à partir de l'ordre naturel en échangeant, dans chacun, t paires d'items aléatoirement choisis. Le tournoi pondéré est alors calculé selon la procédure majoritaire classique. Si t est faible par rapport à n , la tendance générale est alors l'ordre naturel, qui a toutes les chances d'être médian.

L'essai porte sur deux familles d'ordres ; des ordres à 100 (resp. 300) items, sur lesquels on effectue 20 (resp. 100) transpositions avant de calculer le tournoi pondéré. Les décompositions sont en 5 (resp. 10) classes.

n	m	t	q	$BestH$	$Comp_R$	$Comp_M$	$OrdNat$
100	10	20	5	81.7	73.1	66.3	111.9
100	20	20	5	22.6	21.1	19.5	23.9
100	30	20	5	5.3	5.1	5.2	5.9
300	30	100	10	2065	1721	1434	1555
300	50	100	10	800	677	558	544
300	100	100	10	56.1	50.6	49.1	45.9

TABLE 2 : Valeurs moyennes des écarts obtenus par les heuristiques sur des tournois à écart borné.

- 15 Plus le nombre d'ordres est grand, plus le tournoi est transitif et l'ordre naturel proche de l'optimal. La seconde méthode de décomposition s'avère très efficace quand on s'en éloigne.

2.5 Le classement de Shanghai : un tournoi à polémique

Un des plus célèbres et controversés problèmes d'Agrégation des Préférences est celui du classement des Universités fait à Shanghai. Il y a 500 universités et 6 critères de comparaison que je ne veux, ni ne peux, mettre en cause ici. Pour une critique, à laquelle j'adhère pleinement, je renvoie à l'article de Billaut et al.(4). A titre d'illustration de la méthode de décomposition, je considère les notes données par Shanghai aux 150 premières universités dans son classement final de l'année 2010¹. Parmi leurs six critères, quatre ont une importance double, si bien que l'on a l'équivalent de 10 critères. Je traite donc les mêmes données que celles qui donnent lieu à l'ordre final de Shanghai.

A partir de ces notes, je commence par construire un tournoi majoritaire en comparant les universités deux par deux, c'est à dire en comptant le nombre de critères qui place l'université x avant l'université y . En cas d'égalité des notes, aucune préférence n'est enregistrée. Finalement, le tournoi majoritaire est calculé et donc $T(x, y)$ est égal au nombre de critères qui placent x avant y , moins ceux qui placent y avant x (sur les 10 en tenant compte des poids). Ce tournoi est donc "naturellement" pondéré.

On optimise donc W_o , l'indice de Kemeny égal à la somme des poids des arcs à inverser pour rendre le tournoi transitif. BestH donne un ordre dont l'écart est 994, alors que l'ordre de Shanghai est à 1827 ! Commençons par donner les valeurs de W_o suivant le nombre de classes choisi pour la décomposition selon les deux méthodes de décompositions (régulières et calculées) :

1. dont les données m'ont été transmises par les auteurs de (4) que je remercie vivement.

q	5	6	7	8	9	10	11	12	13
$Comp_R$	941	924	946	928	903	929	933	925	921
$Comp_M$	887	918	870	868	864	882	923	872	891

TABLE 3 : Somme des poids des arcs-retour selon le nombre de classes cherchées avec les méthodes $Comp_R$ et $Comp_M$.

On commencera par noter que le poids des arcs à inverser en calculant les classes est toujours inférieur à celui obtenu en considérant un découpage régulier de l'ordre induit par la meilleure heuristique. Ensuite que toutes les valeurs de W_o sont inférieures à celle de la meilleure heuristique (994). Enfin que le minimum est obtenu (864) pour 9 classes calculées et que l'ordre obtenu est nettement supérieur à celui de Shanghai, selon W_o .

Quelles sont les différences entre ces deux classements ? Elles ne sont pas très importantes pour les 20 premières, encore que Stanford, classée 3-ième par Shanghai, permute avec le MIT classé 4-ième (le MIT l'emporte selon 6 critères sur 10), que l'University of Chicago passe du 9-ième rang au 20-ième et que The University of Tokyo, classée 20-ième gagne 7 places. Il y a donc une des universités du top 20 qui perd ce privilège, c'est la Johns Hopkins University (18-ième) qui passe au rang 21. Shanghai ne classe précisément que les 100 premières universités. Il y a des changements importants dans le milieu du classement et vers la fin. Voici les écarts de plus de 10 places :

En gain :

- 47 : University of California Irvine → 35
- 68 : University of Florida → 53
- 75 : Osaka University → 63
- 95 : Texas A&M University - College Station → 71
- 96 : University of Virginia → 75

En perte :

- 9 : University of Chicago → 20
- 42 : Karolinska Institute → 54
- 45 : University of Paris Sud (Paris 11) → 72
- 53 : Vanderbilt University → 67
- 56 : Technical University Munich → 78
- 67 : Uppsala University → 79
- 71 : Ecole Normale Supérieure - Paris → 120
- 72 : The Hebrew University of Jerusalem → 86
- 76 : University of Oslo → 113
- 79 : Stockholm University → 117
- 85 : University of Nottingham → 98
- 86 : Michigan State University → 96
- 88 : McMaster University → 99
- 90 : Ghent University → 104
- 91 : Indiana University Bloomington → 107
- 93 : University of Bonn → 121
- 97 : Case Western Reserve University → 119
- 100 : University of Birmingham → 113

Au delà du 100-ième rang, le classement indique des lots de 50 universités sans donner un rang précis à celles qui sont placées entre 101-ième et 150-ième. Cela fait 23 Universités, sur les 100 premières, dont le rang varie d'au moins 10 places. Dix d'entre elles passent au dessous du

centième rang. C'est donc que 10 universités classées entre 101 et 150 entrent dans ce Top100 : Il s'agit, dans l'ordre calculé de :

- Emory University → 77
- University of California, Santa Cruz → 78
- 5 — University of California, Riverside → 82
- Georgia Institute of Technology → 84
- University of Iowa → 85
- Weizmann Institute of Science → 92
- The University of Queensland → 93
- 10 — Tokyo Institute of Technology → 95
- Oregon State University → 99
- University of Alberta → 100

Que penser de ces importants changements ? Simplement que le classement de Shanghai est le résultat d'une fonction de score qui n'est pas explicite sur leur site. Comme on le sait depuis longtemps, à partir du même scrutin, c'est la façon de dépouiller qui fait le gagnant. Ici, c'est le même phénomène que l'on observe et, plutôt que de faire des efforts dispendieux pour améliorer leur classement, les universités devraient imposer un choix de critères plus large et plus convainquant et une méthode de consensus claire.

2.6 Conclusion

20 La stratégie de décomposition est toujours gagnante et donc il est préférable d'assembler des ordres obtenus sur les sous-tournois en un ordre unique, plutôt que de traiter le tournoi d'un seul bloc. De plus, pour des tournois "presque transitifs", les ordres obtenus par décomposition restent proches de l'ordre optimal.

25 Donc, pour un tournoi particulier, je commencerai par chercher le nombre optimal de classes à l'aide du découpage régulier qui est très rapide, puisqu'il n'y a même pas de distance à calculer. Et autour de cette valeur, je testerai les algorithmes de partitionnement. Un dernier essai sur un tournoi à 1000 items a donné par une décomposition régulière en 15 classes le plus faible écart, et ce très nettement.

III DES ORDRES OPTIMAUX

30 Construire un ordre total π dont l'écart au tournoi $W_o(\pi)$ est minimum (le problème de Kemeny (15)), revient à rendre le tournoi transitif par retournement d'arcs, ceux dont la somme des poids est minimum. Ce problème du *minimum feedback arc set* a une longue histoire algorithmique, qui débute avec Slater (18), Remage & Thomson (19), Bermond (3) pour les tournois non pondérés, puis tout un groupe centré autour de la Maison des Sciences de l'Homme avec
35 (16; 11; 2; 8; 14) pour les tournois pondérés.

Depuis Hudry (13), le problème a été prouvé NP-difficile. Le calcul s'effectue par une méthode de *Branch & Bound* (11) dont nous rappelons tout d'abord le principe. On développe une arborescence dont les sommets sont des *sections commençantes*. Ce sont des débuts d'ordres totaux qui peuvent se prolonger en un ordre total optimal. C'est à dire que le poids de cette section est
40 au plus égal à la borne supérieure B_o calculée par l'une des heuristiques précédentes.

Une feuille de l'arborescence courante est donc une chaîne (o_1, o_2, \dots, o_k) sur une partie $X_s \subset X$ et notons $Y = X - X_s$ l'ensemble des éléments non classés. Le poids W_o^k de cette section

est égal à la somme des poids des arcs-retour internes à cette section plus la somme des poids des arcs dont l'origine y est hors cette section et l'extrémité dans cette section.

$$W_o^k(o_1, o_2, \dots, o_k) = \sum_{1 \leq i < j \leq k} w(o_j, o_i) + \sum_{y \in Y} \sum_{j=1, \dots, k} w(y, o_j).$$

Cette valeur est inférieure ou égale au poids de tout ordre total ayant cette section commençante, puisqu'elle ne tient pas compte de l'ordre non encore déterminé sur Y .

A chaque étape, on répète les opérations :

- Trouver une feuille de l'arborescence de poids minimum,
- Prolonger cette section de toutes les façons possibles par tous les éléments $y \in Y$ tels que

$$W_o^{k+1}(o_1, o_2, \dots, o_k, y) \leq B_o.$$

- 5 En prolongeant l'arborescence, à chaque itération, par la feuille de poids minimum, on aboutit à un ordre total sur X dont l'écart est minimum. Sa valeur est déterminée par le premier ordre total obtenu. Si l'on veut tous les ordres médians, il suffit de prolonger l'arborescence sans dépasser cet écart optimal.

Si le tournoi contient beaucoup de circuits, cette méthode peut devenir inapplicable et ce dès que n atteint quelques dizaines d'items (2). Pour deux raisons ; l'une est la taille de l'arborescence qui peut dépasser le million de noeuds ² et l'autre le temps de calcul, à cause de la recherche de la feuille de l'arborescence de poids minimum que l'on prolonge d'abord pour être sûr d'aboutir à un ordre total optimum. Cette difficulté a été contournée par Woirgard (21) en adaptant une structure de données plus élaborée.

15 De façon plus efficace, pour obtenir tous les ordres totaux à écart minimum du tournoi, au lieu de développer l'arborescence à partir d'une feuille de moindre poids (stratégie du meilleur d'abord), on la développe en largeur d'abord en prolongeant la section (o_1, o_2, \dots, o_k) de tous les éléments de Y acceptables. Ceci évite d'avoir à chercher à chaque pas la feuille de poids minimum. La taille de l'arborescence dépend fortement de la qualité de B_o . De fait, on énumère tous les ordres totaux dont l'écart est inférieur à B_o , borne que l'on met à jour au fil de l'énumération, s'il y a lieu. On peut même fixer une borne inférieure à B_o pour laquelle l'arborescence n'aboutit à aucun ordre si elle est inférieure à l'écart minimum, et ce très rapidement.

3.1 Le tournoi Jobim 2016

Nous prenons comme exemple tout au long de ce paragraphe le choix de conférenciers invités pour le colloque national de bioinformatique, Jobim. Chaque membre du comité de programme désigne les conférenciers qu'il voudrait faire venir et il les range dans un ordre préférentiel. Ceux-ci sont à prendre dans un ensemble ouvert. Les membres du comité ne se sont pas trop dispersés et ont nommé 10 candidats ; nous avons retenu ceux qui étaient dans au moins la moitié des ordres. Il est resté $n = 7$ candidats (les 3 autres étant éliminés). Comme scrutin, nous avons repris les ordres donnés par les membres du comité. Pour chacun, on a enlevé les éliminés et ajouté, en fin de classement et tous ex-aequo, les candidats non retenus pour obtenir un préordre total.

1. $(x_1 \prec x_2 \prec x_3 \prec x_4 \prec x_5, x_6, x_7)$

2. Pour chaque noeud, selon une structure de données déjà élaborée dans (11), il suffit de mémoriser 3 entiers.

2. $(x_5 \prec x_1 \prec x_2 \prec x_4 \prec x_6 \prec x_3, x_7)$
3. $(x_7 \prec x_6 \prec x_2 \prec x_3 \prec x_1, x_4, x_5)$
4. $(x_5 \prec x_3 \prec x_4 \prec x_1, x_2, x_6, x_7)$
5. $(x_4 \prec x_7 \prec x_5 \prec x_6 \prec x_2 \prec x_1, x_3)$
6. $(x_6 \prec x_1 \prec x_7 \prec x_2 \prec x_3, x_4, x_5)$

TABLE 4 : Préordres totaux des 6 membres du comité de programme.

Le tournoi majoritaire résultant de ces préférences, avec le poids des arcs, est donné dans la Table 5

W	x_1	x_2	x_3	x_4	x_5	x_6	x_7
x_1	-	1	1	1	0	0	1
x_2	0	-	4	2	0	0	0
x_3	0	0	-	1	0	0	0
x_4	0	0	0	-	0	2	2
x_5	1	0	1	0	-	1	0
x_6	1	1	2	0	0	-	0
x_7	0	1	1	0	1	0	-

TABLE 5 : Tournoi majoritaire résultant des préordres de la Table 4.

On constate que tous les candidats sont dominés au moins une fois. Il n'y a donc pas de *vainqueur de Condorcet*. Pour déterminer un ordre total ou une bipartition de poids minimum on applique l'heuristique gloutonne, la taille du tournoi ne nécessitant pas une décomposition. Ici, le sommet le moins dominé est x_5 ; un ordre commençant par x_5 n'aura qu'un seul arc-retour de poids 1 (x_7, x_5). Ce sommet étant "placé", on retient x_1 qui n'entraîne qu'un autre arc-retour de poids 1 (x_6, x_1). Et ainsi de suite. On aboutit à l'ordre glouton :

$$O_g = (x_5 \prec x_1 \prec x_2 \prec x_4 \prec x_6 \prec x_7 \prec x_3),$$

et $W_o(O_g) = 5$ avec $\{(x_7, x_5), (x_6, x_1), (x_6, x_2), (x_7, x_2), (x_3, x_4)\}$, comme arcs-retour, tous de poids 1, donc $B_o = 5$. Un ordre optimal est au plus à écart 5 du tournoi.

La procédure de Borda appliquée aux préordres initiaux (en donnant un rang moyen aux candidats à égalité) conduit à deux paires d'ex-aequo $\{x_1, x_5\}$ et $\{x_4, x_7\}$. En respectant les orientations des arcs du tournoi, on aboutit à :

$$O_b = (x_2 \prec x_5 \prec x_1 \prec x_6 \prec x_4 \prec x_7 \prec x_3)$$

qui présente 6 arcs-retour et $W_o(O_b) = 7$, un écart supérieur à O_g .

La procédure de recherche d'un ordre unique aboutit à un autre ordre O_m à écart 5, ce qui confirme l'optimalité de B_o .

$$O_m = (x_4 \prec x_7 \prec x_5 \prec x_6 \prec x_1 \prec x_2 \prec x_3).$$

On lance donc la procédure d'énumération qui donne les 13 ordres médians suivant :

1. $O_1 = (x_1 \prec x_2 \prec x_4 \prec x_7 \prec x_5 \prec x_6 \prec x_3)$
2. $O_2 = (x_1 \prec x_4 \prec x_7 \prec x_5 \prec x_6 \prec x_2 \prec x_3)$
3. $O_3 = (x_4 \prec x_5 \prec x_6 \prec x_1 \prec x_7 \prec x_2 \prec x_3)$

4. $O_4 = (x_4 \prec x_7 \prec x_5 \prec x_6 \prec x_1 \prec x_2 \prec x_3)$
5. $O_5 = (x_5 \prec x_1 \prec x_2 \prec x_4 \prec x_6 \prec x_7 \prec x_3)$
6. $O_6 = (x_5 \prec x_1 \prec x_2 \prec x_4 \prec x_7 \prec x_6 \prec x_3)$
7. $O_7 = (x_5 \prec x_1 \prec x_4 \prec x_6 \prec x_7 \prec x_2 \prec x_3)$
8. $O_8 = (x_5 \prec x_1 \prec x_4 \prec x_7 \prec x_6 \prec x_2 \prec x_3)$
9. $O_9 = (x_5 \prec x_4 \prec x_6 \prec x_1 \prec x_7 \prec x_2 \prec x_3)$
10. $O_{10} = (x_5 \prec x_6 \prec x_1 \prec x_2 \prec x_3 \prec x_4 \prec x_7)$
11. $O_{11} = (x_5 \prec x_6 \prec x_1 \prec x_2 \prec x_4 \prec x_7 \prec x_3)$
12. $O_{12} = (x_5 \prec x_6 \prec x_1 \prec x_7 \prec x_2 \prec x_3 \prec x_4)$
13. $O_{13} = (x_7 \prec x_5 \prec x_6 \prec x_1 \prec x_2 \prec x_3 \prec x_4)$

TABLE 6 : Ordres médians à écart 5 du tournoi de la Table 5.

3.2 Simulations

Est-ce que ces difficultés sont fréquentes ou exceptionnelles ? Pour en avoir le coeur net, nous avons tiré au hasard des profils d'ordres totaux. Pour obtenir des chiffres crédibles pour les invités d'un colloque, nous avons fixé à $n = 20$ le nombre de candidats et à $m = 31$ le nombre d'experts, un nombre impair pour éviter les égalités et donc les arcs de poids nul.

Le plus simple est de partir d'ordres aléatoires, de calculer le tournoi majoritaire, puis d'énumérer tous les ordres d'écart minimum et de compter le nombre moyen de solutions en répétant 100 fois ce tirage. Mais en tirant des ordres aléatoires, on simule une situation bien improbable ; qu'il n'y ait pas de consensus véritable entre les experts non plus que de renommée relative entre les candidats. Pour graduer ce degré d'entente entre les experts, nous sommes partis de l'ordre naturel et simulons les votes en effectuant t échanges de deux items tirés au hasard, pour tous les ordres. Ainsi, si $t = 0$ tout le jury est d'accord, et quand t croît on s'approche du désordre aléatoire qui correspond à $t = 30$.

Pour ces quatre jeux d'essais, $t = 10, 15, 20, 30$, nous tirons donc 100 profils et comptons la valeur moyenne d'écart donnée par l'heuristique gloutonne (O_g), l'ordre de Borda (O_b), un ordre médian (O_m), le nombre moyen d'ordres optimaux ($NbOpt$), le nombre maximum ($NbMax$), la taille moyenne de l'arbre ($Tree$), ainsi que son maximum ($Tree_{Max}$).

t	O_g	O_b	O_m	$NbOpt$	$NbMax$	$Tree$	$Tree_{Max}$
10	9.9	14.8	9.2	2.1	18	187	6695
15	23.2	27.2	20.5	3.3	28	2116	53544
20	35.0	38.7	30.2	4.0	40	12331	964554
30	42.5	47.0	36.7	4.1	40	47766	1391501

TABLE 7 : Ecart moyen des heuristiques, d'un ordre médian, les nombres moyens et maximum d'ordres optimaux et les nombres moyens et maximum de noeuds de l'arbre de recherche.

La différence entre l'écart donné par l'heuristique et l'écart optimal paraît faible, mais il y a beaucoup d'ordres entre ces deux valeurs, ce qui souligne la supériorité (en moyenne) de O_g sur O_b . Le nombre moyen de solutions optimales reste limité, mais le nombre maximum observé est important et justifie les paragraphes ci-dessous pour décider d'un classement unique des candidats. Enfin la taille de l'arbre est importante, mais les calculs sont très rapides, quelques secondes pour les 100 problèmes les plus difficiles.

3.3 Choisir parmi les ordres médians

En présence de plusieurs ordres médians on aimerait n'en garder qu'un pour établir un ordre consensus unique. La tâche n'est pas facile. Sur les 13 ordres optimaux du tournoi Jobim, il y a 4 premiers, x_1, x_4, x_5 et x_7 , dont deux d'entre eux peuvent être derniers. Tous les candidats, 5 sauf x_3 , sont dans les deux premiers rangs. Voici quelques solutions :

3.3.1 Le plus central des ordres médians

De tous ces ordres, quel est le plus central ? C'est celui dont la somme des distances aux autres ordres médians est minimum. Il suffit de calculer la distance de Kendall (nombre de paires d'items en désaccord entre deux ordres) et d'en faire les sommes ligne par ligne :

δ	1	2	3	4	5	6	7	8	9	10	11	12	13	Sum
1	0	4	9	8	5	4	8	7	10	10	8	11	10	94
2	4	0	5	4	7	6	4	3	6	10	8	9	8	74
3	9	5	0	3	6	7	3	4	1	7	5	6	9	65
4	8	4	3	0	9	8	6	5	4	10	8	9	6	80
5	5	7	6	9	0	1	3	4	5	5	3	6	9	63
6	4	6	7	8	1	0	4	3	6	6	4	7	8	64
7	8	4	3	6	3	4	0	1	2	6	4	5	8	54
8	7	3	4	5	4	3	1	0	3	7	5	6	7	55
9	10	6	1	4	5	6	2	3	0	6	4	5	8	60
10	10	10	7	10	5	6	6	7	6	0	2	3	6	78
11	8	8	5	8	3	4	4	5	4	2	0	3	6	60
12	11	9	6	9	6	7	5	6	5	3	3	0	3	73
13	10	8	9	6	9	8	8	7	8	6	6	3	0	88

TABLE 8 : Distance de Kendall entre les ordres médians de la table 6.

C'est O_7 le plus central, d'un courte tête devant O_8 avec lequel il n'a qu'un seul désaccord sur la paire (x_6, x_7) .

3.3.2 Le médian des médians

15 Si l'on veut rester dans l'esprit des ordres à écart minimum du tournoi, on peut refaire un tournoi avec les ordres médians considérés comme des votes. Cette fois-ci, le tournoi majoritaire résultant est :

W	x_1	x_2	x_3	x_4	x_5	x_6	x_7
x_1	-	13	13	7	0	0	9
x_2	0	-	13	1	0	0	0
x_3	0	0	-	0	0	0	1
x_4	0	0	7	-	0	5	9
x_5	9	11	13	5	-	13	5
x_6	1	7	13	0	0	-	1
x_7	0	3	11	0	0	0	-

TABLE 9 : Tournoi majoritaire des ordres médians de la Table 6.

20 On calcule donc les ordres médians du *tournoi des médians* qui donne un ordre unique, O_7 , le même que précédemment. Mais il n'est nullement garanti qu'on aboutisse à un seul ordre, ni même que la procédure répétée itérativement converge ; elle peut aboutir à un cycle. De plus, le ou les ordres obtenus ne sont pas nécessairement médians pour le tournoi initial !

3.3.3 Le plus souvent dans les k premiers

Si l'on avait un seul invité, on prendrait x_5 parce qu'il est le plus souvent le premier dans l'ensemble des ordres médians. Et en seconde position vient celui qui, en dehors de x_5 est le plus souvent premier ou second. C'est x_1 . Et ainsi de suite ; on place au rang k un candidat non placé qui est le plus souvent (plus que les autres) à un rang inférieur ou égal à k . Cette procédure fonctionne comme un scrutin uninominal à un tour qui est répété k fois. Ici, on aboutit encore au même ordre O_7 .

Ce n'est pas équivalent à la procédure de Borda appliquée aux ordres médians, même si ici elle donne le même résultat, car on ne tient aucun compte des mauvais classements.

10 IV A LA RECHERCHE D'UNE SÉLECTION

Dans ce paragraphe, nous cherchons à sélectionner les k meilleurs éléments parmi n en minimisant le nombre de préférences qui vont à l'encontre de ce choix. Précisons qu'ici le paramètre k est fixé. Les arcs-retour entre les sélectionnés S ou entre les rejetés R ne comptent pas, puisqu'ils ne contredisent en rien la sélection finale. On cherche donc deux classes et on ne compte que les arcs-retour qui vont de la classe R vers S .

$$W_p(S|R) = \sum_{x \in S, y \in R} w(y, x)$$

Entre deux classes complémentaires de sommets dans un graphe orienté, on parle de *cocircuit*. On cherche donc un cocircuit dont les parties sont de cardinal fixé et dont le poids des arcs-retour est minimum.

15 Du point de vue méthodologique, nous allons voir que ce problème est très différents de celui de l'ordre consensus. Souvent, le problème de sélection est traité via un classement de tous les items, en retenant les k éléments qui sont les mieux classés. Nous verrons que, pour minimiser W_p , c'est souvent une erreur !

Pour les problèmes de sélection, on a souvent recours à un *vote par approbation*. Chaque votant désigne, sans restriction du nombre, les candidats qu'il approuve ; les candidats sélectionnés sont ceux qui sont le plus souvent nommés. Nous verrons également qu'il s'agit d'une très mauvaise procédure.

Comme précédemment, un ordre total sur X dont l'écart au tournoi soit faible est nécessaire car il fournit une borne supérieure au critère W_p . Cet ordre permet de fixer le poids minimum d'un cocircuit d'arcs-retour sur une classe S de cardinal k , notée B_p^k .

25 Pour illustrer ce paragraphe, nous reprenons le tournoi des conférenciers invités au colloque Jobim. Examinons tout d'abord les ordres donnés par les heuristiques classiques plus la procédure par approbation :

Selon l'ordre glouton $O_g = (x_5 \prec x_1 \prec x_2 \prec x_4 \prec x_6 \prec x_7 \prec x_3)$, une bipartition avec deux invités, x_1 et x_5 donnent $W_p^2(x_1, x_5 | x_2, x_3, x_4, x_6, x_7) = 2$ (avec les arcs-retour $\{(x_6, x_1), (x_7, x_5)\}$), donc $B_p^2 = 2$.

L'ordre de Borda $O_b = (x_2 \prec x_5 \prec x_1 \prec x_6 \prec x_4 \prec x_7 \prec x_3)$ suggère une partition à deux invités, avec $W_p^2(x_2, x_5 | x_1, x_3, x_4, x_6, x_7) = 4$, le double du précédent.

Pour un vote par approbation nous considérons que les listes des candidats approuvés pour sélectionner k candidats s'arrêtent aux k premiers. Pour un candidat, c'est x_5 qui figure le

plus souvent au premier rang. Pour deux candidats, c'est x_1 qui est le plus souvent dans les 2 premières places et pour le second sélectionné, il faut choisir entre x_5, x_6 et x_7 qui sont nommés 2 fois ; nous avons vu que seule la paire $\{x_1, x_5\}$ est optimale. Si l'on voulait sélectionner 3 candidats, sur les trois premiers rangs, x_1, x_2, x_5 et x_7 sont tous les quatre nommés 3 fois ; il faut donc écarter l'un d'entre eux, mais un seul triplet $\{x_1, x_5, x_7\}$ est optimal et il y en a un second $\{x_1, x_5, x_6\}$ qui ne peut être détecté par approbation.

4.1 Une sélection optimale

Ce problème semble encore plus difficile³ que celui de l'ordre médian, car la construction de cette bipartition n'est pas cumulative. Bien sûr, on est tenté de prendre comme "premier" invité le candidat le moins dominé. Mais il se peut qu'un candidat plus dominé puisse être retenu par la suite avec celui qui le domine fortement. La cause en est la non monotonie de la fonction de poids W_p quand le nombre de sélectionnés augmente, alors que W_o l'était quand on prolongeait les sections commençantes.

Prenons l'exemple du petit tournoi pondéré de la Table 10 :

W	x_1	x_2	x_3	x_4	x_5	x_6
x_1	-	5	0	1	1	1
x_2	0	-	5	1	1	1
x_3	5	0	-	1	1	1
x_4	0	0	0	-	1	1
x_5	0	0	0	0	-	1
x_6	0	0	0	0	0	-

TABLE 10 : Tournoi pour illustrer la non monotonie de W_p^k quand k augmente.

S'il n'y a qu'un invité, c'est x_4 de poids 3, s'il y en a deux, c'est l'une des trois paires $\{x_1, x_2\}$, $\{x_1, x_3\}$, $\{x_2, x_3\}$ de poids 5. Mais si l'on en prend trois, c'est $\{x_1, x_2, x_3\}$ car cette partie domine tous les autres et $W_p^3(x_1, x_2, x_3 | x_4, x_5, x_6) = 0$.

Donc, si l'on veut une partie à 1 élément, il suffit de comparer les sommes colonnes de la table du tournoi. Pour 2 ou 3 éléments, il faut effectuer quelques boucles imbriquées. Mais si l'on veut une partie à $k > 3$ éléments, je ne vois pas de méthode optimale autre que l'énumération des parties à k éléments parmi N .

4.2 Des bornes efficaces

Heureusement, cette énumération ne porte pas nécessairement sur toutes les parties de X à k éléments. D'abord un ordre total construit à l'aide d'une heuristique fournit une borne supérieure B_p^k au poids du cocycle retour. Ensuite, pour chaque item x , on peut établir deux bornes, $B_S(x)$ (resp. $B_R(x)$) correspondant aux poids minimum des arcs-retour si $x \in S$ (resp. $x \in R$). Si x est sélectionné ($x \in S$), il peut être extrémité d'éventuels arcs-retour. Il y en a au plus $N - k$ dont le poids est supérieur ou égal à la somme des $N - k$ plus petites valeurs des arcs d'extrémité x , c'est à dire les $N - k$ plus petites valeurs de la colonne x de la table W :

$$B_S(x) = \sum_{i=1, \dots, N-k} \text{Min}\{w(y_i, x)\}_{y_i \neq x}.$$

3. Non pas au sens de la théorie de la complexité, mais du calcul pratique

De même, si x est rejeté ($x \in R$), il peut être à l'origine de k arcs-retour, soit au minimum la somme des k plus petites valeurs de la ligne de x .

$$B_R(x) = \sum_{i=1, \dots, k} \text{Min}\{w(x, y_i)\}_{y_i \neq x}.$$

Ces deux bornes se calculent facilement par tri des lignes et des colonnes de W . Elles permettent souvent d'affecter x à l'une ou l'autre classe si B_S ou B_R dépassent la borne supérieure B_p^k connue pour une bipartition avec k sélectionnés. Comme nous le verrons dans les simulations ci-dessous, c'est ce qui advient pour plus de la moitié des candidats. On peut donc procéder à une énumération sur un nombre restreint d'items parmi lesquels on ne cherche plus qu'un nombre petit de candidats pour compléter ceux qui sont sélectionnés d'office.

Pour une énumération efficace des parties de cardinal fixé d'un ensemble, Nijenhuis & Wilf (17) proposent un algorithme qui, pour passer d'une partie à la suivante, supprime et ajoute un seul élément. Ainsi, le poids de la nouvelle partie peut être calculé à partir du poids de l'ancienne en $O(n)$. Mais la complexité de cette énumération reste liée à $\binom{n}{k}$.

Pour le tournoi Jobim, s'il n'y a qu'un seul invité, x_5 est le seul qui reçoive un arc de poids minimum. Si l'on veut deux invités, x_1 et x_5 n'en reçoivent que deux, et c'est la seule paire qui réalise ce score. Mais s'il y a trois invités, les difficultés reprennent : Il y a deux triplets de score 3, $\{x_1, x_5, x_6\}$ et $\{x_1, x_5, x_7\}$. Et le second n'est même pas un début d'ordre médian !

4.3 Simulations again

Ici aussi, nous avons voulu (i) comparer les heuristiques, (ii) tester la difficulté du calcul et (iii) savoir si les bornes permettent d'affecter directement des candidats. Pour les mêmes tournois que précédemment, avec 31 juges et 20 candidats, nous cherchons à sélectionner 5 invités. Nous avons aussi testé une procédure par approbation *App* qui considère comme seuls nommés les 5 premiers de chaque ordre et qui retient comme sélectionnés les 5 items les plus souvent cités. Nous mesurons le poids moyen des cocircuits-retour obtenus par *App*, par l'heuristique gloutonne, par l'ordre de Borda, puis l'énumération fournit la valeur optimale. Nous indiquons de plus le nombre moyen de candidats dont l'affectation est fixée par les bornes (*NbFix*), le nombre moyen de solutions optimales et le maximum obtenu sur 100 tirages.

t	<i>App</i>	O_g	O_b	O_{opt}	<i>NbFix</i>	<i>NbSol</i>	<i>NbMax</i>
10	8.2	1.8	4.1	1.7	18	1.2	3
15	23.4	5.6	7.1	5.3	15	1.2	3
20	31.7	9.3	10.8	8.6	12	1.2	3
30	37.3	11.4	12.8	10.7	11	1.2	3

TABLE 11 : Valeurs moyennes des poids des cocircuits-retour, du nombre moyen d'item fixés et des nombres moyen et maximum de classes optimales.

Enfin, la procédure par approbation donne des résultats médiocres et l'heuristique gloutonne est proche de l'optimum et fait nettement mieux que la procédure de Bordat. On notera qu'à contrario des ordres, il y a peu de bipartitions optimales, le plus souvent une seule, même pour des classements aléatoires. De façon imprévue, l'énumération porte sur moins de la moitié des candidats, parfois aucun car ils sont tous affectés d'office.

4.4 Une étude de cas

Déterminer si les k items sélectionnés constituent une section commençante d'un ordre médian est une question délicate. D'abord parce qu'il peut y avoir beaucoup d'ordres médians qui n'ont pas nécessairement les mêmes k premiers éléments, et qu'il peut y avoir plusieurs classes d'items qui minimisent le poids des cocircuits-retour. Nous allons illustrer cette problématique à l'aide d'un exemple réel.

1	Toulouse	39	28	33	39	14	22	21	41	41
2	Grenoble	36	25	29	35	21	26	41	40	35
3	Montpellier	28	33	31	33	10	41	32	30	36
4	Marseille-Aix	37	29	37	37	17	27	4	37	37
5	Lyon	40	25	41	39	7	20	16	22	40
6	Bordeaux	38	38	41	30	12	16	8	39	32
7	Nantes	31	37	39	29	13	12	7	34	39
8	Rennes	26	32	37	31	18	36	9	26	34
9	Nice	35	20	33	40	4	28	6	37	33
10	Paris	41	12	40	41	1	13	1	27	38
11	Strasbourg	33	9	35	36	8	23	15	38	30
12	Nancy	25	11	26	25	26	32	37	22	27
13	Rouen	32	17	28	26	25	14	23	30	28
14	Lille	34	7	38	32	3	34	3	25	31
15	Clermont-Fd	16	32	30	17	32	39	17	13	24
16	Poitiers	8	40	19	28	40	31	24	33	11
17	Dijon	29	32	25	9	28	25	21	16	25
18	Caen	23	37	14	17	17	38	34	8	22
19	Angers	14	16	34	10	20	24	26	13	22
20	Orleans	17	20	11	14	39	1	26	9	29
21	Tours	21	34	22	5	28	6	27	2	26
22	St-Etienne	25	14	27	20	32	22	21	13	11
23	Besançon	15	35	13	9	39	29	11	17	15
24	Metz	22	14	12	22	37	30	29	13	8
25	Brest	13	25	21	14	33	20	31	3	13
26	Pau	2	39	6	7	35	10	37	25	16
27	LaRoche	19	41	23	17	3	40	10	4	12
28	Avignon	30	26	8	12	5	18	40	6	19
29	Nimes	10	3	17	5	37	38	14	33	14
30	Chambery	3	25	3	2	24	12	38	33	20
31	Toulon	27	4	7	11	15	34	2	22	23
32	Perpignan	18	25	16	21	30	10	13	16	9
33	Mulhouse	20	11	4	23	29	3	40	25	7
34	LeMans	12	16	24	34	35	8	6	14	2
35	Amiens	5	20	20	24	6	35	21	18	4
36	Limoges	9	8	15	19	41	16	33	19	1
37	Reims	6	6	9	27	23	17	22	30	5
37	LeHavre	11	28	10	19	10	4	30	8	6
39	Douai-Lens	7	1	1	3	22	2	37	1	18
40	Reunion	4	6	2	6	19	8	28	35	3
41	Valenciennes	1	2	5	1	12	5	12	6	18

TABLE 12 : Classement des 41 villes universitaires selon 9 critères pondérés.

Les étudiants d'un cours de recherche opérationnel ont décidé de classer les villes universitaires françaises. Il y en a 41. Le classement a été effectué selon 9 critères pondérés, à savoir la qualité de l'offre Culturelle (2), de l'Environnement (2), des Etudes (3) de son ouverture Internationale (3), des facilités de Logement (2), des offres de Sorties (2), de la pratique des Sports (2), de la qualité des Transports (2) et des offres d'Emploi (6). Les poids sont les valeurs indiquées entre parenthèses. Par la procédure de Borda, ils sont parvenus au classement de la Table 12, basé sur les variables de rang qui admettent de rares ex-aequo.

Les critères sont ordonnés suivant les préférences décroissantes ; Paris est la meilleure ville (41) pour la culture et l'international, mais la dernière (1) pour le logement et le sport. Ces variables de rang permettent de construire le tournoi pour lequel les préférences sont basées sur la somme de poids des critères. Globalement, Toulouse est avant Grenoble puisqu'elle est mieux classée pour 6 critères (de poids 18) alors que Grenoble n'est préférée que pour 3 critères de poids 6. L'arc (1,2) est donc de poids 12.

Pour ce tournoi, l'heuristique de décomposition en 5 classes a donné un ordre d'écart 69. Et l'énumération a donné 585 ordres optimaux d'écart 63. L'arborescence construite n'avait que 294 525 noeuds. Il est remarquable que tous ces ordres soient identiques sur les 14 premières villes à savoir (1, 5, 7, 10, 4, 3, 2, 8, 6, 9, 14, 11, 13, 12). On peut donc comparer aisément ce début d'ordre avec ce que l'on obtient en cherchant à sélectionner les k villes universitaires les plus attractives aux yeux des étudiants.

Pour $k = 1$ ou $k = 2$, ce sont d'abord Toulouse puis Toulouse et Lyon qui sont les seules sélectionnées, alors que Lyon n'arrive que 5-ème dans le classement étudiant. Pour $k = 3$, il y a 2 classes optimales, $\{1, 5, 4\}$ et $\{1, 5, 10\}$ et pour $k = 4$, c'est leur union $\{1, 5, 4, 10\}$. Assurément, Paris mérite mieux que son classement étudiant à la 10-ème place et Nantes, 3-ème dans tous les ordres médians, n'est toujours pas apparue, non plus que pour $k = 5$ qui sélectionne de façon unique $\{1, 5, 2, 4, 10\}$. Nantes apparaît pour $k = 6$ qui donne deux classes optimales $\{1, 5, 2, 4, 7, 10\}$ et $\{1, 5, 3, 4, 7, 10\}$. Ce n'est que pour $k = 9, \dots, 14$ que l'ordre optimal commun sur les k premiers constitue une classe optimale unique pour laquelle il n'y a aucun arc-retour.

Mais, pour des effectifs de 3 à 8 items, les classes optimales ne sont jamais des sections commençantes d'un ordre médian.

V CONCLUSIONS

Dans ce texte, nous avons essentiellement réalisé trois tâches nouvelles :

- calculé de meilleures bornes supérieures de l'écart d'un ordre total à un tournoi grâce à la décomposition du tournoi et à l'optimisation locale ;
- séparé les deux problèmes de classement (ordre total) et de sélection d'un nombre fixé de candidats ;
- proposé une méthode optimale de sélection dans le cas où n est petit, à l'aide de bornes efficaces.

Pour le colloque Jobim 2016, il n'y avait qu'un seul conférencier invité pour la session *System Biology*. Compte tenu de la convergence des procédures de décision, le choix est clair, le comité a invité x_5 . Comme il n'était pas disponible, c'est x_1 qui a fait l'exposé invité.

D'une manière générale, nous préconisons la démarche suivante⁴ :

- 5 — S'entendre sur les candidats, à l'issue d'un premier choix libre ou d'une discussion entre les membres du comité de programme, et sur le nombre d'invités ;
- Faire voter le comité qui exprime un ensemble d'ordres totaux, sur les candidats, si possible en évitant les ex-aequo ;
- Construire le tournoi pondéré par les préférences majoritaires ;
- 10 Pour le problème de construction d'un ordre unique ;
 - Enumérer toutes les bipartitions optimales ;
 - Choisir la plus centrale ;
- Pour le problème de sélection de $k > 3$ candidats ;
 - Calculer les bornes supérieure d'appartenance à l'une ou l'autre classe ;
 - 15 — Enumérer les bipartitions des candidats dont l'affectation n'est pas fixée.

La principale vertu de cette approche est qu'elle tient au mieux compte des classements initiaux, puisqu'elle contredit un nombre minimum de préférences individuelles et qu'on ne peut faire moins. Même si les algorithmes sont de complexité élevée ils s'appliquent à des données de cette taille, courantes dans bien des situations évoquées dans l'introduction. Pour la sélection,
20 si on ne veut qu'un seul ou deux ou trois sélectionnés les solutions optimales sont toujours calculables.

La procédure d'énumération ne demande pas de place mémoire et peut s'envisager pour des problèmes plus importants. Nous obtenons un groupe de 10 sélectionnés parmi 30 candidats, avec 100 votants assez divergents, puisque les ordres ont été générés avec 50 transpositions
25 aléatoires, en moins d'une minute de calcul. Il y a de quoi choisir les meilleurs crus de l'année, ou les "entrants" d'une section du CNRS, autres domaines d'application de cette méthode de sélection.

Références

- 30 [1] Barthélemy J.P., Monjardet B. (1981) The Median Procedure in Cluster Analysis and Social Choice Theory, *Mathematical Social Sciences* 1, 235-267.
- [2] Barthélemy J.P., Guénoche A., Hudry O. (1989). Median linear orders : Heuristics and Branch and Bound algorithm. *European Journal of Operational Research* 42, 555-579.
- [3] Bermond J.C. (1972) Ordre à distance minimale d'un tournoi et graphes partiels sans circuits maximaux, *Mathématiques et Sciences humaines* 37, 5-25.
- 35 [4] Billaut J.C., Bouyssou D., Vincke P. (2010) Should you believe in the Shanghai ranking ? An MCDM view. *Scientometrics* 84(1), 237-263, DOI : 10.1007/s11192-009-0115-x.
- [5] Borda J.C. (1784) Mémoire sur les élections au scrutin, in *Histoire de l'Académie Royale des Sciences pour 1781* 1784.
- [6] Caritat M.J.A.N., Marquis de Condorcet. (1785) *Essai sur l'application de l'analyse de la probabilité des*
40 *décisions rendues à la pluralité des voix*.
- [7] Charon I., Germa A., Hudry O. (1992) Utilisation des scores dans les méthodes exactes déterminant les ordres médians de tournois, *Mathématiques et Sciences humaines* 119, 53-74.

4. Les programmes sont disponibles sur simple demande à l'auteur.

- [8] Charon I., Guénoche A., Hudry O., Woïrgard F. (1996) A Bonsai Branch and Bound method applied to voting theory, *Proceedings of "Ordinal and Symbolic Data Analysis"*, OSDA'95, E. Diday et al. (Eds.), Springer Verlag, 309-318.
- [9] Charon I., Hudry O., Woïrgard F. (1996) Ordres médians et ordres de Slater des tournois, *Mathématiques et Sciences Humaines* 133, 23-56, 1996.
- [10] Charon I., Hudry O. (2001) The noising methods : a generalization of some metaheuristics, *European Journal of Operational Research* 135(1), 86-101.
- [11] Guénoche A. (1977) Un algorithme pour pallier l'effet Condorcet, *R.A.I.R.O. Recherche Opérationnelle* 11(1), 77-83.
- [12] Hansen P., Mladenović N. (2001) Variable neighborhood search : Principles and Applications, *European Journal of Operational Research* 130(3), 449-467.
- [13] Hudry O. (1989) *Recherche d'ordres médians : complexité, algorithmique et problèmes combinatoires*, Thèse de l'ENST, Paris.
- [14] Hudry O. (2012) On the computation of median linear orders, of median complete preorders and of median weak orders, *Mathematical Social Sciences* 64, 2-10.
- [15] Kemeny J.G. (1959) Mathematics without numbers, *Daedalus* 88, 577-591.
- [16] Monjardet B. (1973) Tournoi et ordre médian pour une opinion, *Mathématiques et Sciences humaines* 43, 55-73.
- [17] Nijenhuis A., Wilf H.S. (1975) *Combinatorial algorithms*, Academic Press.
- [18] Slater P. (1961) Inconsistencies in a schedule of paired comparisons, *Biometrika* 48, 303-312.
- [19] Remage R., Thomson W.A. (1966) Maximum likelihood paired comparison rankings, *biometrika*, 53, 143-149.
- [20] Smith A.F.M., Payne C.D. (1974) An algorithm for determining Slater's i and all nearest adjoining orders, *British Journal of Mathematical and Statistical Psychology* 27, 49-52.
- [21] Woïrgard F. (1997) *Recherche et dénombrement des ordres médians des tournois*, Thèse de l'ENST.