Revue française d'automatique, d'informatique et de recherche opérationnelle. Recherche opérationnelle

Odile Mignotte

Méthode de recherche d'un extremum d'une fonction monodimensionnelle

Revue française d'automatique, d'informatique et de recherche opérationnelle. Recherche opérationnelle, tome 9, n° V2 (1975), p. 113-123.

<http://www.numdam.org/item?id=RO_1975__9_2_113_0>

R.A.I.R.O.
(9e année, juin 1975, V-2, p. 113 à 123)

METHODE DE RECHERCHE D'UN EXTREMUM D'UNE FONCTION MONODIMENSIONNELLE (*)

par Odile MIGNOTTE (1)

Résumé. — *On cherche à déterminer l'extremum d'une fonction unimodale sur un intervalle — une nouvelle méthode de recherche est proposée, dont les performances sont comparables aux meilleures méthodes connues (recherche de Fibonacci ou du Nombre d'Or). Il s'agit d'une méthode dynamique, en ce sens que la stratégie adoptée varie en fonction des informations recueillies durant les essais antérieurs. Cette variation repose sur des considérations de nature probabiliste.*

Je tiens à exprimer ma gratitude à M. le Professeur Vignes pour ses conseils et ses encouragements.

INTRODUCTION

Soit f une fonction réelle d'une variable réelle, unimodale sur un intervalle $[a, b]$, $\varepsilon > 0$ étant fixé, il s'agit de déterminer un intervalle de longueur ε qui contienne un point m tel que $f(m)$ soit maximal (ou minimal).

On choisit $n + 1$ points $x_0 = a < x_1 < x_2 < \cdots < x_{n-1} < x_n = b$ de l'intervalle initial. La comparaison des $f(x_i)$ permet de situer m dans l'intervalle de type $[x_i, x_{i+1}]$ ou $[x_i, x_{i+2}]$. Et ainsi de suite.

Pour des points x_i régulièrement espacés nous montrons que les cas optimaux sont $n = 3$ et $n = 4$. Pour ces valeurs de n, nous chercherons la meilleure disposition possible des x_i. Notre recherche sera illustrée par de nombreux exemples calculés sur machine.

(*) Reçu avril 1974.

(1) CNAM, Chaire d'Informatique.

I. PRÉSENTATION DE LA MÉTHODE

1. Préliminaires

Nous nous limiterons à la recherche d'un maximum. Si on calcule $f(a)$, $f(x)$, $f(b)$ pour un certain $x \in]a, b[$, f étant unimodale on a évidemment

$$f(x) < f(a) \Rightarrow m \in [a, x] \,; \quad f(x) < f(b) \Rightarrow m \in [x, b] \,.$$

Dans les autres cas on ne peut situer m par rapport à x. Ainsi dans les hypothèses de notre étude, il est nécessaire de calculer au moins deux valeurs de f en des points x_1 et x_2 intérieurs à $[a, b]$. On aura donc $n \geqslant 3$.

Remarque

En pratique il faut tenir compte de la précision de la machine et de celle des calculs. On s'imposera donc un certain seuil s et on considérera que l'inégalité $f(x) < f(y)$ (respectivement $f(x) > f(y)$) a lieu si et seulement si $\overline{f(x)} < \overline{f(y)} - s$ (respectivement $\overline{f(x)} > \overline{f(y)} + s$) où $\overline{f(z)}$ désigne la valeur approchée de $f(z)$, si tel est le cas on écrira $f(x) \ll f(y)$ (respectivement $f(x) \gg f(y)$). Si $f(y)$ appartient à l'intervalle $[f(x) - s, f(x) + s]$ il y a indétermination, nous arrêtons alors l'exécution du programme et indiquons à l'utilisateur les dernières bornes calculées pour m. Ce cas d'indétermination sera exclu de l'étude théorique.

2. Présentation de l'algorithme

Posons $y_j = f(x_j), j = 0, \ldots, n$. Soit k, le plus petit entier positif, s'il existe, tel que $y_k \ll y_{k-1}$ et $k = n + 1$ si un tel entier n'existe pas. Chaque valeur de k correspond à un événement que nous noterons E_k. L'intervalle réduit est le suivant :

$$m \in \begin{cases} [x_{k-2}, x_k] & \text{si} \quad k \neq 1 \quad \text{et} \quad k \neq n \\ [x_{k-1}, x_k] & \text{si} \quad k = 1, \quad \text{ou} \quad k = n \,. \end{cases}$$

On suppose $f(a)$ et $f(b)$ connus. Notons C_k le nombre de calculs et R_k le rapport

$$\frac{\text{longueur du nouvel intervalle}}{\text{longueur de l'intervalle précédent}} \cdot$$

On a :

$$C_k = k \quad \text{pour} \quad k \leqslant n - 1 \quad \text{et} \quad C_n = C_{n+1} = n - 1$$

tandis que $R_k = \dfrac{2}{n}$ pour $k \neq 1$ et $k \neq n$,

$$R_1 = R_n = \frac{1}{n} \,.$$

REMARQUE

Il est aisé de voir graphiquement que dès que le cas « malchanceux » $\left(R_k = \frac{2}{n}\right)$ s'est produit le cas « heureux » $\left(R_k = \frac{1}{n}\right)$ ne peut plus se produire.

II. MAJORATION DU NOMBRE DE CALCULS

1. Cas *n* pair

D'après ce qui précède, à chaque itération $n - 1$ calculs suffisent toujours ($n - 2$ suffiront si lors de l'itération précédente $R_k = \frac{2}{n}$, le point du milieu est conservé, ce que l'on supposera dans le calcul qui suit). Soient $v_0 = v$, $v_1, \ldots, v_k$ les longueurs successives des intervalles déterminés par cet algorithme. On a :

$$v_j \leqslant \frac{2}{n} v_{j-1} = \frac{2}{n}\left(\frac{2}{n} v_{j-2}\right) = \cdots = \left(\frac{2}{n}\right)^j v .$$

Le nombre total K d'itérations nécessaires est déterminé par $v_k \leqslant \varepsilon < v_{k-1}$. D'où la majoration

$$K \leqslant \left[\frac{\log (v/\varepsilon)}{\log (n/2)}\right] + 1$$

où $[x]$ désigne la partie entière de x (i.e. $[x] = p \in \mathbb{Z}$ tel que $p \leqslant x < p + 1$). Comme le problème qui nous intéresse ne se pose que si (v/ε) est assez grand, on posera pour simplifier $K = A/\log (n/2)$, où $A = \log (v/\varepsilon)$. Le nombre N maximum d'opérations est donc donné par

$$N = A(n - 2)/\log (n/2) .$$

Pour déterminer la meilleure valeur de N en fonction du choix du nombre de points x_i, il s'agit donc d'étudier la fonction :

$$g(x) = \frac{x - 2}{\mathrm{Log}\left(\frac{x}{2}\right)}$$

et de déterminer le minimum de la fonction g pour x entier > 4. On trouve :

$$g'(x) = \frac{h(x)}{\left(\mathrm{Log}\,\frac{x}{2}\right)^2}$$

où
$$h(x) = \mathrm{Log}\left(\frac{x}{2}\right) - 1 + \frac{2}{x}$$

qui s'annule pour $x = 2$ et est toujours positive pour $x > 2$. Donc $g(x)$ est croissante et la meilleure valeur pour n pair sera 4.

2. Cas *n* impair

Pour $n \geqslant 5$ l'étude précédente montre que le cas n impair $\geqslant 5$ ne peut être meilleur que le cas $n = 4$, même en supposant que l'on conserve l'un des points (ce qui est incompatible avec une distribution régulière des points).

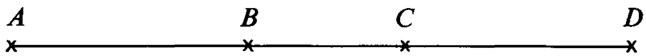
III. CAS $N = 3$

1. Méthode du nombre d'or (1)

Les points sont répartis de telle sorte que le rapport

$$\frac{v_n}{v_{n+1}} = \tau = \frac{1 + \sqrt{5}}{2}$$

soit conservé lors de chaque itération. (v_k étant la longueur de l'intervalle obtenu après k itérations.) $v_k = v_0/\tau^{k-1}$. B et C sont symétriques.



Le rapport dans lequel C divise AD est le même que celui dans lequel B divise AC, ce qui répond au souhait de conserver l'un des points pour l'itération suivante. Un seul calcul de la fonction par itération suffit pour réduire l'intervalle dans le rapport $1/\tau$. Ceci fait tout l'intérêt de la méthode.

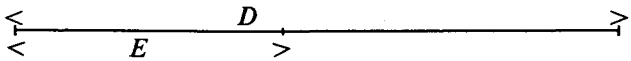
2. Méthode de Fibonacci

Cette méthode se différencie en ceci qu'elle ne vise pas à réduire l'intervalle de façon constante à chaque itération. Par contre elle cherche à conserver un point de l'intervalle précédent. Le point calculé au cours de l'itération est alors placé symétriquement au point conservé.

Il est aisé de noter que sur une seule itération, la meilleure réduction possible est obtenue pour une « pseudo-dichotomie », c'est-à-dire deux points situés presque au milieu de l'intervalle. Mais cette position ne permet évidemment pas de conserver l'un des points à l'étape suivante.

(1) Le nombre d'Or est la racine positive de $\tau^2 - \tau - 1 = 0$.

Soit D la longueur du dernier intervalle (précision demandée) et E l'intervalle qui sépare les deux derniers points :



$$
\begin{aligned}
L_k &= \text{longueur du } k\text{-ième intervalle} \\
L_n &= D \leqslant \varepsilon \text{ (notation précédente)} \\
L_{n-1} &= 2\,D - E = F_2\,D - F_0\,E \\
L_{n-2} &= 3\,D - E = F_3\,D - F_1\,E \\
L_{n-3} &= L_{n-2} + L_{n-1} = F_4\,D - F_2\,E \\
&\vdots \\
L_2 &= F_{n-1}\,D - F_{n-3}\,E \\
L_1 &= F_n\,D - F_{n-2}\,E \\
L_0 &= F_{n+1}\,D - F_{n-1}\,E = F_{n+1}\,L_n - F_{n-1}\,E = V = \text{intervalle initial} \\
D &= (L_0 + F_{n-1}\,E)/F_{n+1}\,.
\end{aligned}
$$

L'inégalité évidente

$$\frac{F_{n-1}}{F_{n+1}} \leqslant \frac{1}{2}$$

montre qu'il suffit de choisir n tel que

$$F_{n+1} \geqslant \frac{V}{\varepsilon - \dfrac{E}{2}}$$

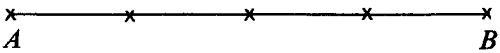
pour que l'on ait $D \leqslant \varepsilon$. Pour que n soit le plus petit possible (ce que l'on souhaite) il faut prendre E aussi petit que le permet la précision des calculs et en particulier E petit devant ε.

IV. CAS $n = 4$. MÉTHODE PROPOSÉE

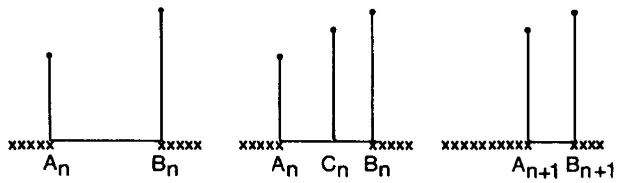
1. Phases de l'étude

1re *Phase*

3 points au maximum sont à calculer. Ceci se fait de gauche à droite ou de droite à gauche

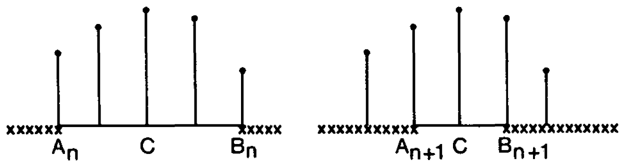


Nous partons de l'extrémité où la valeur de la fonction est la plus grande. Si au premier point calculé, la fonction a une valeur inférieure à celle de l'extrémité de départ, on peut prendre cette nouvelle valeur comme deuxième borne et l'intervalle est alors divisé par 4. Lorsque l'une des extrémités est le maximum recherché, cette méthode est meilleure que celles du nombre d'Or et de Fibonnacci.



2e *Phase*

Dès que le premier point calculé n'a plus une valeur inférieure à celle de l'extrémité de départ, l'extremum recherché ne pourra plus être encadré par 2 points consécutifs, il en faudra 3.



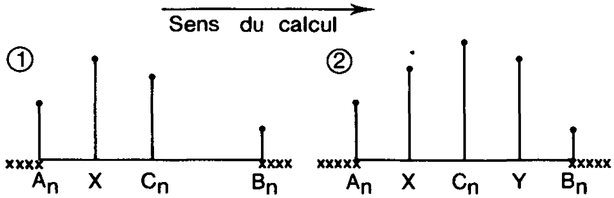
Soit $M = \max(f(A_n), f(B_n))$.

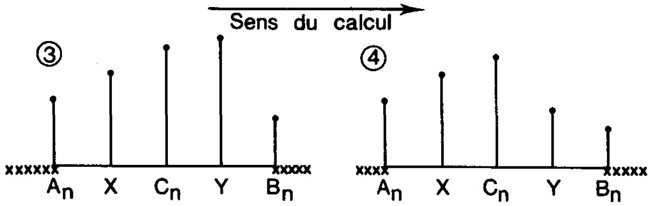
Tous les points compris entre M et $f(C)$ sont supérieurs à M (hypothèse d'unimodalité) donc il est exclu de revenir à la phase 1.

Ordre de calcul des points.

Dans cette phase, si l'on excepte l'initialisation, ce que nous ferons, il y aura deux points à calculer par itération (la valeur aux bornes est connue et la valeur médiane également).

Soient C_n la valeur au centre et A_n, B_n les valeurs aux bornes au cours de la n-ième itération.





Cas ① X est calculé mais pas Y, l'intervalle devient $A_n\ C_n$.

Cas ② et ④ X et Y sont calculés, l'intervalle devient XY.

Cas ③ X et Y sont calculés, l'intervalle devient $C_n\ B_n$.

Des considérations probabilistes nous amènent à adopter la tactique suivante : nous supposons que l'extrémité dont la valeur est la plus grande est en général plus proche que l'autre du maximum. Par conséquent nous supposons que le cas ① où le calcul de Y est inutile sera plus fréquent que le cas ③ :

— dans un premier temps nous utilisons l'ordre de calcul qui vient d'être exposé ;

— tant que cela s'avère judicieux nous poursuivons la même tactique ;

— dès que cet ordre paraît mauvais nous adoptons l'ordre inverse, etc...

2. Nombre moyen d'opérations au cours de la 2e phase

Le calcul qui suit ne tient pas compte des améliorations apportées par la première phase ni de l'ordre du calcul dans la deuxième phase.

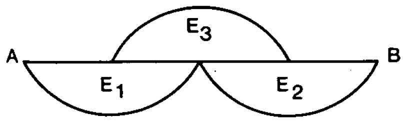
Nombre maximum d'itérations :

$$K = \frac{\text{Log}\,(V\varepsilon^{-1})}{\text{Log}\,2}\,.$$

Nombre moyen de calculs

$$\overline{N} = K \times \sum p_i\, O_{pi}$$

Notations :



Nombre de calculs nécessaires et probabilités.

E_i	0_{p_i}	p_i
E_1	1	$\frac{3}{8}$
E_2	2	$\frac{3}{8}$
E_3	2	$\frac{1}{4}$

De plus l'initialisation de la recherche nécessite trois calculs : deux pour les bornes et un pour le point du milieu. D'où

$$\bar{N} = K \times \sum_i p_i \, 0_{p_i} + 3$$

$$\bar{N} = \frac{\text{Log}\,(V\varepsilon^{-1})}{\text{Log}\,2}\left[\frac{3}{8} + \frac{3}{4} + \frac{1}{2}\right] + 3$$

$$\bar{N} = 3 + \frac{13}{8}\,\frac{\text{Log}\,(V\varepsilon^{-1})}{\text{Log}\,2}.$$

3. Comparaison avec la méthode du nombre d'or

Pour cette méthode, le nombre de calculs est donné par la formule :

$$\bar{N} = 1 + \frac{\text{Log}\,(V\varepsilon^{-1})}{\text{Log}\,(1{,}62)} \quad \text{car} \quad \frac{v}{\tau^{N-1}} = \varepsilon\,.$$

Reprenons nos trois intervalles à savoir E_1, E_2 et E_3. La question que nous voulons résoudre est la suivante : quelles valeurs devraient avoir les p_i pour que notre formule soit, en moyenne, équivalente à la formule du nombre d'Or ?

Au cours de notre recherche, nous tenons le plus grand compte de tous les renseignements obtenus, et nous abordons chaque nouvelle itération en tentant de partir de l'extrémité la plus favorable. Considérons la figure précédente et supposons que $f(A)$ est supérieur à $f(B)$. Nous pensons que dans ces conditions, la probabilité d'avoir l'extremum dans l'intervalle E_1 est supérieure à celle de le trouver dans l'intervalle E_2. Celle de le trouver dans l'intervalle E_3 n'étant pas changée.

Les nouvelles valeurs des p_i deviennent alors :

$$p(E_1) = p$$
$$p(E_2) = \frac{3}{4} - p$$
$$p(E_3) = \frac{1}{4}.$$

Nous cherchons à obtenir :

$$\overline{N} = K \times p_i 0_{p_i} + 3 = \frac{\text{Log}(V\varepsilon^{-1})}{\text{Log}(1{,}62)} + 1.$$

D'où

$$\frac{\text{Log}(V\varepsilon^{-1})}{\text{Log } 2} p + \frac{1}{2} + 2 \times \left(\frac{3}{4} - p\right) + 3 = \frac{\text{Log}(V\varepsilon^{-1})}{\text{Log}(1{,}62)} + 1 .$$

Lorsque $V\varepsilon^{-1}$ est grand, les termes constants sont négligeables, d'où la relation

$$\frac{(2-p)}{\text{Log } 2} \triangleq \frac{1}{\text{Log}(1{,}62)}.$$

Soit $$p \triangleq 0{,}56.$$

V. APPLICATIONS NUMÉRIQUES

1° **Etude et comparaison** de nos résultats avec la méthode de Fibonacci sur les exemples étudiés par M. Hasles [1] :

$F(X) = (\text{Log}(X) - 3x + 3)^2$ sur $(0, 1\,E - 30, E + 9)$ $\varepsilon = E - 6$

$F_1(X) = (ABS(X - \text{ALPHA}))\frac{1}{100}$ sur $(0, 1)$ $\varepsilon = E - 10$

$F_2(X) = \text{Log}(X)$ sur $(E - 25, E - 5)$
$= X$ sur $)E - 5, E5)$ $\varepsilon = E - 10$

$F_3(X) = -X$ sur $(-E4, E10)$
$= \text{Log}(X)$ sur $)E10, 1)$ $\varepsilon = E - 10$
$= X$ sur $)1, E4)$

TABLEAU COMPARATIF

		FIBONACCI	NOS RÉSULTATS
F		73	53
F_1	ALPHA = 0,45	49	53
	ALPHA = 0,925	49	50
F_2		73	28
F_3		70	67
Total		314	251

Sur ces exemples, notre méthode se révèle supérieure à celle de Fibonacci.

2° Etude d'une famille de polynômes

$$P(X) = X^6 - AX^5 + 3X^3 + 2X + 1$$

pour A variant de 120 à 149. L'intervalle initial : 50.

Nous donnons ci-après un extrait des résultats : une colonne sur trois. La première ligne indique le nombre de calculs pour la méthode de Fibonacci.

L'ensemble des résultats porte sur 1 200 tests totalisant 40 000 calculs.

L'étude des résultats tend à montrer que les deux méthodes ont des performances équivalentes. Un examen approfondi des résultats montre que la première phase fait gagner beaucoup de points. Le nombre de calculs nécessités par la méthode que nous proposons se trouve diminué de deux si les valeurs de la fonction aux deux bornes sont connues à l'avance. Ces valeurs ne sont pas utilisées par les méthodes de Fibonacci et du nombre d'Or.

CONCLUSION

La méthode qui vient d'être exposée donne expérimentalement des résultats comparables à ceux de la méthode de Fibonacci. Elle possède par rapport à cette dernière et par rapport à toutes les autres du même type (nombre d'Or, pseudo-dichotomie) l'originalité de s'adapter à l'information recueillie, au fur et à mesure du calcul des différents points de la fonction, et il lui arrive d'être beaucoup plus rapide. L'ordre de calcul des points est déterminé au début de chaque itération, en fonction des résultats trouvés au cours des itérations précédentes. Notre méthode est dynamique. En fait elle s'apparente aux méthodes qui utilisent la dérivée, — dont ici nous ne supposons pas l'existence — remplacée ici par la corde.

A	8	13	17	21	26	30	34	39	43	47	52	56	60
120	5	6	8	9	111	12	14	15	17	18	20	21	23
121	6	10	14	18	24	28	32	36	42	46	50	54	59
122	6	11	16	20	25	28	33	38	41	46	49	53	58
123	7	11	16	20	24	29	34	39	43	48	53	58	62
124	7	11	16	21	26	31	36	42	46	50	55	61	66
125	6	9	12	15	19	23	27	32	37	41	44	47	
126	7	12	16	21	24	28	32	38	43	47	50	55	59
127	8	12	16	21	25	29	35	40	44	48	54	58	63
128	8	12	16	22	27	32	35	41	45	48	52	57	61
129	8	13	17	22	27	32	35	39	43	49	53	58	62
130	7	10	13	16	20	26	32	37	42	48	52	55	60
131	8	12	17	22	26	31	35	38	43	46	50	53	58
132	9	13	18	22	27	32	36	41	45	49	53	57	61
133	8	13	18	22	26	30	35	40	46	51	56	60	64
134	9	14	18	22	27	31	36	39	44	49	52	57	61
135	10	16	22	28	34	38	43	48	53	57	61	66	70
136	9	14	18	22	28	33	37	42	45	49	53	57	62
137	8	13	18	22	27	32	37	43	46	49	54	60	65
138	9	13	18	22	26	30	34	37	43	47	52	57	62
139	8	12	17	22	27	31	35	40	45	49	50	57	62
140	7	10	13	16	20	24	30	34	38	41	45	49	55
141	8	13	17	22	25	30	33	37	41	45	50	55	60
142	9	13	17	22	26	31	35	41	45	50	56	61	
143	9	13	17	23	28	34	39	42	47	52	57	60	64
144	9	14	18	23	27	32	37	41	46	50	54	58	63
145	8	11	14	17	20	25	30	35	40	45	48	52	55
146	9	13	18	23	27	31	36	41	45	50	54	58	63
147	10	14	19	23	28	32	37	41	44	49	53	57	62
148	9	14	19	23	27	30	34	39	44	45	53	58	62
149	10	15	19	23	28	33	37	42	45	50	55	59	
gain (–) ou perte (+)	+1	-23	-15	-8	-20	-11	+1	-17	-1	+6	-19	-12	+3

BIBLIOGRAPHIE

[1] J. HASLES, *Algorithmes de recherche de l'optimum d'une fonction d'une seule variable*, Institut Français du Pétrole, sept. 1970.

[2] J. VIGNES, Thèse. Etude et mise en œuvre d'algorithmes de recherche d'un extremum d'une fonction de plusieurs variables, février 1969.

[3] WILDE, *Méthodes de recherches d'un optimum*, Dunod, 1966.