UMONS
Université de Mons

Faculté des Sciences

Aide multicritère à la décision

Résumé

Résumé réalisé par: Anthony Rouneau
Section: 2ème Bloc Master en Sciences Informatiques
Images: Proviennent du cours de M. Pirlot.

Faculté des Sciences • Université de Mons • Place du Parc 20 • B-7000 Mons

Contents

1 Introduction

1.1 Types de problèmes

- Make or buy – Décider si on achète à un tiers ou si on produit nous même. Si on produit ça coute plus cher mais ça nécessite l'infrastructure et les machines.
- Management – Définir un ordre de priorité dans les projets internes.
- Choix d'équipement – Retourne un objet.
- Choix de candidats – Retourne un rangement parmi les candidats.
- Classement de réponse à un appel d'offre – Retourne un tri en paquet.
- Élections par votes – Les critères sont définis par les votants.
- Trouver localisation optimale – Problème d'optimisation.
- Gestion de portefeuille – Objectif : optimisation globale.

1.2 Problème de décision

Un problème de décision doit rester subjectif. En effet, on doit respecter les préférences des parties prenantes. Voici quelques mots de vocabulaire :

- Décideur(s) – Responsable(s) de la décision.
- Parties prenantes – personnes ayant un intérêt dans la décision.
  - Peuvent ajouter des contraintes.
  - Peuvent montrer de la résistance dans la prise de décision.
  - Peuvent ajouter des informations utiles pour la prise de décision.
- Problématique – Choisir, ranger, trier, ...
- Les actions ou alternatives – Les choix disponibles.
- Les critères – Les qualités qui vont affecter le choix. Si les critères sont bien fixés, le choix guidé par l'aide à la décision doit correspondre avec un choix idéal pour le décideur.

Il y a deux phases principales dans la résolution d'un tel problème :

- Structuration – Définition des étapes de résolution.
- Agrégation – Établissement d'un modèle de préférence, validation des paramètres, ...

Deux méthodes principales peuvent être utilisée pour établir un classement dans les alternatives :

- Synthétique – Classement général, établi en fonction des critères.
- Comparaison par pair – On compare les alternatives deux à deux selon les critères.

Dans le cas de critères discrets, des problèmes mathématiques peuvent survenir. En effet si on classe des critères avec un entier entre 1 et 5, 2 ne vaut pas toujours le double de 1 (Comme si on notait A, B, C, D, ...). Malgré tout, on va considérer que 2 est le double de 1 dans les calculs. On peut se retrouver alors avec des échelles et des unités tout à fait hétérogènes.

1.3 Dominance

On dit que A domine B si l'objet A obtient de meilleurs résultats sur tous les critères par rapport à l'objet B. Si le modèle est bon, on ne devrait alors jamais sélectionner B dans la procédure de décision.

2 Critique de la somme pondérée

2.1 Super-critère par somme pondérée

On parle de super-critère par somme pondérée pour un re-codage d'un critère existant en prenant en compte le poids accordé à ce critère. Ce re-codage u vaut ceci :

$$u(a) = \sum_{i=1}^{n} k_i g_i(a), \text{ avec } a \text{ une des alternatives}$$

avec n le nombre de critères

avec g_i l'évaluation du critère i

avec k_i le poids accordé au critère i, négatif si le critère est à minimiser

Le principe est donc de re-coder de cette manière toutes les alternatives afin de donner un "score" pondéré à chacune d'elle. Ensuite, on peut comparer les alternatives entre elles selon leur super-critère :

$$a \succsim b \Leftrightarrow u(a) \geqslant u(b) \Leftrightarrow \sum_{i=1}^{n} k_i(g_i(a) - g_i(b)) \geqslant 0$$

2.2 Normalisation

Il existe plusieurs manières de normaliser. Selon la méthode utilisée et selon les alternatives prises dans la normalisation, le résultat de la prise de décision peut être différente.

Normaliser en fonction de la meilleures alternative

$$g'_i(a) = \frac{g_i(a)}{\max g_i}$$

Normaliser pour que le min vaille 0 et le max vaille 1

$$g'_i(a) = \frac{g_i(a) - \min g_i}{\max g_i - \min g_i}$$

Normaliser au sens statistique

$$g'_i(a) = \frac{g_i(a) - \mu_{g_i}}{\sigma_{g_i}}$$

2.3 Poids vs. taux de change

Le poids des critères ne représentent pas leur importance. Ce sont des taux de substitution. Un désavantage de k_i unités sur le critère j peut être compensé par un avantage de k_j unités sur le critère i. En effet, c'est pour garder l'importance d'un critère que l'on va adapter le poids. Par exemple, si on parlait de critères temporels et que les secondes avec un poids k_s, on accorderait un poids valant $k_s \cdot 60$ aux minutes afin que les minutes gardent leur importance, et $k_s \cdot 3600$ aux heures.

2.4 Préférence non-linéaire

En pratique, il arrive souvent que les préférences ne soient pas linéaires. Exemple : on aimerait bien plus trouver une voiture entre 15000 et 20000 euros qu'entre 20000 et 30000. On va donc accorder plus d'importance aux voitures se trouvant dans le premier intervalle. La somme pondérée ne permet pas d'exprimer des préférences non-linéaires. En pratique, il faudra re-coder le super-critère (pas celui de la somme pondérée) pour qu'il soit non-linéaire.

2.5 Indépendance préférentielle

La somme pondérée raisonne selon "ceteris paribus", qui veut dire "toutes choses restant égales par ailleurs". Ça veut dire qu'en cas d'égalité entre plusieurs critères, ce sont les critères inégaux qui influenceront le choix. Ça veut dire également que les préférences doivent être indépendantes afin d'être utilisées dans une somme pondérées. Ce raisonnement n'est pas toujours pertinent. Prenons comme exemple un problème dans lequel on doit obtenir un ordre de préférence entre frite/pâtes (premier critère) et mayo/bolo (deuxième critère). soit $g_1(\text{frites}) = 0.4$, $g_1(\text{pates}) = 0.7$, $g_2(\text{mayo}) = 0.6$ et $g_2(\text{bolo}) = 0.5$. On se retrouve alors avec frites mayo $>$ frites bolo, ce qui semble correct, mais par contre, on a que pates mayo $>$ pates bolo, ce qui ne fera pas plaisir aux parties prenantes. Une solution serait de regrouper les deux critères en un seul, car ils sont dépendants.

2.6 Écarts minimes

Même si il y a seulement une différence d'un centième entre deux alternatives, la somme pondérée va trancher. il faudrait donc faire une analyse de sensibilité, c'est-à-dire faire varier les valeurs des poids et des évaluations pour voir si le même résultat en ressort.

2.7 Résumé

Conditions de validité d'une somme pondérée	
Normalisation	une éventuelle normalisation des évaluations doit être choisie conformément au sens du problème
Poids	les poids sont des taux de substitution et doivent être déterminés en se référant aux évaluations
Linéarité	les échelles sont supposées linéaires (en termes de préférence) ou doivent être transformées
Indépendance	les préférences doivent être indépendantes
Précision	les écarts, aussi minimes soient-ils, peuvent avoir un impact

3 Fonctions de valeur additive

Le principe ici est de calculer le super-critère des alternatives selon une somme de fonctions de re-codage. On va supposer que chaque critère a sa propre fonction de re-codage.

$$u(a) = \sum_{i=1}^{n} u_i(g_i(a))$$

ou encore, si on accorde des poids aux critères :

$$u'(a) = \sum_{i=1}^{n} k_i \cdot u'_i(g_i(a))$$

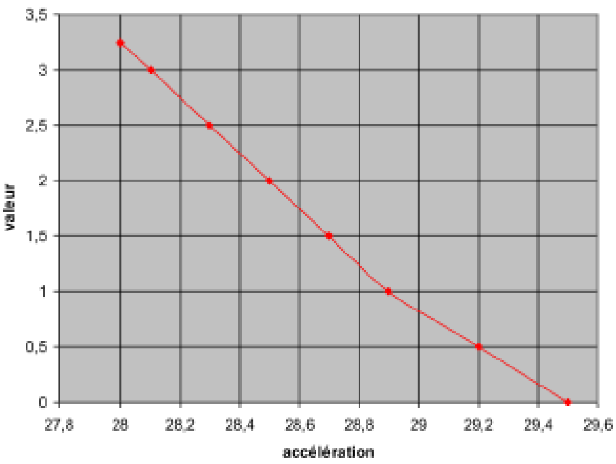
$$\text{avec } u'_i(g_i, \min) = 0,\ u'_i(g_i, \max) \text{ et } \sum_{i=1}^{n} k_i = 1$$

3.1 Méthodes directes

3.1.1 Jugements d'indifférence

Le principe ici est de demander aux parties prenantes d'estimer des valeurs pour lesquelles deux alternatives seraient indifférentes (équivalentes). Pour ce faire, on va avoir besoin d'un critère de base, pour lequel on prendra un intervalle de valeur. On part de la meilleure (resp. la moins bonne) valeur de cet intervalle et d'une valeur marginale pour le critère à évaluer. On va ensuite à chercher une valeur pour le critère à évaluer tel que l'alternative ayant cette valeur et la moins bonne (resp. la meilleure) valeur pour le critère de base soit indifférente à la première. Soit deux critères à minimiser : le prix et le temps d'accélération, voici un exemple de jugement d'indifférence :

$$
\begin{array}{ll}
(16500, 29.5) \sim (17500, x_1) & x_1 = 29.2 \\
(16500, 29.2) \sim (17500, x_2) & x_2 = 28.9 \\
(16500, 28.9) \sim (17500, x_3) & x_3 = 28.7 \\
(16500, 28.7) \sim (17500, x_4) & x_4 = 28.5 \\
(16500, 28.5) \sim (17500, x_5) & x_5 = 28.3 \\
(16500, 28.3) \sim (17500, x_6) & x_6 = 28.1
\end{array}
$$



Il ne manque plus qu'à normaliser la courbe pour que la valeur soit comprise entre 0 et 1 (ou 0 et 100). Il est alors simple de calculer les taux de substitution (poids) :

$$\frac{k2}{k1} = \frac{u_1'(16500) - u_1'(17500)}{u_2'(29.2) - u_2'(29.5)}$$

On fait pareil avec tous les critères (toujours en fonction du critère 1), et on finit par choisir k_1 pour que l'équation $\sum_{i=1}^{n} k_i = 1$ soit respectée.

En résumé, la technique est bonne mais présente quelques inconvénients :

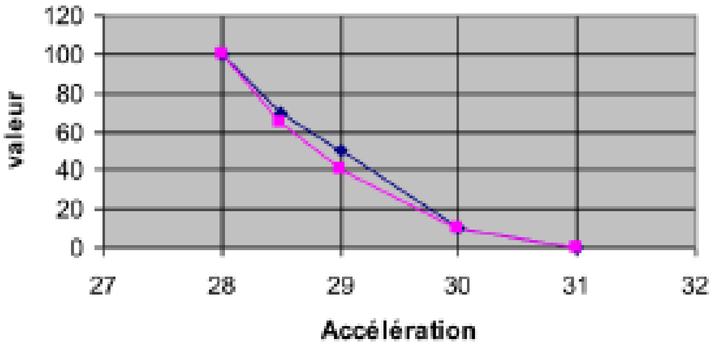
Et pourtant, pas si évidente

- Certaines questions peuvent être difficiles à répondre.
  En particulier pour des échelles "abstraites", comme le Freinage et la Tenue de route.
- Des biais sont possibles, par exemple dans le choix de l'intervalle initial (au milieu, en haut ou en bas des échelles de valeur)
- Toutefois, il est possible de vérifier la cohérence,
  par des questions redondantes
- Cette méthode ne peut pas s'utiliser avec des échelles discrètes
  il faut au moins une échelle continue ou très détaillée

3.1.2 Direct rating

SMART Cette technique est très simple : on prend deux valeurs extrêmes, et on leur attribue une valeur d'utilité (pour u) : 0 pour la moins bonne, et 100 pour la meilleure. On va ensuite interroger le décideur pour trouver les évaluations intermédiaires.

$$\begin{array}{lllll}
28 & \rightarrow & 100 & & \\
31 & \rightarrow & 0 & & \\
29 & \rightarrow & ??? & \rightarrow & 50 \\
28.5 & \rightarrow & ??? & \rightarrow & 70 \\
30 & \rightarrow & ??? & \rightarrow & 10
\end{array}$$



Une fois qu'on a effectué cette partie pour tous les critères, on donne 10 points (poids) au critère le moins important, et on se base dessus pour assigner des poids aux autres critères. Il reste ensuite à normaliser les poids pour qu'ils somment à 1.

Swing Weighting Le principe ici est de faire une évaluations des alternatives. On part de la pire alternative possible, ce sera la référence. Ensuite, on demande au décideur quel premier critère permettrait d'améliorer cette alternative, puis quel second critère améliorerait l'alternative, etc... On obtiendra donc un ordre d'importance parmi les critères. Il suffit ensuite de pondérer ces critères selon l'ordre choisi, tout en respectant les contraintes précédentes (poids somment à 1, ...).

3.2 Fonction de valeur additive (UTA)

Le but ici est de fournir une méthode automatisée pour trouver la fonction de re-codage ainsi que le poids de chaque critère. Pour trouver des fonctions de re-codage non-linéaires, on va donner une fonction linéaire par morceaux. Ces morceaux étant séparés par des points de cassure.

- Le décideur fournit un sous-ensemble d'alternatives qu'il est capable de ranger directement (ensemble d'apprentissage ou *learning set*):

$$a_1 \succ a_2 \succ \ldots \succ a_p$$

- L'analyste décide du nombre de points de cassure $n_i - 1$ de chaque fonction de valeur marginale u_i. Les points de cassure sont notés

$$g_{ij}, \quad \text{pour } j = 0, \ldots, n_i$$

  Par défaut, ils sont également espacés dans le domaine de variation $[g_{i0}, g_{in_i}]$ de chaque critère.

L'apprentissage se fait ensuite selon un programme linéaire qui va chercher à maximiser l'écart entre les alternatives sélectionnées. Pour ce faire, l'apprentissage va chercher la position de ces points de cassure.

Contraintes:

$$\begin{cases} u_i(g_{ij}) - u_i(g_{i(j-1)}) & \geq & 0 & i = 1, \ldots n; \quad j = 1, \ldots, n_i \\ \sum_i u_i(g_{in_i}) & = & 1 & \\ u(a_k) - u(a_{k+1}) & \geq & s & \text{pour } k = 1, \ldots, p-1 \end{cases}$$

Objectif: $\max s$

Cependant, avec cette formulation, il est possible qu'on ne trouve pas de solution avec le nombre de points de cassure choisis et/ou les alternatives choisies. Pour résoudre ce problème, on va ajouter des pénalités qui feront en sorte qu'il existe toujours une solution. Ces pénalités sont à minimiser, et dans l'idéal restent à 0 (ce qui veut dire qu'il n'y a pas besoin de pénalités pour que ça fonctionne).

Formulation "sans échec"

Contraintes:

$$\begin{cases} u_i(g_{ij}) - u_i(g_{i(j-1)}) & \geq & 0 & i = 1, \ldots n; \quad j = 1, \ldots, n_i \\ \sum_i u_i(g_{in_i}) & = & 1 & \\ u(a_k) - u(a_{k+1}) + z_k & \geq & s & \text{pour } k = 1, \ldots, p-1 \end{cases}$$

Objectif: $\max s - M \sum_k z_k$
avec $M \gg 0$

3.2.1 Théorème de représentation

Les préférences du décideur doivent respecter une liste de contraintes afin de pouvoir être représentée par une fonction de valeur additive. Si elle ne correspondent pas aux conditions suivantes, il faut choisir un autre modèle de représentation.

Théorème

- Soit $\succsim$ une relation binaire sur $X = \prod_{i=1}^{n} X_i$ avec $n \geq 3$
- Supposons que $\succsim$ vérifie la Solvabilité Restreinte et qu'au moins 3 critères soient Essentiels
- La préférence $\succsim$ peut être représentée par une FVA si et seulement si
- la préférence est un préordre complet qui satisfait la propriété d'Indépendance Préférentielle et la propriété Archimédienne
- De plus, la représentation est unique à une transformation affine positive près des fonctions de valeur marginales:

$$x \succsim y \quad \Leftrightarrow \quad \sum_{i=1}^{n} u_i(x_i) \geq \sum_{i=1}^{n} u_i(y_i) \quad \Leftrightarrow \quad \sum_{i=1}^{n} u_i'(x_i) \geq \sum_{i=1}^{n} u_i'(y_i)$$

implique $u_i' = \alpha u_i + \beta_i$, pour un certain nombre $\alpha > 0$ et certains nombres β_i

$\succsim$ est un préordre complet
i.e. un rangement complet avec possibilité d'ex aequo
i.e. une relation transitive et complète

Solvabilité restreinte

- "Il y a toujours assez de valeurs pour équilibrer les jugements d'indifférence "
- plus précisément: pour tout i, pour tous $a_i, b_i \in X_i$, pour tous $x, z \in X$,

$$(a_i, z_{-i}) \succsim x \succsim (b_i, z_{-i}) \quad \Rightarrow \quad \exists c_i \in X_i \text{ t. q. } x \sim (c_i, z_{-i})$$

Essentialité

- "Le critère i est essentiel s'il a un certain impact"
- plus précisément: le critère i est essentiel si il existe $x_i, y_i \in X_i$ et $a \in X$ tels que

$$(x_i, a_{-i}) \succ (y_i, a_{-i})$$

Hypothèse 2: Indépendance préférentielle

- pour tout i, pour tous $v_i, w_i \in X_i$, pour tous $x, y \in X$,

$$(x_1, \ldots, x_{i-1}, v_i, x_{i+1}, \ldots, x_n) \succsim (y_1, \ldots, y_{i-1}, v_i, y_{i+1}, \ldots, y_n)$$

iff

$$(x_1, \ldots, x_{i-1}, w_i, x_{i+1}, \ldots, x_n) \succsim (y_1, \ldots, y_{i-1}, w_i, y_{i+1}, \ldots, y_n)$$

- En bref, avec des notations évidentes:

$$(v_i, x_{-i}) \succsim (v_i, y_{-i}) \quad \text{iff} \quad (w_i, x_{-i}) \succsim (w_i, y_{-i})$$

Hypothèse 3: propriété archimédienne

- Il n'y a pas d'alternatives infiniment désirables ni infiniment indésirables
- On peut formuler précisément cette propriété à l'aide de séquences standard

3.2.2 Méthode des échanges égaux

Le but de cette méthode est d'éliminer les alternatives dominées et de supprimer des critères en compensant par d'autres critères. Prenons par exemple les alternatives suivantes (2 critères sont à minimiser, les autres à maximiser), on peut retirer *e* car est dominée par *b* et *a* est presque dominée par *d*, on peut donc retirer *a* et *e*. Ensuite, on va chercher à supprimer un critère, et celui qui a le plus de valeur en commun est *Accès*. On va donc chercher à compenser le gain en *Accès* par un gain en *Clients*. Pour trouver δ, on demande au décideur pour quel gain en *Clients*, *c* et *c′* seront équivalent à ses yeux. Une fois qu'on a déterminé cette valeur, on remplace *c* par *c′* et on supprime le critère *Accès* dont toutes les valeurs sont égales.

	a	*b*	*c*	*d*	*e*
(min) *Accès*	45	25	20	25	30
(max) *Clients*	50	80	70	85	75
(max) *Services*	*A*	*B*	*C*	*A*	*C*
(max) *Surface*	80	70	50	95	70
(min) *Coût*	1850	1700	1500	1900	1750

	c	*c′*
Accès	20	**25**
Clients	70	$\mathbf{70} + \delta$
Services	*C*	*C*
Surface	50	50
Coût	1500	1500

trouver δ tel que $c' \sim c$

	b	*c′*	*d*
Clients	80	78	85
Services	*B*	*C*	*A*
Surface	70	50	95
Coût	1700	1500	1900

On recommence le procédé ((1) chercher les dominances et retirer les alternatives dominées et (2) supprimer des critères en compensant par d'autres critères) jusqu'à n'avoir qu'une seule possibilité restante. Cependant :

Il y a aussi quelques difficultés

- Ce processus convient quand les alternatives sont peu nombreuses
  - autrement, il y a beaucoup de questions à poser
- Le résultat n'est pas un modèle de préférence
  - si de nouvelles alternatives apparaissent,
il faut relancer le processus de questionnement
- Sous quelles hypothèses, la méthode est-elle valide?
  - il faut que la préférence du décideur soit représentable par une fonction de valeur additive
  - en particulier, il faut que le raisonnement "toutes choses égales par ailleurs" soit permis

4 Méthode de Saaty (AHP)

Le principe de cette méthode est d'effectuer des comparaisons par paires. Soit par paires d'alternatives, auquel cas on obtient un classement au bout de la méthode, soit par paires de critère, ce qui permet d'évaluer le poids de chaque critère. La comparaison par paire (pour un critère i dans le cas d'une alternative) va donner une matrice α_i. Une case $\alpha_i(a, b)$ de cette matrice veut dire que l'alternative (ou le critère) a est $\alpha_i(a, b)$ plus important que b dans la décision finale. Dans le cas d'une comparaison entre alternatives, on a alors que

$$\alpha_i(a, b) \approx \frac{u_i(a)}{u_i(b)}$$

et dans le cas d'une comparaison de critères, on a que

$$\alpha(\text{crit}_1, \text{crit}_2) \approx \frac{k1}{k2}$$

Une telle matrice est réciproque, c'est-à-dire que $\alpha_i(a,b) = \frac{1}{\alpha_i(b,a)}$. On dit qu'elle est parfaitement cohérente si $\forall a,b,c : \alpha_i(a,b) = \alpha_i(a,c) \cdot \alpha_i(c,b)$. A partir de cette matrice, on peut estimer $u_i(a), \forall a$. Pour ce faire, on peut utiliser la méthode des moindres carrés, ou alors on peut utiliser la méthode des vecteurs propres expliquée ci-dessous.

4.1 Vecteurs propres

Cette méthode des vecteurs propres approxime u_i par le vecteur propre normalisé w_i. La composante (de ligne) $w_i(a)$ de w_i vaut :

$$w_i(a) = \lim_{k \to \infty} \frac{\alpha_i^k(a,b)}{\sum_c \alpha_i^k(c,b)}$$

On a donc que w_i est approximé par une colonne de la k$^{\text{ème}}$ puissance de la matrice α_i, normalisé par la somme des éléments de cette colonne. C'est aussi vrai pour la somme des colonnes, normalisée, de la puissance de la matrice

Avec $k = 2$, on a déjà une bonne approximation ! Voici les étapes pour calculer l'approximation de u_i en cas de comparaison d'alternatives, ou des poids k en cas de comparaison de critères :

1. Prendre le carré de la matrice de comparaison par paire.
2. Sommer les lignes de cette nouvelles matrice, et normaliser ces sommes afin qu'elles somment à 1.
3. Recommencer à l'étape 1 en utilisant la matrice carrée comme entrée dans le cas ou la différence entre le vecteur calculé à cette étape et le vecteur précédente est insignifiante.

Voici une illustration de l'étape 2 :

								Normalisation
3.0000	+	1.7500	+	8.0000	=	12.7500		0.3194
5.3332	+	3.0000	+	14.0000	=	22.3332		0.5595
1.1666	+	0.6667	+	3.0000	=	4.8333		0.1211
						39.9165		1.0000

On peut donner une interprétation à l'élément dans la ligne de a et la colonne de b dans α_i^2:

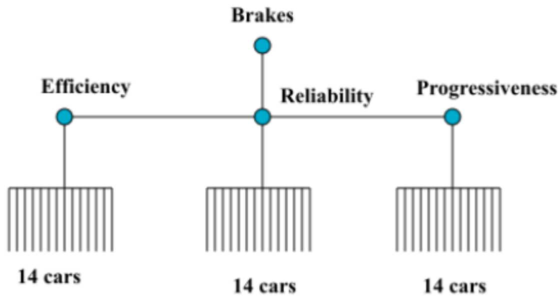
$$\alpha_i^2(a,b) = \sum_c \alpha_i(a,c) \times \alpha_i(c,b)$$

C'est donc la somme des estimations du rapport $u_i(a)/u_i(b)$ obtenues en passant par tous les chemins de longueur 2 joignant a et b

En normalisant, on obtient une sorte de moyenne de ces rapports, ce qui corrige les imprécisions.

4.2 Profondeur d'arbre

Tout ceci est valable pour un arbre à trois niveaux : But, Critères, Alternatives. Cependant, il pourrait y avoir plus de niveaux (sous-critères).



S'il y a plus de 3 niveaux, en chaque nœud on compare par paires les enfants, on en tire un vecteur de priorités et on l'utilise pour agréger par la formule de la somme pondérée les priorités calculées au niveau des enfants. La procédure doit donc être appliquée en partant du bas de la hiérarchie.
Elle finit par l'évaluation globale, celle du "Goal".

4.3 Évaluations verbales

Pour établir les comparaisons par paires, on pourrait faire appel à une évaluation verbale : Bon, moins bon, etc... Cependant, on doit ensuite procéder à un re-codage de ces termes en nombres. Et selon l'échelle choisie, les résultats seront très différents. Il ne faut pas utiliser les comparaisons verbales en pratique. Mieux vaut utiliser une échelle fixée.

4.4 Critique de l'AHP

1. La méthode des comparaisons par paires fonctionne bien pour évaluer des distances, par exemple (voir séance d'exercices). Pas sûr qu'on puisse en tirer argument pour justifier l'évaluation de préférences ou de l'importance
2. La façon d'agréger les priorités par une somme pondérée est peu défendable. En particulier, l'importance des critères est estimée sans faire référence aux échelles de ces critères. C'est une des critiques que nous avions faites à la somme pondérée et AHP tombe sous le coup de cette critique.
3. En particulier, si on enlève ou qu'on ajoute une alternative, sans changer les comparaisons par paires des autres alternatives, il arrive que le rangement final des alternatives de départ soit modifié. Ce phénomène, connu sous le nom de *rank reversal* est illustré dans les exercices.

5 Surclassement

On dit que l'alternative a surclasse l'alternative b

- Si l'ensemble des critères sur lesquels a est meilleure que b est suffisamment important (concordance). (Possibilité de faire des coalitions de critères et de poser des conditions sur les critères à prendre). En pratique on accorde des poids (qui somment à 1) à chaque critère, et il faut que la somme des poids des critères où a est meilleur que b dépasse le seuil λ. Ces poids ne sont pas des taux de substitution.
- Sans que a soit inacceptablement moins bonne que b sur un critère (veto).

Une relation de surclassement

- n'est pas toujours complète: il se peut que

$$a \not\succ b \quad \text{et} \quad b \not\succ a;$$

  on dit que a et b sont *incomparables*
- n'est pas toujours transitive : il se peut que

$$a \succ b \quad \text{et} \quad b \succ c \quad \text{et} \quad a \not\succ c$$

  Dans ce cas, il se peut que $c \succ a$ (cycle) ou que $c \not\succ a$ (incomparabilité).

5.1 Procédures de vote

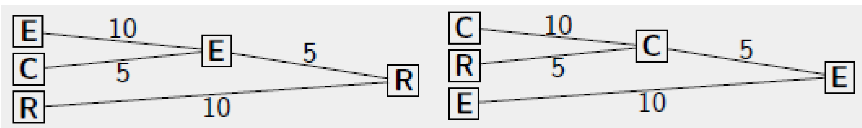
En utilisant le vote, on va montrer qu'il n'existe pas de procédure idéale pour la comparaison par paire.

5.1.1 Vote par majorité simple

C'est le vote le plus simple : on demande aux gens de voter pour leur favoris, celui qui obtient la majorité gagne. Cependant, le résultat peut ne pas contenter la majorité des personnes (dû à leur préférences secondaires).

5.1.2 Vote par agenda

Les votes par agenda sont très sensibles à leur "agenda". Un agenda est un ordre dans lequel on effectue un vote. Les exemples suivant ont été construit avec les même votant (et donc les même préférences). Selon l'ordre dans lequel on effectue le vote (d'abord E-C puis E-R, ou C-R puis C-E), le résultat diffère.



5.1.3 Vote en deux tours

Procédure où le premier tour sert à se concentrer sur deux candidats uniquement. Les votes qui allaient aux candidats éliminés reviennent à la deuxième préférence de ces votants (selon les candidats restants). Le problème est que cette procédure de vote est non-séparable. On ne peut pas effectuer ça par région, car le résultat du premier tour pourrait différer selon les zones.

5.1.4 Vote par score (Borda)

On donne un score entre 1 et X pour ses X premières préférences. Ensuite, le total des scores obtenu pour chaque candidat détermine le vainqueur. Le problème d'une telle procédure est que les partisans des candidats pourraient tenter de pénaliser un concurrent direct en le classant dernier, et les résultats sont au final biaisés de cette manière.

5.1.5 Vote par circonscription

S'il faut obtenir une majorité de voix dans la majorité des circonscriptions, on peut gagner l'élection avec 26% des voix globales car la taille des circonscriptions diffère.

5.1.6 Représentation proportionnelle

Si on veut proposer un certain nombre de places pour un certain nombre d'élu, une approche proportionnelle par la règle du plus grand reste donne des résultats peu logiques...

Composition d'un parlement à la proportionnelle

2 100 votants, 3 partis et 20 sièges

Votes	Sièges: règle du plus grand reste
A: 928	A: 8,84 $\Rightarrow$ 8 sièges + **1** = 9
B: 635	B: 6,05 $\Rightarrow$ 6 sièges + **0** = 6
C: 537	C: 5,11 $\Rightarrow$ 5 sièges + **0** = 5

où $8,84 = \frac{928}{2100} \times 20$

Si l'on augmente le nombre de sièges

Pour 21 sièges	Pour 22 sièges
A: 9,28 $\Rightarrow$ 9 sièges	A: 9,72 $\Rightarrow$ 9 sièges + **1**
B: 6,35 $\Rightarrow$ 6 sièges	B: 6,65 $\Rightarrow$ 6 sièges + **1**
C: 5,37 $\Rightarrow$ 5 sièges **+1**	C: 5,63 $\Rightarrow$ 5 sièges **!!!**

5.1.7 Coalitions

Le poids d'un parti n'est pas égal à son nombre de siège ou d'actions.

Si les décisions sont prises à la majorité absolue

Soit une assemblée de 100 membres, composée de 3 partis

- A : 45 sièges
- B : 15 sièges
- C : 40 sièges

- un parti seul ne peut rien
- toute coalition de 2 partis gagne

Tous les partis ont donc le même pouvoir !

Pour les décisions prises à la majorité qualifiée des 2/3

- un parti seul ne peut rien
- la seule coalition gagnante est **A** et **C**

B n'a aucun pouvoir, et les autres sont obligés de s'entendre

5.1.8 Théorème d'Arrow

Considérons 5 propriétés "naturelles"

en supposant les votes les plus riches: des classements

- Universalité: *accepter toutes les configurations de votes*
- Transitivité et complétude: *fournir une liste ordonnée des candidats*
- Unanimité: *respecter des préférences unanimes*
- Indépendance des alternatives non concernées: *ne regarder que si a est préféré à b ou pas*
- Absence d'un dictateur: *ne pas accepter de dictateur*

Théorème d'Arrow (1951)

Pour plus de 2 candidats et plusieurs votants,
il n'existe **aucune procédure satisfaisant ces 5 principes**.

Par exemple, le problème de la technique de Borda est qu'elle ne respecte pas l'indépendance des alternatives non-concernées. Si un candidat mal placé se retire, il se peut que l'ordre des candidats restant change.

5.1.9 Théorème de Gibbard et Satterthwaite

Théorème de Gibbard et Satterthwaite (1973)

Quand il y a au moins 3 candidats,
toute procédure de choix universelle et sans dictateur est manipulable

- procédure de choix :
  fournit le nom d'un candidat sur base des rangements de tous les candidats par les votants
- manipulable:
  dans certains cas, un votant a intérêt à ne pas être sincère et donc, à ne pas voter pour son candidat favori

5.2 Ranger selon des comparaisons par paires

Par défaut, une relation de surclassement vérifie toutes les propriétés du théorème d'Arrow, sauf la transitivité et complétude. Il faut donc retravailler le surclassement afin de trouver un rangement à partir de celle-ci. Ça s'appelle la phase d'exploitation. Voici un exemple de matrice de surclassement. Une case (i, j) de la matrice valant 1 signifie que l'alternative i surclasse l'alternative j. On peut voir qu'il existe un cycle de surclassement, et que le tableau ne respecte donc pas toutes les propriétés d'Arrow.

Cars	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1	1	0	0	0	1	1	0	1	0	0	0	0	1
2	0	1	0	1	0	1	0	1	1	0	0	0	1	1
3	1	1	1	1	1	1	1	1	1	1	0	1	1	1
4	1	0	0	1	0	1	0	0	0	0	0	0	1	0
5	1	1	0	1	1	1	0	0	0	1	0	0	1	0
6	0	0	0	0	0	1	0	0	0	0	0	0	1	0
7	1	1	0	1	1	1	1	1	1	1	0	0	1	1
8	1	0	0	1	1	1	1	1	1	0	0	0	1	1
9	0	1	0	1	1	1	0	0	1	0	0	0	1	1
10	1	1	0	1	0	1	1	1	1	1	1	0	1	1
11	1	1	1	1	1	1	1	1	1	1	1	1	1	1
12	1	1	0	1	1	1	1	1	1	1	0	1	1	1
13	1	0	0	0	0	0	0	0	0	0	0	0	1	0
14	0	1	0	1	1	1	0	1	1	0	0	0	1	1

Plusieurs méthodes existent pour la phase d'exploitation.

5.2.1 Sommes simples

On peut soit sommer les lignes, auquel cas le meilleur est le maximum ou sommer les colonnes, auquel cas le meilleur est le minimum.

5.2.2 ELECTRE

ELECTRE I

a surclasse b

s'il y a concordance et non discordance:

Concordance

$$c(a,b) = \sum_{j : g_j(a) \geq g_j(b)} p_j \quad \geq \lambda$$

où λ est le seuil de concordance (≥ 0.5 si les poids sont normalisés)

Non-discordance il n'y a de veto sur aucun critère, donc:

$$\forall j, \quad g_j(a) \geq g_j(b) - v_j$$

ELECTRE II

Surclassement fort

a surclasse fortement b ($aS^F b$) si

- $c(a,b) = \sum_{j:g_j(a) \geq g_j(b)} p_j \quad \geq \lambda^F$
- $P^+(a,b) = \sum_{j:g_j(a) > g_j(b)} p_j \quad \geq P^+(b,a) = \sum_{j:g_j(b) > g_j(a)} p_j$
- $\forall j, \quad g_j(a) \geq g_j(b) - v_j^F$

Surclassement faible

a surclasse faiblement b ($aS^f b$) si a ne surclasse pas fortement b et

- $c(a,b) = \sum_{j:g_j(a) \geq g_j(b)} p_j \quad \geq \lambda^f$
- $P^+(a,b) = \sum_{j:g_j(a) > g_j(b)} p_j \quad \geq P^+(b,a) = \sum_{j:g_j(b) > g_j(a)} p_j$
- $\forall j, \quad g_j(a) \geq g_j(b) - v_j^f$

avec $\lambda^F > \lambda^f$ et $v_j^F < v_j^f$ pour tout j

Distillation Processus itératif qui vise à retirer les circuits. A chaque itération, on retire de la matrice (et on attribue une place) l'alternative (ou les alternatives en cas d'égalité) qui est surclassée par le moins d'autres alternatives (qui est battue par le moins de monde). On retire donc du meilleur au moins bon.

Décantation Processus itératif qui vise à retirer les circuits. A chaque itération, on retire de la matrice (et on attribue une place) l'alternative (ou les alternatives en cas d'égalité) qui surclasse le moins d'autres alternatives (qui bat le moins de monde). On retire donc du moins bon au meilleur.

Rangement moyen Le principe est d'appliquer une distillation, une décantation et de sommer le rang obtenu par les alternatives lors de ces deux procédures. Après normalisation, on obtient un rangement des alternatives.

5.3 Fonctions de valeur additive VS surclassement

Le pour et le contre des fonctions de valeur:

- plus usuel
- requiert des évaluations quantitatives

Le pour et le contre du surclassement:

- ne force pas à évaluer quantitativement
- moins concluant (circuits)

6 Méthodes de tri

6.1 Ranger VS trier

Ranger, c'est ...

- ordonner les alternatives de la meilleure à la moins bonne
- avec des *ex aequo* éventuels
- dans le cas de candidats à un poste, par exemple, il se peut qu'aucun candidat ne convienne pour le poste, même le meilleur
- un rangement donne une évaluation relative et non absolue

Trier, c'est ...

- assigner chaque alternative à une catégorie
- les catégories sont prédéfinies et ordonnées
- dans le cas de candidats à un poste, par exemple, les candidatures sont classées en 3 catégories: *adéquat, potentiel, non adéquat*
- un tri ordonné produit une évaluation absolue

Après avoir trié ...

- il faut parfois ranger
- le fait d'avoir trié d'abord facilite le rangement
- on range seulement les alternatives de chaque catégorie sans remettre en cause le rangement induit par les catégories
- dans le cas de candidats à un poste, tous les candidats adéquats sont rangés avant les candidats potentiels, qui sont eux-mêmes rangés avant les non adéquats
- on peut utiliser une des méthodes de rangement vues précédemment
- il n'est pas toujours nécessaire de ranger toutes les alternatives

6.2 Fonction de valeur

On peut utiliser des fonctions de valeurs avec des seuils. Par exemple, si l'alternative dépasse le seuil, on le met dans la catégorie A, et sinon dans la catégorie B.

Ces fonctions de valeurs et ces seuils peuvent s'apprendre de manière automatisée par la méthode UTA.

6.3 Surclassement

MR-Sort fonctionne sur le principe suivant:

1. Pour chaque catégorie (sauf la plus mauvaise), déterminer les caractéristiques minimales d'une alternative qui appartient à cette catégorie: on l'appelle profil limite (inférieur) de la catégorie
2. Règle d'affectation : une alternative est affectée à une catégorie si elle surclasse le profil limite de la catégorie sans surclasser celui de la catégorie au-dessus
3. Surclasser : une alternative surclasse un profil si
   - elle est au moins aussi bonne que le profil sur un ensemble de critères suffisamment important
   - sans être inacceptablement moins bonne que le profil sur aucun critère

Remarquons que

Parmi les méthodes basées sur les coalitions suffisantes de critères,

- MR-Sort ne nécessite pas de phase d'exploitation,
- contrairement aux méthodes de surclassement utilisées pour ranger (sur base de comparaisons par paires)

7 Décision dans le risque et l'incertain

7.1 Décision dans l'incertain

La décision dans l'incertain, c'est quand

- les évaluations des alternatives ne sont pas sûres
- elles dépendent de la réalisation de scénarios qui ont chacun une probabilité de survenir

Le principe pour résoudre ce genre d'exercice est de calculer les probabilités, et ensuite de calculer l'espérance de gain pour chaque action considérée.

Exemple Illustrons par un exemple :

Un problème d'urnes

Il y a deux types d'urnes

U_1 une urne U_1 contient 4 boules rouges R et 6 boules noires N

U_2 une urne U_2 contient 9 boules rouges R et 1 boule noire N

On tire une urne au hasard parmi 1000 urnes dont 800 sont de type U_1 et 200 sont de type U_2
Au départ, il y a 3 actions :

A_1 parier que l'urne tirée est de type U_1

A_2 parier que l'urne tirée est de type U_2

A_0 ne pas parier

On remarque qu'en sommant le nombre total de boules rouges, et en sommant le nombre total de boules noires, il y en a 5000 de chaque sorte : #R = $(800 \cdot 4 + 200 \cdot 9) = 5000$, et #N = $(800 \cdot 6 + 200 \cdot 1) = 5000$.

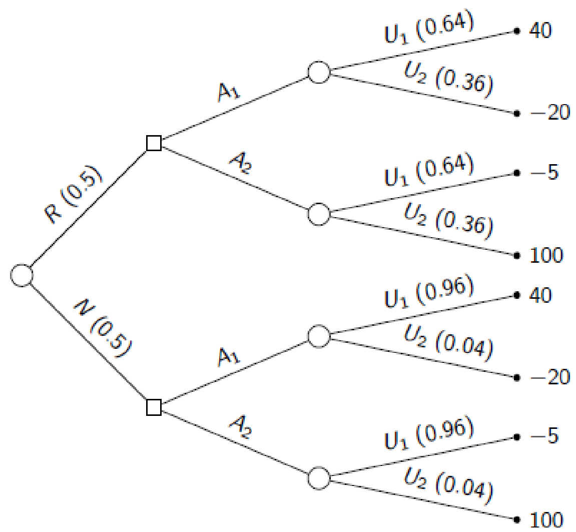
É	Actions		Probabilité
	A_1	A_2	
U_1	40 €	-5€	0.8
U_2	-20 €	100€	0.2

De plus, imaginons que l'on puisse payer pour obtenir des informations. L'exemple prendra l'action d'information e_1 qui consiste à tirer au hasard une boule de l'urne choisie pour 8 euros. Pour calculer les probabilités des résultats d'une telle action, il faut utiliser la théorie probabiliste suivante :

Probabilités conditionnelles

- Définition : $p(U_1|R) = \frac{p(U_1 \text{ et } R)}{p(R)}$
- $p(R) = P(R|U_1) \cdot p(U_1) + p(R|U_2) \cdot p(U_2)$ (Théorème des probabilités totales)
- $p(U_1|R) = \frac{p(R|U_1) \cdot p(U_1)}{p(R)}$ (Théorème de Bayes ou de probabilité des causes)

$p(U_1) = 0.8$ | $p(U_2) = 0.2$ | $p(R|U_1) = 0.4$ | $p(N|U_1) = 0.6$ | $p(R|U_2) = 0.9$ | $p(N|U_2) = 0.1$



Ensuite, pour calculer l'espérance de gain, on va partir des feuilles de l'arbre, et calculer le gain moyen pour chaque action, en comptant les probabilités calculées précédemment.

- Au nœud A_1 si R : gain espéré $= 40 \times 0.64 + (-20) \times 0.36 = 25.6 - 7.2 = 18.4$
- Au nœud A_2 si R : gain espéré $= (-5) \times 0.64 + 100 \times 0.36 = -3.2 + 36 = 32.8$

⇨ Donc, si on a obtenu R, il faut choisir l'action A_2

- Au nœud A_1 si N : gain espéré $= 40 \times 0.96 + (-20) \times 0.04 = 38.4 - 0.8 = 37.6$
- Au nœud A_2 si N : gain espéré $= (-5) \times 0.96 + 100 \times 0.04 = -4.8 + 4 = -0.8$

⇨ Donc, si on a obtenu N, il faut choisir l'action A_1

Ce qui signifie que pour l'action E1 :

- Au nœud e_1 : gain espéré $= 32.8 \times 0.5 + 37.6 \times 0.5 = 16.4 + 18.8 = 35.2$

  Le gain espéré de la branche e_1 est donc 35.2

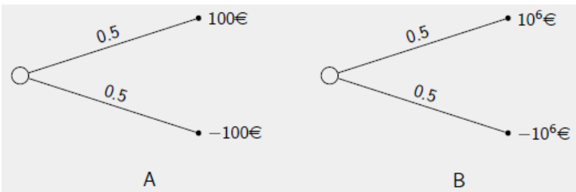
  La politique optimale associée consiste à choisir

  - l'action A_2 si le résultat de l'action d'information est R
  - l'action A_1 si le résultat de l'action d'information est N

  Le gain espéré, déduction faite du coût de e_1 est donc 35.2 -8 $=$ 27.2

7.2 Décision dans le risque

L'espérance n'est pas une mesure qui représente au mieux le choix du décideur lorsque celui peut y perdre. En effet, les loteries suivantes ont la même espérance, mais il peut paraître très risqué de pouvoir perdre le montant d'argent de la loterie B.



7.2.1 Espérance-Variance

Une première idée est de rajouter la variance dans le calcul, cependant, là encore, il peut y avoir des choix ambigus.

Le risque est lié à la variance des résultats

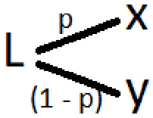
- À moyennes égales, la perception du risque est liée à la différence entre les gains ou entre le gain et la perte, donc à la variance des évaluations
- Le critère espérance-variance évalue une loterie L par :

$$F_\lambda(L) = E(L) - \lambda\sigma(L)$$

où $E(L)$ est l'espérance de gain, $\sigma(L)$ est l'écart-type des résultats et λ un coefficient d'autant plus grand que le décideur a de l'aversion pour le risque

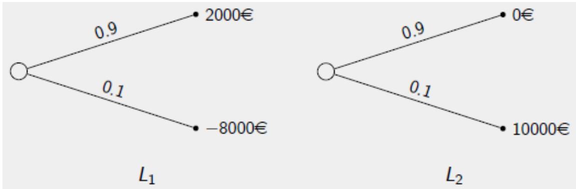
Ce critère a été proposé par Harry Markowitz en 1952 pour comparer des portefeuilles financiers

Rappel :



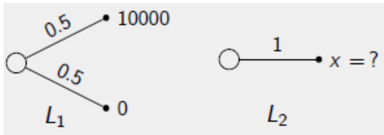
$$E(L) = p \cdot x + (1-p) \cdot y$$
$$\sigma(L) = \sqrt{p \cdot (x - E(L))^2 + (1-p) \cdot (y - E(L))^2}$$

Les loteries suivantes ont le même critère espérance-variance.



7.2.2 Utilité espérée

L'utilité espérée va dépendre de l'aversion au risque du décideur. Celle-ci s'évalue en posant des questions. Les questions sont du style à proposer des valeurs pour x au décideur pour qu'il trouve L1 équivalent à L2. On va proposer alternativement des valeurs hautes et des valeurs basses afin de trianguler le prix idéal pour le décideur.



Une fois qu'on a trouvé x, on place x à 0.5 de la fonction d'utilité. On va chercher des valeurs intermédiaires. On va donc reposer la même question, mais cette fois ci en fixant les valeurs de L1 entre 0 et x et ensuite entre x et 10000. On répète ainsi de suite jusqu'à avoir le degrés de précision recherché.

Von Neumann et Morgenstern

Cette technique est réalisable si la comparaison entre loterie respecte 5 axiomes amenés par Von Neumann et Morgenstern :

Notations :

- X = ensemble de toutes les conséquences possibles d'un problème de décision (par ex. $X = \mathbb{R}$)
- $\mathcal{L}(X)$ = toutes les loteries (finies) dont les lots appartiennent à X (loteries à un ou plusieurs niveaux = arbres)
- $\succsim$= préférence du décideur sur les loteries

Axiomes de von Neumann et Morgenstern

1. $\succsim$ est un préordre complet sur l'ensemble de toutes les loteries $\mathcal{L}(X)$
2. Réduction : on peut toujours ramener une loterie à plusieurs étages à une loterie à un étage qui lui est indifférente
3. Monotonie de $\succsim$
4. Continuité de $\succsim$
5. Indépendance de $\succsim$

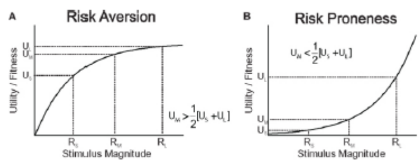
Si on accepte les 5 axiomes,
il existe une fonction $u : X \to \mathbb{R}$ telle que pour toutes loteries $L, L' \in \mathcal{L}(X)$,

$$L \succsim L' \quad \text{ssi} \quad \sum_{x \in X} p^L(x)u(x) \geq \sum_{x \in X} p^{L'}(x)u(x).$$

La fonction d'utilité u est unique à une transformation affine positive près, i.e., si u et v satisfont l'inéquation ci-dessus, alors il existe $\alpha > 0$ et $\beta \in \mathbb{R}$ tels que $u = \alpha v + \beta$.

Aversion VS Goût

On notera alors qu'une fonction d'utilité concave (gauche) représente une aversion pour le risque tandis qu'une fonction d'utilité convexe (droite) représente un goût pour le risque. Une attitude neutre se traduit alors par une fonction d'utilité linéaire. Le décideur peut toujours changer de comportement au cours de la fonction d'utilité (point d'inflexion).



La façon de poser la question au décideur influencera son aversion ou son goût pour le risque. En effet, le décideur doit se baser sur un point de référence qui peut être le gain/perte ou la modification de son capital (10000 => 10050 ou 10000 => 9950).

Évaluation du risque

On dira en général qu'un risque s'évalue selon sa fréquence $\times$ sa gravité

Table 3.4 Risk Assessment Matrix Using a Risk Index

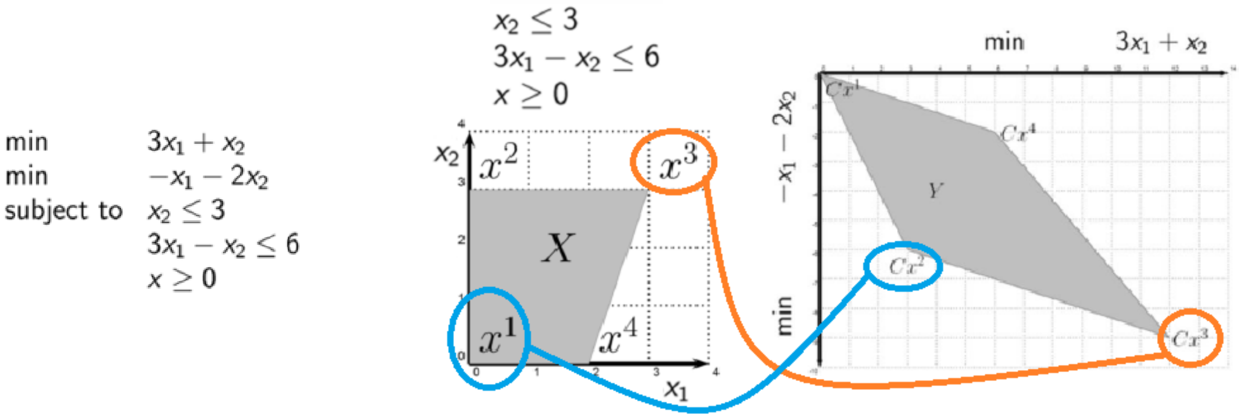
Likelihood level	Consequence level			
	I Catastrophic	II Critical	III Marginal	IV Negligible
A: Frequent	1	3	7	13
B: Probable	2	5	9	16
C: Occasional	4	6	11	18
D: Remote	8	10	14	19
E: Improbable	12	15	17	20

Criteria based on Risk Index:
1-5 = Unacceptable;
6-9 = Undesirable (project management decision required);
10-17 = Acceptable with review by project management; and
18-20 = Acceptable without review.

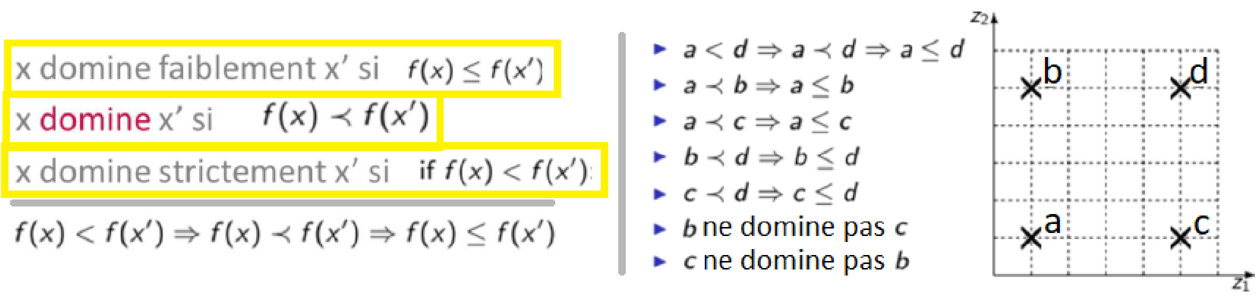
Wiggins, 1985

8 Optimisation Multi-Objectif

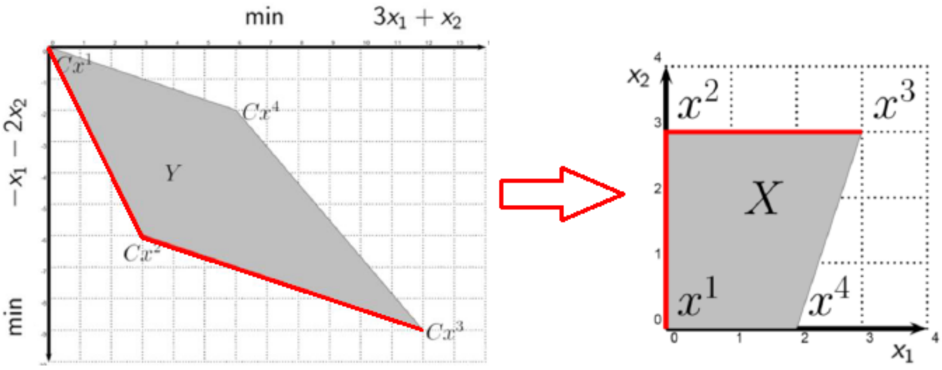
Le plus simple est d'expliquer la résolution sur un exemple de minimisation. Si jamais on a une maximisation, on doit repasser en minimisation ! La représentation du milieu contient les solutions admissibles. La représentation toute à droite est l' espace des critères, et regroupe l'image de l'ensemble des solutions admissibles. Pour passer de l'un à l'autre, on prend les coins (ou plus généralement : les points) et on les représente selon leur images en fonction des critères. Deux points ont été liés, on pourrait faire de même avec les deux autres...



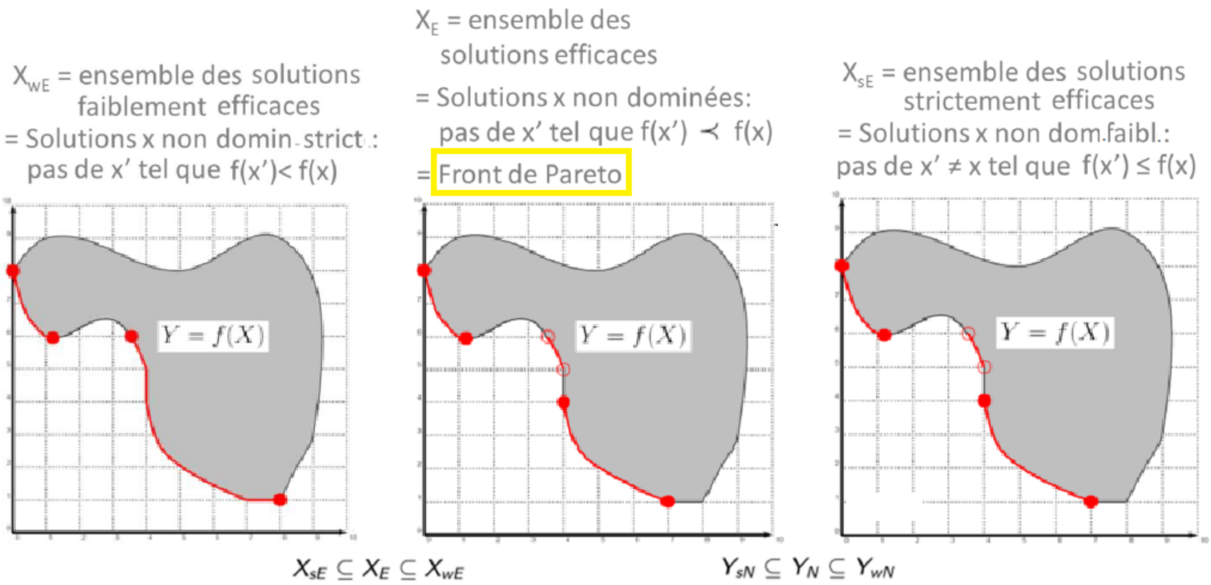
Pour choisir une solution, on va avoir recours à une comparaison de vecteurs.



Par cette notion de dominance, on peut trouver le front de Pareto de notre exemple.

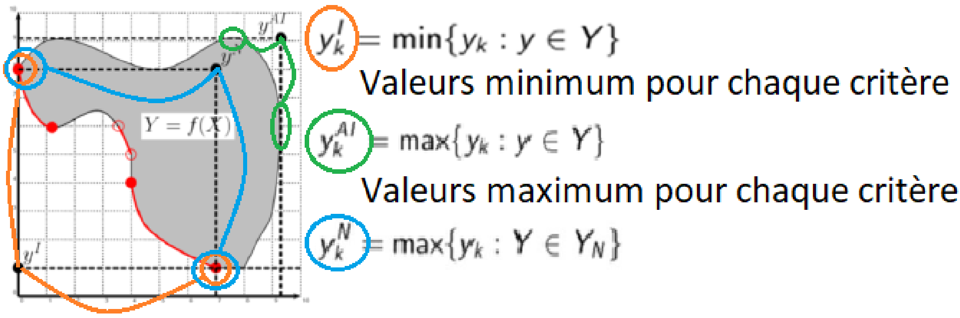


En général, voilà les différences entre les sortes de dominances :



8.1 Points spéciaux

On peut distinguer trois points spéciaux : le point idéal (y^I), le point anti-idéal (y^{AI}) et le point Nadir (y^N).



On a $\forall y \in Y_N$:

$$y_k^I \leqslant y_k$$

$$y_k \leqslant y_k^N$$

8.2 Optimum lexicographique

$$y1 <_{lex} y2 \quad \text{si} \quad y1_q < y2_q \quad \text{où} \quad q = \min\{k : y1_k \neq y2_k\}$$

Autrement dit, on doit penser comme l'ordre alphabétique : on regarde 'lettre' par 'lettre' jusqu'à tomber sur une différence, et puis on compare cette lettre : $(2,3,4) <_{lex} (2,3,5)$. On peut éventuellement ranger les dimensions (critères) des solutions par ordre d'importance ! Une solution x est un optimum lexicographique si $\forall x' : f(x) <_{lex} f(x')$.
optimum lexicographique $\Rightarrow$ solution efficace.

8.3 Résolution d'un problème

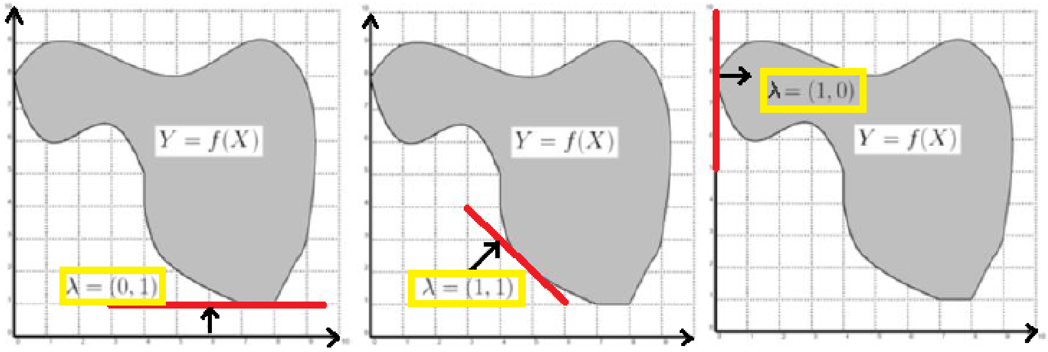
On a besoin des contraintes suivantes :

- $X_E \neq \emptyset$
- $y^I \neq y^N$

Notre méthode doit être correcte (les solution trouvées sont efficaces) et complète (on trouve toutes les solution efficaces). Pour ce faire, on peut faire une somme pondérée des critères avec un paramètre λ qui est un vecteur de poids :

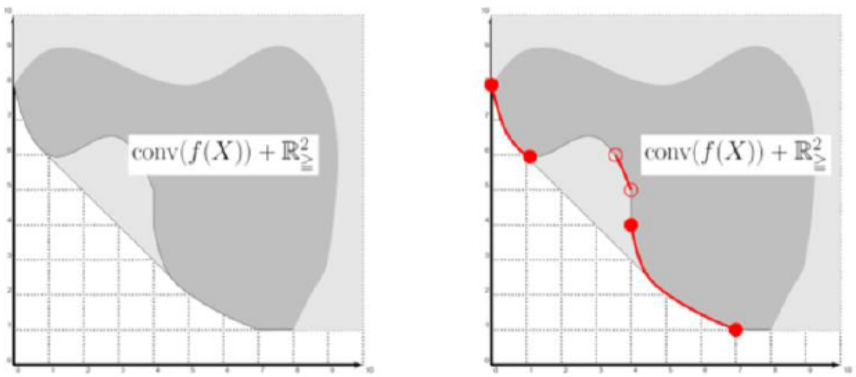
$$sol = \min\{\sum_{k=1}^{p} \lambda_k f_k(x) : x \in X\}, \text{avec } \lambda \succ 0$$

$\lambda \succ 0$ signifie qu'au moins une des composantes de lambda doit être supérieure à 0.

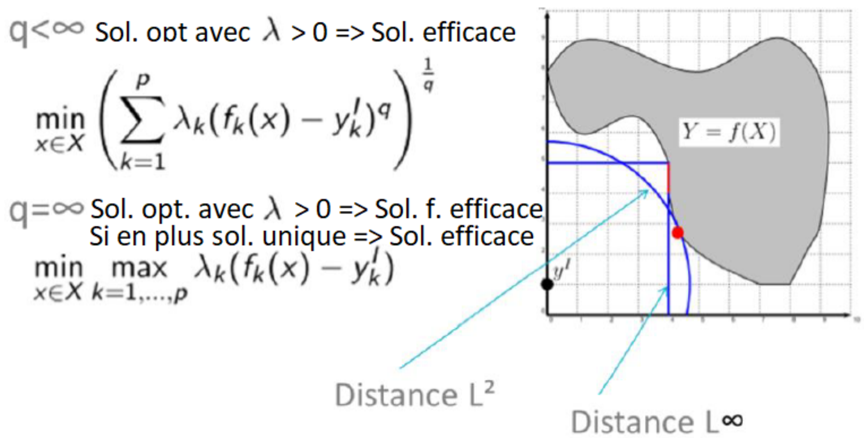


On peut prouver qu'une solution optimale de la formule ci-dessus est faiblement efficace. Si la solution optimale est unique, alors elle est efficace tout court. De plus, si une solution est optimale avec $\lambda > 0$ (toutes les composantes > 0), alors la solution est efficace (qu'elle soit unique ou non).

Théorème de Geoffrion (sol. correctes) Si $Y = f(X)$ est convexe, alors si x est faiblement efficace, $\exists \lambda \succ 0$ t.q. x est solution optimale de $\min\{\sum_{k=1}^{p} \lambda_k f_k(x) : x \in X\}$. (Rappel: un polygone est convexe si n'importe quelle ligne tracée en son sein est entièrement comprise dans le polygone. Ce n'est pas le cas des exemple plus haut.)
Dans le cas non-convexe, les solutions efficaces supportées sont les solution efficaces x telles que $f(x)$ appartiennent à l'enveloppe convexe du polygone.



Compromis Le principe est de trouver un point à une distance minimale du point idéal.



8.4 Préférences du décideur

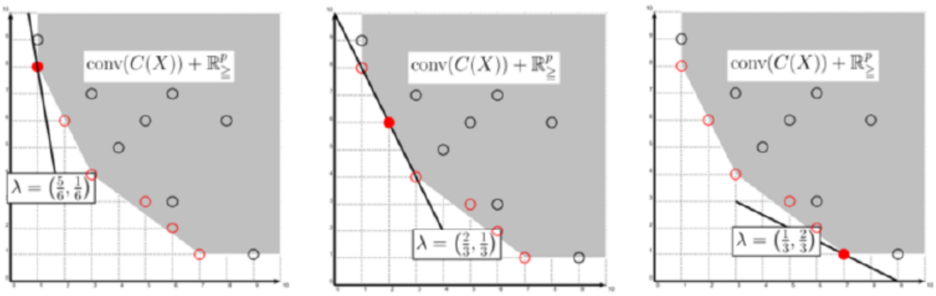
Le principe est de construire une fonction d'utilité pour chaque critère afin de "pondérer" les solutions trouvées. On peut également lui proposer des solutions afin qu'il donne un avis sur le résultat selon chaque critère.

Méthode de Steuer and Choo Cette méthode peut s'appliquer à des problèmes non linéaires. Le principe est de donner des solutions efficaces (calculée grâce à une distance d'un point idéal ou utopique) au décideur et de lui demander laquelle il préfère. On recommence ensuite avec des poids et des distances proches de la solution choisie précédemment. Une fois que le décideur est satisfait, on peut s'arrêter.

8.5 Optimisation combinatoire (discrète) multi-objectifs

Le principe est ici de fonctionner avec l'enveloppe convexe des solutions:

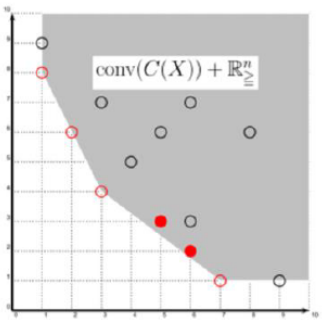
x appartient à X_{SE} si il existe $\lambda>0$ tel que pour tout x de X: $\lambda^T C\hat{x} \leqq \lambda^T Cx$



Point extrême $\rightarrow X_{SE1}$ Non extrême $\rightarrow X_{SE2}$ Point extrême $\rightarrow X_{SE1}$

Les solutions à l'intérieur de cette enveloppe sont considérés comme "non-supportés".

x dans X_{NE} : à l'intérieur de $\mathrm{conv}(Y) + \mathbb{R}^p_{\geqq}$



9 Théorie des jeux

Dans la théorie des jeux, on a des joueurs et des actions que ces joueurs peuvent effectuer.

9.1 Vocabulaire

Un jeu non-coopératif

est jeu dans lequel les joueurs ne peuvent pas conclure des accords contraignants

Jeu à stratégies simultanées vs. jeu séquentiel

- simultané : les actions sont décidées en même temps
- séquentiel : on joue les uns après les autres

Jeu symétrique

Chaque joueur est confronté à la même situation pour déterminer sa stratégie ; en particulier, la matrice des payoffs est symétrique

Jeu infiniment répété vs. jeu à horizon déterminé

Certains jeux finissent nécessairement, d'autres pas. Parfois le nombre de coups est déterminé à l'avance ou bien une règle de fin est spécifiée.

Jeux itérés vs. non itérés

Le même jeu peut être joué plusieurs fois ou une seule fois. Après chaque partie, il y a un vainqueur. Un joueur peut tenir compte de la stratégie tenue par l'adversaire lors de la partie précédente, par exemple en adoptant celle-ci dans la nouvelle partie.

- Stratégie pure : choisir une action parmi celles proposées.
- Stratégie mixte : choisir une action selon des probabilités (e.g. on lance un dés pour déterminer quelle action jouer...).
- Équilibre de Nash : paire de stratégie de laquelle aucun des deux joueurs n'a un intérêt personnel à changer unilatéralement de stratégie.
- Jeux de coordination : les joueurs ont intérêt à adopter tous les deux la même action.
- Jeux d'anti-coordination : les joueurs ont intérêt à adopter des actions différentes.

9.2 Dominance

On peut appliquer le principe de dominance par ligne ou par colonne. Prenons comme un exemple un dilemme du prisonnier (à minimiser) (à gauche : un exemple; à droite : la formule générale) :

		Col	
		avoue	n'avoue pas
Lig	avoue	10, 10	1, 20
	n'avoue pas	20, 1	2, 2

pour $x > y > z > w$

	ils font X	ils ne font pas X
je fais X	z, z	x, w
je ne fais pas X	w, x	y, y

Sur l'exemple, la ligne entourée domine la ligne non-entourée pour le joueur Lig (Scores : 10 < 20 et 1 < 2), et la colonne entourée domine la colonne non-entourée pour le joueur Col (Scores : 10 < 20 et 1 < 2). L'intersection est alors (10, 10) ce qui représente un assez mauvais score pour les deux joueurs... Le plus logique aurait été (2, 2) qui est un assez bon score pour les deux joueurs => Dans un dilemme du prisonnier, la dominance conduit à un résultat sous-optimum.

Dans un jeu séquentiel, la dominance se calcule selon un principe de récursion arrière (on part de la fin pour arriver au début, toujours en prenant l'action qui nous intéresse le plus ce tour-ci).

Dans un jeu à n joueurs, le principe reste le même. Prenons 3 comme exemple. 3 joueurs ont chacun deux actions possible :

- J1 – A, B
- J2 – F, G
- J3 – X, Y

Voici la liste des payoffs :

Actions jouées (J1 J2 J3)	**Score J1**	**Score J2**	**Score J3**	**Gardé par**
AFX	1	6	9	/
AFY	2	5	8	/
AGX	3	9	9	/
AGY	4	8	7	/
BFX	2	1	3	J1, J2, J3
BFY	3	2	2	J1, J2
BGX	4	0	7	J1
BGY	5	1	2	J1

On va alors résonner joueur par joueur de manière rationnelle (chaque joueur suppose que les autres joueurs maximisent leur propre score et jouent donc de la même manière que le premier). Le joueur 1 va préférer jouer l'action B car dans tous les cas (quelques soient les actions que ses adversaires considèrent), l'action B réussi mieux au joueur 1. Maintenant qu'on sait que le joueur 1 va choisir l'action B, on va regarder les actions plausibles du joueur 2 en fonction. On préfère choisir le F car les cas "BG" sont moint intéressant que les cas "BF" pour le joueur 2. Enfin, il reste au joueur 3 à sélectionner une des deux actions restantes. Il va préférer X car l'action lui rapporte 3 tandis que "BFY" ne lui rapporte que 2.

9.3 Minimax

Le critère du minimax s'applique aux jeux à somme nulle. Ces jeux ont une seule notion de score. Ce qui est gagné par un joueur est perdu par l'autre. Le but d'un joueur est donc de maximiser le score et celui de l'autre joueur est de minimiser le score.

Équilibre pur

On dit alors qu'une paire de stratégies pures en équilibre est une paire qui réalise le résultat maximum de sa colonne et le minimum de sa ligne. Dans l'exemple ci-contre, les entourés sont les maximums de colonne et les encadrés sont les minimum de ligne. Un seul membre fait parti des deux ensembles : c'est l'équilibre selon le critère minimax. Il peut y avoir plusieurs paires de stratégies en équilibre, mais alors elles se situent soit dans la même ligne, soit dans la même colonne. Il se peut qu'il n'y ait pas d'équilibre.

	C_1	C_2	C_3
L_1	3	$\boxed{-3}$	7
L_2	4	5	6
L_3	2	7	$\boxed{-1}$

Équilibre mixte On peut également chercher des stratégies mixtes en équilibre.

Théorème du minimax

Théorème

Tout jeu de deux personnes à somme nulle a une solution, càd une paire de stratégies en équilibre. S'il y en a plusieurs, elles donnent la même espérance de gain.

Preuve

Considérons un jeu sous forme standard, càd

	C_1	C_2
L_1	a	d
L_2	c	b

avec $a > c$, $b > d$, $b > c$ et $a, b, c, d > 0$

Soit une paire de stratégies mixtes : $(L_1 : p, L_2 : 1-p)$ et $(C_1 : q, C_2 : 1-q)$. On a :

$$\begin{aligned} EU_L &= apq + dp(1-q) + c(1-p)q + b(1-p)(1-q) \\ &= (a-d-c+b)pq + (d-b)p + (c-b)q + b \end{aligned}$$

Posons $K = a-d-c+b$, $L = -(d-b)$, $M = -(c-b)$ et $N = b$.
À cause des hypothèses faites sur a, b, c, d, on a que $K, L, M, N > 0$.

$$\begin{aligned} EU_L &= Kpq - Lp - Mq + N \\ &= K[(p - M/K)(q - L/K)] + (NK - LM)/K \end{aligned}$$

En prenant $q = L/K$, C est sûr que $EU_L \leq (NK - LM)/K$. De son côté, L peut s'assurer que $EU_L \geq (NK - LM)/K$ en prenant $p = M/K$.

Si les deux joueurs choisissent les stratégies ci-dessous, aucun des deux n'a intérêt à s'en écarter unilatéralement.

$$\begin{array}{ll} (L_1 : M/K & , \ L_2 : (1 - M/K)) \\ (C_1 : L/K & , \ C_2 : (1 - L/K)) \end{array} \quad \text{ou :} \quad \begin{array}{lll} L_1 & : & \dfrac{|c-b|}{a-d-c+b} \\ C_1 & : & \dfrac{|d-b|}{a-d-c+b} \end{array}$$

9.4 Équilibre de Nash

Un équilibre de Nash est un équilibre duquel aucun des deux joueurs n'a intérêt à changer unilatéralement d'action. Dans l'exemple suivant, les deux équilibres purs sont mis en évidence. Depuis l'équilibre encadré, le joueur L n'a pas intérêt à passer de l'action L_1 à L_2 car il y perdrait 1, et le joueur C n'a pas intérêt à passer de l'action C_2 à C_1 car il y perdrait 99.

		p	$(1 - p)$
		C_1 : go	C_2 : stop
q	L_1 : go	$-100, -100$	$\boxed{1, -1}$
$(1 - q)$	L_2 : stop	$-1, 1$	$0, 0$

Il existe également un équilibre de Nash mixte dans l'exemple ci-dessus. Il se calcule de la manière suivante (EU veut dire utilité espérée) :

<u>Joueur C</u> (on regarde aux actions que peut faire L et au score que ce dernier pourrait obtenir) :
$EU(L_1) = -100 \cdot p + 1 \cdot (1 - p)$
$EU(L_2) = -1 \cdot p + 0 \cdot (1 - p)$
$EU(L_1) = EU(L_2)$ (Nash) $\Leftrightarrow -100p + 1 - p = -p \Leftrightarrow p = \frac{1}{100}$
On a donc que le joueur C doit joueur l'action C_1 dans $\frac{1}{100}$ des cas, et l'action C_2 dans $\frac{99}{100}$ des cas.

<u>Joueur L</u> (on regarde aux actions que peut faire C et au score que ce dernier pourrait obtenir) :
$EU(C_1) = -100 \cdot q + 1 \cdot (1 - q)$
$EU(C_2) = -1 \cdot q + 0 \cdot (1 - q)$
$EU(C_1) = EU(C_2)$ (Nash) $\Leftrightarrow -100p + 1 - q = -q \Leftrightarrow q = \frac{1}{100}$
On a donc que le joueur L doit joueur l'action L_1 dans $\frac{1}{100}$ des cas, et l'action L_2 dans $\frac{99}{100}$ des cas.

9.5 Solution de Nash

Dans un jeu en coopération, les joueurs ont intérêt à trouver un équilibre, sans quoi leur score pourrait être mauvais. Dés lors, une solution de Nash peut être trouvée en utilisant des fonctions d'utilité.

Axiomes de Nash

Des joueurs rationnels devraient normalement accepter les principes suivants :

- les stratégies choisies devraient être des équilibres de Nash
- les stratégies choisies ne dépendent pas de la fonction d'utilité choisie pour représenter la préférence de chaque joueur
- indépendance des alternatives irrelevantes : on peut enlever ou ajouter des stratégies qui ne sont préférées par aucun joueur sans influencer les stratégies choisies
- symétrie : les stratégies des joueurs sont identiques si et seulement si leurs fonctions d'utilité sont identiques

- Si ces axiomes sont acceptés par les joueurs, Nash montre que le partage doit se faire en maximisant le produit des utilités des joueurs :

$$\max_{(x_1,\ldots,x_n)} u_1(x_1) \cdot u_2(x_2) \cdot \ldots \cdot u_n(x_n)$$

où $\sum_{i=1}^{n} x_i = 100\$$ (exemple)
- on appelle cette solution, *solution de Nash*
- Exemple : 2 joueurs et les utilités sont linéaires, càd $u_1(x) = u_2(x) = x$, alors la solution optimale est $x_1 = x_2 = 50$

Commentaires sur les axiomes

- les axiomes 1 et 2 impliquent que la solution de Nash est Pareto-optimale
- l'axiome de symétrie (Ax. 4) signifie que les joueurs ont des capacités de négociation égales, puisque la seule chose qui compte pour le partage est la forme de leur fonction d'utilité

9.6 Stratégie évolutionnaire stable

En comparant deux stratégies mixtes, une stratégie mixte est dite évolutionnaire stable si celle-ci a une espérance d'utilité plus élevée que l'autre stratégie.