Séminaire Delange-Pisot-Poitou. Théorie des nombres

Jean-Louis Nicolas

Des exemples de programmation non linéaire en théorie des nombres

Séminaire Delange-Pisot-Poitou. Théorie des nombres, tome 14, n° 1 (1972-1973), exp. n° 10, p. 1-11

<http://www.numdam.org/item?id=SDPP_1972-1973__14_1_A8_0>

Séminaire DELANGE-PISOT-POITOU
(Théorie des nombres)
14e année, 1972/73, n° 10, 11 p.

18 décembre 1972

DES EXEMPLES DE PROGRAMMATION NON LINÉAIRE EN THÉORIE DES NOMBRES

par Jean-Louis NICOLAS

1. Problèmes de programmation mathématique.

Le problème général de programmation est le suivant : on cherche à maximiser, ou minimiser une fonction de k variables :

$$z = f(x_1 , x_2 , \dots , x_k)$$

lorsque ces variables sont soumises à des contraintes :

$$(1) \qquad g_\alpha(x_1 , x_2 , \dots , x_k) < \text{ ou } = A_\alpha \ ; \ \alpha \in I$$

Lorsque la fonction z et les contraintes g_α sont linéaires, la programmation est dite linéaire. Si les variables sont réelles, l'ensemble des solutions "possibles" du problème, c'est-à-dire le sous-ensemble de $\underset{\sim}{R}^k$, satisfaisant aux contraintes (1), est une partie convexe de $\underset{\sim}{R}^k$, et le problème a une solution qui est un des sommets de ce convexe. Lorsque les variables et les contraintes sont nombreuses, il n'est pas facile de trouver la solution. On utilise alors l'algorithme du simplexe.

Si on limite les variables à ne prendre que des valeurs entières, on a un problème de programmation linéaire en nombres entiers. On peut alors adapter l'algorithme du simplexe pour résoudre le problème (cf. GREENBERG, Integer programming [2]).

Lorsque la fonction f ou les contraintes g_α ne sont pas linéaires, on dit que le problème de programmation est non linéaire. Si les variables sont réelles, rappelons dans un cas simple la solution théorique du problème : S'il n'y a qu'une contrainte de la forme :

$$(2) \qquad g(x_1 , x_2 , \dots , x_k) = A \ ,$$

f a un extremum relatif à la condition nécessaire :

$$(3) \qquad \frac{f'_1}{g'_1} = \frac{f'_2}{g'_2} = \dots = \frac{f'_k}{g'_k}$$

en notant $f'_1 = \partial f / \partial x_1$. Les relations (2) et (3) permettent de calculer les variables x_i .

Une autre façon d'obtenir les relations (3) est celle des multiplicateurs de Lagrange : Supposons que, pour une valeur de ρ réelle, la fonction $f - \rho g$ ait un minimum absolu dans $\underset{\sim}{R}^k$ au point $x^* = (x_1^* , x_2^* , \dots , x_k^*)$, alors x^* est une solution du problème de programmation :

$$\text{minimiser } z = f(x_1 , x_2 , \dots , x_k)$$

lorsque les variables x_i sont soumises à la contrainte (2) pour la valeur

$$A = g(x^*) .$$

En effet, quelque soit x vérifiant $g(x) = A = g(x^*)$, on aura :

$$f(x) - \rho g(x) \geqslant f(x^*) - \rho g(x^*) ,$$

ce qui entraîne

$$f(x) \geqslant f(x^*) + \rho[g(x) - g(x^*)] \geqslant f(x^*)$$

Pour que la fonction $f - \rho g$ ait un minimum, il faut que :

$$\frac{\partial f}{\partial x_i} - \rho \frac{\partial g}{\partial x_i} = 0 \tag{4}$$

En comparant avec (3), on peut interpréter le multiplicateur de Lagrange ρ comme la valeur commune des rapports f'_i/g'_i .

Enfin, dans le cas de programmation non linéaire en nombres entiers, il existe un algorithme qui s'appelle "programmation dynamique" que nous allons exposer. On veut résoudre le problème suivant :

$$\text{(5)} \quad \begin{cases} \text{Trouver des variables } x_1 , x_2 , \ldots , x_k \text{ entières et positives,} \\ \text{vérifiant } \sum_{i=1}^{k} a_i x_i \leqslant A , \quad a_i > 0 , \quad a_i \text{ entiers} \\ \text{qui minimisent } z = f(x) = \sum_{i=1}^{k} f_i(x_i) . \end{cases}$$

On suppose ici qu'il n'y a qu'une seule contrainte linéaire à coefficients entiers et que la fonction à optimiser est séparable.

Remarquons d'abord que le nombre N de solutions possibles est fini : On doit avoir $0 \leqslant x_i \leqslant \frac{A}{a_i}$ pour chaque i .

On définit, pour $j = 1 , 2 , \ldots , k$ et $0 \leqslant M \leqslant A$,

$$\Delta_j(M) = \min_{x_1, x_2, \ldots, x_j, \sum_i a_i x_i \leqslant M} \sum_i f_i(x_i) , \quad 1 \leqslant i < j . \tag{6}$$

On voit que $\Delta_k(A)$ donnera la solution du problème (5).

D'autre part, les $\Delta_j(M)$ se calculent par récurrence.

$$\Delta_1(M) = \min_{x_1 \leqslant M/a_1} f_1(x_1)$$

$$\Delta_j(M) = \min_{0 \leqslant x_j \leqslant M/a_j} [f_j(x_j) + \Delta_{j-1}(M - a_j x_j)]$$

$$= \min[f_j(0) + \Delta_{j-1}(M) , \ f_j(1) + \Delta_{j-1}(M - a_j) , \ \ldots]$$

2. Théorie des nombres.

RAMANUJAN [9] dit qu'un nombre N est hautement composé si tout nombre $n < N$ a strictement moins de diviseurs que N .

Appelons $d(n)$ le nombre de diviseurs de n . On sait que la fonction d est

multiplicative (si m et n sont premiers entre eux, $d(mn) = d(m)\,d(n)$) et si la décomposition en facteurs premiers de n est $n = \prod p_i^{\alpha_i}$, alors $d(n) = \prod_i (\alpha_i + 1)$.

On a donc : N est hautement composé si, et seulement si,,

$$n < N \rightarrow d(n) < d(N)$$

Soit un entier a ; cherchons $\max_{n \leqslant a} d(n)$. Soit n^* le plus petit entier où $d(n)$ atteint ce maximum. Le nombre n^* est hautement composé. Et si l'on écrit

$$n = 2^{x_1}\, 3^{x_2} \ldots p_k^{x_k} \ldots \quad \text{avec} \quad x_i \geqslant 0 ,$$

n^* est solution du problème de programmation suivant :

$$(7) \qquad \begin{cases} x_1 \log 2 + x_2 \log 3 + \ldots + x_k \log p_k \leqslant A = \log a \\ \max z = (x_1 + 1) \ldots (x_k + 1) . \end{cases}$$

Rappelons les résultats obtenus sur les nombres hautement composés (cf. [5],,où l'on trouvera d'autres références) : Une solution n^* du problème (7) vérifie $x_1 \geqslant x_2 \geqslant \ldots \geqslant x_k \geqslant \ldots$ Pour le plus grand k, tel que $x_k = 0$, on a $x_k = 1$, sauf pour $n^* = 4$ et $n^* = 36$. Enfin, si l'on pose :

$$Q(x) = \text{Card}\{\text{nombres hautement composés} \leqslant x\} ,$$

on a :

$$(\log x)^{c_1} \leqslant Q(x) \leqslant (\log x)^{c_2} ,$$

pour x assez grand, avec $c_1 > 1$, et la conjecture :

$$\frac{\log Q(x)}{\log \log x} = 1 + \frac{1}{4}\left(\frac{\log 15}{\log 2} - 3\right) = 1{,}227 .$$

D'autre part, on a étudié le même problème, en remplaçant la fonction $d(n)$ par les fonctions :

$$\sigma(n) = \text{somme des diviseurs de } n , \text{ et } \frac{\sigma(n)}{n} \quad \text{(cf. [1])},$$

et des généralisations avec les fonctions :

$$\sigma_r(n) = \sum_{d \mid n} \frac{1}{d^r} \quad (r = 0 , \ \sigma_r(n) = d(n) ; \qquad r = -1 , \ \sigma_r(n) = \sigma(n)$$

et

$$d_t(n) = \text{nombre de manières de décomposer } n \text{ en } t \text{ facteurs} .$$

avec $t = 2$, $d_t(n) = d(n)$ (cf. [7] et [8]).

Désignons par $g(n)$ l'ordre maximum d'un élément du groupe des permutations S_n. On peut montrer que (cf. [4] et [6]) :

$$(8) \qquad g(n) = \max_{\ell(j) \leqslant n} j ,$$

où la fonction ℓ est ainsi définie :

si $n = \prod p_i^{\alpha_i}$, $\alpha_i \geqslant 1$, alors :

$$\ell(n) = \sum_i p_i^{\alpha_i} \quad \text{et} \quad \ell(1) = 0$$

(ainsi $\ell(180) = \ell(4 \cdot 9 \cdot 5) = 4 + 9 + 5 = 18$).

Le problème de calculer $g(n)$ est un problème de programmation

$$(9) \qquad \begin{cases} \ell(2^{x_1}) + \ell(3^{x_2}) + \ldots + \ell(p_k^{x_k}) \leqslant n \\ \max j = x_1 \log 2 + x_2 \log 3 + \ldots + x_k \log p_k \end{cases}$$

avec $\ell(p^x) = p^x$ si $x \geqslant 1$ et $\ell(p^0) = 0$.

Remarquons que dans [6], pour faire une table de $g(n)$ jusqu'à $n = 8.000$, l'algorithme utilisé était un algorithme de programmation dynamique.

Rappelons quelques propriétés de la fonction $g(n)$ définie par (8) : $g(n)$ est croissante, mais non strictement croissante, et il existe des intervalles aussi grands que l'on veut sur lesquels $g(n)$ est constante. D'autre part, les exposants x_i dans la décomposition en facteurs premiers de $g(n)$, qui sont la solution du problème (9) ne sont pas tout à fait décroissants : pour $i < i'$, on a

$$x_{i'} \leqslant x_i + 1 ,$$

et l'on a une infinité de fois $i < i'$ et $x_{i'} = x_i + 1$.

Pour démontrer ces propriétés, ainsi que les propriétés des nombres hautement composés, nous avons utilisé la méthode des nombres hautement composés supérieurs de RAMANUJAN, que nous exposerons au paragraphe 3.

RAMANUJAN dit qu'un nombre N est hautement composé supérieur s'il existe $\varepsilon > 0$ tel que, pour tout entier M , on ait :

$$\frac{d(M)}{M^{\varepsilon}} \leqslant \frac{d(N)}{N^{\varepsilon}} .$$

On peut montrer que de tels nombres existent, et sont solution du problème (7) pour certaines valeurs de A . Par la méthode des "bénéfices", que l'on exposera au paragraphe 4, on peut obtenir des renseignements sur les solutions du problème (7) pour d'autres valeurs de A . Ces méthodes peuvent s'appliquer à d'autres problèmes de programmation.

3. La méthode des nombres hautement composés supérieurs de RAMANUJAN.

Soit le problème de programmation en nombres entiers :

$$(10) \qquad \begin{cases} g(x) = \sum_{i=1}^{k} a_i x_i \leqslant A \ ; \quad a_i \geqslant 0 \ ; \quad x_i \geqslant 0 \ ; \quad x_i \text{ entier} \\ \min z = f(x) = \sum_{i=1}^{k} f_i(x_i) \ , \quad f_i \text{ convexe} \end{cases}$$

La différence entre ce problème et le problème de programmation dynamique (5) est: les a_i peuvent ne pas être entiers, mais les fonctions f_i doivent être convexes, c'est-à-dire

$$f_i(n-1) - f_i(n) \geqslant f_i(n) - f_i(n+1) \quad \text{pour } n \text{ entier} \geqslant 1 .$$

A chaque nombre A correspond une solution $x(A) = x_1(A)\ ,\ x_2(A)\ ,\ \ldots\ ,\ x_k(A)$, avec $z = z(A) = f(x(A))$. On dira que A est une borne intéressante du problème si $g(x(A)) = A$. Les bornes intéressantes, rangées par ordre croissant, forment une suite discrète $A_1\ ,\ A_2\ ,\ \ldots\ ,\ A_n\ ,\ \ldots$ sous-ensemble de l'ensemble discret des valeurs prises par la fonction g .

En utilisant la méthode des multiplicateurs de Lagrange, exposée précédemment, mais cette fois avec des valeurs entières des variables, on peut construire explicitement une sous-suite A_{n_j} de la suite A_n des bornes intéressantes, avec la solution correspondante du problème (10).

Construction des bornes hautement intéressantes du problème (10). - Étant donné ρ réel, positif, on cherche si la fonction $f(x) + \rho g(x)$ a un minimum absolu en x .

Pour qu'une fonction séparable $\varphi(x) = \sum_i \varphi_i(x_i)$ ait un minimum en

$$x = x_1\ ,\ x_2\ ,\ \ldots\ ,\ x_k\ ,$$

il faut et il suffit que, pour tous les i , $\varphi_i\ f_i$ ait un minimum en x_i . Pour que $f(x) + \rho g(x)$ ait un minimum en x , il faut que $f_i(x_i) + \rho a_i\ x_i$ ait un minimum en x_i , ce qui entraîne

$$f_i(x_i + 1) + \rho a_i(x_i + 1) \geqslant f_i(x_i) + \rho a_i\ x_i$$

et

$$f_i(x_i - 1) + \rho a_i(x_i - 1) \geqslant f_i(x_i) + \rho a_i\ x_i$$

ce qui entraîne

$$f_i(x_i - 1) - f_i(x_i) \geqslant \rho a_i \geqslant f_i(x_i) - f_i(x_i + 1)\ . \tag{11}$$

Comme les fonctions f_i ont été supposées convexes, les inégalités (11) entraînent :

$$\frac{f_i(0) - f_i(1)}{a_i} \geqslant \frac{f_i(1) - f_i(2)}{a_i} \geqslant \ldots \geqslant \frac{f_i(x_i - 1) - f_i(x_i)}{a_i} \geqslant \rho \geqslant \frac{f_i(x_i) - f_i(x_i)}{a_i}\ . \tag{12}$$

Le nombre ρ étant fixé, les inégalités (12) déterminent la valeur de x_i (si les inégalités sont strictes).

Les inégalités (12) entraînent également que, pour tout entier n ,

$$f_i(n) - \rho a_i\ n \geqslant f_i(x_i) - \rho a_i\ x_i\ . \tag{13}$$

Par exemple, pour $n < x_i$, on a

$$\frac{f_i(n) - f_i(x_i)}{a_i} = \frac{f_i(n) - f_i(n + 1)}{a_i} + \ldots + \frac{f_i(x_i - 1) - f_i(x_i)}{a_i} \geqslant (n - x_i)\ \rho\ .$$

L'inégalité (13) montre qu'en $x = x_1\ ,\ x_2\ ,\ \ldots\ ,\ x_k$, la fonction $f(x) + \rho g(x)$ a un minimum absolu, qui fournit une solution au problème (10) pour la valeur $A = g(x)$.

En effet, quelque soit y vérifiant $g(y) < g(x) = A$, on a

$$f(y) + \rho g(y) \geqslant f(x) + \rho g(x) ,$$

ce qui entraîne

$$f(y) \geqslant f(x) + \rho[g(x) - g(y)] \geqslant f(x) .$$

En faisant varier ρ , on obtiendra diverses solutions du problème (10) pour les valeurs de A : A_{n_1} , A_{n_2} , ... , A_{n_j} , qui forment une sous-suite de l'ensemble des bornes intéressantes.

Définition. - Les éléments de cette sous-suite seront appelés bornes hautement intéressantes du problème (10).

EXEMPLE n° 1 : Les nombres hautement composés supérieurs. - On a :

$$\begin{cases} g(x) = \sum_{i=1}^{k_1} \log p_i \, x_i \text{ , où } p_i \text{ désigne le } i\text{-ième nombre premier} \\ \min f(x) = \sum_{i=1}^{k} - \log(x_i + 1) . \end{cases}$$

Les inégalités (12) deviennent :

$$\frac{\log 2}{\log p_i} \geqslant \frac{\log 3/2}{\log p_i} \geqslant \frac{\log 4/3}{\log p_i} \geqslant \ldots \geqslant \frac{\log(x_i/(x_i - 1))}{\log p_i} \geqslant \rho \geqslant \frac{\log(x_i + 1)/x_i}{\log p_i}$$

dont les valeurs numériques sont :

(14)

$p_i = 2$	1	0,585	0,415	0,322	0,263
3	0,631	0,369	0,262	0,203	0,166
5	0,431	0,252	0,179	0,139	0,113
7	0,356	0,208	0,148	0,115	0,094
11	0,289	0,169	0,119	0,093	0,076

On voit sur le tableau précédent que si l'on choisit $\rho > 1$, on obtient $x_i = 0$ pour tout i . Si l'on choisit $\rho = 0,3$, on trouve : $x_1 = 4$; $x_2 = 2$; $x_3 = 1$; $x_4 = 1$; $x_5 = 0$.

(15)

ρ	x	$e^{g(x)} = N$	$e^{-f(x)} = d(N)$
1	0,0,0,0,0	1	1
0,631	1,0,0,0,0	2 = 2	2
0,585	1,1,0,0,0	2.3 = 6	4
0,431	2,1,0,0,0	4.3 = 12	6
0,415	2,1,1,0,0	4.3.5 = 60	12
0,369	3,1,1,0,0	8.3.5 = 120	16
0,356	3,2,1,0,0	8.9.5 = 360	24
0,322	3,2,1,1,0	8.9.5.7 = 2520	48
0,289	4,2,1,1,0	16.9.5.7 = 5040	60
	4,2,1,1,1	16.9.5.7.11 = 55440	120

EXEMPLE n° 2 : Le stockage des pièces de rechange d'un sous-marin. - Dans le livre de HADLEY ([3], p. 362), un problème de stockage des pièces de rechange pour un sous-marin, et traité par la programmation dynamique, revient à minimiser la fonction :

$$f = \sum_{j=1}^{3} f_j = \sum_{j=1}^{3} \pi_j \varphi_j \quad \text{avec} \quad \pi_j = \not{\$}\ 800\ ;\ 600\ ;\ 1300$$

et

$$\varphi_j(x) = \begin{cases} \mu_j P(x-1\ ,\ \mu_j) - x\, P(x\ ,\ \mu_j) & \text{si } x \geqslant 1 \\ \mu_j & \text{si } x = 0 \end{cases}$$

avec $\mu_j = 4\ ;\ 2\ ;\ 1$. La fonction $P(x\ ,\ \mu)$ est la fonction cumulative de Poisson :

$$P(x\ ,\ \mu) = \sum_{n=x}^{\infty} \frac{\mu^n e^{-\mu}}{n!}\ ,$$

et la contrainte est $x_1 + 2x_2 + 2x_3 \leqslant A = 10$.

Les valeurs de la fonction $P(x\ ,\ \mu)$ sont données par les tables (cf. [10]). On a :

	μ	x = 0	x = 1	x = 2	x = 3	x = 4	x = 5	x = 6	x = 7
j = 1	4	1	0,9817	0,9084	0,7619	0,5665	0,3712	0,2149	0,1107
j = 2	2	1	0,8646	0,5940	0,3232	0,1428			
j = 3	1	1	0,6320	0,2642	0,0802				

On en déduit la table des valeurs de $(f_i(x_i - 1) - f_i(x_i))/a_i$ qui vaut

$$\frac{\pi_i}{a_i}[\varphi_i(x_i - 1) - \varphi_i(x_i)] = \frac{\pi_i}{a_i} P(x\ ,\ \mu_i)\ .$$

(16)

	x = 1	x = 2	x = 3	x = 4	x = 5	x = 6	x = 7
i = 1	785,36	726,72	609,52	453,20	296,96	171,92	88,56
i = 2	259,38	178,20	96,96	42,84			
i = 3	410,80	171,73	52,13				

On obtient alors le tableau des bornes hautement intéressantes :

(17)

ρ	x	A = g(x)	f(x)
	0 0 0	0	5700
785,36	1 0 0	1	4914
726,72	2 0 0	2	4188
609,52	3 0 0	3	3578
453,20	4 0 0	4	3125
410,80	4 0 1	6	2285
296,96	5 0 1	7	1989
259,38	5 1 1	9	1470
178,20	5 2 1	11	1114
171,92	6 2 1	12	942
171,73	6 2 2	14	599

4. La méthode des "bénéfices".

L'inconvénient principal de la méthode exposée au paragraphe précédent est de ne pas fournir toutes les bornes intéressantes : Dans l'exemple n° 2, le tableau (17) ne donne pas la solution pour $A = 5 , 4 , 8 , 10 , 13$. Nous allons voir comment on peut y remédier. Les hypothèses sont toujours celles du problème (10).

Soit une borne hautement intéressante A , associée à un nombre ρ , et la solution x correspondante du problème (10). Soit $\xi = (\xi_1 , \xi_2 , \dots , \xi_k)$ un vecteur à coordonnées entières, alors on a :

(18) $$f(x + \xi) = f(x) - \rho g(\xi) + \text{bén}(\xi)$$

avec $\text{bén}\ \xi = \sum_{i=1}^{k} \text{bén}\ \xi_i$ et $\text{bén}\ \xi_i = f_i(x_i + \xi_i) - f_i(x_i) + \rho a_i\ \xi_i$.

Calcul du bénéfice.

Si $\xi_i > 0$,

$$\xi_i = 1 \text{ , on a } \text{bén}(\xi_i = 1) = a_i\left[\rho - \frac{f_i(x_i) - f_i(x_i + 1)}{a_i}\right]$$

où le crochet fait intervenir les termes de l'inégalité (12) que l'on avait mis en tables dans les exemples (tableaux (14) et (16)).

$\xi_i = 2$, on a : $\text{bén}(\xi_i = 2) = \text{bén}(\xi_i = 1) + a_i[\rho - \dfrac{f_i(x_i + 1) - f_i(x_i + 2)}{a_i}]$

etc.

Si $\xi_i < 0$, on a :

$$\text{bén}(\xi_i = -1) = a_i[\frac{f_i(x_i) - f_i(x_i - 1)}{a_i} - \rho]$$

$$\text{bén}(\xi_i = -2) = \text{bén}(\xi_i = -1) + a_i[\frac{f_i(x_i - 2) - f_i(x_i - 1)}{a_i} - \rho]$$

etc.

Exemple numérique. - Dans l'exemple 1 précédent, prenons $\rho = 0,3$; on avait $x_1 = 4$, $x_2 = 2$; $x_3 = x_4 = 1$, $x_5 = 0$, et on a :

$$\text{bén}(\xi_2 = 1) = (0,3 - 0,262) \log 3 \fallingdotseq 0,042$$

$$\text{bén}(\xi_2 = 2) = \text{bén}(\xi_2 = 1) + (0,3 - 0,203) \log 3 = 0,042 + 0,107 = 0,151 .$$

Remarque. - Le bénéfice, défini par (18) dépend non seulement de la solution x hautement intéressante mais aussi de ρ . Dans l'exemple numérique précédent, $\rho = 0,31$ donnait la même solution x , mais pas les mêmes valeurs de bénéfice.

THÉORÈME. - On a, pour tout ξ_i , $\text{bén}\ \xi_i \geqslant 0$.

Comme $f(x) + \rho g(x)$ est un minimum, on a :

$$f(x + \xi) + \rho g(x + \xi) \geqslant f(x) + \rho g(x) ,$$

et (18) donne le résultat.

THÉORÈME de majoration des bénéfices (cf. [5], proposition 2). - Soient

$$y = (y_1 , y_2 , \dots , y_k) \text{ et } y'$$

deux solutions possibles du problème (10), et soit z une solution exacte du même problème, avec la valeur $A = g(z)$, et vérifiant l'une ou l'autre des conditions :

(i) $g(y) \leqslant g(z) \leqslant g(y')$

(ii) $f(y') \leqslant f(z) \leqslant f(y)$

Soit x une solution hautement intéressante du problème (10) associée à ρ , par rapport à laquelle on définit les bénéfices. Alors on a :

(19) $$\text{bén}(z - x) \leqslant \text{bén}(y - x) + \rho[g(y') - g(y)]$$

et

(20) $$\text{bén}(z - x) \leqslant \text{bén}(y' - x) + f(y) - f(y') .$$

Démonstration. - L'hypothèse (i) entraîne :

$$g(y) \leqslant g(z) = A \Rightarrow f(y) \geqslant f(z) , \text{ car } z \text{ minimise } f .$$

L'hypothèse (ii) entraîne :

$$f(y') \leqslant f(z) \Rightarrow g(y') \geqslant A = g(z)$$

Dans les deux cas, on a :

$$\text{bén}(z - x) = f(z) - f(x) + \rho[g(z) - g(x)]$$

$$\text{bén}(z - x) \leqslant f(y) - f(x) + \rho[g(y') - g(x)] \qquad (21)$$

Le membre de droite de (21) est égal au membre de droite de (19) et de (20).

Dans la pratique, on cherchera des solutions possibles y , y' , y'' , ... entre deux solutions hautement intéressantes consécutives, et on majorera avec le théorème précédent le bénéfice d'une solution intéressante éventuelle. On éliminera alors les valeurs de ξ_i qui auraient un trop grand bénéfice, il ne restera plus qu'un petit nombre de valeurs de $\xi = z - x$ à essayer.

Exemple n° 2 avec $\rho = 250$: la solution hautement intéressante est

$$x_1 , x_2 , x_3 = 5 , 1 , 1 .$$

Le tableau des bénéfices s'obtient à partir du tableau (16) :

(22)

	$\text{bén}(\xi_i = -2)$	$\text{bén}(\xi_i = -1)$		$\text{bén}(\xi_i = 1)$	$\text{bén}(\xi_i = 2)$
$i = 1$	249	46		78	240
$i = 2$	—	18		144	451
$i = 3$	—	321		157	553

On cherche une solution z telle que $g(z) = 10$. On a une solution possible $y = 6 , 1 , 1$. On applique le théorème avec la condition (i) et $y = y'$. On obtient :

$$\text{bén}(z - x) \leqslant \text{bén}(y - x) = 78 .$$

Les composantes de $z - x$, si $\text{bén}(z - x) < 78$, doivent être : $\xi_1 = 0$ ou -1 , $\xi_2 = 0$ ou -1 , $\xi_3 = 0$. Dans tous les cas, on aurait $g(z) < g(x)$. La solution du problème de l'exemple n° 2 pour $A = 10$ est donc $x = 6 , 1 , 1$.

On pourrait montrer de même que, pour $A = 13$, la solution est $5 , 2 , 2$.

BIBLIOGRAPHIE

[1] ALAOGLU (L.) and ERDOS (P.). - On highly composite and similar numbers, Trans. Amer. math. Soc., t. 56, 1944, p. 448-469.

[2] GREENBERG (H.). - Integer programming. - New York, Academic Press 1971 (Mathematics in Science and Engineering, 76).

[3] HADLEY (G.). - Non linear and dynamic programming. - Reading, Palo Alto London, Addison-Wesley publishing Company, 1964 (Addison-Wesley series in management science and economics).

[4] NICOLAS (J.-L.). - Ordre maximal d'un élément du groupe S_n des permutations et highly composite numbers, Bull. Soc. math. France, t. 97, 1969, p. 129-191.

[5] NICOLAS (J.-L.). - Répartition des nombres hautement composés de Ramanujan. - Canad. J. Math., t. 23, 1971, p. 116-130.

[6] NICOLAS (J.-L.). - Calcul de l'ordre maximum d'un élément du groupe symétrique S_n, R. I. R. O., 3e année, n° R-2/1969, p. 43-50.

[7] PILLAÏ (S.). - Highly abundant numbers, Bull. Calcutta math. Soc., t. 35, 1943, p. 141-156.

[8] PILLAÏ (S.). - Highly composite number, J. Indian math. Soc., t. 8, 1944, p. 61-74.

[9] RAMANUJAN (S.). - Highly composite numbers, Proc. London math. Soc., Series 2, t. 14, 1915, p. 347-400 ; and Collected papers, p. 78-128. - Cambridge, at the University Press, 1927.

[10] Tables of individual and cumulative terms of Poisson distribution. - Princeton, D. Van Nostrand Company, 1962.

(Texte reçu le 12 mars 1973)

Jean-Louis NICOLAS
Tour Mexico
65 rue du Javelot
75013 PARIS