Donald E. Knuth

Mariages stables

LES PRESSES DE L'UNIVERSITÉ DE MONTRÉAL

Mariages stables et leurs relations avec d'autres problèmes combinatoires

Introduction
à l'analyse mathématique
des algorithmes

COLLECTION DE LA CHAIRE AISENSTADT

Fondateur : Lucien Le Cam, Université de Berkeley

Directeur : Anatole Joffe, directeur du Centre de recherches mathématiques de l'Université de Montréal

Dans la même collection

Robert Hermann, *Physical Aspects of Lie Group Theory*, 1974

Mark Kac, *Quelques problèmes mathématiques en physique statistique*, 1974

S. R. de Groot, *la Transformation de Weyl et la fonction de Wigner: une forme alternative de la mécanique quantique*, 1975

Jacques Louis Lions, *Sur quelques questions d'analyse, de mécanique et de contrôle optimal*, 1976

Cette collection est consacrée à la publication des conférences données, depuis 1970, au Centre de recherches mathématiques de l'Université de Montréal, dans le cadre de la Chaire Aisenstadt. C'est grâce à la générosité de Monsieur André Aisenstadt, docteur en physique théorique de l'Université de Zurich, que le Centre de recherches mathématiques peut inviter des chercheurs prestigieux et publier, aux Presses de l'Université de Montréal, le texte de leur conférence.

Mariages stables et leurs relations avec d'autres problèmes combinatoires

Introduction à l'analyse mathématique des algorithmes

Donald E. Knuth

Édition revue et corrigée

1976
Les Presses de l'Université de Montréal
C.P. 6128, succ. «A», Montréal, Qué., Canada H3C 3J7

ISBN 2-7606-0529-9

Dépôt légal, 3e trimestre 1976 — Bibliothèque nationale du Québec

AVANT-PROPOS

Ces dix dernières années ont vu le développement rapide d'un domaine qui a reçu le nom d'Informatique. A mon avis, il conviendrait d'appeler "Algorithmique" cette discipline dont l'objet principal n'est pas l'étude de l'information elle-même, mais celle des processus de traitement de l'information, ou algorithmes. Quoi qu'il en soit, sous un nom ou un autre, ce domaine s'est avéré passionant à plus d'un titre.

Le but de cet ouvrage est d'introduire le lecteur à l'Analyse des Algorithmes, en se basant sur des exemples plutôt qu'en faisant le tour des principaux résultats théoriques. Cette forme de présentation saura, je l'espère, lui donner une idée des méthodes utilisées dans ce domaine, et illustrer les rapports étroits qu'il entretient avec de nombreuses autres branches des mathématiques. Le problème des mariages stables semble idéal dans ce contexte, car il ne fait appel à aucune connaissance préalable en Algorithmique, et conduit naturellement à illustrer les techniques essentielles de l'Analyse des Algorithmes. Ce problème me permet aussi de montrer combien l'Analyse des Algorithmes peut être intéressante en elle-même, en plus de son importance bien connue sur le plan pratique.

Le niveau du discours dans cet ouvrage est élémentaire et n'exige aucune connaissance préalable en analyse algorithmique ou en mariage. Mon but est d'informer le lecteur plutôt que de l'impressionner.

Je tiens à exprimer toute ma reconnaissance à Monsieur André Aisenstadt, dont la générosité m'a permis d'aller à l'Université de Montréal donner la série de conférences dont ce livre est tiré. Je veux aussi remercier Monsieur le professeur Anatole Joffe, pour l'honneur qu'il m'a fait en m'invitant, et pour son hospitalité qui a rendu le séjour si agréable pour ma famille comme pour moi-même.

Je regrette de ne pas encore pouvoir maîtriser la langue française. Cependant cette lacune m'a valu le grand plaisir de travailler en liaison étroite avec Pascale Rousseau et François Trochu, qui ont si soigneusement traduit le texte des conférences que j'ai données en anglais. Je voudrais également remercier un de mes étudiants, Bernard Mont-Reynaud, pour son aide dans la préparation du texte final, et Madame Micheline Marano et Mademoiselle Johanne Marcoux pour leur excellente typographie.

Donald E. Knuth

Stanford, Californie
Mai 1976

N.B. A l'occasion du second tirage de ce livre, j'ai corrigé quelques erreurs typographiques qui s'étaient glissées dans le premier tirage. Toutefois, j'attends toujours des solutions aux problèmes qui sont posés dans ce livre!

D.E.K., octobre 1980

... pour avoir appris,
la cuisine,
qui retient les petits maris,
qui s'débinent.

PREMIÈRE CONFÉRENCE

INTRODUCTION, DÉFINITIONS ET EXEMPLES

Soient H et F deux ensembles finis de n éléments; H est l'*ensemble des hommes* $A_1, A_2, \ldots, A_n$ et F l'*ensemble des femmes* $a_1, a_2, \ldots, a_n$. Un *couplage* est une bijection de H sur F, c'est-à-dire un ensemble de n mariages monogames entre les hommes et les femmes. Dans toute la suite, les deux ensembles H et F jouent un rôle symétrique, et il ne faut attacher aucune signification spéciale au fait que les lettres minuscules désignent les femmes.

Supposons que chaque homme ait un ordre de préférence pour les femmes et chaque femme un ordre de préférence pour les hommes. La *liste* des préférences de chacun est donc constituée par deux tableaux de n^2 éléments.

Un couplage est *instable* si un homme A et une femme a, non mariés entre eux, se préfèrent mutuellement à leurs conjoints. Cette liaison dangereuse se produit lorsque:

A est marié avec b;
a est mariée avec B;
A préfère a à b;
a préfère A à B.

(Les opinions de b et B n'importent pas ici.) Un couplage est *stable* si cette situation ne se produit pas.

EXEMPLE 1. Mariage de 4 hommes (A,B,C,D) avec 4 femmes (a,b,c,d).

Hommes	Ordres de préférence
Anatole	c b d a
Barnabé	b a c d
Camille	b d a c
Dominique	c a d b

Femmes	Ordres de préférence
antoinette	A B D C
brigitte	C A D B
cunégonde	C B D A
donatienne	B A C D

Notons que ces préférences ne changent pas avec le temps.

Le couplage ($\underline{A}a,B\underline{b},Cc,Dd$) est instable car A et b se préfèrent mutuellement (ce que l'on indique en soulignant A et b). Faisons divorcer Aa et Bb; remarions A avec b; alors les deux autres, a et B, n'ont plus qu'à se marier entre eux. On obtient ainsi le couplage ($A\underline{b},Ba,\underline{C}c,Dd$) qui lui aussi est instable à cause de b et C. Procédant ainsi jusqu'à l'obtention d'un couplage stable, on obtient la séquence

$\underline{A}a$	$B\underline{b}$	Cc	Dd	instable
$A\underline{b}$	Ba	$\underline{C}c$	Dd	instable
$A\underline{c}$	Ba	Cb	$\underline{D}d$	instable
Ad	Ba	Cb	Dc	♡ stable ♡

Pour vérifier qu'on a bien un couplage stable, il suffit de dresser la liste des épouses que chaque homme préférerait à

la sienne:

A : préférerait c, puis b
B : préférerait b
C : a déjà son meilleur choix
D : a déjà son meilleur choix

et la liste des époux que chaque femme préférerait au sien:

a : préférerait A
b : a déjà son meilleur choix
c : préférerait C, puis B
d : préférerait B.

On ne peut améliorer le choix de A, car ni c, ni b ne préfèrent A à leurs conjoints. On ne peut non plus améliorer le choix de B puisque b a obtenu son meilleur partenaire. Le couplage considéré est donc stable.

On arrive au même résultat en se plaçant du point de vue des femmes, c'est-à-dire en essayant d'améliorer les choix de a, c, ou d. On constate que les hommes qu'elles préféreraient ne sont pas, eux, d'accord. (C'est la vie.)

Le procédé qui consiste à choisir des divorces successifs, de la façon indiquée plus haut, ne conduit pas nécessairement à un couplage stable. Regardons en effet l'exemple suivant:

EXEMPLE 2.

Choix des hommes	Choix des femmes
A : b a c	a : A C B
B : liste arbitraire	b : C A B
C : a b c	c : liste arbitraire

Aa Bb Cc

Ab Ba Cc

Ac Ba Cb

Ac Bb Ca

Aa Bb Cc

La séquence des divorces et remariages mène à un cycle au lieu d'aboutir à une solution stable. Cependant de telles solutions existent: (Ab,Bc,Ca) et (Ac,Bc,Cb).

EXEMPLE 3. (Choix en permutation circulaire)

Choix des hommes	Choix des femmes
A : a b c d e	a : B C D E A
B : b c d e a	b : C D E A B
C : c d e a b	c : D E A B C
D : d e a b c	d : E A B C D
E : e a b c d	e : A B C D E

Les cinq couplages stables sont

Aa	Bb	Cc	Dd	Ee
Ab	Bc	Cd	De	Ea
Ac	Bd	Ce	Da	Eb
Ad	Be	Ca	Db	Ec
Ae	Ba	Cb	Dc	Ed

Dans la première solution stable, chaque homme est marié avec son meilleur choix, ce qui correspond au moins bon choix des femmes. La deuxième solution stable améliore le sort de chaque femme d'un rang tandis que les hommes obtiennent leur deuxième choix. Ainsi de suite jusqu'à la dernière solution

qui correspond au meilleur choix des femmes et au moins bon choix des hommes. Ce conflit d'intérêt soulève le problème de la définition d'un critère pour le choix d'une solution stable. On peut se placer du point de vue des femmes ou bien du point de vue des hommes.

Montrons qu'il existe seulement cinq solutions stables. A doit être marié à a, b, c, d ou e. Supposons qu'une solution stable contienne Aa, alors elle contient Ee. En effet, si E était marié avec b, c ou d, alors on voit sur les tableaux que

E préfère a,

a préfère E;

le couplage serait instable. On écrit donc

$$Aa \rightarrow Ee.$$

Procédant ainsi, on obtient:

$$Aa \Rightarrow Ee \rightarrow Dd \rightarrow Cc \Rightarrow Bb \rightarrow Aa.$$

Pour une solution stable contenant Ab alors on doit avoir Ee ou Ea; mais l'on vient de voir que $Ee \rightarrow Aa$. On obtient donc la suite d'implications

$$Ab \rightarrow Ea \Rightarrow De \rightarrow Cd \rightarrow Bc \rightarrow Ab.$$

De même si Ac apparaît dans la solution stable, on a

$$Ac \Rightarrow Eb \rightarrow \text{etc.}$$

Ceci est un exemple où il y a exactement n solutions. On peut se demander si ces n solutions existent toujours. L'existence d'au moins une solution stable sera établie plus loin; il existe dans la littérature un exemple où pour $n = 8$, on a 9 solutions stables.

Si on est tenté de croire qu'il existe au plus $2n$ solutions stables, ou même n^2, l'exemple suivant montre qu'il peut en exister beaucoup plus.

EXEMPLE 4. (n pair)

Choix des hommes	Choix des femmes
1 : 1 2 ...	1 : 1
2 : 2 1 ...	2 : 2
3 : 3 4 ...	3 : 3
4 : 4 3 ...	4 : 4
n-1 : n-1 n ...	n-1 : n-1
n : n n-1...	n : ..... n

Supposons que chaque paire d'hommes (1 et 2, 3 et 4,...) se marie à son premier choix ou à son second choix. Les cercles ci-dessus illustrent un tel schéma de mariage, le numéro encerclé étant celui du conjoint. On peut construire $2^{n/2}$ schémas différents.

Chaque couplage ainsi obtenu est stable. En effet quand deux hommes sont mariés à leur deuxième choix, ils ne peuvent obtenir leur meilleur choix, car il correspond au dernier choix des femmes.

On voit ici que la recherche systématique des solutions stables peut conduire à envisager un nombre exponentiel de cas possibles, ce qui, pour n assez grand, est trop long, même pour un ordinateur.

Généralisation au problème de l'admission de n étudiants dans m universités

La k-ième université admet n_k étudiants; sans nuire à la

généralité, on a l'égalité $n_1+n_2+\ldots+n_m = n$. (Si le nombre d'étudiants excède les possibilités d'admission dans les universités, on crée une université fictive, dernier choix de tous les étudiants. Si le nombre d'étudiants est inférieur au nombre de places dans les universités, on crée des étudiants fictifs, derniers choix pour toutes les universités. On est ramené ainsi au cas de l'égalité $n_1+n_2+\ldots+n_m = n$.)

Chaque étudiant a un ordre de préférence pour les universités et chaque université un ordre de préférence pour les étudiants. On parle d'*admission stable* quand il n'y a pas d'étudiant A, non admis à l'université a, tel que A préfère a à son université et a préfère A à un de ses étudiants.

Ce problème peut se ramener à celui du couplage stable en remplaçant la k-ième université par n_k "places", chaque place ayant le même ordre de préférence pour les étudiants.

Dans la pratique (par exemple pour la répartition des étudiants dans les universités iranaises et celle des internes dans les hôpitaux américains), ce problème est formulé, et résolu à l'aide d'un ordinateur, en utilisant un algorithme pour déterminer un couplage stable.

Généralisation: listes incomplètes

Considérons le problème des couplages stables quand les hommes n'ont pas nécessairement classé toutes les femmes et quand les femmes n'ont pas nécessairement classé tous les hommes.

On cherche des solutions stables avec la condition supplémentaire que chaque personne soit mariée avec une personne figurant sur sa liste de préférence (on peut imaginer qu'une personne préfère mourir plutôt que d'épouser quelqu'un qui n'est pas sur sa liste).

Considérons par exemple les listes incomplètes

$$
\begin{array}{llll}
A : a & & & a : C\ A\ B \\
B : a\ a\ b & & & b : B\ A\ C \\
C : c\ a & & & c : A\ B\ C
\end{array}
$$

Le seul couplage possible est $(Aa, \underline{B}\underline{b}, C\underline{c})$, mais il est instable à cause de B et c.

Le théorème d'existence des couplages stables pour les listes complètes ne se généralise donc pas au cas des listes incomplètes.

Notations

$$
\begin{array}{rcl}
a \in A & \longleftrightarrow & a \text{ est sur la liste de préférence de } A \\
A \in a & \longleftrightarrow & A \text{ est sur la liste de préférence de } a \\
aAb & \longleftrightarrow & A \text{ préfère } a \text{ à } b \text{ ou } b \notin A \\
AaB & \longleftrightarrow & a \text{ préfère } A \text{ à } B \text{ ou } B \notin a
\end{array}
$$

Alors $(A_1a_1, A_2a_2, \ldots, A_na_n)$ est un couplage stable si et seulement si:

(i) $a_k \in A_k$ et $A_k \in a_k$ pour $1 \le k \le n$;

(ii) Il n'y a pas de j et k tels que $A_ja_kA_k$ et $a_kA_ja_j$.

Une approche pour résoudre un problème de couplage d'ordre n consisterait à le réduire à un problème d'ordre n-1. On peut démontrer facilement qu'un couplage stable contenant A_na_n existe si et seulement si $a_n \in A_n$, $A_n \in a_n$, et s'il existe un couplage stable pour le système B d'ordre n-1 défini par les propriétés suivantes:

$$
\begin{array}{rcl}
b_i \in B_j & \longleftrightarrow & a_i \in A_j \text{ et non } (A_ja_nA_n \text{ et } a_nA_ja_i) \\
B_i \in b_j & \longleftrightarrow & A_i \in a_j \text{ et non } (a_jA_na_n \text{ et } A_na_jA_i) \\
b_iB_jb_k & \longleftrightarrow & a_iA_ja_k \text{ ou } b_k \notin B_j \\
B_ib_jB_k & \longleftrightarrow & A_ia_jA_k \text{ ou } B_k \notin b_j
\end{array}
$$

pour $1 \le i,j,k \le n$-1.

Conversion de listes incomplètes en listes complètes

On peut fabriquer une liste complète à partir d'une liste incomplète en ajoutant un nouvel homme V, le *veuf*, et une nouvelle femme v, la *veuve*. Le veuf V sera la dernière préférence de v, et v la dernière préférence de V. La femme a_k classera V le dernier sur sa liste de préférence (peut-être incomplète), puis classera après V, dans un ordre indifférent, tous les hommes qui n'étaient pas sur sa liste.

L'homme A_k classera v après sa liste de préférence (peut-être incomplète), puis classera après v, dans un ordre indifférent, toutes les femmes qui n'étaient pas sur sa liste.

Une application de la remarque précédente donne le résultat suivant:

THEOREME. Il existe un couplage stable tel que V est marié à v pour le système complet si et seulement si il existe un couplage stable pour le système incomplet.

A partir de résultats qui seront donnés dans la conférence suivante, on pourra même montrer:

THEOREME. S'il existe un couplage stable avec V marié à v pour le système complet, alors pour tous les couplages stables de ce système, V est marié avec v.

Par conséquent, il suffit pour décider de l'existence d'un couplage stable dans le système incomplet d'obtenir une seule solution stable au système complet. Si V n'est pas marié à v alors il n'existe pas de solution stable au système incomplet; si V est marié à v, alors la solution stable sans Vv est une solution stable du système incomplet.

EXERCICE 1.1. Etant donné l'ensemble d'hommes $A_1, A_2, \ldots, A_n$, soit a_k la femme qui correspond au meilleur choix "réaliste" pour l'homme A_k, c'est-à-dire celle qu'il préfère parmi les partenaires avec lesquelles il réalise un couplage stable. Toutes les listes de préférence sont complètes. (On utilise ici l'existence, non encore établie, d'au moins un couplage stable.)

(a) Prouver que $j \neq k$ implique $a_j \neq a_k$.

(b) Prouver que $(A_1 a_1, \ldots, A_n a_n)$ est un couplage stable.

EXERCICE 1.2. Trouver tous les couplages stables pour les choix

A :	e	d	c	b	a		a :	A	B	C	D	E
B :	a	e	d	c	b		b :	B	C	D	E	A
C :	b	a	e	d	c		c :	C	D	E	A	B
D :	c	b	a	e	d		d :	D	E	A	B	C
E :	d	c	b	a	e		e :	E	A	B	C	D

EXERCICE 1.3. (J.S. Hwang). On suppose que la matrice de préférence des hommes est un carré latin (chaque colonne contient chacune des femmes). Montrer que les couplages définis par les colonnes de ce carré latin sont tous stables, si et seulement si la matrice de préférence des femmes est le carré latin dual (défini par la condition: a est le j-ième choix de A si et seulement si A est le $(n+1-j)$-ième choix de a, pour tout a et A).

DEUXIÈME CONFÉRENCE

EXISTENCE D'UN COUPLAGE STABLE; ALGORITHME FONDAMENTAL

Un algorithme fondamental permettant de construire un couplage stable est développé; cet algorithme constitue en lui-même une preuve par construction de l'existence d'au moins une solution stable. On a vu dans la première conférence qu'une séquence de divorces au hasard n'aboutit pas toujours à un couplage stable. Dans l'algorithme fondamental, les divorces sont remplacés par de nombreuses fiançailles. Les hommes jouent tour à tour le rôle du prétendant, font des avances aux femmes qui acceptent ou refusent selon leur préférence. On verra que l'on obtient toujours une solution stable.

L'algorithme utilise trois variables k, X, x et deux constantes n, Ω.

n : nombre d'hommes = nombre de femmes

k : nombre de couples (à l'essai) déjà constitués

X : prétendant

x : femme à qui le prétendant fait une avance

Ω : homme imaginaire (très indésirable)

Description de l'algorithme

$k \leftarrow 0$; fiancer toutes les femmes (temporairement) avec Ω;
tant que $k < n$ faire
début $X \leftarrow (k+1)$-ième homme;
  tant que $X \neq \Omega$ faire
  début $x \leftarrow$ meilleur choix restant sur la liste de X;
    si x préfère X à son fiancé
    alors début fiancer X et x;
          $X \leftarrow$ fiancé précédent de x
      fin;
    si $X \neq \Omega$ alors retirer x de la liste de X
  fin;
  $k \leftarrow k+1$
fin; célébrer n mariages.

Pour ceux qui ne sont pas familiers avec un language du type algol, remarquons que la notation $k \leftarrow 0$ signifie que la variable k prend la valeur 0; et qu'une série d'instructions entre le mot début et le mot fin correspondant constitue une seule séquence. Cette séquence est répétée, quand elle figure dans le programme après l'instruction tant que (...) faire, aussi longtemps que la condition (...) est satisfaite.

EXEMPLE. Pour comprendre cet algorithme, faisons-le fonctionner sur l'exemple 1.

A :	a	b	b	a		a :	A	B	D	C	Ω
B :	b	a	c	d		b :	C	A	D	B	Ω
C :	b	d	a	c		c :	C	B	D	A	Ω
D :	c	a	d	b		d :	B	A	C	D	Ω

Remarquons que nous avons ajouté Ω à la fin de la liste de préférence de chaque femme.

Au début de l'exécution de l'algorithme, k prend la valeur zéro et chaque femme est fiancée à Ω.

Premier cycle

Après avoir vérifié que $k < n$, la variable X prend la valeur A (le premier homme). Puis, on vérifie que $X \neq \Omega$ et on exécute ensuite l'instruction:

$x \leftarrow$ meilleur choix restant pour X.

C'est-à-dire

$x \leftarrow a$.

A cette étape, nous indiquerons les valeurs que prennent les trois variables k, X et x de la façon suivante:

$$k = 0$$
$$X = A$$
$$x = a$$

Puisque a préfère A à Ω, on fiance A et a. La valeur de X devient donc Ω, le fiancé précédent de a. Comme $X = \Omega$, on n'exécute pas la commande

retirer x de la liste de X.

On passe alors à l'instruction

tant que $X \neq \Omega$ faire.

Comme $X = \Omega$, on sort de la séquence des "avances" qui commence à cette instruction. L'indice k est alors augmenté ($k \leftarrow k+1$).

Deuxième cycle

Puisque $k < n$, on exécute alors l'instruction

$$X \leftarrow (k+1)\text{-ième homme}$$

et X prend la valeur B; B devient le nouveau prétendant. On vérifie que $X \neq \Omega$ et on exécute l'instruction:

$$x \leftarrow \text{meilleur choix restant pour } X.$$

C'est-à-dire

$$x \leftarrow b.$$

On se trouve dans la situation:

$$k = 1$$
$$X = B$$
$$x = b.$$

La proposition de B, étant la première véritable avance faite à b, le couple Bb est formé; X devient Ω, et on augmente k à nouveau.

<u>Troisième cycle</u>

Ce cycle est différent des deux précédents. On vérifie que $k = 2 < n$, alors $X \leftarrow C$. Le meilleur choix restant sur la liste de C est b, alors $x \leftarrow b$ et

$$k = 2$$
$$X = C$$
$$x = b.$$

Mais b préfère C à son ami B. On fiance donc Cb; X devient l'ancien ami, B. Après avoir vérifié $X \neq \Omega$, on retire x de la liste de X (c'est-à-dire b de la liste de B).

X ayant la valeur B, l'instruction

<u>tant que</u> $X \neq \Omega$ <u>faire</u>

entraîne une nouvelle fois l'exécution de la séquence des "avances" avec B comme nouveau prétendant. Le meilleur choix restant pour B est a, alors $x \leftarrow a$ et:

$$k = 2$$
$$X = B$$
$$x = a.$$

La femme a préfère B à Ω, on accouple donc Ba; X devient l'ancien ami Ω. On sort alors de la séquence des "avances" qui commence à l'instruction <u>tant que</u> $X \neq \Omega$ <u>faire</u>. On augmente k à nouveau.

<u>Quatrième cycle</u>

On vérifie que $k = 3 < n$, alors $X \leftarrow D$ et la nouvelle situation est

$$k = 3$$
$$X = D$$
$$x = a.$$

La femme a préfère D à son partenaire actuel A. Elle change de partenaire; A devient le nouveau prétendant X. Puis a est retiré de la liste de A. Le meilleur choix restant sur la liste de A est b, donc

$$k = 3$$
$$X = A$$
$$x = b.$$

Mais b est déjà avec C et ne lui préfère pas A. Il n'y a pas de changement de partenaire et b est retiré de la liste de A, qui fait donc de nouvelles avances au meilleur choix restant sur sa liste:

$$k = 3$$
$$X = A$$
$$x = d.$$

La femme d accepte l'avance de A, puisqu'elle préfère A à l'indésirable Ω. Alors $X \leftarrow \Omega$ et on sort de la séquence des "avances". On augmente k.

Terminaison

On a atteint $k = n = 4$; on termine donc l'itération

tant que $k < n$ faire.

On peut maintenant célébrer quatre mariages. Le couplage obtenu est stable, comme nous allons le montrer.

REMARQUE. On peut se demander pourquoi l'algorithme comporte deux tests $X \neq \Omega$ dans les instructions

tant que $X \neq \Omega$ faire..., et
si $X \neq \Omega$ alors... .

Mais on évite ainsi d'utiliser une instruction aller à.

Preuve de l'algorithme

Ce n'est que récemment (depuis cinq à dix ans) que l'on a commencé à démontrer rigoureusement les résultats qui concernent les algorithmes. D'habitude, l'intuition produit un algorithme, dont le fonctionnement est vérifié expérimentalement avec l'ordinateur. Pour prouver la validité d'un algorithme et évaluer ses performances avec rigueur, il est nécessaire de préciser la signification des variables de l'algorithme; de faire apparaître leurs relations à chaque étape de l'algorithme; enfin de raisonner par récurrence sur l'algorithme.

Remarquons d'abord les quelques résultats suivants, valables pendant l'exécution de l'algorithme.

POINT 1. Si une femme a est retirée de la liste de A, aucun couplage stable ne peut contenir Aa.

DEMONSTRATION. Quand l'algorithme effectue l'opération "retirer x de la liste de X" et que $x = a$, $X = A$, deux situations ont pu se présenter:

(i) L'homme A a fait une avance à a, mais cette dernière lui préfère son ami B.

(ii) La femme a avait A comme ancien ami, mais elle venait de le quitter (après une avance plus intéressante de la part de B).

Dans les deux cas, a préfère B à A et B préfère a à toutes les autres femmes restant sur sa liste.

Si un couplage stable contenait Aa, alors B aurait dû être marié à quelqu'un qu'il préfère à a. Mais toutes les partenaires préférables à a ne figurent plus sur la liste de B. On arrive donc à une contradiction par récurrence sur l'algorithme (admettant que le point 1 soit vrai à chacune des étapes précédant celle considérée).

POINT 2. Si A préfère a à sa fiancée, c'est que a l'a rejeté pour un autre.

POINT 3. Deux femmes ne peuvent être les petites amies du même homme (sauf si cet homme est Ω).

POINT 4. La situation d'une femme n'empire jamais au cours de l'algorithme.

POINT 5. La liste de préférence de chaque homme ne devient jamais vide.

DEMONSTRATION. Autrement cet homme aurait été rejeté par toutes les femmes à cause du point 2. Alors chaque femme aurait un petit ami qui n'est pas Ω, à cause des points 4 et 3. Il existerait alors au moins n autres hommes, en plus de A. Contradiction.

A cause du point 5, l'algorithme est *bien défini*. En effet toutes les autres opérations sont d'une façon évidente bien définies; seul le point (5) nécessitait une démonstration.

L'algorithme *se termine* en un nombre fini d'étapes car après chaque proposition, une des listes de préférence devient plus courte ou bien k augmente d'une unité.

POINT 6. *Le couplage obtenu est stable.*

DEMONSTRATION. Si A n'est pas marié avec a, et si A préfère a à son épouse, c'est que a l'a rejeté (point 2) et est mariée avec quelqu'un qu'elle lui préfère (point 4).

En fait, l'algorithme donne le *résultat optimal pour chaque homme*. Chaque homme est marié de la meilleure façon possible: il n'y a pas de couplage stable dans lequel il aurait une épouse qu'il préfèrerait à celle qu'il a déjà. (C'est une conséquence du point 1. Remarquez qu'on n'a pas utilisé cet argument pour prouver la validité de l'algorithme, mais seulement pour cette propriété plus forte.)

Puisque la solution est caractérisée comme étant optimale pour les hommes, elle est indépendante de la numérotation de ceux-ci, même si l'algorithme leur fait jouer tour à tour le rôle de prétendant dans un ordre particulier.

En outre, le résultat constitue aussi la *pire solution pour les femmes*. (Dans tout couplage stable, chaque femme obtient un choix égal ou supérieur à celui que lui attribue l'algorithme.)

DEMONSTRATION. Supposons Aa mariés par l'algorithme, mais Ba et Ab mariés dans une autre solution stable où a préfère A à B; alors A doit préférer b à a, ce qui contredit le fait que Aa est la meilleure solution pour A.

Le même algorithme peut être utilisé pour obtenir la meilleure solution pour les femmes (et la pire pour les hommes). Les femmes doivent alors faire les avances.

Conflit d'intérêt

Le fait que "le meilleur pour les hommes est le pire pour les femmes" est un cas particulier d'un résultat beaucoup plus général.

THEOREME. Si un couplage stable contient le couple Aa, et un autre contient Ab et Ba, alors

$$bAa \quad \text{et} \quad AaB,$$

ou bien

$$aAb \quad \text{et} \quad BaA.$$

(En d'autres termes, tout autre couplage stable est meilleur pour un des conjoints et moins bon pour l'autre.)

DEMONSTRATION. Par définition de la stabilité, la situation de A et a ne peut empirer dans la seconde solution. Il reste donc à montrer qu'elle ne peut s'améliorer pour les deux en même temps.

Soit $A = X_0$, $a = x_0$, $b = x_1$ et supposons que A préfère b à a, c'est-à-dire $x_1 X_0 x_0$. Puisque le premier couplage stable correspond au pire choix pour X_0, c'est que x_1 a obtenu un meilleur choix.

Soit X_1 le conjoint de x_1 dans ce premier couplage stable; donc $X_1 x_1 X_0$. Puisque le second couplage correspond au pire partenaire pour x_1, c'est que X_1 a obtenu un meilleur choix. Soit x_2 le conjoint de X_1 dans ce second couplage stable; donc $x_2 X_1 x_1$, etc.

On obtient la suite

$$X_0 x_0 \quad X_1 x_1 \quad X_2 x_2 \quad \ldots \quad \text{dans le premier couplage stable,}$$

$$X_0 x_1 \quad X_1 x_2 \quad X_2 x_3 \quad \ldots \quad \text{dans le second,}$$

où

$$x_{k+1} X_k x_k \quad \text{et} \quad X_{k+1} x_{k+1} X_k \quad \text{pour tout} \quad k \geq 0.$$

Puisqu'il y a un nombre fini de personnes, il existe des entiers j et k, $j < k$ tels que $X_j = X_k$. Soit j le plus petit entier ayant cette propriété, et, pour ce j, soit k le plus petit des entiers tel que $X_j = X_k$ et $k > j$. On a $x_j = x_k$. De plus, $j = 0$ sinon $X_{k-1} x_k = X_{k-1} x_j$ apparaîtrait dans le second couplage de même que $X_{j-1} x_j$. (D'où $X_{j-1} = X_{k-1}$, ce qui contredit le fait que j est le plus petit des entiers j tel que $X_j = X_k$.) Donc $X_{k-1} x_0$ apparaît dans le second couplage. Mais $x_0 = a$, donc $X_{k-1} = B$. Etant donné que $X_k x_k X_{k-1}$, on a prouvé AaB.

<u>COROLLAIRE</u>. Si un couplage stable est au moins aussi bon qu'un autre du point de vue de chacun des hommes, le second est au moins aussi bon que le premier du point de vue de chacune des femmes.

<u>Démonstration du théorème énoncé pendant la première conférence</u>

S'il existe un couplage stable contenant Aa dans lequel a est le dernier choix de A et A le dernier choix de a, alors *tous* les couplages stables contiennent Aa.

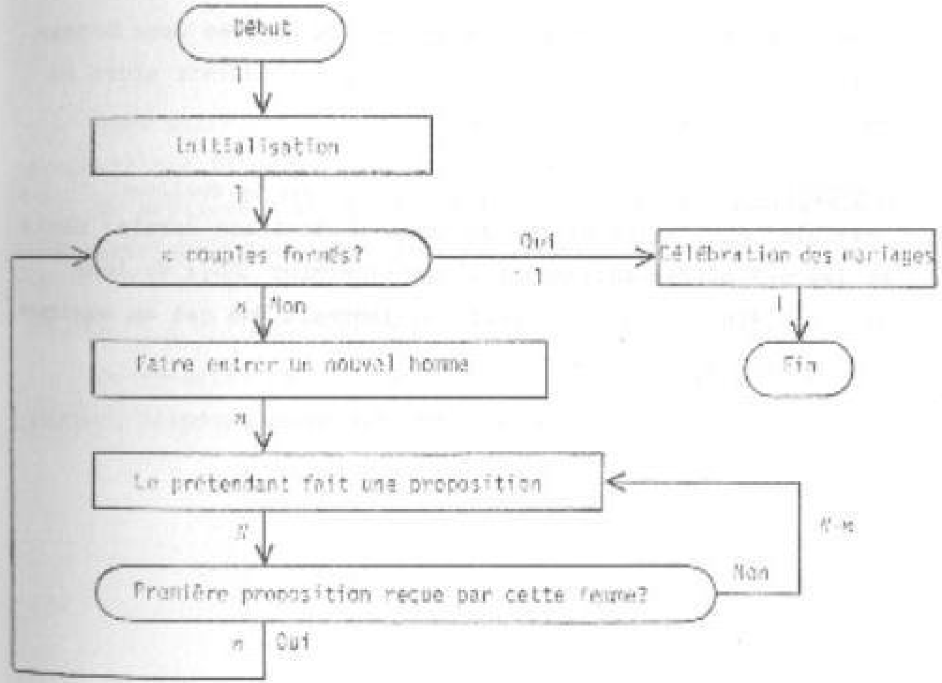
<u>DEMONSTRATION</u>. Le pire couplage stable du point de vue des femmes contient Aa; c'est aussi la meilleure solution pour les

hommes.

Remarquons que ce théorème s'applique aux listes complétées à l'aide des éléments V et v.

<u>Analyse de l'algorithme</u>

En combien d'étapes cet algorithme se terminera-t-il? Voyons ce qui se passe à l'aide de l'organigramme:



L'indice le long d'une flèche indique le nombre total d'opérations effectuées par l'algorithme. Le nombre d'hommes ou de femmes est n qui est une constante connue. Le nombre total de propositions des prétendants est noté N, et ce nombre dépend de la structure des listes de préférence.

Il est intéressant de connaître la valeur moyenne de N lorsque les listes de préférence sont construites au hasard. Ceci fera l'objet des prochaines conférences. Une autre valeur intéressante est celle que N peut atteindre dans la pire situation. Cette question est étudiée dans les exercices qui suivent.

EXERCICE 2.1. Prouver qu'au plus un homme obtient son moins bon choix (avec l'algorithme fondamental).

[Conséquence: S'il existe un couplage stable avec deux hommes (ou plus) qui obtiennent leur pire choix, il existe alors au moins deux couplages stables.]

EXERCICE 2.2. Le résultat de l'exercice 1 est le meilleur possible: dans certains cas un homme obtient son dernier choix et les n-1 autres obtiennent l'avant-dernier choix de leur liste. [Ainsi le nombre total des propositions est au maximum $n + (n-1)(n-1) = n^2 - n + 1$.]

TROISIÈME CONFÉRENCE

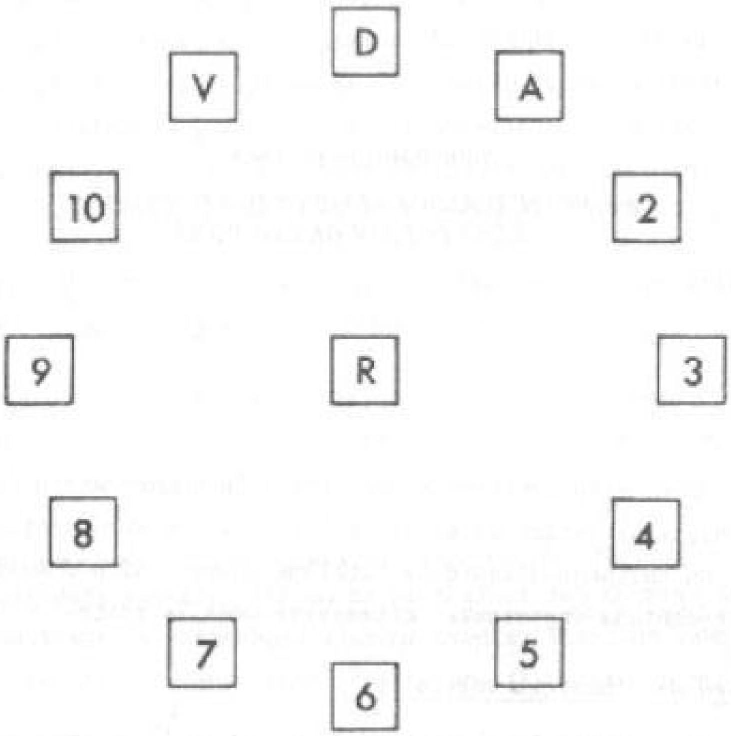
PRINCIPE D'AJOURNEMENT DES DÉCISIONS; COLLECTION DE COUPONS

Dans cette conférence, on estime le nombre moyen de propositions faites au cours de l'algorithme fondamental.

On envisage d'abord le problème suivant afin d'introduire une technique importante, nécessaire pour la suite.

Le jeu de l'horloge solitaire

On répartit les 52 cartes d'un jeu en 13 paquets de 4 cartes, disposés comme sur une horloge:



Chaque heure indique le numéro d'un paquet, sauf pour

- le paquet à 1 heure, appelé "As",
- le paquet à 11 heures, appelé "Valet",
- le paquet à 12 heures, appelé "Dame",
- le paquet au centre, appelé "Roi".

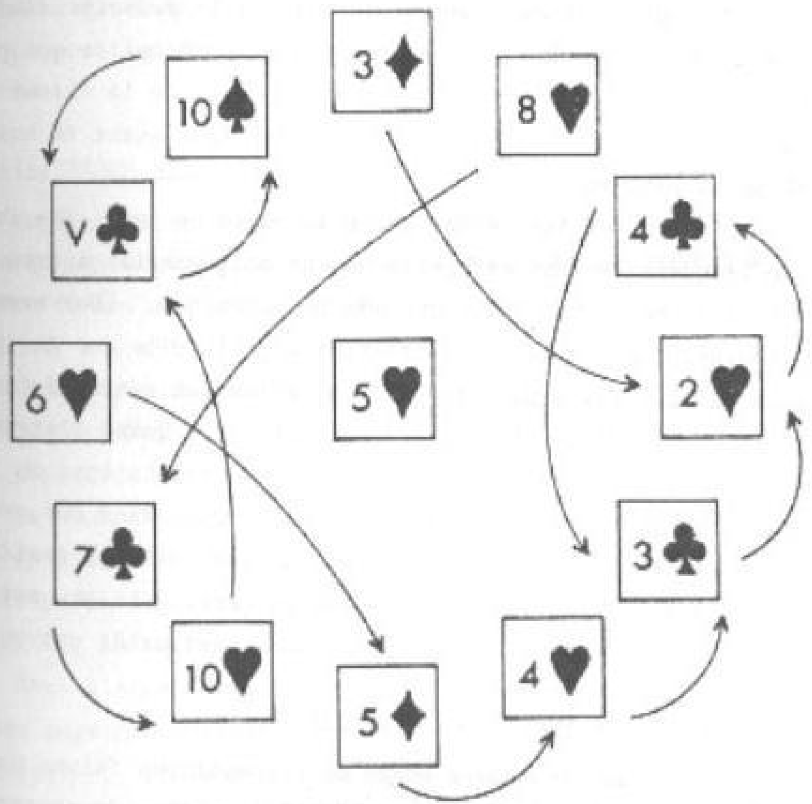
Le jeu consiste à retourner successivement toutes les cartes selon la règle suivante:

- on retourne d'abord la première carte du paquet "Roi";
- on la dispose à son emplacement sur l'horloge (si c'est un 7, près du paquet à 7 heures);
- puis on recommence la même opération avec le paquet correspondant (on retourne la première carte du paquet à 7 heures; c'est un As par exemple; on le dispose près du

paquet "As", et ainsi de suite).

Le jeu s'arrête quand on arrive à un paquet dont toutes les cartes sont épuisées. Ceci se produit lorsque les quatre Rois sont retournés. On gagne si toutes les cartes sont à découvert.

D'après un théorème de la théorie des graphes, il y a une façon plus rapide de savoir si l'on gagne ou si l'on perd: tout dépend des dernières cartes de chaque paquet. Envisageons par exemple la distribution des dernières cartes:



Les arcs indiquent les positions où chaque carte disposée sur le cercle doit être déplacée. On obtient ainsi un *graphe orienté*.

Si ce graphe contient un cycle ou plus, alors le joueur a perdu: on ne peut retourner toutes les cartes. S'il n'y a pas de cycle, le joueur va gagner, indépendamment de la position des autres cartes.

La probabilité de gagner est 1/13. Il y a deux façons de le montrer:

(i) une méthode difficile consiste à énumérer tous les arrangements du jeu tels qu'aucun cycle n'apparaisse entre les dernières cartes des paquets disposés sur le cercle.

(ii) une méthode plus simple utilise le *principe d'ajournement des décisions*: "Ne faites pas aujourd'hui ce qui peut être fait demain." En particulier, le choix de la valeur d'une carte retournée n'a jamais besoin d'être fait avant le moment où on la retourne.

On retourne les cartes selon la règle du jeu. Une fois que le quatrième Roi est retourné, on découvre les autres cartes, s'il en reste, selon une règle quelconque. (Par exemple, retourner toutes les cartes restant dans la pile des As, puis dans la pile des deux, etc.) On a maintenant retourné toutes les cartes du jeu selon un ordre aléatoire. Quand l'arrangement initial des cartes parcourt les 52! permutations du jeu, l'arrangement obtenu en retournant les cartes dans cet ordre parcourt aussi les 52! permutations du jeu. Il est facile alors de dénombrer les arrangements gagnants. Le jeu est gagné si et seulement si la dernière carte retournée est un Roi.

Etude du nombre moyen N de propositions

On cherche le nombre moyen de propositions faites par les prétendants successifs au cours de l'algorithme de construction d'un couplage stable. On suppose que la matrice de préférence des femmes est donnée, et que celle des hommes, en revanche, est aléatoire. Il y a, pour chaque homme, $n!$ permutations des

n femmes, donc $(n!)^n$ matrices de préférence équiprobables. Comment dénombrer le nombre moyen de propositions de l'algorithme?

On utilise le principe d'ajournement des décisions. Au lieu de choisir à l'avance sa liste de préférence parmi les $n!$ possibles, chaque prétendant se décide au dernier moment: à chaque nouvelle proposition, il choisit au hasard la femme à qui cette proposition s'adresse. Ceci consiste à choisir aléatoirement une carte parmi les cartes restantes.

Mais il reste difficile de calculer ainsi le nombre moyen $\bar{N}$ de propositions. Il est nécessaire de simplifier le problème.

Simplification du problème

On suppose que les hommes sont *amnésiques*. Le moment venu de faire une nouvelle avance à une femme, ils ont oublié à quelles femmes ils ont déjà fait une proposition.

Ceci revient à mélanger le jeu complet avant que le prétendant ne tire une carte pour faire une nouvelle proposition. L'ordre dans lequel les cartes apparaissent *pour la première fois* constitue une permutation aléatoire.

Le nombre moyen $\bar{N}$ de propositions sera le nombre moyen de de propositions faites par les hommes amnésiques, diminué du nombre moyen d'avances redondantes. En étudiant le problème simplifié, on obtiendra donc une borne supérieure pour $\bar{N}$.

Dans l'algorithme fondamental, le nombre de couples formés augmente chaque fois qu'une femme reçoit sa première proposition. L'algorithme se termine dès que chaque femme a été demandée au moins une fois. Avec des hommes amnésiques, les propositions forment une suite aléatoire que l'on arrête dès que chaque femme a été représentée.

Regardons ce qui se passe sur un exemple. Soient quatre hommes A, B, C, D, et quatre femmes a, b, c, d, dont la liste de préférence est donnée arbitrairement

a : C A D B
b : B D A C
c : C A B D
d : C D B A

Considérons la suite aléatoire des propositions

$d, b, d, b, c, c, b, c, d, a,$

qui va servir à tous les prétendants successifs. On fait fonctionner l'algorithme fondamental, chaque prétendant faisant ses avances aux femmes dans l'ordre de la liste

A propose à $d \Rightarrow$ formation du couple Ad à l'essai.
B propose à $b \Rightarrow$ formation du couple Bb à l'essai.
C propose à $d \Rightarrow$ formation du couple Cd à l'essai.
puisque d préfère C à A,
A devient le nouveau prétendant.
A propose à b; pas de nouveau couple formé, puisque b préfère B à A,
A propose à $c \Rightarrow$ formation du couple Ac.

Ceci revient à construire la liste partielle de préférence suivante:

A : d b c
B : b
C : d
D :

On continue ainsi jusqu'à ce qu'une avance soit faite à la dernière femme apparaissant pour la première fois sur la liste, c'est-à-dire a.

D propose à c; refus de c.
D propose à b; refus de b.
D propose à c; refus de c à nouveau.

Cette avance est la première proposition redondante.

D propose à d; refus de d.
D propose à $a \Rightarrow$ formation du couple Da.

Le tableau partiel des listes de préférence est finalement

A : d b c
B : b
C : d
D : c b d a

Dix propositions ont été faites, dont une redondante.

Remarquons qu'il est inutile de construire des listes de préférence complètes, puisque avec l'algorithme fondamental, on obtient souvent un couplage stable avant d'avoir consulté toute la matrice de préférence.

La longueur de la suite nécessaire pour obtenir ainsi un couplage stable correspond au nombre de propositions faites au cours de l'algorithme. La longueur moyenne de telles suites correspond donc au *nombre moyen* $\bar{N}$ *de propositions* (avec amnésie). Une suite permet d'obtenir un couplage stable quand toutes les femmes y ont apparues au moins une fois. Calculer la distribution de la longueur d'une telle suite constitue le problème de la *collection de coupons*.

Collection de coupons

On suppose qu'il y a n coupons différents, et que chaque fois qu'on achète une boîte de détergent, on obtient un coupon. Combien de boîtes doit-on acheter, en moyenne, pour obtenir les n coupons?

<u>EXEMPLE</u>. Ici $n = 10$; on cherche tous les chiffres $0,1,2,\ldots,9$, dans les développements décimaux de π et de e.

$$\underline{3}\ \underline{1}\ \underline{4}\ 1\ \underline{5}\ \underline{9}\ \underline{2}\ \underline{6}\ 5\ 3\ 5\ \underline{8}\ 9\ \underline{7}\ 9$$
$$3\ 2\ 3\ 8\ 4\ 6\ 2\ 6\ 4\ 3\ 3\ 8\ 3$$
$$2\ 7\ 9\ 5\ \underline{0}$$

$$\underline{2}\ \underline{7}\ \underline{1}\ \underline{8}\ 2\ 8\ 1\ 8\ 2\ 8\ \underline{4}\ \underline{5}\ \underline{9}\ \underline{0}\ 4\ 5$$
$$2\ \underline{3}\ 5\ 3\ \underline{6}$$

Précisons d'abord les notations. Soit p_k la probabilité qu'exactement k boîtes soient nécessaires. Posons

$$\begin{aligned} q_1 &= p_1 + p_2 + p_3 + p_4 + \ldots \\ q_2 &= \phantom{p_1 +{}} p_2 + p_3 + p_4 + \ldots \\ q_3 &= \phantom{p_1 + p_2 +{}} p_3 + p_4 + \ldots \\ q_k &= p_k + p_{k+1} + \ldots \end{aligned}$$

Alors q_k est la probabilité qu'au moins $\underline{k}$ boîtes soient nécessaires, et on a

$$q_1 = 1; \quad p_k = q_k - q_{k+1}.$$

Le nombre moyen de boîtes utilisées est par définition

$$p_1 + 2p_2 + 3p_3 + \ldots = q_1 + q_2 + q_3 + \ldots$$

On décompose le problème en n étapes successives. A l'étape m, $0 \le m < n$, nous sommes déjà en possession de m coupons, et on cherche le nombre moyen de boîtes qu'il faut acheter pour obtenir un coupon supplémentaire. La probabilité d'acheter une boîte ou plus est évidemment 1. (Il faut acheter au moins une boîte pour trouver un nouveau coupon.) La probabilité de trouver, chaque fois que l'on achète une nouvelle boîte, un des m coupons déjà en notre possession est m/n. Donc pour l'étape m

$$\begin{aligned} q_1 &= 1 \\ q_2 &= \frac{m}{n} \\ q_3 &= \left(\frac{m}{n}\right)^2 \\ &\ldots; \end{aligned}$$

$$q_1 + q_2 + q_3 + \ldots = 1 + \frac{m}{n} + \left(\frac{m}{n}\right)^2 + \ldots = \frac{n}{n-m}.$$

Le nombre moyen de boîtes à acheter pour obtenir les n coupons est donc

$$\begin{aligned} &\frac{n}{n-0} + \frac{n}{n-1} + \frac{n}{n-2} + \ldots + \frac{n}{n-(n-1)} \\ &= n\left(1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}\right) = nH_n \end{aligned}$$

où H_n est la somme des n premiers termes de la *série harmonique* $1, \frac{1}{2}, \frac{1}{3}, \ldots$.

<u>Conclusion</u>

On a trouvé une *borne supérieure pour le nombre moyen de propositions* $\bar{N}$ faites au cours de l'algorithme fondamental permettant d'obtenir un couplage stable: En effet, la longueur moyenne d'une suite aléatoire de femmes telle que les n femmes apparaissent toutes au moins une fois, correspond au nombre moyen de boîtes à acheter pour obtenir les n coupons, soit nH_n. Mais

$$H_n = \ln n + \gamma + \frac{1}{2n} - \frac{1}{12n^2} + \epsilon,$$

où $0 < \varepsilon < \frac{1}{120n^4}$, γ est la constante d'Euler et ln désigne le logarithme népérien. Donc, pour n assez grand, le nombre moyen de propositions $\bar{N}$ est asymptotiquement équivalent à $n \ln n$.

On peut raffiner ce résultat en considérant un modèle avec *amnésie partielle*.

<u>Amnésie partielle</u>

On suppose maintenant que chaque homme a toujours une très mauvaise mémoire, mais qu'il peut néanmoins se souvenir du nom de la femme qui vient de le rejeter. Ce modèle donne une meilleure borne supérieure pour le nombre moyen de propositions.

Quand une proposition a été faite à m femmes, $0 \le m < n$, le nombre moyen de propositions à faire pour proposer à une $(m+1)$-ième femme peut être calculé en modifiant légèrement le schéma de la collection de coupons:

$$q_1 = 1$$

$$q_2 = \frac{m}{n}$$

$$q_3 = \frac{m}{n}\,\frac{m-1}{n-1}$$

$$q_4 = \frac{m}{n}\left(\frac{m-1}{n-1}\right)^2$$

$$\ldots;$$

$$q_1 + q_2 + q_3 + \ldots = 1 + \frac{m}{n}\left(1 + \frac{m-1}{n-1} + \left(\frac{m-1}{n-1}\right)^2 + \ldots\right)$$

$$= \frac{n}{n-m} - \frac{m}{n(n-m)}$$

La probabilité de faire une proposition ou plus est évidemment $q_1 = 1$. La probabilité que la première avance soit faite à une des m femmes est $\frac{m}{n}$. La probabilité pour que les autres avances soient faites à une des $m-1$ femmes parmi les $n-1$ restantes quand on retire de la liste la femme qui vient de rejeter son prétendant, est $\frac{m-1}{n-1}$.

Le nombre moyen $\bar{N}$ de propositions se calcule facilement:

$$\bar{N} = \sum_{0\le m<n} \frac{n^2-m}{n(n-m)} = \sum_{1\le m\le n} \frac{n^2-(n-m)}{nm}$$

$$= (n-1)\left(\sum_{1\le m\le n} \frac{1}{m}\right) + \frac{1}{n}\left(\sum_{1\le m\le n} 1\right)$$

$$= (n-1)H_n + 1.$$

On obtient donc le résultat suivant:

<u>THEOREME</u>. Pour toute matrice de préférence des femmes, le nombre moyen de propositions faites au cours de l'algorithme aboutissant à la solution optimale pour les hommes est au plus $(n-1)H_n + 1$.

Ainsi chaque homme fait au plus approximativement $\ln n$ propositions, en moyenne. Une femme reçoit approximativement le même nombre de propositions. Elle obtient donc un choix au moins à peu près à $n/\ln n$ du début de sa liste.

<u>EXERCICE 3.1</u>. Etant donné la matrice de préférence des femmes

$$\begin{aligned} a &: A\ B\ C \\ b &: B\ C\ A \\ c &: C\ A\ B, \end{aligned}$$

considérer les $216 = (3!)^3$ choix possibles pour les hommes. Montrer qu'il y a exactement

48	matrices	demandant	3	propositions,
72	"	"	4	" ,
60	"	"	5	" ,
30	"	"	6	" ,

6 matrices demandant 7 propositions.

[Ainsi le nombre moyen de propositions est $4 + 5/12$; la borne supérieure est $2^{H}_3 + 1 = 4 + 8/12$.]

EXERCICE 3.2. Etant donné la suite de femmes $d\ b\ d\ b\ a\ c\ b\ c\ d\ a$, déterminer la probabilité pour que k propositions redondantes soient faites, pour chaque $k = 0,1,2,\ldots$, si les hommes sont amnésiques et la matrice de préférence des femmes aléatoire.

QUATRIÈME CONFÉRENCE

DÉVELOPPEMENTS THÉORIQUES: APPLICATION AUX PLUS COURTS CHEMINS

Aujourd'hui nous allons examiner les points suivants:

1. Revue rapide de la théorie des probabilités discrètes.
2. Etude de la variance dans le problème de la collection de coupons.
3. Algorithme fondamental de couplage stable: étude du cas le plus défavorable.
4. Application à la détermination du *plus court chemin sur un graphe*.

1. Théorie des probabilités discrètes

Fonctions génératrices

Soit X une *variable aléatoire discrète*, prenant ses valeurs dans l'ensemble $\mathbb{N}$ des entiers positifs ou nuls. Soit p_k la probabilité que $X = k$; donc $p_k \geq 0$, et $p_0 + p_1 + p_2 + \ldots = 1$.

La fonction génératrice pour cette densité de probabilité est la série

$$P(z) = p_0 + p_1 z + p_2 z^2 + \ldots = \sum_{k\geq 0} p_k z^k.$$

Remarquons que $P(1) = 1$.

La valeur moyenne $\bar{X}$ et la variance $V(X)$ de X peuvent être calculées à partir des valeurs des dérivées de la fonction génératrice à $z = 1$. Nous avons

$$P(z) = p_0 + p_1 z + p_2 z^2 + \ldots = \sum_{k\geq 0} p_k z^k;$$

$$P'(z) = p_1 + 2p_2 z + 3p_3 z^2 + \ldots = \sum_{k\geq 0} k p_k z^{k-1};$$

$$P''(z) = 2p_2 + 6p_3 z + 12 p_4 z^2 + \ldots = \sum_{k\geq 0} k(k-1) p_k z^{k-2}.$$

Soit $E(X)$ la valeur moyenne $\bar{X}$ de la variable aléatoire X; on a

$$E(X) = \sum_{k\geq 0} k p_k = P'(1).$$

La variance $V(X)$ de la variable aléatoire X est définie comme étant la valeur moyenne de $(X-E(X))^2$, soit

$$\begin{aligned} V(X) &= \sum_{k\geq 0} (k-E(X))^2 p_k \\ &= \sum_{k\geq 0} k^2 p_k - 2E(X) \sum_{k\geq 0} k p_k + E(X)^2 \sum_{k\geq 0} p_k \\ &= \sum_{k\geq 0} k(k-1) p_k + \sum_{k\geq 0} k p_k - 2E(X)^2 + E(X)^2 \\ &= P''(1) + P'(1) - P'(1)^2. \end{aligned}$$

Signification de la variance

Soit $\sigma = \sqrt{V(X)}$ l'écart type de la variable aléatoire X pour la densité de probabilité p_k, $k \geq 0$.

Pour tout $m \geq 1$, la probabilité que $|X-E(X)| > m\sigma$ est inférieure à $1/m^2$.

Ce résultat est indépendant de la densité de probabilité considérée. On l'appelle l'*inégalité de Tchébichev* (résultat dû aussi à J. Bienaymé). Ainsi dans 99% des cas les inégalités suivantes sont vérifiées:

$$E(X) - 10\sigma \leq X \leq E(X) + 10\sigma.$$

DEMONSTRATION. Soit p la probabilité pour que $|X-E(X)| > m\sigma$. Alors si $p > 0$,

$$\begin{aligned} V(X) &= E((X-E(X))^2) \\ &> p(m\sigma)^2 + (1-p)\cdot 0 \\ &= pm^2 V(X), \end{aligned}$$

c'est-à-dire, $p < 1/m^2$.

Variables aléatoires indépendantes

Soit Y une autre variable aléatoire, telle que la probabilité que $Y = k$ soit q_k. La fonction génératrice correspondante est

$$Q(z) = \sum_{k\geq 0} q_k z^k.$$

Les variables aléatoires X et Y sont indépendantes si la probabilité que $X = j$ et $Y = k$ est $p_j q_k$, pour tout j et k.

Si X et Y sont indépendantes, la probabilité que $X+Y = m$ est

$$r_m = p_0 q_m + p_1 q_{m-1} + \ldots + p_{m-1} q_1 + p_m q_0.$$

Dans ce cas, la fonction génératrice de la nouvelle variable aléatoire $X+Y$ est le produit des fonctions génératrices de X et Y; en effet:

$$\sum_{m\geq 0} r_m z^m = \sum_{m\geq 0}\left(\sum_{\substack{j,k\geq 0\\ j+k=m}} p_j q_k\right) z^m$$

$$= \left(\sum_{j\geq 0} p_j z^j\right)\left(\sum_{k\geq 0} q_k z^k\right)$$

$$= P(z)Q(z).$$

Pour toute fonction génératrice $P(z)$, on écrit

$$e(P) = P'(1)$$
$$v(P) = P''(1) + P'(1) - P'(1)^2.$$

On peut montrer que

$$e(PQ) = e(P) + e(Q),$$
$$v(PQ) = v(P) + v(Q).$$

DEMONSTRATION. On a

$$(PQ)'(z) = P'(z)Q(z) + P(z)Q'(z)$$
$$(PQ)''(z) = P''(z)Q(z) + 2P'(z)Q'(z) + P(z)Q''(z).$$

Donc

$$(PQ)'(1) = P'(1) + Q'(1)$$

$$\begin{aligned}(PQ)''(1) &+ (PQ)'(1) - (PQ)'(1)^2\\ &= P''(1) + 2P'(1)Q'(1) + Q''(1) + P'(1) + Q'(1)\\ &- P'(1)^2 - 2P'(1)Q'(1) - Q'(1)^2\\ &= P''(1) + P'(1) - P'(1)^2 + Q''(1) + Q'(1) - Q'(1)^2.\end{aligned}$$

Distribution cumulée

Soit $q_k = p_k + p_{k+1} + \ldots$ la probabilité que $X \geq k$. La fonction génératrice $Q(z) = q_0 + q_1 z + q_2 z^2 + \ldots$ ne correspond à aucune densité de probabilité, car $\sum q_k > 1$ sauf si $p_0 = 1$.

Puisque $p_k = q_k - q_{k+1}$, on a

$$\begin{aligned}P(z) &= \sum_{k\geq 0}(q_k - q_{k+1})z^k\\ &= \sum_{k\geq 0} q_k z^k - z^{-1}\sum_{k\geq 0} q_{k+1} z^{k+1}\\ &= Q(z) - z^{-1}(Q(z) - Q(0));\end{aligned}$$

donc, puisque $Q(0) = 1$,

$$zP(z) = (z-1)Q(z) + 1.$$

En dérivant cette relation, on obtient

$$P(z) + zP'(z) = Q(z) + (z-1)Q'(z)$$
$$2P'(z) + zP''(z) = 2Q'(z) + (z-1)Q''(z),$$

d'où

$$e(P) = Q(1) - 1 \tag{1}$$

$$v(P) = 2Q'(1) + Q(1) - Q(1)^2. \tag{2}$$

2. Variance dans le problème de la collection de coupons

Soit $P_m(z)$ la fonction génératrice de la variable aléatoire: "nombre de boîtes qu'il faut acheter pour avoir un $(m+1)$-ième coupon quand on en a déjà m". Les variables aléatoires correspondantes étant indépendantes, on a

$$P(z) = P_0(z)P_1(z)\ \ldots\ P_{n-1}(z),$$

$$e(P) = e(P_0) + e(P_1) + \ldots + e(P_{n-1}) \tag{3}$$

$$v(P) = v(P_0) + v(P_1) + \ldots + v(P_{n-1}), \tag{4}$$

d'où:

$$Q_m(z) = 1 + z + \frac{m}{n}z^2 + (\frac{m}{n})^2 z^3 + \ldots = 1 + \frac{nz}{n-mz}$$

$$P_m(z) = \frac{(n-m)z}{n-mz}$$

$$Q_m(1) = 1 + \frac{n}{n-m}, \quad Q'_m(1) = \frac{n^2}{(n-m)^2}$$

$$e(P_m) = \frac{n}{n-m}, \quad v(P_m) = \frac{mn}{(n-m)^2}.$$

On obtient donc à l'aide de (1), (2), (3) et (4)

$$e(P) = \sum_{1\le m\le n} \frac{n}{m} = nH_n = n \ln n + O(n);$$

$$v(P) = \sum_{1\le m\le n} \frac{n(n-m)}{m^2} = n^2(1 + \frac{1}{2^2} + \ldots + \frac{1}{n^2}) - n(1 + \frac{1}{2} + \ldots + \frac{1}{n})$$

$$= \frac{\pi^2}{6}n^2 - n \ln n + O(n).$$

La notation $O(f(n))$ représente une fonction dont le quotient par $f(n)$ reste borné pour tout n assez grand.

Amélioration par rapport à l'inégalité de Tchébichev

Puisque l'écart type est $\sqrt{v(P)} = O(n)$, on déduit de l'inégalité de Tchébichev que le nombre moyen de propositions est presque toujours $nH_n + O(n)$ quand les hommes font des avances aléatoires avec amnésie. Mais cette inégalité s'applique à des distributions arbitraires, et dans des cas particuliers on peut souvent obtenir des résultats plus forts.

Considérons par exemple la distribution engendrée par $P_m(z)$. La moyenne est $n/(n-m)$, et l'écart type $\sqrt{mn}/(n-m)$; l'inégalité de Tchébichev nous dit donc que la probabilité d'acheter plus de $\sigma n/(n-m)$ boîtes avant d'avoir le $(m+1)$-ième coupon est au plus $m/n(\sigma-1)^2$, pour tout $\sigma > 1$. En fait nous

savons que la probabilité d'acheter plus de $\sigma n/(n-m)$ boîtes est

$$q_{1+\sigma n/(n-m)} = (\frac{m}{n})^{\sigma n/(n-m)} \le e^{-\sigma},$$

puisque $e^x \ge 1+x$ pour tout x, et on peut prendre $x = -(n-m)/m$. Quand σ est grand, cette estimation est bien meilleure que celle donnée par l'inégalité de Tchébichev.

Considérons maintenant le processus de collection des coupons dans son ensemble. La probabilité qu'il faille acheter plus de σnH_n boîtes est au plus égale à la probabilité qu'il y ait au moins une étape m dans laquelle plus de $\sigma n/(n-m)$ boîtes sont achetées; cette seconde probabilité est au plus $ne^{-\sigma}$. Avec le choix $\sigma = C \ln n$, on obtient le résultat suivant:

THÉORÈME. La probabilité d'acheter plus de $CnH_n \ln n$ boîtes, avant de compléter la collection de n coupons, est au plus $1/n^{C-1}$.

L'inégalité de Tchébichev, qui ne donne qu'une borne supérieure de l'ordre de $1/C^2(\log n)^4$, ne suffirait pas à prouver ce théorème.

Nous avons maintenant démontré que le nombre de propositions n'est nettement supérieur à nH_n que rarement, même avec des hommes amnésiques. Mais qu'en est-il dans le cas le plus défavorable: combien peut-il y avoir de propositions? (Cette question n'a de sens qu'en l'absence d'amnésie, bien sûr.)

3. Algorithme fondamental: étude du cas le plus défavorable

On résout maintenant les exercices de la deuxième conférence.

Un homme au plus arrive à la fin de sa liste, puisque au moment où il fait une avance à la n-ième femme l'algorithme s'arrête. (Toutes les femmes ont été nommées.)

Les n-1 autres hommes font chacun au plus n-1 propositions. Le nombre total de propositions est donc au plus $n + (n-1)^2 = n^2-n+1$.

Remarquons qu'une femme (la dernière citée) reçoit une seule proposition. Les n-1 autres femmes doivent recevoir n propositions chacunes. Le couplage stable ainsi obtenu est optimal à la fois pour les hommes et pour les femmes, donc unique.

Voici un exemple où ceci se produit:

A :	b	c	d	e	a		a :	liste arbitraire				
B :	c	d	e	b	a		b :	B	C	D	E	A
C :	d	e	b	c	a		c :	C	D	E	A	B
D :	e	b	c	d	a		d :	D	E	A	B	C
E :	b	c	d	e	a		e :	E	A	B	C	D

A propose à b; formation du couple Ab à l'essai.
B propose à c; formation du couple Bc à l'essai.
C propose à d; formation du couple Cd à l'essai.
D propose à e; formation du couple De à l'essai.
E propose à b; formation du couple Eb à l'essai, puisque b préfère E à A;
A devient le nouveau prétendant.
A propose à c; formation du couple Ac à l'essai, puisque c préfère A à B;
B devient le nouveau prétendant.
B propose à d; formation du couple Bd à l'essai, puisque d préfère B à C;
C devient le nouveau prétendant, et ainsi de suite... jusqu'à ce que a reçoive sa première proposition de la part de A.

L'algorithme s'arrête quand l'homme A est arrivé à la fin de sa liste. Les hommes B, C, D et E ont fait chacun quatre

propositions, et sont mariés respectivement avec b, c, d et e. On voit que ces femmes ont obtenu leur meilleur choix. Quelle que soit la liste de préférence de la femme a, le couplage stable obtenu est optimal pour les femmes.

4. Plus court chemin dans un graphe

Soient n villes $1,2,3,\ldots,n$, reliées entre elles par des chemins de longueurs données. Soit L_{ij} la longueur du chemin menant de la ville i à la ville j. (Si l'on ne peut allor directement de i à j, on notera $L_{ij} = \infty$.) Il n'est pas nécessaire que $L_{ij} = L_{ji}$.

Un algorithme important, dû à E.W. Dijkstra, donne la plus courte distance de la ville 1 à chacune des autres villes. Notons que cet algorithme peut être appliqué à d'autres questions ne relevant pas de la circulation routière: temps minimal pour exécuter une série de tâches, etc.

Définissons d'abord les variables utilisées au cours de l'algorithme. (On dira que la distance entre la ville 1 et une autre ville j est connue quand on aura déterminé le plus court chemin de 1 à j sur le graphe.)

A : ensemble des villes dont la plus courte distance à la ville 1 est connue.
B : ensemble des autres villes.
d_i = plus courte distance de 1 à i, quand on passe uniquement par les villes de A.
k : ville passant de B dans A.

Description de l'algorithme

$d_1 \leftarrow 0$; $A \leftarrow \{1\}$; $B \leftarrow \{2,\ldots,n\}$;
<u>pour</u> $i \in B$ <u>faire</u> $d_i \leftarrow L_{1i}$;
<u>tant que</u> B est non vide <u>faire</u>

<u>début</u> $k \leftarrow$ ville de B telle que $d_k = \min\{d_i | i \in B\}$;

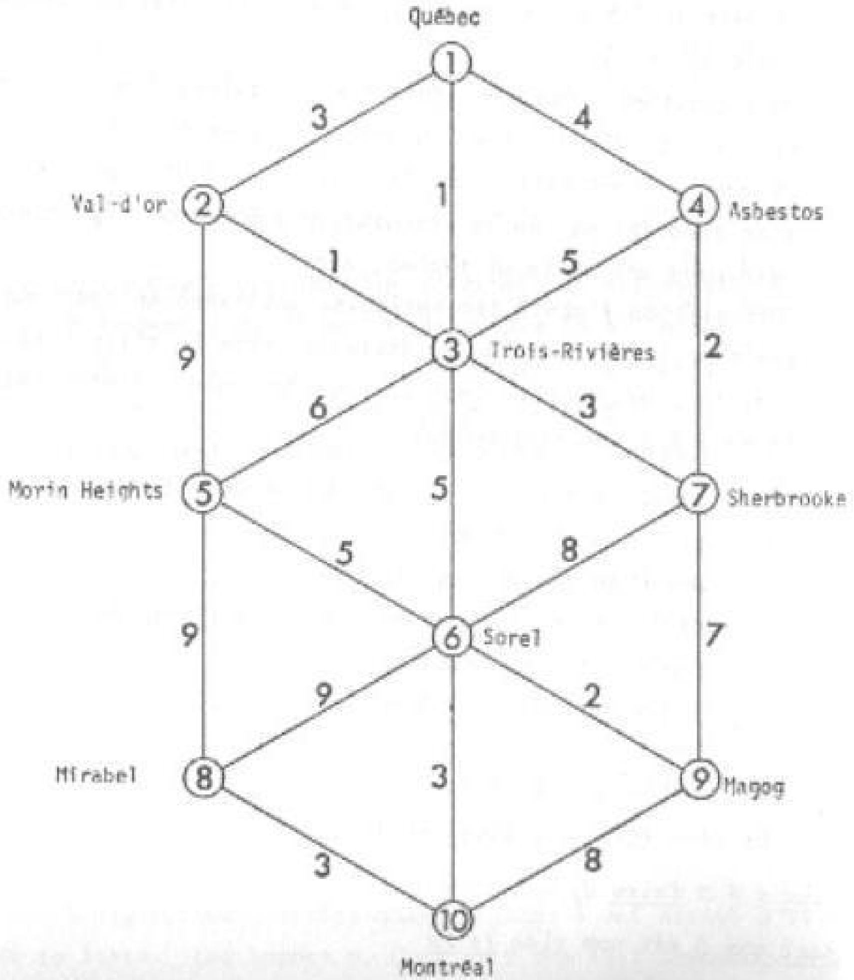
$B \leftarrow B\backslash\{k\}$; $A \leftarrow A \cup \{k\}$;

<u>pour</u> $i \in B$ <u>faire</u> $d_i \leftarrow \min(d_i, d_k + L_{ki})$

<u>fin</u>.

(L'instruction $B \leftarrow B\backslash\{k\}$ retire l'élément k de l'ensemble B.)

<u>EXEMPLE</u>. Etudions le fonctionnement de l'algorithme sur l'exemple suivant:



Les villes sont numérotées de 1 (Québec) à 10 (Montréal). Les distances sont indiquées par un nombre le long de l'arc reliant deux villes.

<u>Initialisation</u>

Aucun chemin de Québec à une autre ville n'est encore connu

$$d_1 \leftarrow 0; \quad A \leftarrow \{1\}; \quad B \leftarrow \{2,\ldots,10\}.$$

Il s'agit de déterminer le plus court chemin de Québec aux 9 villes de $B = \{2,3,\ldots,10\}$.

Les seules villes reliées directement à Québec sont Val-d'or, Trois-Rivières et Asbestos, donc $d_2 \leftarrow 3$; $d_3 \leftarrow 1$; $d_4 \leftarrow 4$; et $d_i \leftarrow \infty$, pour les autres villes.

<u>Première étape</u>

Puisque B n'est pas vide, on exécute la séquence d'instructions comprises entre <u>début</u> et <u>fin</u>.

La ville la plus proche de Québec est Trois-Rivières, donc

$$k \leftarrow 3; \quad B \leftarrow B\backslash\{3\}; \quad A \leftarrow A \cup \{3\}.$$

(On retire 3 de l'ensemble B et on l'inclut dans A, parce qu'on connaît la plus courte distance de Québec à la ville 3 comme il sera démontré par la suite.)

Pour toutes les autres villes de B (ici pour les villes 2,4,5,...,10), on exécute l'instruction

<u>pour</u> $i \in B$ <u>faire</u> $d_i \leftarrow \min(d_i, d_3 + L_{3i})$,

c'est-à-dire qu'on cherche les plus courts chemins passant par une ville de $A = \{1,3\}$ et menant aux villes de B. On obtient le tableau

Villes de B	Villes de A
$d_2 = 2$	$d_1 = 0$
$d_4 = 4$	$d_3 = 1$
$d_5 = 7$	
$d_6 = 6$	
$d_7 = 4$	

(Pour les autres villes de B, d_i reste ∞.)

Deuxième étape

L'ensemble B est toujours non vide: $B = \{2,4,5,\ldots,10\}$. La ville de B la plus proche de Québec en passant uniquement par des villes de A est donc la ville ②, et $k \leftarrow 2$; $B \leftarrow B\setminus\{2\}$; $A \leftarrow A \cup \{2\}$. Le nouveau tableau obtenu est

Villes de B	Villes de A
$d_4 = 4$	$d_1 = 0$
$d_5 = 7$	$d_2 = 2$
$d_6 = 6$	$d_3 = 1$
$d_7 = 4$	

(Le chemin pour aller à Morin Heights via Val-d'or n'est pas plus court que celui via Trois-Rivières, donc d_5 reste inchangé. Pour les villes de B ne figurant pas sur le tableau, $d_i = \infty$.)

Troisième étape

L'ensemble B est toujours non vide: $B = \{4,5,\ldots,10\}$. La ville de B la plus proche de Québec (en passant uniquement par des villes de A) est maintenant la ville ④, et $k \leftarrow 4$; $B \leftarrow B\setminus\{4\}$; $A \leftarrow A \cup \{4\}$.

On omet le tableau puisqu'il n'y a aucune amélioration des distances d_i, $i \in B$. On a simplement fait passer ④ dans A.

Quatrième étape

$$k \leftarrow 7; \quad B \leftarrow B\setminus\{7\}; \quad A \leftarrow A \cup \{7\}.$$

On peut maintenant atteindre ⑨ ($d_9 = 11$) mais aucune autre distance d_i, $i \in B$, n'est améliorée.

Cinquième étape

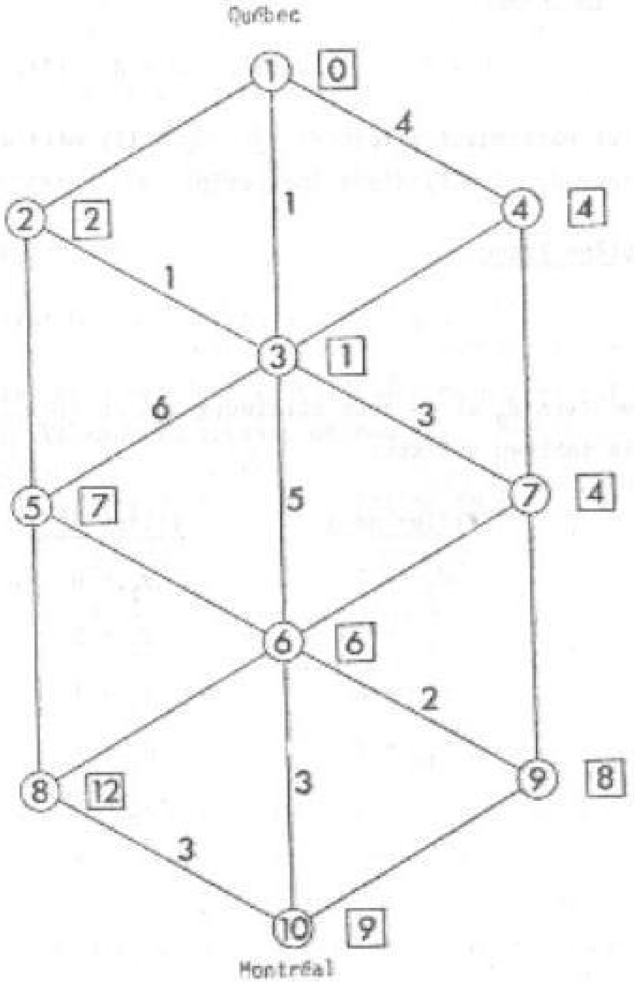
$$k_6 \leftarrow 6; \quad B \leftarrow B\setminus\{6\}; \quad A \leftarrow A \cup \{6\}.$$

On améliore d_9 et on peut atteindre ⑧ et ⑩, comme le montre le tableau suivant:

Villes de B	Villes de A
$d_5 = 7$	$d_1 = 0$
$d_8 = 15$	$d_2 = 2$
$d_9 = 8$	$d_3 = 1$
$d_{10} = 9$	$d_4 = 4$
	$d_6 = 6$
	$d_7 = 4$

Montrons maintenant que d_k est la plus courte distance de Québec à la ville k, au moment où k passe de B dans A. S'il y avait un plus court chemin, il devrait nécessairement passer par une autre ville de B que k. Mais toutes les autres villes de B sont au moins aussi loin de Québec que k, et toutes les distances sont positives ou nulles.

Quand l'algorithme se termine, le tableau des valeurs d_i, $i = 1,2,\ldots,10$, est:



Les nombres encadrés sont les distances des plus courts chemins de Québec à la ville considérée. On a indiqué la longueur du chemin entre deux villes, seulement si cet arc est parcouru dans la solution optimale.

Le temps moyen d'exécution de l'algorithme de Dijkstra est difficile à déterminer exactement. Envisageons alors le modèle suivant:

Pour chaque ville i, choisissons aléatoirement les villes j qui lui seront reliées sur le graphe: rangeons ces villes dans l'ordre des L_{ij} croissant (pour chaque ville i). On obtient une permutation aléatoire.

Par exemple, on choisit le nombre d'arcs issus de chaque ville selon une certaine loi de probabilité. On détermine ensuite aléatoirement les villes reliées entre elles par ces arcs. Supposons que les distances sont des variables aléatoires indépendantes ayant même densité de probabilité.

Dressons une liste de préférence pour chaque ville i, en rangeant les autres villes selon l'ordre croissant des L_{ij}. Ces listes sont analogues à celles de l'algorithme de couplage stable. On pourra ainsi étudier le temps moyen d'exécution de l'algorithme de Dijkstra, à l'aide des résultats obtenus dans l'étude du problème de la collection de coupons.

L'algorithme de Dijkstra peut être modifié de façon que l'opération $d_i \leftarrow \min(d_i, d_k + L_{ki})$ soit retardée jusqu'au moment où elle devient nécessaire. On obtient un algorithme de la forme suivante:

$A \leftarrow \{1\}$; $B \leftarrow \{2,\ldots,n\}$;

tant que B non vide faire

début choisir une ville i dans A, dont la liste est non vide;

$k \leftarrow$ première ville sur la liste de i;
effacer k de la liste de i;
si $k \in B$ alors faire passer k de B dans A

fin.

Remarquons l'analogie avec l'algorithme de couplage stable; la ville k joue le rôle de la femme à qui on fait une avance. Le critère de refus n'influence pas l'analyse qu'on veut faire. L'important est que la variable k soit choisie aléatoirement; ainsi l'analyse du problème de la collection de coupons s'applique à l'algorithme de Dijkstra de la même façon qu'au problème de couplage stable: le nombre moyen d'étapes, dans cette seconde formulation de l'algorithme de Dijkstra, est au plus nH_n. On peut montrer que l'algorithme, utilisé n fois pour trouver les plus courtes distances de la ville i à la ville j, pour tout i et j, demande un nombre moyen d'étapes de l'ordre de $O(n \log n)^2$.

EXERCICE 4.1. Combien de propositions sont faites dans l'algorithme de couplage stable quand tous les hommes ont exactement la même liste de préférence pour les femmes? (Les choix des femmes sont arbitraires.) Le couplage stable obtenu est-il unique?

EXERCICE 4.2. Combien de propositions sont faites en moyenne si la matrice de préférence des hommes est

$$\begin{array}{ll} A: & a\ b\ c\ d \\ B: & b\ c\ a\ d \\ C: & c\ a\ b\ d \\ D: & a\ b\ c\ d \end{array}$$

et si celle des femmes est aléatoire?

EXERCICE 4.3. Soit X une variable aléatoire discrète, et p_k la probabilité que $X = k$. La fonction génératrice de Dirichlet est

$$P(z) = \sum_{k\geq 1} \frac{p_k}{k^z}.$$

Remarquer que $P(0) = 1$.

(a) Exprimer $E(X)$, $V(X)$, et $E(\ln X)$ en fonction de $P(z)$ et de ses dérivées.

(b) Soit Y une variable aléatoire indépendante de X. Sa fonction génératrice de Dirichlet est $Q(z)$. Quelle est la fonction génératrice de Dirichlet du produit $X \cdot Y$?

Exprimer $E(XY)$ et $V(XY)$ en fonction de $E(X)$, $E(Y)$, $V(X)$ et $V(Y)$.

CINQUIÈME CONFÉRENCE

RECHERCHE DANS UNE TABLE PAR ADRESSAGE DISPERSÉ: COMPORTEMENT MOYEN DE L'ALGORITHME FONDAMENTAL

Adressage dispersé

Soit une unité d'information x dont on cherche l'emplacement dans une table. Une méthode conventionnelle consiste à ranger les informations de la table dans un certain ordre, puis à rechercher systématiquement l'unité d'information x. (Par exemple, chercher un nom dans le bottin téléphonique.)

Une technique plus efficace, appelée *hashing* en anglais et *hachage* ou *adressage dispersé* en français, est utilisée sur un ordinateur. Si l'on dispose de n mots-mémoire, où n est plus grand que le nombre de données, alors certains emplacements sont vides, d'autres contiennent les données. Supposons l'ensemble de toutes les données possibles extrêmement grand, beaucoup plus grand que le nombre de mots-mémoire n. (Il y a par exemple 26^6 noms de 6 lettres, mais seulement quelques noms doivent apparaître dans la table.) On associe à chaque unité d'information un nombre compris entre 1 et n, un nombre qui désigne son emplacement dans la table.

Dans ce but, on utilise une fonction $h(x)$ qui associe à un nom x une permutation de $[1,2,\ldots,n]$

$$x \to h(x) = h_1(x)\ h_2(x)\ \dots\ h_n(x).$$

En pratique, il est possible de procéder de telle sorte que chaque permutation obtenue soit aléatoire (avec la probabilité $1/n!$), indépendante des permutations associées avec les autres noms de la table.

Pour trouver x dans la table, on cherche dans les cases $h_1(x)$, $h_2(x)$,... jusqu'à ce qu'on trouve x (recherche couronnée de succès), ou jusqu'à ce qu'on arrive à une case vide (recherche vaine). Dans le second cas, on peut ranger le nom x dans cette case vide. Ainsi pourra-t-on le retrouver par la suite.

EXEMPLE. Ici $n = 9$ et la table contient six noms dont les permutations sont

$$\begin{array}{l}
x_1 \to \underline{3}\ 2\ 7\ 8\ 6\ 9\ 1\ 5\ 4 \\
x_2 \to \underline{1}\ 5\ 3\ 7\ 6\ 2\ 9\ 4\ 8 \\
x_3 \to \underline{4}\ 8\ 1\ 2\ 5\ 9\ 6\ 3\ 7 \\
x_4 \to \underline{1}\ \underline{5}\ 4\ 9\ 7\ 8\ 3\ 6\ 2 \\
x_5 \to \underline{9}\ 8\ 6\ 4\ 1\ 3\ 5\ 7\ 2 \\
x_6 \to \underline{4}\ \underline{1}\ \underline{7}\ 9\ 8\ 6\ 5\ 2\ 3
\end{array}$$

Après avoir inséré les six noms dans la table, on obtient

1	2	3	4	5	6	7	8	9
x_2		x_1	x_3	x_4		x_6		x_5

Si on cherche x_4 par exemple, on le trouve en deux étapes. On a besoin seulement des nombres soulignés dans chaque permutation. Si on cherche un emplacement pour un nom x_7 dont la

permutation est

$$x_7 \to 7\ 1\ 8\ \dots ,$$

on trouve après trois essais que x_7 n'est pas enregistré. Si on le place dans la case 8, on peut le retrouver en trois étapes.

Temps moyen de recherche d'une information

Quand on ajoute un $(m+1)$-ième nom à la table, quel est le nombre moyen d'essais avant de trouver une case vide où le placer? (Ceci donne le nombre moyen d'étapes pour le retrouver ensuite.)

On peut formuler cette question d'une manière équivalente: Quel est le nombre moyen d'étapes d'une recherche vaine, quand la table contient m noms?

Soit q_k la probabilité qu'un nombre d'étapes $\geq k$ soit nécessaire. On a

$$q_1 = 1$$

$$q_2 = \frac{m}{n}$$

$$q_3 = \frac{m}{n}\,\frac{m-1}{n-1}$$

$$q_4 = \frac{m}{n}\,\frac{m-1}{n-1}\,\frac{m-2}{n-2}$$

$$\dots$$

$$q_k = \frac{m!}{(m-k+1)!}\,\frac{(n-k+1)!}{n!}$$

$$= \binom{n-k+1}{n-m}\Big/\binom{n}{m}, \qquad 1 \leq k \leq m+1$$

où $\binom{n}{m} = \frac{n!}{m!\,(n-m)!}$ sont les coefficients binômiaux. La valeur moyenne du nombre d'étapes nécessaires pour placer le $(m+1)$-ième nom est

$$q_1 + q_2 + q_3 + \ldots$$

$$= \left[\binom{n}{n-m} + \binom{n-1}{n-m} + \ldots + \binom{n-m}{n-m}\right] / \binom{n}{m}$$

$$= \binom{n+1}{n-m+1} / \binom{n}{m} = \frac{n+1}{n+1-m} .$$

Si la table est à moitié remplie, alors deux étapes en moyenne sont suffisantes pour établir qu'un nom n'est pas dans la table.

La variance peut être calculée d'une manière analogue. On obtient

$$\frac{(n+1)m(n-m)}{(n+1-m)^2(n+2-m)} .$$

Lorsqu'il y a m noms dans la table, le nombre moyen d'étapes pour trouver un des noms présents dans la table (choisi au hasard) est

$$\frac{1}{m}\left(\frac{n+1}{n+1} + \frac{n+1}{n} + \frac{n+1}{n-1} + \ldots + \frac{n+1}{n+2-m}\right)$$

$$= \frac{n+1}{m}\left(H_{n+1} - H_{n+1-m}\right).$$

Soit $\alpha = \frac{m}{n+1}$, alors le nombre moyen d'étapes peut s'écrire quand $n \to \infty$

$$\frac{1}{\alpha} \ln \left(\frac{1}{1-\alpha}\right) + O\left(\frac{1}{n}\right).$$

Par exemple, lorsque la mémoire est remplie à 90%, le nombre moyen d'étapes est seulement de l'ordre de $\frac{10}{9} \ln 10 \approx 2.56$, même si m et n sont très grands ($m = 900\ 000$; $n = 1\ 000\ 000$).

<u>Liaison avec le problème de couplage</u>

Une borne supérieure pour le nombre moyen de propositions

a été obtenue. La véritable valeur peut en fait, être beaucoup plus petite.

Calculons le nombre moyen de propositions dans un cas particulier, quand les femmes ont toutes la même liste de préférence, et que les préférences des hommes sont aléatoires.

Le nombre total de propositions ne change pas quand on modifie la numérotation des hommes. Supposons que le prétendant X soit successivement le premier préféré des femmes, leur second préféré, etc. Chaque prétendant propose donc aux femmes jusqu'à ce qu'une nouvelle femme soit nommée. *Tout se passe comme dans l'adressage dispersé.*

Retournons à l'exemple précédent en supposant qu'il y ait 9 hommes et 9 femmes. Les six premiers hommes $(X_1, X_2, \ldots, X_6)$ sont déjà "casés": la case qu'ils occupent correspond au numéro de leur amie. Le 7-ième homme propose aux femmes 7 et 1, qui refusent ses avances (les cases 7 et 1 sont occupées). La femme 8, nommée pour la première fois, acceptera X_7.

On peut utiliser les résultats précédents. Le nombre moyen de propositions faites par le $(m+1)$-ième homme est $\frac{n+1}{n+1-m}$. Donc le nombre moyen total de propositions est:

$$(n+1)\left(\frac{1}{n+1} + \frac{1}{n} + \frac{1}{n-1} + \ldots + \frac{1}{2}\right)$$

$$= 1 + (n+1)(H_n - 1)$$

$$= (n+1)H_n - n.$$

Ce nombre est inférieur de $n - 2H_n + 1$ à la borne supérieure obtenue pour le modèle avec amnésie partielle. Nous sommes conduits à faire la conjecture suivante: Le nombre moyen de propositions, quand les préférences des femmes sont fixées et que celles des hommes sont aléatoires, est toujours $\geq (n+1)H_n - n$. En fait, étant donnée n'importe quelle matrice de préférence pour les femmes, la probabilité que le nombre de propositions faites par les hommes soit $\geq k$ est minimale quand les femmes ont la même liste de préférence. (Evident pour $k \leq n+2$; on peut le montrer pour $k = n+3$.)

Valeur asymptotique du nombre moyen de propositions dans l'algorithme fondamental

D'après la conjecture ci-dessus, on peut s'attendre à ce que le nombre moyen de propositions soit voisin de la borne supérieure $(n-1)H_n + 1$, plutôt que de notre hypothétique borne inférieure $(n+1)H_n - n$. En réalité, ceci peut être montré, et l'on arrive au résultat le plus important de cette série de conférences.

THEOREME. Etant données des matrices de préférence aléatoire pour les hommes et pour les femmes, le nombre moyen de propositions dans l'algorithme du couplage stable est

$$nH_n + O(\log n)^4.$$

Jusqu'ici nous avons procédé par dénombrement (approche combinatoire). Dans la démonstration du théorème fondamental, une approche probabiliste (aussi appelée approche hongroise puisque développée par Erdős, Rényi, etc.) est utilisée. On majorera la probabilité des cas peu fréquents, de façon à ne considérer que les situations les plus fréquentes.

Soit $M = 3n(\ln n)^2 - n$. On définit la condition

Condition C: Il y a au plus M propositions.

(Nous verrons plus loin comment M a été choisi.)

Soit p la probabilité que la Condition C soit satisfaite. Supposons que les hommes soient amnésiques. Lorsqu'on exécute l'algorithme fondamental, chaque femme classe un homme au

moment où ce dernier lui fait une première proposition. A ce moment elle classe le nouveau prétendant au hasard parmi les hommes qui lui ont déjà fait des avances. Soit N le nombre de vraies propositions et R le nombre de propositions redondantes. Les variables N et R sont aléatoires, mais ne sont pas indépendantes; quand N est grand, R tend à l'être. On connaît la valeur moyenne et la variance de $N + R$, à partir des résultats sur la collection de coupons.

La probabilité $1-p$ que $N > M$ est inférieure à la probabilité que $N + R > M$. Du théorème énoncé dans la quatrième conférence, on déduit que $1-p = O(1/n^2)$. Le nombre moyen $\bar{N}$ de véritables propositions est donc

$$\begin{aligned}\bar{N} &= p \cdot (\bar{N} \text{ quand C est vraie}) + (1-p) \cdot (\bar{N} \text{ quand C est fausse}) \\ &\leq (\bar{N} \text{ quand C est vraie}) + (1-p)n^2 \\ &= (\bar{N} \text{ quand C est vraie}) + O(1).\end{aligned}$$

Il reste à montrer que le nombre moyen de propositions quand C est vraie est $nH_n + O(\log n)^4$. Ceci résultera du fait que

$$\bar{R}\ (\text{quand C est vraie}) = O(\log n)^4,$$

car $\bar{N} + \bar{R} = nH_n$.

Soient N_A et R_A le nombre de propositions faites par A. La distribution de ces variables ne dépend pas de A; donc $\bar{N} = n\bar{N}_A$ et $\bar{R} = n\bar{R}_A$. Montrons que le nombre de propositions redondantes faites par un homme A est $O((\log n)^4/n)$.

Soit p_k la probabilité que $N_A = k$, quand la Condition C est vérifiée. Si $N_A = k$, la valeur moyenne de R_A est

$$f(k) = \frac{0}{n} + \frac{1}{n-1} + \dots + \frac{k-1}{n-(k-1)},$$

car le nombre moyen de propositions redondantes faites entre la m-ième et la $(m+1)$-ième vraie proposition est

$$\frac{m}{n} + \left(\frac{m}{n}\right)^2 + \ldots = \frac{m}{n-m}\,.$$

Le nombre moyen $\bar{R}_A$ de propositions redondantes faites par A, quand C est vraie, est donc:

$$\begin{aligned} \bar{R}_A &= \sum_{k\geq 1} p_k\, f(k) \\ &= (q_1-q_2)f(1) + (q_2-q_3)f(2) + (q_3-q_4)f(3) + \ldots \\ &= q_1 f(1) + q_2(f(2)-f(1)) + q_3(f(3)-f(2)) + \ldots \\ &= q_1\frac{0}{n} + q_2\frac{1}{n-1} + q_3\frac{2}{n-2} + \ldots + q_n\frac{n-1}{1}\,, \end{aligned}$$

où $q_k = p_k + p_{k+1} + \ldots$, et $q_{n+1} = 0$. Pour majorer $\bar{R}_A$, on va majorer q_k, la probabilité que $N_A \geq k$ quand C est vraie, par la probabilité que $N_A + R_A \geq k$ quand C est vraie:

$$q_k \leq (1-\varepsilon)^{k-1},$$

où $1-\varepsilon$ est une majoration de la probabilité qu'une avance de A à une femme aléatoire a soit rejetée, étant données toutes les propositions faites à a par les autres hommes. Autrement dit, la probabilité que a accepte A est au moins ε.

Si les n femmes reçoivent respectivement $m_1, m_2, \ldots, m_n$ avances des autres hommes pendant l'exécution de l'algorithme, la probabilité que A soit accepté par une femme aléatoire est

$$\frac{1}{n}\left(\frac{1}{m_1+1} + \frac{1}{m_2+1} + \ldots + \frac{1}{m_n+1}\right).$$

Le minimum de la fonction $\frac{1}{m_1+1} + \frac{1}{m_2+1} + \ldots + \frac{1}{m_n+1}$, sous la contrainte $m_1 + m_2 + \ldots + m_n \leq M$, est atteint quand $m_i = M/n$, pour $i=1,2,\ldots,n$. Donc la probabilité qu'une offre soit acceptée est au moins

$$\frac{1}{n}\left(\frac{n}{M+n} + \ldots + \frac{n}{M+n}\right) = \frac{n}{M+n} = \frac{1}{3(\ln n)^2}\,,$$

que nous prenons pour valeur du paramètre ε.

Récapitulation

Voici où nous en sommes: on veut montrer que $\bar{R}_A = O((\log n)^4/n)$, et on sait que

$$\bar{R}_A \leq \sum_{1\leq k\leq n} (1-\varepsilon)^{k-1}\frac{k-1}{n+1-k},$$

où $\varepsilon = 1/(3(\ln n)^2)$. Terminons la démonstration en utilisant une technique classique:

$$\begin{aligned} \bar{R}_A &\leq \sum_{1\leq k\leq n/2} \frac{(1-\varepsilon)^{k-1}k}{n/2} + \sum_{n/2<k\leq n} (1-\varepsilon)^{n/2} n \\ &\leq \frac{2}{n}\sum_{k\geq 1} (1-\varepsilon)^{k-1}k + \frac{n^2}{2}e^{-\varepsilon n/2} \\ &= \frac{2}{n\varepsilon^2} + \frac{n^2}{2}\exp(-n/(6(\ln n)^2)) = O((\log n)^4/n). \end{aligned}$$

La démonstration du théorème fondamental est donc achevée.

Résumé

Notre analyse du nombre moyen de propositions dans l'algorithme de couplage stable a permis d'illustrer les techniques fondamentales utilisées pour l'analyse des algorithmes:

1) Dénombrement et sommation discrète.
2) Application des fonctions génératrices.
3) Analyse combinatoire du cas le plus défavorable.
4) Approche probabiliste (aussi appelée hongroise).

Remarque finale au sujet de l'adressage dispersé

Dans la pratique, il est trop long de calculer une permutation $h_1(x), h_2(x), \ldots, h_n(x)$ vraiment aléatoire. On calcule $h(x)$ et $\delta(x)$, où $1 \le h(x) \le n$, $\delta(x)$ est premier avec n, et on utilise la permutation $h(x)$, $h(x) + \delta(x)$, $h(x) + 2\delta(x), \ldots$ (modulo n). Cette technique est appelée "*double hashing*" en anglais.

Bien que cette méthode n'utilise que des "progressions arithmétiques aléatoires", il a été démontré récemment par L. Guibas et E. Szemerédi que le comportement asymptotique est le même que si de véritables permutations aléatoires étaient utilisées. Ceci est un théorème difficile, obtenu par une application élaborée de l'approche probabiliste.

SIXIÈME CONFÉRENCE

IMPLANTATIONS ET VARIANTES DE L'ALGORITHME FONDAMENTAL

Après la description de l'algorithme fondamental, on écrit maintenant un *programme* qui pourra être soumis à l'ordinateur. Ce travail d'implantation nécessite, pour être efficace, de combiner diverses structures de l'information représentant les données.

Si l'on se reporte à l'algorithme fondamental de la deuxième conférence, on voit que les opérations à traduire sont:

[1] $x \leftarrow$ meilleur choix restant sur la liste de X.

[2] x préfère-t-il X à son fiancé?

[3] Fiancer X et x.

[4] Retirer x de la liste de X.

Dans l'opération [1], on cherche le premier élément x restant sur la liste de préférence de X. Dans l'opération [4], on supprime un élément de cette liste. Heureusement cet élément n'est pas choisi arbitrairement, c'est le premier de la liste. Choisissons ici une structure telle que ces opérations soient d'exécution facile. On représente la matrice des préférences des hommes directement par un tableau à deux dimensions

$$\begin{pmatrix} HC[1,1] & HC[1,2] & \dots & HC[1,n] \\ \vdots & \vdots & & \vdots \\ HC[n,1] & HC[n,2] & \dots & HC[n,n] \end{pmatrix},$$

où $HC[X,j]$ indique le nom de la j-ième préférée de X.

On utilise aussi un tableau unidimensionnel $H[1],\dots,H[n]$, où $H[X]$ est un pointeur indiquant la position du premier élément restant dans la liste de X; quand X est fiancé, il l'est à $HC[X,H[X]]$. Les opérations [1] et [4] s'écrivent alors comme suit:

[1] : $x \leftarrow HC[X,H[X]]$;

[4] : $H[X] \leftarrow H[X] + 1$.

L'opération [2] consiste à savoir si un nouvel homme X ne serait pas préférable au partenaire actuel de la femme x. On donne donc aux listes des femmes une représentation différente de celle des hommes (sinon l'opération nécessiterait une recherche trop longue).

Soit $P[x,X]$ le nombre servant à évaluer la préférence de x pour X; c'est-à-dire, $P[x,X] = n$ si X est son premier choix, 1 si c'est le pire, 0 si $X = \Omega$. Soit $F[x]$ le partenaire actuel de x. Alors l'opération [2] s'écrit:

[2] : $P[x,X] > P[x,F[x]]$.

L'opération [3] est facile à décrire. Nous sommes donc prêts à donner une version concrète de l'algorithme.

```
        k, X, x, t :  entier;
        HC :  tableau [1:n,1:n] de entier;
        P :  tableau [1:n,0:n] de entier;
        H,F :  tableau [1:n] de entier;
1.      entrée des données HC et P;
2.      pour X ← 1 à n faire H[X] ← 1;
3.      k ← 0; pour x ← 1 à n faire F[x] ← 0;
4.      tant que k < n faire
5.      début X ← k+1;
6.            tant que X ≠ 0 faire
7.            début
8.                x ← HC[X,H[X]];
9.                si P[x,X] > P[x,F[x]]
10.               alors début t ← F[x]; F[x] ← X;
11.                           X ← t;
12.                     fin;
13.               si X ≠ 0 alors H[X] ← H[X]+1
14.           fin;
15.           k ← k+1;
16.     fin; imprimer le couplage stable (HC[1,H[1]],...,
                                         HC[n,H[n]]).
```


Améliorations

A partir de cette première transcription, on essaie d'apporter des améliorations à la nouvelle version de l'algorithme en éliminant les instructions qui ne sont pas strictement nécessaires.

a) insérer l'opération $H[X] \leftarrow H[X]+1$ avant l'instruction 8. On évite ainsi de répéter le test: si $X \neq 0$. Il faut alors changer $H[X] \leftarrow 1$ à la ligne 2 en $H[X] \leftarrow 0$.

b) introduire un nouveau tableau $Q[x] \leftarrow P[x,F[x]]$, ce qui évite d'avoir à calculer $F[x]$ et $P[x,F[x]]$ pour chaque nouvelle proposition faite à la femme x. Naturellement $Q[x]$ doit être modifié chaque fois qu'on change $F[x]$, mais ceci n'arrive pas aussi souvent. (L'opération $x \leftarrow HC[X,H[X]]$ pourrait être modifiée de la même façon, mais cela n'est pas une amélioration.)

c) si le compilateur ne reconnaît pas les expressions communes, on peut changer la séquence

```
H[X] ← H[X]+1;
x ← HC[X,H[X]]
```

en

```
t ← H[X]+1;
H[X] ← t;
x ← HC[X,t].
```

Cette amélioration n'est pas aussi importante que les deux précédentes.

Initialisation du tableau P

Supposons que la liste de préférence d'une femme soit $(3,2,4,1) = FC[\mathbf{x},\cdot]$, alors $P[\mathbf{x},\cdot]$ prend les valeurs $(1,3,4,2)$; par exemple, 3 est le meilleur choix de $\mathbf{x}$, donc $P[\mathbf{x},3] = 4$. Pour faire cette affectation de valeurs il suffit d'utiliser les instructions suivantes

```
pour x ← 1 à n faire
pour k ← 1 à n faire P[x,FC[x,k]] ← n+1-k.
```

Si pour économiser la mémoire, on doit faire partager à FC et P le même tableau, on peut utiliser l'algorithme suivant (fondé sur la structure cyclique des permutations):

```
pour x ← 1 à n faire
début pour k ← 1 à n faire FC[x,k] ← -FC[x,k];
      pour k ← 1 à n faire
      si FC[x,k] < 0
      alors début j ← k; X ← -FC[x,k];
                  tant que X ≠ k faire
                  début t ← -FC[x,X];
                        FC[x,X] ← n+1-j;
                        j ← X; X ← t
```



```
                  fin;
                  FC[x,k] ← n+1-j;
            fin;
fin.
```

Maintenant $P[x,X] = FC[x,X]$.

Ecrivons à nouveau le programme après les améliorations a et b:

```
k,X,x,t :  entier
HC,P :  tableau [1:n,1:n] de entier;
H,F,Q :  tableau [1:n] de entier;
entrée des données HC et P;
pour X ← 1 à n faire H[X] ← 0;
k ← 0; pour x ← 1 à n faire F[x] ← Q[x] ← 0;
tant que k < n faire
début X ← k+1;
      tant que X ≠ 0 faire
      début H[X] ← H[X]+1;
            x ← HC[X,H[X]];
            si P[x,X] > Q[x]
            alors début t ← F[x]; F[x] ← X;
                        Q[x] ← P[x,X]; X ← t;
                  fin;
      fin;
      k ← k+1
fin; imprimer le couplage stable (HC[1,H[1]],...,
                                  HC[n,H[n]]).
```

Mariage forcé entre A et a

Modifions le programme afin de trouver une solution stable dans laquelle A et a sont mariés, et qui parmi ces solutions (s'il en existe) soit optimale pour les hommes.

D'après la théorie exposée dans les deux premières conférences, on utilise l'algorithme fondamental avec les n-1 hommes restant quand on supprime A, en excluant les deux situations suivantes:

[1] Pour toute femme b que A préfère à a, on interdit à b d'épouser un homme qu'elle ne préférerait pas à A.

[2] Pour tout homme B que a préfère à A, on interdit à B d'épouser une femme qu'il ne préférerait pas à a.

Pour éliminer la situation [1], on fait commencer l'algorithme en faisant faire à A une proposition (non sincère) aux femmes b de la situation [1]:

```
tant que HC[A,H[A]] ≠ a faire
début t ← HC[A,H[A]]; Q[t] ← P[t,A];
      H[A] ← H[A]+1;
fin;
```

Pour éliminer la situation [2], on interrompt l'algorithme si une proposition d'un homme tel que B est acceptée par a. On est conduit au programme modifié suivant:

```
k,X,x,t :  entier;
HC,P :  tableau [1:n,1:n] de entier;
H,F,Q :  tableau [1:n] de entier;
entrée des données HC et P;

pour X ← 1 à n faire H[X] ← 0;
k ← 0; pour x ← 1 à n faire F[x] ← Q[x] ← 0;
H[A] ← 1;
tant que HC[A,H[A]] ≠ a faire
début t ← HC[A,H[A]];  Q[t] ← P[t,A];
      H[A] ← H[A]+1;
fin;
F[a] ← A; Q[a] ← P[a,A];
```



```
tant que k < n faire
début X ← k+1;
      si X ≠ A alors
      tant que X ≠ 0 faire
      début H[X] ← H[X]+1;
            si H[X] > n alors aller à terminé;
            x ← HC[X,H[X]];
            si P[x,X] > Q[x]
            alors début t ← F[x]; F[x] ← X;
                        Q[x] ← P[x,X]; X ← t;
                        si X = A alors aller à terminé;
                  fin;
      fin;
      k ← k+1;
fin; imprimer le couplage stable (HC[1,H[1]],...,
                                  HC[n,H[n]]);
terminé:
```

Généralisation à plusieurs mariages forcés

Si l'on veut trouver le couplage stable optimal pour les hommes tel que les hommes d'un sous-ensemble S soient mariés avec des femmes qu'on leur assigne, il suffit de modifier quelque peu la séquence où l'on présente à A la femme a qui lui est destinée.

```
pour tout A ⊂ S faire
début a ← épouse destinée à A; H[A] ← 1;
      tant que HC[A,H[A]] ≠ a faire
      début t ← HC[A,H[A]];
            si Q[t] < P[t,A]
            alors début si F[t] ≠ 0 alors aller à terminé;
                        Q[t] ← P[t,A];
```

```
                fin;
                H[A] ← H[A]+1;
        fin;
        si Q[a] > P[a,A] alors aller à terminé;
        P[a] ← A; Q[a] ← P[a,A];
    fin;
```

On remplace aussi les instructions "si $X \neq A$" par "si $X \notin S$", et "si $X = A$" par "si $X \in S$".

Recherche d'un couplage stable équitable

Les différents algorithmes considérés jusqu'à présent favorisent les hommes; et si l'on y échange le rôle des hommes et des femmes, ils deviennent favorables aux femmes. Une telle injustice est trop choquante par les temps qui courent. Pouvons-nous donc trouver une solution qui traite les deux sexes d'une manière équitable?

On peut énumérer toutes les solutions stables et choisir le couplage le plus heureux selon un certain critère. Ceci risque d'être long si le nombre de couplages stables est grand. On ne sait pas en général s'il existe un grand nombre ou un petit nombre de solutions. Il semble donc préférable d'employer une autre méthode.

Choisissons par exemple le couple Aa au hasard. On exécute l'algorithme modifié donnant un couplage stable contenant Aa. S'il en existe un, alors on répète la même opération sur un problème réduit de dimension n-1, en tenant compte du fait que A est marié à a. S'il n'y a pas de couplage stable contenant Aa, alors on choisit un autre couple que Aa, et on essaie à nouveau.

Il est possible de développer cette approche de telle sorte que l'algorithme du couplage stable soit exécuté au plus

n^2 fois, c'est-à-dire avec un nombre moyen d'étapes polynomial (de l'ordre de $O(n^4)$, souvent $O(n^3 \log n)$).

Stan Selkow a inventé un algorithme moins aléatoire que le précédent et qui aboutit à une solution *optimale équitable* au sens qu'il minimise le regret de la personne la plus malheureuse. Pour un couplage M définissons le regret d'une personne par la distance de son conjoint au début de sa liste de préférence; soit $U(M)$ le maximum des regrets des $2n$ individus. Il s'agit de construire un couplage stable M qui minimise $U(M)$. L'algorithme est développé de la façon suivante: les solutions optimales pour les hommes et les femmes permettent d'obtenir des bornes supérieures et inférieures pour le regret de tous les individus dans les couplages stables; on répète les opérations suivantes: Former tous les mariages entre des individus dont les bornes supérieures et inférieures sont égales; on obtient ainsi une réduction du système. Quand on ne peut réduire davantage le système on choisit un individu au hasard parmi ceux dont les bornes supérieures sont les plus élevées. Sans nuire à la généralité supposons que cet individu soit la femme a, autrement le rôle des hommes et des femmes est permuté dans la description suivante. Soit A le plus mauvais choix de a parmi les hommes non éliminés; remarquons que a est le meilleur choix possible pour A parmi les femmes non éliminées. Supprimons a de la liste de A et déterminons une nouvelle solution optimale pour les hommes: ceci élève la borne inférieure de regret de A et de même pour au moins un autre homme. Simultanément la borne supérieure du regret de a et aussi, d'au moins une autre femme est diminuée; il est clair que ces opérations itératives réduisent le système à un couplage stable.

Recherche exhaustive des couplages stables

On décrit ici une *procédure récursive toutm* qui dépend d'un paramètre entier, j, entre 0 et n. Au début de la procédure, $(H[1],H[2],\ldots,H[n])$ désigne un couplage stable. A la fin, on redonne aux variables de ce tableau leur valeur initiale. L'algorithme imprime tous les couplages stables $(H'[1],H'[2],\ldots,H'[n])$ tels que:

$$H'[k] = H[k], \text{ pour } 1 \le k \le j$$
$$H'[k] \ge H[k], \text{ pour } j < k \le n.$$

(Aucun homme ne fait un meilleur choix, et les hommes de 1 à j ne changent pas de partenaire.)

Pour obtenir toutes les solutions de notre problème, on exécute d'abord l'algorithme fondamental. Mais au moment d'imprimer la solution optimale pour les hommes, on utilise *toutm*(0) à la place.

Le programme est écrit ci-dessous:

```
procédure toutm (j: entier);
x,X,y,t : entier;
SH,SF,SQ : tableau [1:n] de entier;
si j = n alors imprimer le couplage stable (H[1],...,
                                            H[n])
sinon début pour t ← 1 à n faire
            début SH[t] ← H[t]; SF[t] ← F[t];
                  SQ[t] ← Q[t];
            fin;
continuer : toutm(j+1);
changer :  X ← j+1;
           y ← HC[X,H[X]]; {cette partenaire sera
                                interdite à X}
proposer : H[X] ← H[X]+1;
           si H[X] > n alors aller à terminer;
           x ← HC[X,H[X]];
           si F[x,X] > Q[x]
           alors début t ← F[x];
                       si t ≤ j alors aller à terminer;
                       F[x] ← X; Q[x] ← F[x,X];
                       si x = y alors aller à continuer;
                       X ← t
                 fin;
           aller à proposer;
terminer : pour t ← 1 à n faire
           début H[t] ← SH[t]; F[t] ← SF[t];
                 Q[t] ← SQ[t]
           fin;
      fin.
```


Il est plus difficile ici de prouver la validité de cet algorithme, mais l'approche reste la même qu'auparavant. Par récurrence sur l'algorithme, on suppose que les exécutions de "*toutm*$(j+1)$" sont correctes.

Puis, on montre qu'avant d'exécuter l'opération "$H[X] \leftarrow H[X]+1$", tous les couplages stables désirés où X est accouplé avec son choix $H[X]$, ont déjà été imprimés. (Ainsi trouve-t-on tous les couplages stables.)

Enfin, tous les couplages obtenus sont stables. Soit un homme A préférant a à sa partenaire, alors a préfère son ami à A, car la qualité des compagnons des femmes s'améliore au cours de l'exécution de l'algorithme.

Passage à un algorithme non récursif

Cet algorithme peut être rendu non récursif à l'aide de quelques changements:

```
début
   x,X,y,t,j :  entier;
   SH,SF,SQ :  tableau [0:n,1:n] de entier;
   HC,P :  tableau [1:n,1:n] de entier;
   H,F,Q :  tableau [1:n] de entier;
           j ← 0;
   toutm :  si j = n alors imprimer le couplage stable (H[1],...,
                                                          H[n])
           sinon début pour t ← 1 à n faire
                       début SH[j,t] ← H[t]; SF[j,t] ← F[t];
                             SQ[j,t] ← Q[t];
                       fin;
   continuer :         j ← j+1; aller à toutm;
   changer :           {même séquence d'instructions}
   proposer :          {même séquence d'instructions}
   terminer :          pour t ← 1 à n faire
                       début H[t] ← SH[j,t]; F[t] ← SF[j,t];
                             Q[t] ← SQ[j,t]
                       fin;
                 fin;
           j ← j-1; si j ≥ 0 alors aller à changer;
fin.
```

On peut montrer que le temps d'exécution de ce programme est $O(n^2 S)$ lorsqu'il y a S couplages stables.

SEPTIÈME CONFÉRENCE

PROBLÈMES DE RECHERCHE

Les questions suivantes sont loin d'être du même niveau que les fameux problèmes présentés par Hilbert dans sa célèbre conférence à Paris en 1900, mais elles semblent dignes d'intérêt, et susceptibles d'être résolues en un temps fini.

PROBLEME 1. Etudier le nombre moyen de changements de partenaire d'une femme au cours de l'algorithme fondamental. (Une femme x change d'ami chaque fois qu'on modifie $F[x]$.)

Ce nombre est-il très inférieur au nombre moyen de propositions? Il est assez naturel de supposer que ce nombre est d'ordre $n \log \log n$. En effet une femme accepte toujours la première offre, accepte la seconde offre avec probabilité $\frac{1}{2}$,... la k-ième offre avec probabilité $\frac{1}{k}$. Si cette femme reçoit k propositions, elle change d'ami en moyenne $1 + 1/2 + 1/3 + \ldots + 1/k = H_k$ fois. On sait que cette femme reçoit en moyenne H_n propositions donc le nombre moyen de changements de partenaire est d'ordre $\log \log n$.

PROBLEME 2. Au cours de la cinquième conférence, on a émis une conjecture sur une borne inférieure du nombre moyen de propositions $\bar{N}$, lorsque la matrice de préférence des femmes est donnée. Cette conjecture est-elle exacte?

PROBLEME 3. Si la matrice de préférence des hommes est donnée et si celle des femmes est aléatoire, le nombre moyen de propositions est-il maximal lorsque tous les hommes ont la même liste de préférence pour les femmes?

PROBLEME 4. Y-a-t-il une manière efficace de calculer exactement le nombre moyen de propositions si la matrice de préférence des femmes est donnée? En d'autres termes, peut-on trouver d'une façon efficace ce nombre exact en étudiant la structure de la matrice de préférence des femmes? Peut-être ce nombre dépend-il seulement de propriétés simples des matrices de préférence, par exemple combien de fois deux femmes classent deux hommes différemment? On obtiendrait ainsi élégamment la solution de l'exercice 3.1.

PROBLEME 5. Trouver les matrices de préférence d'ordre n des hommes et des femmes qui maximisent le nombre de couplages stables.

Pour $n = 4$ par exemple, le nombre maximal connu est 10. Cette situation se présente pour les deux matrices de préférence suivantes:

A :	a	b	c	d	a :	D	C	B	A
B :	b	a	d	c	b :	C	D	A	B
C :	c	d	a	b	c :	B	A	D	C
D :	d	c	b	a	d :	A	B	C	D

où tous les couplages stables contiennent chacun A, B, C et D mariés respectivement à

a	b	c	d
b	a	c	d
a	b	d	c
b	a	d	c
b	d	a	c
c	a	d	b
c	d	a	b
c	d	b	a
d	c	a	b
d	c	b	a.

PROBLEME 6. Trouver une méthode pour décrire la structure de l'ensemble des couplages stables (pour des matrices de préférence données), de telle sorte qu'on puisse caractériser les solutions sans avoir besoin de les énumérer.

Remarques sur ce problème

La théorie suivante, due à John Conway, montre la présence de telles structures, mais bien des questions restent en suspens.

THEOREME. Si $M = (Aa, Bb, \ldots, Zz)$ et $M' = (Aa', Bb', \ldots, Zz')$ sont deux couplages stables, alors $M \vee M' = (A \max_A(a,a'),$ $B \max_B(b,b'), \ldots, Z \max_Z(z,z'))$ est aussi un couplage stable. (Ici $\max_A(a,a')$ représente la préférée de A parmi $\{a,a'\}$.)

DEMONSTRATION. Montrons d'abord que $M \vee M'$ est un couplage, c'est-à-dire que les mariages sont monogames.

Supposons par exemple que

$$a = \max_A(a,a') = \max_B(b,b') = b'.$$

Alors B est marié avec a dans M'. Comme M' est un couplage stable et que aAa', on a BaA. Mais M est stable, donc on a aussi bBa, ce qui contredit le fait que $\max_B(b,a) = a$.

Enfin, le couplage $M \vee M'$ est stable. Supposons en effet que A et $\max_B(b,b') = b$, par exemple, préfèrent être ensemble. Alors A préfère b à $\max_A(a,a')$, donc bAa. Mais AbB, ce qui contredit le fait que M est stable.

COROLLAIRE 1. Avec les hypothèses du théorème précédent, le couplage

$$M \wedge M' = (A \min_A(a,a'), B \min_B(b,b'), \ldots, Z \min_Z(z,z'))$$

est aussi un couplage stable.

DEMONSTRATION. Il suffit de permuter les rôles des hommes et des femmes, et d'appliquer le théorème de la seconde conférence.

Les opérations $\vee$ ("max") et $\wedge$ ("min") sont associatives, commutatives, idempotentes et distributives:

$$(M \wedge M') \wedge M'' = M \wedge (M' \wedge M''),$$

$$(M \wedge M') \vee M'' = (M \vee M'') \wedge (M' \vee M''),$$

$$M \wedge M = M, \text{ etc.}$$

Il en résulte:

COROLLAIRE 2. Etant données des matrices de préférence pour les hommes et pour les femmes, l'ensemble des couplages stables est un treillis distributif.

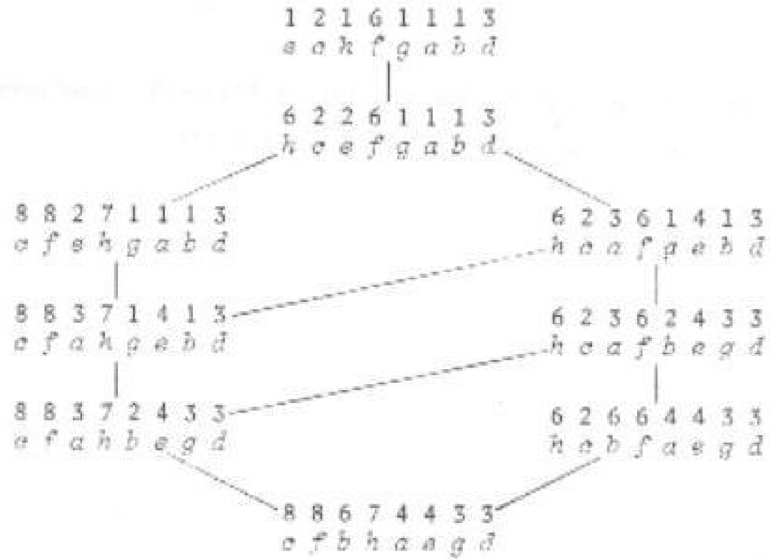
La solution optimale pour les hommes est le "max" de tous les couplages stables, et la solution optimale pour les femmes est le "min".

Etudions le treillis des couplages stables sur quelques exemples. On trouve dans la littérature l'exemple suivant pour 8 hommes et 8 femmes:

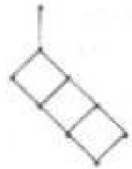
A :	e	g	a	b	f	h	d	c
B :	b	c	g	e	d	a	h	f
C :	h	e	a	d	f	b	c	g
D :	c	b	g	d	a	f	h	e
E :	g	b	e	a	c	f	h	d
F :	a	f	g	e	h	d	b	c
G :	b	e	g	f	c	d	h	a
H :	c	h	d	e	g	b	f	a

a :	E	C	G	F	A	B	H	D
b :	H	F	C	E	G	B	A	D
c :	A	E	F	B	D	H	G	C
d :	H	G	C	B	D	A	E	F
e :	F	D	G	C	H	A	B	E
f :	B	H	E	D	F	C	G	A
g :	G	E	B	A	H	F	D	C
h :	G	D	A	E	B	C	F	H.

Les neuf couplages stables peuvent être disposés selon le treillis ci-dessous. (Un couplage est représenté par les noms des partenaires de A, B, C, D, E, F, G et H; le nombre au-dessus de chaque nom correspond au rang de la femme sur la liste du conjoint.)

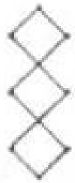


On peut représenter cette structure par le treillis distributif suivant:



Si l'on parcourt le treillis de haut en bas on s'aperçoit que le rang de préférence que les hommes donnent à leurs épouses est croissant.

La représentation par un treillis des dix couplages stables du problème 5 est



Le treillis des n couplages stables de l'exemple avec permutation circulaire de la première conférence est

$$\begin{array}{ccccc} a & b & c & d & e \\ b & c & d & e & a \\ c & d & e & a & b \\ d & e & a & b & c \\ e & a & b & c & d. \end{array}$$

Voici un exemple où le treillis est une ligne droite, mais où les hommes ne changent pas tous de partenaires d'un couplage à l'autre:

A	:	a c b				a	:	B A C
B	:	b c a				b	:	C B A
C	:	c b a				c	:	A B C

$$\begin{array}{ccc} a & b & c \\ a & c & b \\ c & a & b. \end{array}$$

Voici un exemple où la structure du treillis est plus "riche":

A :	a	b	c	d	e	f		a :	E	F	D	B	C	A
B :	b	c	a	e	f	d		b :	F	D	E	C	A	B
C :	c	a	b	f	d	e		c :	D	E	F	A	B	C
D :	d	e	f	a	b	c		d :	B	C	A	E	F	D
E :	e	f	d	b	c	a		e :	C	A	B	F	D	E
F :	f	d	e	c	a	b		f :	A	B	C	D	E	F

On vérifie facilement que

$$\begin{array}{ll} Aa \to Cc \to Bb \to Aa, & Dd \to Ff \to Ee \to Dd, \\ Ab \to Ca \to Bc \to Ab, & De \to Fd \to Ef \to De, \\ Af \to Ce \to Bd \to Af, & Dc \to Fb \to Ea \to Dc, \\ Ae \to Cd \to Bf \to Ae, & Db \to Fa \to Ec \to Db. \end{array}$$

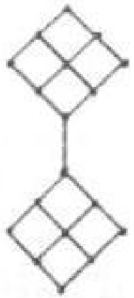
De plus,

$$Ac \to Ff,\ Fd, \text{ ou } Fe;$$
$$Ac, Ff \to Ee, Dd, Ba, Cb;$$
$$Ac, Fd \to Ef, De, Ba, Cb;$$
$$Ac, Fe \to Ba, Df, Cb, Ed.$$

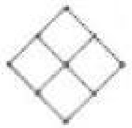
Ainsi

$$Ac \to Ba \to Cb \to Ac,$$

et les couplages stables ont la structure suivante:



Ceci peut être interprété comme la *somme directe* de | et de deux treillis

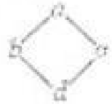
 ,

dont chacun est le *produit direct* des deux treillis

Peut-on obtenir ainsi tous les treillis distributifs? Quelles propriétés particulières ont les treillis associés aux carrés latins duaux, comme dans l'exercice 1.3? (La plupart des exemples ci-dessus sont de ce type.)

On pourrait vouloir généraliser le problème de la façon suivante: Au lieu d'avoir des ensembles d'hommes et de femmes totalement ordonnés, on aurait simplement un ordre de treillis. Mais cela ne marche pas très bien:

Supposons que tous les hommes classent les femmes selon l'ordre du treillis



,

et que les femmes classent les hommes selon le même treillis:

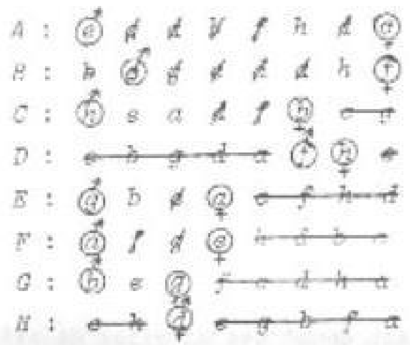
.

Il y a deux couplages stables (Aa, Bb, Cc, Dd) et (Aa, Bc, Cb, Dd). Mais $M \vee M' = (Aa, Ba, Ca, Dd)$ n'est plus un couplage et on n'a donc pas une structure de treillis pour les couplages stables.

Intersection des intervalles

Si on utilise l'algorithme fondamental pour trouver la solution optimale pour les hommes et la solution optimale pour les femmes, on obtient des "intervalles" de choix pour chaque homme et pour chaque femme. Dans tous les couplages stables, chaque homme est marié à une femme de son intervalle; de même chaque femme est mariée à un homme de son intervalle.

Considérons l'exemple des 8 hommes et des 8 femmes donné précédemment.



a	:	*E*	*C*	*G*	*F*	*A*	*B*	*H*	*D*
b	:	*H*	*F*	*C*	*E*	*G*	*B*	*A*	*D*
c	:	*A*	*E*	*F*	*B*	*D*	*H*	*G*	*C*
d	:	*H*	*G*	*C*	*B*	*D*	*A*	*E*	*F*
e	:	*E*	*D*	*G*	*C*	*H*	*A*	*B*	*F*
f	:	*B*	*H*	*E*	*D*	*F*	*C*	*G*	*A*
g	:	*G*	*E*	*B*	*A*	*H*	*F*	*D*	*C*
h	:	*G*	*D*	*A*	*E*	*B*	*C*	*F*	*H*

Ici ♂ représente la solution optimale pour les hommes, ♀ la solution optimale pour les femmes, " — " un mariage exclus car la femme est en dehors de l'intervalle de l'homme et finalement " / " un mariage exclus car l'homme est en dehors de l'intervalle de la femme.

Par exemple, l'homme *A* ne peut épouser la femme *b* parce qu'il n'est pas dans l'intervalle de *b*.

Cette observation améliore considérablement la vitesse de l'algorithme de recherche exhaustive et des algorithmes "équitables" présentés à la sixième conférence. Cependant dans certains cas exceptionnels, une telle observation n'a pas d'effets (quand la solution optimale pour les hommes leur donne le premier choix tandis que les femmes obtiennent leur dernier choix, et quand la solution optimale pour les femmes leur donne le premier choix, tandis que les hommes obtiennent leur dernier choix).

L'intersection des intervalles est une condition nécessaire mais non suffisante de stabilité. Dans l'exemple ci-dessus, il n'y a pas de couplage stable contenant *Bh*, mais le couplage *Ac Bh Ce Df Eb Fa Gg Hd* se trouve dans l'intersection des intervalles.

<u>PROBLEME 7</u>. Etant donné des matrices de préférence aléatoires,

trouver le nombre moyen asymptotique de couplages stables. Si $n = 2$, ce nombre est $1 + 1/8$; et si $n = 3$, on a $1 + \frac{1139}{3888}$. Dans le cas général, ce nombre est $n!p_n$ où p_n est la probabilité que $A_1a_1, A_2a_2, \ldots, A_na_n$ soit stable.

(1) $$p_n = \frac{1}{n!^n} \sum_{(x_{i,j})} \prod_{1 \le j \le n} \frac{\ell_j!\,(n-1-\ell_j)!}{1+c_j}$$

(2) $$p_n = \sum_{(x_{i,j})} \prod_{1 \le j \le n} \frac{(-1)^{\ell_j}}{(1+\ell_j)(1+c_j)}$$

où, pour ces deux cas, la somme se fait sur les $2^{n(n-1)}$ matrices $(x_{i,j})$ où $x_{i,j} = 0$ ou 1, et où $x_{i,i} = 0$. Ici $\ell_i = \sum_j x_{i,j}$ est la somme de la ligne i, et $c_j = \sum_i x_{i,j}$ est la somme de la colonne j.

On a trouvé la formule (1) en comptant les matrices de préférence telles que $A_1a_1, \ldots, A_na_n$ soit stable et que $a_jA_ia_i \iff x_{i,j} = 1$. Pour obtenir la formule (2), on a posé $x_{i,j} = 1 \iff a_jA_ia_i$ et $A_ia_jA_j$ (cas d'instabilité). On additionne toutes les matrices, puis on soustrait celles avec un cas d'instabilité, puis on additionne celles avec deux cas d'instabilité, etc.; on obtient ainsi la probabilité que $A_1a_1, \ldots, A_na_n$ soit stable. (Il serait intéressant de prouver que (1) = (2) à l'aide de manipulations algébriques.)

Voici une autre formule pour p_n:

(3) $$\int_0^1\!\int_0^1 \cdots \int_0^1 dx_1\,dx_2 \ldots dx_n\,dy_1\,dy_2 \ldots dy_n \prod_{\substack{1 \le i,j \le n \\ i \ne j}} (1 - x_iy_j).$$

Par exemple,

$$P_3 = \int_0^1\int_0^1\int_0^1\int_0^1\int_0^1\int_0^1 da\, db\, dc\, dA\, dB\, dC\ (1-Ab)(1-Ac)(1-Ba)(1-Bc)$$
$$\times (1-Ca)(1-Cb).$$

DEMONSTRATION. On remarque qu'on peut exprimer le produit

$$\prod_{\substack{1\le i,j\le n\\ i\ne j}} (1-x_i y_j) = \sum_{(e_{ij})} \prod_{1\le j\le n} (-x_j)^{\ell_j} y_j^{c_j}$$

et l'on obtient la formule (2). Le comportement asymptotique de ces trois formules semble plutôt délicat. On sait que $n!p_n$ doit être ≥ 1 car il existe toujours un couplage stable; mais ceci ne découle même pas d'une façon évidente des formules (1), (2), (3).

PROBLEME 8. Etant donné des matrices de préférence pour les hommes et les femmes, construisons un "graphe orienté pour les divorces". Ce graphe aura $n!$ sommets, un pour chaque permutation $p_1,\ldots,p_n$ de $\{1,2,\ldots,n\}$.

Si $(A_1a_{p_1},\ldots,A_na_{p_n})$ est un couplage instable dans lequel A_i et a_{p_j} se préfèrent mutuellement à leurs époux respectifs $(i\ne j)$, alors le graphe des divorces contient un arc du sommet $(p_1,\ldots,p_n)$ à $(q_1,\ldots,q_n)$, où $(q_1,\ldots,q_n)$ est la permutation $(p_1,\ldots,p_n)$ avec p_i et p_j interchangés. Les sommets desquels ne part aucun arc représentent les couplages stables.

Existe-t-il un chemin partant de n'importe quel point et menant vers un couplage stable? Si oui, y-en-a-t-il un qui serait assez petit?

Ce graphe orienté aurait peut-être des propriétés intéressantes le mettant en relation avec la structure en treillis des couplages stables.

PROBLEME 9. Existe-t-il un algorithme donnant un couplage

stable, où le nombre d'opérations croît moins vite que n^2 dans le pire cas? (On ne compte pas le temps d'entrée des matrices de préférence des femmes et des hommes.)

PROBLEME 10. Existe-t-il un lien intéressant entre le problème du couplage stable et le problème d'attribution? (Etant donnée une matrice a_{ij}, trouver une permutation maximisant $\sum_{1\le j\le n} a_{jp_j}$.)

PROBLEME 11. Peut-on généraliser le problème du couplage stable à trois ensembles d'objets (par exemple hommes, femmes, chiens)?

PROBLEME 12. (Un problème unisexe: stabilité des compagnons de chambre.)

Chaque individu d'un ensemble de $2n$ personnes classe les $2n-1$ autres selon un certain ordre de préférence. Trouver un algorithme efficace (temps d'exécution d'ordre polynômial dans le pire cas) permettant d'obtenir, si c'est possible, un ensemble stable de n couples. (Stable signifie ici qu'il n'y a pas deux personnes séparées qui voudraient toutes les deux se retrouver.)

Par exemple, on peut vérifier que le système de préférences

$$\begin{array}{ccccccccc}
A & : & B & E & D & F & G & H & C\\
B & : & C & F & A & G & H & E & D\\
C & : & D & G & B & H & E & F & A\\
D & : & A & H & C & E & F & G & B\\
E & : & F & A & H & B & C & D & G\\
F & : & G & B & E & C & D & A & H\\
G & : & H & C & F & D & A & B & E\\
H & : & E & D & G & A & B & C & F
\end{array}$$

n'a que trois solutions stables: $\{AB,CD,FG,HE\}$, $\{BC,DA,EF,GH\}$, $\{AE,BF,CG,DH\}$.

Quelquefois il n'y a pas de solution stable:

A : B C D
B : C A D
C : A B D
D : arbitraire.

(Toute personne associée avec D voudra changer.) Peut-être pourra-t-on montrer qu'un tel problème est NP-complet.

Résumé des conférences

Le problème des couplages stables nous a introduits aux méthodes de l'analyse des algorithmes. On a fait appel à des notions tirées de domaines divers des mathématiques:

analyse combinatoire,
théorie des probabilités,
analyse,
algèbre,

en plus de sujets purement informatiques:

structures de données,
structures de contrôle,
complexité du calcul.

Il reste beaucoup de problèmes passionnants à explorer.

SOLUTIONS DES EXERCICES

1.1. (a) Si le couplage stable $(A_j a_j, A_k b, \ldots)$ donne le meilleur choix de A_j, et $(A_k a_j, A_j c, \ldots)$ est optimal pour A_k, on a $a_j A_k b$, d'où $A_j a_j A_k$; d'autre part $a_j A_j c$, d'où $A_k a_j A_j$. Contradiction.

(b) Supposons $a_k A_j a_j$ et $A_j a_k A_k$, une situation d'instabilité. Soit $(A_k a_k, A_j b, \ldots)$ un couplage stable optimal pour A_k. Alors $a_j A_j b$, par définition de a_j (puisque $b \neq a_j$). Par transitivité on a $a_k A_j b$, et $A_j a_k A_k$ contredit la stabilité.

1.2. $(Ae,\ Ba,\ Cb,\ Dc,\ Ed)$ et $(Aa,\ Bb,\ Cc,\ Dd,\ Ee.)$

2.1 et 2.2. Cf. quatrième conférence, troisième partie.

3.2. Sans perte de généralité, on peut supposer AdC. La probabilité d'avoir 0, 1, 2, et 3 propositions redondantes est respectivement 1/9, 7/18, 1/4, et 1/4. Par exemple, il n'y a pas de proposition redondante lorsque BbC, DbB, et CaD (probabilité: 1/12), ou lorsque CbB, DbC, BaD, et CaB (probabilité: 1/36).

4.1. Soit $a\ b\ c \ldots$ la liste de préférence commune. La femme a obtient son meilleur choix, disons A. Les autres hommes font des propositions à b, qui obtient son meilleur choix autre que A, disons B. Les hommes restants font des propositions à c, qui obtient son meilleur choix autre que A, B. Et ainsi de suite. Le nombre total de propositions est $n + (n-1) + \ldots + 1 = n(n+1)/2$. La solution est unique, puisqu'elle est optimale pour les hommes et pour les femmes. (Le nombre moyen de propositions serait inférieur à nH_n si les femmes avaient fait les avances.)

4.2. L'ordre des propositions est toujours a, b, c, a, b, c, ... jusqu'à ce qu'un homme demande d. La probabilité d'avoir 7, 8, 9, 10, 11, 12, et 13 propositions est respectivement 1/4, 1/6, 2/9, 25/108, 1/12, 1/36, et 1/54. La moyenne est donc 8 + 8/9. (Voir le problème 3, septième conférence.)

4.3. (a) $E(X) = P(-1)$, $V(X) = P(-2) - P(-1)^2$, $E(\ln X) = -P'(0)$.
(b) La fonction génératrice est $P(s)Q(s)$. $E(XY) = E(X)E(Y)$, $V(XY) = (V(X)+E(X)^2)(V(Y)+E(Y)^2) - E(X)^2E(Y)^2$.

BIBLIOGRAPHIE

1. Couplages stables

D. GALE et L.S. SHAPLEY, "College admissions and the stability of marriage", *Amer. Math. Monthly* 69 (1962), 9-15.

[Le premier article consacré au sujet.]

D.G. McVITIE et L.B. WILSON, "Stable marriage assignment for unequal sets", *BIT* 10 (1970), 295-309.

D.G. McVITIE et L.B. WILSON, "The application of the stable marriage assignment to university admissions", *Operational Research Quarterly* 21 (1970), 425-433.

D.G. McVITIE et L.B. WILSON, "The stable marriage problem", *Communications of the ACM* 14 (July 1971), 486-492.

[Algorithmes pour la construction de tous les couplages stables.]

L.B. WILSON, "An analysis of the stable marriage assignment algorithm", *BIT* 12 (1972), 569-575.

[Cet article détermine la borne supérieure du problème de l'amnésie partielle.]

T.H.F. BRISSENDEN, "Some derivations from the marriage bureau problem", *Math. Gazette* 58 (1974), 250-257.

[Le lecteur corrigera quelques erreurs.]

L.E. DUBINS et D.A. FREEDMAN, "Machiavelli and the Gale-Shapley algorithm," Amer. Math. Monthly 88

JUAN BULNES et JACOBO VALDES, *Notes on the Complexity of the Stable Marriage Problem*, manuscrit inédit, Stanford University, 1972.

[Etude du cas le plus défavorable.]

2. Collection de coupons

R.E. GREENWOOD, "Coupon collector's test for random digits", *Math. Tables Aids Comput.* 9 (1955), 1-4.

D.E. KNUTH, *Seminumerical Algorithms: The Art of Computer Programming* 2 (Reading, Mass.: Addison-Wesley, 1969), Section 3.3.2E, exercices 7-10.

3. Problème du plus court chemin

E.W. DIJKSTRA, "A note on two problems in connection with graphs", *Numer. Math.* 1 (1959), 269-271.

P.M. SPIRA, "A new algorithm for finding all shortest paths in a graph of positive arcs in average time $O(n^2 \log^2 n)$", *SIAM J. Computing* 2 (1973), 28-32.

4. Technique d'adressage dispersé

W.W. PETERSON, "Addressing for random-access storage", *IBM J. Research and Devel.* 1 (1957), 130-146.

J.D. ULLMAN, "A note on the efficiency of hashing functions", *Journal of the ACM* 19 (1972), 569-575.

D.E. KNUTH, *Sorting and Searching: The Art of Computer Programming* 3 (Reading, Mass.: Addison-Wesley, 1973), Section 6.4.

L.J. GUIBAS, *The Analysis of Hashing Algorithms*, Stanford University, Computer Science Dept. Ph.D. Thesis, 1976.

5. Structures des données et structures de contrôle

D.E. KNUTH, *Fundamental Algorithms: The Art of Computer Programming* 1 (Reading, Mass.: Addison-Wesley, 1968).

[Pour l'analyse du problème de l'horloge voir en particulier les exercices 2.3.4.2-16 et 17.]

D.E. KNUTH, "Structured programming with go to statements", *Computing Surveys* 6 (1974), 261-301.

6. Analyses des algorithmes

D.E. KNUTH, "The analysis of algorithms", *Actes du Congrès International des Mathématiciens 1970*, 3 (Paris: Gauthier-Villars, 1971), 269-274.

D.E. KNUTH, "Mathematical analysis of algorithms", *Proc. IFIP Congress 1971*, 1 (Amsterdam: North-Holland, 1972), 19-27.

R. SEDGEWICK, *Quicksort*, Computer Science Dept., Stanford University, Computer Science Dept., Ph.D. thesis, 1975.

TABLE DES MATIÈRES

35. **Nonparametric Estimation.** Constance VAN EEDEN
36. **K-Théorie.** Max KAROUBI
37. **Differential Complexes.** Joseph J. KOHN
38. **Variétés hilbertiennes : aspects géométriques.** Nicolaas H. KUIPER
39. **Deformations of Compact Complex Manifolds.** Masatake KURANISHI
40. **Grauert's Theorem on Direct Images of Coherent Sheaves.** Raghavan NARASIMHAN
41. **Systems of Linear Partial Differential Equations and Deformation of Pseudogroup Structures.** A. KUMPERA et D.C. SPENCER
42. **Analyse globale.** P. LIBERMANN, K.D. ELWORTHY, N. MOULIS, K.K. MUKHERJEA, N. PRAKASH, G. LUSZTIG et W. SHIH
43. **Algebraic Space Curves.** Shreeram S. ABHYANKAR
44. **Théorèmes de représentabilité pour les espaces algébriques.** Michael ARTIN
45. **Groupes de Barsotti-Tate et cristaux de Dieudonné.** Alexandre GROTHENDIECK
46. **On Flat Extensions of a Ring.** Masayoshi NAGATA
47. **Introduction à la théorie des sites et son application à la construction des préschémas quotients.** Masayoshi MIYANISH
48. **Méthodes logiques en géométrie diophantienne.** Shuichi TAKAHASHI
49. **Index Theorems of Atiyah — Bott — Patodi and Curvature Invariants.** Ravindra S. KULKARNI
50. **Numerical Methods in Statistical Hydrodynamics.** Alexandre CHORIN
51. **Introduction à la théorie des hypergraphes.** Claude BERGE
52. **Automath, a Language for Mathematics.** Nicolaas G. DE BRUIJN
53. **Logique des topos (Introduction à la théorie des topos élémentaires).** Dana SCHLOMIUK
54. **La Série génératrice exponentielle dans les problèmes d'énumération.** Dominique FOATA
55. **Feuilletages : résultats anciens et nouveaux (Painlevé, Hector et Martinet).** Georges H. REEB
56. **Finite Embedding Theorems for Partial Designs and Algebras.** Charles C. LINDNER et Trevor EVANS
57. **Minimal Varieties in Real and Complex Geometry.** H. Blaine LAWSON, Jr
58. **La Théorie des points fixes et ses applications à l'analyse.** Kazimierz GEBA, Karol BORSUK, Andrzej JANKOWSKI et Edward ZHENDER
59. **Numerical Analysis of the Finite Element Method.** Philippe G. CIARLET
60. **Méthodes numériques en mathématiques appliquées.** J.F. Giles AUCHMUTY, Michel CROUZEIX, Pierre JAMET, Colette LEBAUD, Pierre LESAINT et Bertrand MERCIER
61. **Analyse numérique matricielle.** Paul ARMINJON
62. **Problèmes d'optimisation en calcul des probabilités.** Serge DUBUC
63. **Chaînes de Markov sur les permutations.** Gérard LETAC
64. **Géométrie différentielle stochastique.** Paul MALLIAVIN
65. **Numerical Methods for Solving Time-Dependent Problems for Partial Differential Equations.** Heinz-Otto KREISS
66. **Difference Sets in Elementary Abelian Groups.** Paul CAMION
67. **Groups in Physics : Collective Model of the Nucleus; Canonical Transformation in Quantum Mechanics.** Marcos MOSHINSKY
68. **Points fixes pour les applications compactes : espaces de Lefschetz et la théorie de l'indice.** Andrzej GRANAS
69. **Set Theoretic Methods in Homological Algebra and Abelian Groups.** Paul EKLOF
70. **Abelian p-Groups and Mixed Groups.** Laszlo FUCHS
71. **Integral Representations and Structure of Finite Group Rings.** Klaus W. ROGGENKAMP
72. **Homological Invariants of Modules over Commutative Rings.** Paul ROBERTS
73. **Representations of Valued Graphs.** Vlastimil DLAB

Achevé d'imprimer à Cap St-Ignace
aux Ateliers Graphiques Marc Veilleux Inc.
en février 1981.