Monotonie des méthodes de rangement par choix répété

Document de travail
27 janvier 1995

1. Définitions et Notations.

Soit X un ensemble fini tel que |X| = m = 3 (le cas m = 2 est trivial). On note :
F(X) l'ensemble de toutes les relations binaires floues réflexives (ceci uniquement pour ne pas avoir à se préoccuper des boucles) sur X et $N(X) \subset F(X)$ l'ensemble de toutes les relations binaires nettes réflexives sur X.
(Remarque : dans ce qui suit l'inclusion est notée $\subseteq$, l'inclusion stricte étant notée $\subset$).

Si $R \in N(X)$, on notera x R y au lieu de R(x, y) = 1.

Un **préordre complet** sur X est une relation nette complète et transitive. On notera WO(X) l'ensemble de tous les préordres complets sur X. Soit R une préordre complet sur X, on notera $U_k(R)$ la kième classe d'équivalence de ce préordre, c'est-à-dire, pour k = 1, 2, 3, …,
$U_k(R) = \{x \in X[R, k] : x R y, \forall y \in X[R, k]\}$,
où X[R, 1] = X et pour k = 2, 3, ..., $X[R, k] = X[R, k-1] \backslash U_{k-1}(R)$.
Il est clair que $U_1(R)$ est toujours non vide.

Soit $R \in F(X)$ et $Y \subseteq X$. On notera R/Y la restriction de R à Y, c'est-à-dire l'élément de F(Y) tel que, $\forall x, y \in Y$, R/Y(x, y) = R(x, y).

Une **procédure de choix** C est une fonction associant à tout ensemble fini X, à tout sous-ensemble A non vide de X et à tout élément R de F(X) un ensemble C(X, A, R) tel que $C(X, A, R) \subseteq A$ et $C(X, A, R) \neq Ø$.

Une **procédure de rangement** $\geq$ est une fonction associant à tout ensemble fini X et à tout élément R de F(X) un élément $\geq(X, R)$ de WO(X). On notera $\geq(X, R)$ (resp. $>(X, R)$) la partie symétrique (resp. asymétrique) de $\geq(X, R)$.

Étant donné une procédure de choix C, son itération (descendante) définit une procédure de rangement conduisant à un préordre dont la première classe d'équivalence est l'ensemble des éléments sélectionnés par C, la deuxième ceux sélectionnés par C parmi les éléments non sélectionnés à la première étape, etc. Formellement, l'itération descendante d'une procédure de choix C conduit à une procédure de rangement $\geq_C$ telle que pour tout ensemble fini X et pour toute relation $R \in F(X)$, le préordre $\geq_C(X, R)$ est tel que, pour k = 1, 2, …,
$U_k(\geq_C(X, R)) = C(X, X[\geq_C(R), k], R)$.
(Rappelons que $X[\geq_C(R), k]$ désigne l'ensemble X moins les k-1 premières classes d'équivalence de $\geq_C(R)$)

2. Diverses formes de monotonie.

Définition 2.1

On dira qu'une procédure de rangement ≥ est **monotone** si, pour tout ensemble fini X, toute relation $R \in F(X)$ et tous $a, b \in X$:
$[a \geq(X, R)\, b \Rightarrow a \geq(X, R')\, b]$ et $[a > (X, R)\, b \Rightarrow a > (X, R')\, b]$
où R′ est une relation de F(X) telle que :
$[R'/A\backslash\{a\} = R/A\backslash\{a\}$ et, $\forall\, c \in A\backslash\{a\}$, $R'(a, c) = R(a, c)$ et $R'(c, a) = R(c, a)]$ (on dira alors que R′ améliore a).

Remarques

a) Soit ≥ une procédure de rangement monotone. On a, pour tout ensemble fini X, toute relation $R \in F(X)$ et tous $a, b \in X$:
$[a \geq(X, R)\, b \Rightarrow a \geq(X, R')\, b]$ et $[a > (X, R)\, b \Rightarrow a > (X, R')\, b]$
où R′ est une relation de F(X) telle que :
$[R'/A\backslash\{b\} = R/A\backslash\{b\}$ et, $\forall\, c \in A\backslash\{b\}$, $R'(b, c) = R(b, c)$ et $R'(c, b) = R(c, b)]$ (on dira que R′ détériore b).
La définition de la monotonie est donc symétrique par rapport aux améliorations/détériorations.

b) Une définition équivalente de la monotonie consiste à écrire :
$a \geq(X, R)\, b \Rightarrow a \geq(X, R')\, b$,
où R′ améliore a ou détériore b.

c) Lorsque R est une relation de surclassement, la condition de monotonie traduit l'exigence suivante : si les performances de a augmentent, sa position dans le classement ne peut diminuer – l'action a reste strictement avant toutes les actions qu'elle battait déjà et elle ne peut que monter par rapport à celles qui lui étaient indifférentes. Ceci implique que, symétriquement, si les performances de b diminuent, sa position dans le classement ne peut augmenter – les actions qui étaient strictement avant b le restent et la position de b par rapport aux actions qui lui étaient indifférentes ne peut que se dégrader.

Définition 2.2
On dira qu'une procédure de rangement $\geq$ est **faiblement monotone** si, pour tout ensemble fini X, toute relation $R \in F(X)$ et tous $a, b \in X$:

$a \geq(X, R)\ b \Rightarrow a \geq(X, R')\ b$,

où R′ améliore a.

Remarques

a) La faible monotonie est une condition permettant d'avoir :

$a >(X, R)\ b$ et $a \geq(X, R')\ b$,

où R′ améliore a ou détériore b,

ce qui paraît criticable pour une condition de monotonie. On verra cependant qu'il n'est pas simple de trouver des procédures de rangement par choix répété qui soient faiblement monotones.

b) Il est clair que la monotonie implique la faible monotonie.

Le problème posé est le suivant :

Existe-t-il des procédures de rangement par choix répété non triviales qui soient monotones ?

Nous donnerons plus bas un exemple d'une telle procédure de rangement (elle est peu attrayante !). A l'exception de cet exemple nous n'avons que peu de choses à dire à ce sujet. En revanche, nous apporterons davantage d'éléments de réponse à la question :

Existe-t-il des procédures de rangement par choix répété non triviales qui soient faiblement monotones ?

Remarque. La nécessité du qualificatif "non triviale" provient du fait que la méthode de rangement consistant à toujours classer l'ensemble des actions dans une seule classe d'équivalence est monotone.

La notion de monotonie a une contrepartie pour ce qui concerne les procédures de choix.

Définition 2.3

On dira qu'une procédure de choix C est **monotone** si, pour tout ensemble fini X, tout sous-ensemble non vide A de X et toute relation $R \in F(X)$:

$a \in C(X, A, R) \Rightarrow a \in C(X, A, R')$,

où R' améliore a.

Remarques

a) Il évident de montrer que si C est monotone alors :

$b \notin C(X, A, R) \Rightarrow b \notin C(X, A, R')$,

où R' est une relation détériorant b.

b) La monotonie d'une procédure de choix n'exclue pas d'avoir :

$a \in C(X, A, R)$, $b \notin C(X, A, R)$ et $b \in C(X, A, R')$,

où R' améliore a. D'où la définition suivante.

Définition 2.4

On dira qu'une procédure de choix C est **fortement monotone** si elle est monotone et si, pour tout ensemble fini X, tout sous-ensemble non vide A de X, tous $a, b \in X$ et toute relation $R \in F(X)$:

$[a \in C(X, A, R)$ et $b \notin C(X, A, R) \Rightarrow b \notin C(X, A, R')]$,

où R' améliore a.

Définition 2.5

On dira qu'une procédure de choix C est **super monotone** (resp. **super faiblement monotone**) si la procédure de rangement $\geq_C$ définie par son itération descendante est monotone (resp. faiblement monotone).

Remarques

Considérons la procédure de choix consistant dans tout sous-ensemble de X à choisir les éléments dont le score de Flot Net est maximal **dans X**. Cette procédure de choix est fortement monotone. Il est évident que la procédure de rangement itérant une telle procédure de choix est monotone. La difficulté provient ici du fait que l'on a considéré une procédure de choix qui n'est pas "locale", c'est-à-dire dont le résultat ne dépend pas uniquement du comportement de la relation sur le sous-ensemble considéré. La question de savoir s'il existe des procédures de rangement par choix itéré qui sont (faiblement) monotones ne nous semble avoir d'intérêt véritable que pour des procédures locales (il semble que c'est ainsi que Patrice a, implicitement, voulu formuler sa question). De fait, il n'est pas simple de trouver des procédures de choix locales dont l'itération conduise à une procédure de rangement ayant certaines propriétés de monotonie (cf. Thèse de Patrice, Chapitre 5, pages 255-256).

3. Propriétés d'une procédure de choix.

Nous introduisons dans cette section quelques propriétés des procédures de choix, en dehors de celles ayant trait à la monotonie.

Définition 3.1
On dira qu'une procédure de choix C est **locale** (ou, selon l'usage le plus répandu indépendante vis-à-vis des alternatives non concernées) si, pour tous ensembles finis X et Y, tout sous-ensemble non vide $A \subseteq X \times Y$, toute relation $R \in F(X)$ et toute relation $R' \in F(Y)$:
$[R/A = R'/A] \Rightarrow C(X, A, R) = C(Y, A, R')$.

Lemme 3.1. Soit C une procédure de choix locale. Si C est super faiblement monotone alors elle est monotone.

Démonstration. Soit C une procédure de choix locale et non monotone. Il existe donc un ensemble fini X, $A \subseteq X$ et $R \in F(X)$ tels que :
$a \in C(X, A, R)$ et $a \notin C(X, A, R')$,
où R′ améliore a.
La procédure C étant locale, on a :
$a \in C(A, A, R/A)$ et $a \notin C(A, A, R'/A)$, d'où :
$a \in U_1(\geq_C(A, R))$ et $a \notin U_1(\geq_C(A, R'))$, ce qui viole la faible monotonie de $\geq_C$. ❈

Lemme 3.2. Soit C une procédure de choix locale. Si C est super monotone alors elle est fortement monotone.

Démonstration. Soit C une procédure de choix locale et non fortement monotone. Si C est non monotone alors, en vertu du lemme 4.1, elle n'est pas super faiblement monotone et donc pas super monotone. Supposons donc que C soit monotone mais pas fortement monotone. Il existe donc un ensemble fini X, $A \subseteq X$ et $R \in F(X)$ tels que :

$[a \in C(X, A, R)$, $b \notin C(X, A, R)$ et $b \in C(X, A, R')]$,
où R' améliore a.
La procédure C étant locale, on a :
$a \in C(A, A, R/A)$, $b \notin C(A, A, R/A)$ et $b \in C(A, A, R'/A)$, d'où :
$a \in U_1(\geq_C(A, R))$, $b \notin U_1(\geq_C(A, R))$ et $b \in U_1(\geq_C(A, R'))$,
ce qui viole la monotonie de $\geq_C$. ❈

Remarques
a) Pour les procédures de choix locales, la monotonie (resp. la forte monotonie) est donc une condition nécessaire de super faible monotonie (resp. super monotonie).

b) Au vu des lemmes 3.1 et 3.2, la terminologie retenue peut paraître curieuse ! Nous avons cependant voulu baptiser "monotonie" à la fois pour les procédures de choix et de rangement ce qui nous a semblé être la condition la plus "naturelle".

Définition 3.2
On dira qu'une procédure de choix C est **neutre** si, pour tout ensemble fini X, toute relation $R \in F(X)$, tout $A \subseteq X$ et toute permutation σ de X telle que, $\forall x \in X\backslash A$, $\sigma(x) = x$,
$[a \in C(X, A, R) \Leftrightarrow \sigma(a) \in C(X, A, R^{\sigma})]$
où $R^{\sigma} \in F(X)$ est telle que $R^{\sigma}(\sigma(a), \sigma(b)) = R(a, b)$, pour tout $a, b \in X$.

Pour mémoire notons le :

Lemme 3.3. Si C est une procédure de choix neutre et locale, alors pour tout ensemble fini X et toute relation $R \in F(X)$:
$[R(a, b) = R(b, c) = R(c, a)$ et $R(a, c) = R(c, b) = R(b, a)] \Rightarrow C(X, \{a, b, c\}, R) = \{a, b, c\}$.

Démonstration. La procédure C étant locale, $C(X, \{a, b, c\}, R)$ ne dépend que de $R/\{a, b, c\}$. Puisque pour toute permutation σ de $\{a, b, c\}$, on a $R/\{a, b, c\} = R^{\sigma}/\{a, b, c\}$, supposer $a \notin C(X, \{a, b, c\}, R)$ impliquerait que $C(X, \{a, b, c\}, R) = \emptyset$. ❈

Définition 3.3
Soit X un ensemble fini, $R \in F(X)$, $A \subseteq X$ et $a, b \in A$. On dira que "a Couvre b dans A avec R", ce que nous noterons a VS(A, R) b, si :
$R(a, b) > R(b, a)$ et, $\forall c \in A\backslash\{a, b\}$, $R(a, c) \geq R(b, c)$ et $R(c, a) \leq R(c, b)$.

Remarque. Il est clair que VS(A, R) est une relation nette asymétrique et transitive.

Définition 3.3

On dira qu'une procédure de choix C est **compatible avec la relation de couverture stricte** si, pour tout ensemble fini X, tout sous-ensemble non vide A de X, toute relation $R \in F(X)$ et tout a, $b \in A$:

a VS(A, R) b $\Rightarrow$ b $\notin$ C(X, A, R).

Définition 3.5

On notera UC (UnCovered Set), la plus grande procédure de choix compatible avec la relation de couverture stricte, c'est-à-dire, la procédure des choix sélectionnant pour $R \in F(X)$ dans tout sous-ensemble $A \subseteq X$ l'ensemble :

UC(X, A, R) = {a $\in$ A : $\forall$ b $\in$ A, Non(b VS(A, R) a)}.

Remarque. Compte tenu des propriétés de VS(A, R), cet ensemble est toujours non vide. Il s'agit donc bien d'une procédure de choix.

Définition 3.6

Une procédure de choix C vérifie :

α si $[x \in A \subseteq B] \Rightarrow [x \in C(X, B, R) \Rightarrow x \in C(X, A, R)]$,

$\alpha 2$ si $[x \in C(X, A, R)] \Rightarrow [x \in C(X, \{x, y\}, R), \forall y \in A]$,

$\gamma 2$ si $[x \in C(X, \{x, y\}, R), \forall y \in A] \Rightarrow [x \in C(X, A, R)]$,

γ si $[D = A \cup B, x \in C(X, A, R), x \in C(X, B, R)] \Rightarrow [x \in C(X, D, R)]$,

β si [$A \subseteq B$ et $C(X, A, R) \times C(X, B, R)$? Ø] $\Rightarrow$ $[C(X, A, R) \subseteq C(X, B, R)]$,

β+ si [$A \subseteq B$ et $A \times C(X, B, R)$? Ø] $\Rightarrow$ $[C(X, A, R) \subseteq C(X, B, R)]$,

ε si $[A \subseteq B] \Rightarrow$ [Non($C(X, B, R) \subset C(X, A, R)$)],

ε+ si $[C(X, B, R) \subseteq A \subseteq B] \Rightarrow [C(X, A, R) \subseteq C(X, B, R)]$,

O si $[C(X, B, R) \subseteq A \subseteq B] \Rightarrow [C(X, A, R) = C(X, B, R)]$,

Idempotence si [C(C(X, A, R)) = C(X, A, R)],

*PI si $C(X, [C(X, A, R) \cup C(X, B, R)], R) \subseteq C(X, A \cup B, R)$,

PI * si $C(X, A \cup B, R) \subseteq C(X, [C(X, A, R) \cup C(X, B, R)], R)$,

PI si *PI et PI*

pour tout ensemble fini X, toute relation R de F(X) et tous sous-ensembles A, B, D non vides de X.

Remarques

a) On connaît bien les liens entre ces diverses conditions. Par exemple on a (cf. Sen, Moulin, Aizerman) :

$\alpha \Rightarrow \alpha 2$,

$\gamma \Rightarrow \gamma 2$,

[C rationalisable] $\Leftrightarrow$ [α et γ] $\Leftrightarrow$ [$\alpha 2$ et $\gamma 2$],

[C rationalisable par une relation dont la partie asymétrique est transitive] $\Leftrightarrow$ [α, γ, ε] $\Leftrightarrow$ [α, γ, $\varepsilon+$] $\Leftrightarrow$ [α, γ, O],

[C est rationalisable par un préordre complet] $\Leftrightarrow$ [α et β],

$\varepsilon+ \Rightarrow \varepsilon$,

$\beta+ \Rightarrow \beta$, $\varepsilon+$, γ,

b+ $\Rightarrow$ *PI,

$\alpha \Leftrightarrow$ PI*

PI $\Leftrightarrow$ [α, O] $\Leftrightarrow$ [α, $\varepsilon+$] $\Leftrightarrow$ [α, ε],

O $\Rightarrow$ $\varepsilon+$,

O $\Rightarrow$ idempotence,

O $\Leftrightarrow$ $\varepsilon+$ et idempotence,

PI et rationalisation sont indépendantes,

O et $\beta+$ sont indépendantes (cf. Bordes (1979)),

O et β sont indépendantes,

O et γ sont indépendantes,

O et α sont indépendantes,

α et γ sont indépendantes.

b) Pour les procédures de choix conduisant toujours à des singletons, O et γ sont équivalentes. Elles impliquent la rationalisation par un ordre complet et donc toutes les autres conditions. Notons qu'en vertu de lemme 4.3, il n'existe pas de procédures de choix neutres et locales conduisant toujours à des singletons. *Il ne faut donc pas chercher dans les "singletons" une solution au problème de la super monotonie.*

c) La condition O est appelée "Strong Superset Property" par certain auteurs (Moulin, Laffond, etc.) et "Nash" par d'autres (Suzumura). La terminologie O vient ici de Aizerman (O = Outcast). Attention ne pas confondre O avec ce que certains auteurs appellent "Aizerman" qui est ici la propriété $\varepsilon+$ (Aizerman n'utilise jamais $\varepsilon+$) !

d) La condition O est appelée à jouer un rôle important dans la suite. Elle traduit le fait qu'une fonction de choix est insensible à la disparition d'éléments non choisis. Enlever un ou plusieurs ou même tous les éléments non choisis ne perturbe pas le choix.

4. Exemples.

Il n'est pas simple de trouver des procédures de choix non triviales qui soient super monotones ou, même, super faiblement monotones. Nous rappelons le problème en présentant deux exemples. Le premier utilise une procédure de choix fondée sur un score, le second une procédure de choix fondée sur la notion de couverture.

Exemple 1 (Emprunté à Patrice). Soit R la relation nette sur X = {a, b, c, d, e} définie par :
a R b, b R d, b R e, c R a, d R a, d R c, d R e.
Utilisons la procédure de choix consistant à retenir dans tout sous ensemble les éléments de score de Copeland (qualification) maximum dans ce sous-ensemble. Une telle procédure de choix est locale et fortement monotone. Il est facile de constater que l'itération descendante de cette procédure de choix conduit au préordre (en utilisant une notation, allégée, évidente) :
d > c > a > b > e.
Considérons à présent la relation R′ identique à R à l'exception de Non(d R′ a). On aboutit alors au préordre :
(bd) > c > (ae).
On avait [a strictement avant b]. L'action a s'améliore et l'on obtient [b strictement avant a], ce qui montre que C, bien que fortement monotone, n'est pas super faiblement monotone.

Exemple 2 (Emprunté à Philippe). Soit R la relation nette sur X = {a, b, c, d, e} définie par :
b R c, b R d, c R d, d R a, d R e, e R a, e R b, e R c.
Utilisons la procédure de choix UC. Une telle procédure de choix est locale et monotone (il est facile de vérifier qu'elle n'est pas fortement monotone). L'itération descendante de cette procédure de choix conduit au préordre :
(bde) > (ac).
Considérons à présent la relation R′ identique à R à l'exception de a R′ b. On aboutit alors au préordre :
(bcde) > a.
On avait [a indifférente à c]. L'action a s'améliore et l'on obtient [c strictement avant a], ce qui montre que C, bien que monotone, n'est pas super faiblement monotone.

Remarque : On trouvera d'autres exemples dans la Thèse de Patrice, pages 254 et ss et infra.

5. Conditions suffisantes garantissant la super-faible monotonie.

À Liège Patrice a donné un résultat que j'interprète de la manière suivante :
Si C est monotone et vérifie β+ alors :
$a >_C(X, R)\ b \Rightarrow a \geq_C(X, R')\ b$,
où R′ améliore a.

Remarques.
a) La démonstration (telle que donnée par Patrice à Liège) repose sur le principe suivant. En contradiction avec la thèse supposons qu'il existe un ensemble X, une relation $R \in F(X)$, deux éléments $a, b \in X$ tels que :
$a >_C(X, R)\ b$ et $b >_C(X, R')\ a$,
où R′ améliore a.
Or $b >_C(X, R')\ a$ implique l'existence d'un sous-ensemble A de X contenant a et b et pour lequel $b \in C(X, A, R')$ et $a \notin C(X, A, R')$.
Supposons $a \in C(X, \{a, b\}, R')$. Le respect de β+ imposerait alors $a \in C(X, A, R')$ puisque : $[a \in C(X, \{a, b\}, R'), b \in \{a, b\}, \{a, b\} \subseteq A] \Rightarrow [b \in C(X, A, R') \Rightarrow a \in C(X, A, R')]$. On a donc $\{b\} = C(X, \{a, b\}, R')$. Puisque C est monotone, on a alors $\{b\} = C(X, \{a, b\}, R)$. La condition β+ impose alors que dans tout sous-ensemble B où sont présents a et b, $[a \in C(X, B, R) \Rightarrow b \in C(X, B, R)]$. Il est donc impossible que $a >_C(X, R)\ b$.

b) La démonstration ne se généralise pas facilement pour prouver que :
$a \geq_C(X, R)\ b \Rightarrow a \geq_C(X, R')\ b$,
où R′ améliore a.

c) Ce résultat présente, à mon sens, trois faiblesses :
- il n'utilise pas la localité de la procédure de choix. Or, on a vu qu'il est très simple pour une procédure de choix non locale d'être super monotone !
- il concerne une propriété plus faible que la super faible monotonie,
- il utilise une condition (β+) qui est forte (cf. infra).

On a :

Proposition 5.1. Si une fonction de choix est locale, monotone et vérifie la condition O elle est super faiblement monotone.

Démonstration. Supposons que C ne soit pas super faiblement monotone. Alors il existe un ensemble X, une relation $R \in F(X)$, deux éléments $a, b \in X$ tels que :
$[a \geq_C(X, R)\ b$ et $b >_C(X, R')\ a]$,

où R′ améliore a.
La procédure C étant monotone, on sait que $a \notin U_1(\geq_C(X, R))$ (sinon la monotonie impliquerait $a \in U_1(\geq_C(X, R'))$, en contradiction avec $b >_C(X, R')\ a$) et on supposera que $a \in U_r(\geq_C(X, R))$, avec r = 2. Notons que, par construction, $a \notin U_1(\geq_C(X, R'))$.

Posons $A = X \backslash \{a\}$. On a $R/A = R'/A$. La procédure de choix C étant locale, on a :
$C(X, A, R) = C(X, A, R')$.
Puisque $a \notin U_1(\geq_C(X, R))$, on a $C(X, X, R) \subseteq A \subseteq X$. La condition O impose :

$C(X, A, R) = C(X, X, R)$.
Puisque $a \notin U_1(\geq_C(X, R'))$, on a $C(X, X, R') \subseteq A \subseteq X$. La condition O impose :

$C(X, A, R') = C(X, X, R')$.
On a alors $C(X, X, R) = C(X, X, R')$ et donc $U_1(\geq_C(X, R)) = U_1(\geq_C(X, R'))$. Notons en particulier que $b \notin U_1(\geq_C(X, R'))$.
Si $a \in U_2(\geq_C(X, R'))$, il est alors impossible que $b >_C(X, R')\ a$. Supposons donc que $a \notin U_2(\geq_C(X, R'))$.

La procédure C étant locale, il est facile de constater que l'on peut répéter le raisonnement fait sur X en considérant $X[R, 2] = X[R', 2]$. On a donc :
$U_2(\geq_C(X, R)) = U_2(\geq_C(X, R'))$ et $b \notin U_2(\geq_C(X, R'))$.

En itérant le raisonnement précédant, il est facile de constater que la mise en défaut de la super monotonie ne peut se produire que si :
$U_k(\geq_C(X, R)) = U_k(\geq_C(X, R'))$ pour k = 1, 2, …, r-1, (ce qui implique $X[R, r] = X[R', r]$), $a \in X[R, r]$ et $b \in X[R, r]$.
Or, on a, par construction, $a \in U_r(\geq_C(X, R))$ et donc $a \in C(X, X[R, r], R)$. La procédure C étant monotone, on a alors $a \in C(X, X[R, r], R')$ et donc $a \geq_C(X, R')\ b$, ce qui achève la démonstration. ❈

Remarques.
a) En vertu du lemme 4.1, on sait que la monotonie est une condition nécessaire de super faible monotonie pour les procédures locales. La nécessité de O sera discutée plus loin. Intuitivement, ce résultat n'est pas surprenant. En effet, la condition O implique que la disparition d'éléments

non choisis ne perturbe pas le choix et donc, en particulier, l'idempotence. Ces caractéristiques font des éléments choisis des "sortes d'*ex-æquo*" et il n'est pas alors surprenant d'aboutir à la super faible monotonie.

b) Pour des raisons de "symétrie", on peut s'interroger sur la proposition suivante :
"Si une fonction de choix est locale, fortement monotone et vérifie la condition O elle est super monotone". Je conjecture que ce résultat est faux... mais je ne suis pas parvenu à trouver un contre exemple. On verra cependant plus loin qu'il existe bien des procédures de rangement par choix itéré qui sont monotones et, donc, des procédures de choix super monotones.

6. Exemples, Remarques et Commentaires.

Dans cette section, on s'attachera à commenter les résultats et remarques précédents dans le cas particuliers des **Tournois** (relations complètes et antisymétriques). On pourra ainsi profiter des nombreux résultats disponibles sur ces relations. Toutes les relations utilisées dans cette section seront des tournois.

Pour mémoire notons le :
Lemme 6.1. (cf. Moulin (1986)). Si T est un tournoi, alors :
$x \in UC(X, A, T) \Leftrightarrow [\forall\, y \in A\backslash\{x\},\ x\ T\ y \text{ ou } \exists\, z \in A \text{ tel que } x\ T\ z \text{ et } z\ T\ y]$.

Définition 6.1
Soit X un ensemble fini, A un sous-ensemble non vide de X et T un tournoi sur X. On notera $T_A{}^*$ l'élément de N(A) qui est la fermeture transitive de T/A.

On a, $\forall$ a, b $\in$ A,
$a\ T_A{}^*\ b \Leftrightarrow$
$[a\ T\ b \text{ ou } \exists\, k \in 2^* \text{ et } c_1, c_2, \ldots, c_k \in A \text{ tels que } a\ T\ c_1, c_1\ T\ c_2, \ldots, c_{k-1}\ T\ c_k, c_k\ T\ b]$.

Remarques
a) Par construction $T_A{}^*$ est un préordre complet sur A.

b) On notera TC (Top Cycle) la procédure de choix qui sélectionne dans tout sous ensemble non vide A de X les plus grands éléments de $T_A{}^*$, c'est-à-dire :

$TC(X, A, T) = \{a \in A : \forall b \in A, a\, T_A^* b\}$.

A- Examen du résultat de Patrice

Afin de justifier le fait annoncé que β+ est une condition très forte, notons le résultat suivant.

Lemme 6.2. Il n'existe pas de procédure de choix pour les tournois vérifiant β+ qui soit locale et compatible avec la relation de couverture stricte.

Démonstration. Soit X un ensemble fini, A un sous-ensemble non vide de X et T un tournoi sur X. Supposons que a T b et $b \in C(X, A, T)$. La procédure C étant locale et compatible avec la couverture stricte on a nécessairement $\{a\} = C(X, \{a, b\}, T)$. Le respect de β+ impose alors $a \in C(X, A, T)$ puisque :
$[a \in C(X, \{a, b\}, T), b \in \{a, b\}, \{a, b\} \subseteq A] \Rightarrow [b \in C(X, A, T) \Rightarrow a \in C(X, A, T)]$.
On a donc nécessairement $TC(X, A, T) \subseteq C(X, A, T)$. Le lemme sera alors prouvé si l'on montre que TC amène à sélectionner des éléments strictement couverts, c'est-à-dire si $UC(X, A, T) \subset TC(X, A, T)$. Ce résultat est classique. A titre d'exemple considérons le tournoi T sur $X = \{a, b, c, d\}$ tel que :
a T b, a T c, b T c, c T d, d T a, d T b.
On a de façon évidente :
$TC(X, X, T) = X$ et $UC(X, X, T) = \{a, c, d\}$. ❄

On voit donc qu'utiliser β+ comme condition garantissant la super monotonie amène à considérer des procédures de choix peu attrayantes ! Précisons ce point plus avant en introduisant quelques définitions classiques.

Définition 6.2
Une procédure de choix C est **Condorcet Compatible (CC)** si, pour tout ensemble fini X, tout sous-ensemble non vide $A \subseteq X$ et tout tournoi T sur X :
$[a\, T\, b, \forall b \in A] \Rightarrow C(X, A, T) = \{a\}$.

Définition 6.3
Une procédure de choix C est **Condorcet Transitive (CT)** si, pour tout ensemble fini X, tout sous-ensemble non vide $A \subseteq X$ et tout tournoi T sur X :
$[a\, T\, b, b \in C(X, A, T)] \Rightarrow a \in C(X, A, T)$.

Définition 6.4
Une procédure de choix C est **Smith Compatible (SC)** si, pour tout ensemble fini X, tout ensemble non vide $A \subseteq X$, tout ensemble non vide $B \subseteq A$ et tout tournoi T sur X :
$[a \in B, b \in A \backslash B \Rightarrow a\ T\ b] \Rightarrow C(X, A, T) \subseteq B$.

Remarque. Il est clair que la Smith Compatibilité implique la Condorcet Compatibilité. La Smith Compatibilité semble être une condition très naturelle qui est impliquée par le respect de la couverture stricte.

Lemme 6.3 (Moulin (1986)).
TC est la plus petite procédure de choix locale qui soit Condorcet Transitive.
TC est la plus grande procédure de choix locale qui soit Smith Compatible.
TC vérifie β+ et O.

Remarques.
a) Il est clair que CC est une condition de base pour une procédure de choix.

b) En présence de β+, CC implique CT.
(en effet d'après CC, $a\ T\ b \Rightarrow \{a\} = C(X, \{a, b\}, T)$. Si $b \in C(X, A, T)$, on a, d'après β+, $a \in C(X, A, T)$ puisque :
$[a \in C(X, \{a, b\}, T), b \in \{a, b\}, \{a, b\} \subseteq A] \Rightarrow [b \in C(X, A, T) \Rightarrow a \in C(X, A, T)]$.)

c) La procédure TC est donc incluse dans toutes les procédures vérifiant CC et β+. De plus si l'on considère que SC est une condition fondamentale (ce qui est notre avis), alors imposer β+ ne laisse plus qu'un seul choix : prendre les plus grands éléments de la fermeture transitive !

d) La condition O n'a pas le même inconvénient que β+. On peut en effet vérifier SC et O sans vérifier en même temps CT. On peut alors s'intéresser à des procédures qui sont strictement incluses dans TC.

e) Notons que TC est un premier exemple de procédure de choix super faiblement monotone puisqu'elle vérifie O. Une démonstration directe simple montre que TC est non seulement super faiblement monotone mais également super monotone (cf l'article de Philippe dans Theory and Decision).

B- Exemple de procédure de choix non triviale super faiblement monotone.

Afin de montrer la pertinence de la proposition 5.1, donnons un exemple de procédure de choix locale, monotone vérifiant O et compatible avec la relation de couverture stricte. On trouvera plusieurs de ces procédures dans la littérature consacrée aux tournois. Donnons le plus simple (Minimal Covering Set introduit par Dutta (1988)).

Soit X un ensemble fini et T un tournoi sur X. On dira que $Y \subseteq X$ est un ensemble couvrant X pour T si :
$UC(X, Y, T) = Y$ et, $\forall x \in X \backslash Y, x \notin UC(X, Y \cup \{x\}, T)$.

On montre que la famille des ensembles couvrant X pour T est :
- non vide,
- contient UC^{∞} (définie comme l'itération à l'infini de UC, c'est-à-dire UC(UC(UC(....))))
- contient un élément minimal par inclusion que l'on note MC(X, T).

La procédure de choix définie par :
MCS(X, A, T) = MC(A, T/A),
est (cf. Dutta) :
- locale,
- neutre,
- monotone et
- compatible avec la couverture stricte,
- vérifie O et SC (et donc CC).

Remarques.
a) Il est clair que MCS ne vérifie pas β+ (puisque MCS vérifie SC elle devrait alors inclure TC, ce qui est impossible puisque MCS $\subseteq$ UC), ce qui montre la non nécessité de cette condition pour garantir la super faible monotonie.

b) Ni TC ni MCS ne sont rationalisables, il suffit pour s'en convaincre de considérer un circuit sur 3 éléments. La rationalisation n'est donc pas une condition nécessaire pour garantir la super faible monotonie (on verra plus loin pourquoi).

La condition O est-elle cependant nécessaire ?

C- A propos de la nécessité de la condition O.

Remarques

a) Dans la proposition 5.1, il n'est pas possible de remplacer O par ε+. On sait en effet que UC est locale, monotone et vérifie ε+ tout en violant O (UC n'est pas idempotente), ce qui fournit un contre-exemple puisque UC n'est pas super faiblement monotone comme le montre l'exemple suivant.

X = {a, b, x, y, z, t}. Considérons le tournoi T défini par :

a T b, a T x, a T t

b T z, b T x, b T y, b T t

z T a, z T x, z T y, z T t,

x T t,

y T a, y T x,

t T y.

Il est facile de vérifier UC(X, X, T) = {a, b, z} et que UC(X, {x, y, t}, T) = {x, y, t}. On a donc x $\geq_{UC}$(X, T) y. Considérons à présent le tournoi V identique à T à l'exception de x V z et Non(z V x). On vérifiera sans peine que UC(X, X, V) = {a, b, z, y} et que donc y $>_{UC}$(X, V) x, ce qui montre que UC n'est pas super faiblement monotone.

Si O est une condition suffisante de super faible monotonie, ε+ ne l'est pas. Si O se révèle être non nécessaire, il y a donc assez peu de marge de manœuvre pour l'affaiblir.

b) Si l'on n'impose pas la condition SC, O n'est pas nécessaire pour garantir la super faible monotonie. Il existe en effet des procédures de choix locales monotones et super faiblement monotones violant O et SC. Un exemple en est fourni par la procédure définie par :

C(X, A, T) = {a ∈ A : ∃ b ∈ A tel que a T b},

qui sélectionne tous les non perdants de Condorcet. Cette procédure est clairement locale, monotone et super faiblement monotone (elle est même super monotone !). Elle viole O comme le montre l'exemple suivant :

X = {a, b , c},

a T b, a T c, b T c,

C(X, {a, b}, T) = {a} et C(X, {a, b, c}, T} = {a, b}.

Cette procédure viole CC et donc SC. Elle est particulièrement peu attrayante.

c) Les remarques qui précèdent pourraient amener à penser que pour les procédures de choix locales, monotones et SC, O est une condition nécessaire de super faible monotonie. Il n'en est rien ! Pour le montrer, rappelons le résultat classique suivant.

Lemme 6.4. (Moulin (1986))
$UC(X, A, T) = \{a\} \Leftrightarrow [a\ T\ b, \forall\ b \in A]$. Si $|UC(X, A, T)|$? 1 alors $|UC(X, A, T)| = 3$ et il n'existe pas de gagnant de Condorcet dans UC(X, A, T) pour T/UC(X, A, T).

De ce lemme découle le :

Lemme 6.5. Si $|X| = 4$, UC est super faiblement monotone.

Démonstration. Si $|X| = 3$, la démonstration est triviale. Si $|X| = 4$, en vertu du lemme 7.4, trois cas sont possibles pour ce qui concerne $|UC(X, X, T)|$.
i) $|U_1(\geq_{UC}(X, T)| = |UC(X, X, T)| = 1$. Posons $\{a\} = UC(X, X, T)$, a étant un gagnant de Condorcet. Il est impossible d'améliorer a. Mais si on améliore une action quelconque par rapport à un gagnant de Condorcet, elle devient nécessairement non couverte en vertu du Lemme 7.3. Il ne peut donc pas y avoir de violation de la super faible monotonie.
ii) $|U_1(\geq_{UC}(X, T)| = |UC(X, X, T)| = 3$ et donc $|U_2(\geq_{UC}(X, T)| = 1$. Il ne peut y avoir de violation de la super faible monotonie que si on améliore un élément de $U_1(\geq_{UC}(X, T)$. La fonction UC étant monotone, l'élément amélioré reste alors choisi et la super faible monotonie est nécessairement respectée.
iii) $|U_1(\geq_{UC}(X, T)| = |UC(X, X, T)| = 4$. UC étant monotone, il ne peut y avoir de violation de la super faible monotonie. ❄

Lemme 6.6. Si $|X| = 5$, UC est super faiblement monotone.

Démonstration. Posons $X = \{a, b, c, d, e\}$. En vertu du lemme 7.4, quatre cas sont possibles.
i) $|U_1(\geq_{UC}(X, T)| = |UC(X, X, T)| = 1$. Posons $\{a\} = UC(X, X, T)$, a étant un gagnant de Condorcet. Il est impossible d'améliorer a. En vertu du lemme 7.5, il n'y aura pas de violation de la super faible monotonie si on améliore une action de $X\backslash\{a\}$ par rapport à une autre action de $X\backslash\{a\}$ (en effet dans ce cas a reste toujours choisie seule en tête, et tout se passe comme si $|X| = 4$). Mais si on améliore une action quelconque par rapport à un gagnant de Condorcet, elle devient nécessairement non couverte en vertu du Lemme 7.3. Il ne peut donc pas y avoir de violation de la super faible monotonie.
ii) $|U_1(\geq_{UC}(X, T)| = |UC(X, X, T)| = 3$ et donc $|U_2(\geq_{UC}(X, T)| = 1$ et $|U_3(\geq_{UC}(X, T)| = 1$.
Posons $U_1(\geq_{UC}(X, T) = \{a, b, c\}$, $U_2(\geq_{UC}(X, T) = \{d\}$ et $U_3(\geq_{UC}(X, T) = \{e\}$. On a nécessairement :

a T b, b T c, c T a et d T e.

Aucune amélioration de e ne peut conduire à violer la super faible monotonie. Observons que d ne peut battre qu'au plus une action de {a, b, c} car d $\notin$ UC(X, X, T). Si d bat une action de {a, b, c} toute amélioration de d la rendra choisie en tête et il n'y aura pas de violation possible de la super faible monotonie. Supposons donc que d ne batte aucune des actions de {a, b, c}. Puisque e $\notin$ UC(X, X, T), e ne peut battre qu'au plus une action de {a, b, c}.

Si e ne bat aucune des actions de {a, b, c}. Améliorons d par rapport à a. On obtient alors (abc) > d > e.

Si e bat a. Améliorons d par rapport à a. On obtient alors (abc) > d > e. Améliorons d par rapport à b ou c. On obtient alors (abcd) > e.

Dans tous les cas, on a d > e et il n'y a donc pas de violation possible de super faible monotonie.

iii) $|U_1(\geq_{UC}(X, T)| = |UC(X, X, T)| = 4$ et donc $|U_2(\geq_{UC}(X, T)| = 1$. Il ne peut y avoir de violation de la super faible monotonie que si on améliore un élément de $U_1(\geq_{UC}(X, T)$. La fonction UC étant monotone, l'élément amélioré reste alors choisi et la super faible monotonie est nécessairement respectée.

iv) $|U_1(\geq_{UC}(X, T)| = |UC(X, X, T)| = 5$. UC étant monotone, il ne peut y avoir de violation de la super faible monotonie. ❈

Remarque. Dès lors que |X| = 5, la procédure UC est super faiblement monotone (le contre exemple donné au a) est donc de taille minimale). Mais UC peut violer O avec |X| = 5.

Exemple. X = {a, b, c, d, e}.

a T b, a T d,

b T c, b T e,

c T a, c T d, c T e,

d T b, d T e,

e T a.

On a UC(X, X, T) = {a, b, c, d} (e est couvert par c) et UC(X, {a, b, c, d}, T) = {a, b, c} (d est couvert par a).

La condition O n'est donc pas nécessaire pour garantir la super faible monotonie même en imposant la condition SC (qui est, bien sûr, vérifiée par UC).

7. A propos de la super monotonie.

On peut imaginer de nombreuses conditions garantissant la super monotonie d'une procédure de choix. Par exemple, on a :

Si une fonction de choix est locale, fortement monotone et vérifie PI elle est super faiblement monotone.

Démonstration. Puisque PI implique O, on sait d'après la proposition 6.1 que C est super faiblement monotone. Supposons que C ne soit pas super monotone. Alors il existe un ensemble X, une relation $R \in F(X)$, deux éléments $a, b \in X$ tels que :
$[a >_C(X, R)\ b$ et $b \geq_C(X, R')\ a]$,

où R' améliore a.
La procédure C étant fortement monotone, on sait que $a \notin U_1(\geq_C(X, R))$ (sinon la monotonie impliquerait $a \in U_1(\geq_C(X, R'))$ et donc $b \in U_1(\geq_C(X, R')$, ce qui violerait la forte monotonie de C). Notons $K \subset X$, le sous-ensemble conduisant à $a >_C(X, R)\ b$. On a :

$a, b \in K$, $a \in C(X, K, R)$ et $b \notin C(X, K, R)$.
Supposons que $a \in U_1(\geq_C(X, R'))$ et donc $b \in U_1(\geq_C(X, R')$. Puisque PI implique α, on a :

$a \in C(X, K, R')$ et $b \in C(X, K, R')$.
On a donc :
$a \in C(X, K, R)$, $b \notin C(X, K, R)$, $a \in C(X, K, R')$ et $b \in C(X, K, R')$, ce qui viole la monotonie forte de C.
On a donc nécessairement $a \notin U_1(\geq_C(X, R))$ et $a \notin U_1(\geq_C(X, R'))$.
D'après le raisonnement fait à la proposition 6.1, on sait alors que $U_1(\geq_C(X, R)) = U_1(\geq_C(X, R'))$. La procédure C étant locale, il est facile de constater que l'on peut répéter le raisonnement fait sur X en considérant $X[R, 2] = X[R', 2]$. On a alors :
$U_2(\geq_C(X, R)) = U_2(\geq_C(X, R'))$, $a \notin U_2(\geq_C(X, R))$ et $a \notin U_2(\geq_C(X, R'))$.
L'itération du raisonnement conduit alors à une contradiction puisque $a \notin U_r(\geq_C(X, R))$ pour tout r. ❈

Remarques
a) On sait que PI n'implique pas qu'une fonction de choix soit rationalisable. Il est alors tentant de croire à l'intérêt du résultat précédant. Il n'en est rien comme le montre la remarque suivante.

b) Montrons que PI est une condition suffisante particulièrement insatisfaisante. Plaçons-nous dans le cas particulier des tournois. On a :

Lemme 7.1 (Moulin (1986)). Il n'existe pas de procédure de choix pour les tournois qui soit locale et vérifient CC et α.

Démonstration. Soit T un tournoi sur X tel que a T b, b T c et c T a. Posons A = {a, b, c}. La procédure C étant locale, C(X, K, T) ne dépend que de T/K. La condition CC implique alors : {a} = C(X, {a, b}, T), {b} = C(X, {b, c}, T) et {c} = C(X, {c, a}, T). La condition α implique alors a $\notin$ C(X, A, T), b $\notin$ C(X, A, T) et c $\notin$ C(X, A, T), une contradiction. ❈

Ce lemme montre que si l'on tient au caractère local de la procédure et si l'on souhaite vérifier CC, on ne peut en aucun cas imposer à une procédure de choix de vérifier α. Ceci remet non seulement en cause l'intérêt du résultat précédent mais également de toute tentative de voir dans la "rationalisation" une condition suffisante de super monotonie puisque α est une condition nécessaire de rationalisation.

8. Conclusions et Problèmes Ouverts.

La proposition 5.1 permet de trouver de nombreuses procédures de choix super faiblement monotone. Il existe néanmoins de nombreux problèmes ouverts ! Mentionnons-en quelque uns.

a) Peut-on améliorer le résultat concernant la super faible monotonie en considérant une condition moins forte que O ? (On a vu que $\varepsilon+$ n'était pas suffisante – il y a donc assez peu de marge).

b) Peut-on trouver des conditions garantissant la super monotonie d'une procédure de choix locale et fortement monotone ?

c) Peut-on espérer trouver des conditions nécessaires et suffisantes de super faible monotonie (ou de super monotonie) pour les fonctions de choix locales ? Ceci semble cependant difficile car il faudrait s'assurer que le viol de la condition nécessaire va se produire sur un des ensembles appartenant à l'itération descendante de C.

[27 janvier 1995 - Ver 4.3]