

APUNTS DE MATEMÀTIQUES. NÚMERO 3

Mètodes numèrics de minimització de funcions de diverses variables

CARLES BATLLE, MERCÈ CLAVEROL I ESTER SIMÓ

Dept. de Matemàtica Aplicada i Telemàtica

E.U.P. de Vilanova i la Geltrú

Universitat Politècnica de Catalunya

Setembre 1990

Sumari

1	Introducció	1
2	Minimització unidimensional	3
2.1	El mètode de la <i>relació d'or</i>	3
2.2	Mètodes que utilitzen la derivada	10
2.3	Condicions de finalització	13
3	Minimització multidimensional	13
3.1	El mètode del <i>descens màxim</i>	14
3.2	El mètode del <i>gradient conjugat</i>	16
3.3	Un exemple senzill	22
4	El mètode de la cristallització simulada	24
A	Significat geomètric del gradient	31
B	Codi en Pascal per a la funció de Rosenbrook	33

1 Introducció

Imagineu que us donen una funció real f que depèn d'una o més variables reals independents i que us demanen que trobeu els valors de les variables que fan que f assoleixi el seu valor màxim o mínim i que calculeu aquest. Els dos problemes, maximització i minimització, estan trivialment relacionats, ja que sols cal canviar f per $-f$. A més, si us esteu guanyant el pa amb això, es tracte de que ho feu en poc temps, gastant poc i utilitzant pocs recursos. Com que la majoria de vegades l'esforç de càlcul ve determinat per la dificultat en avaluar f i, probablement, les seves derivades, tot això es tradueix en el següent principi fonamental: avalueu f com menys vegades millor.

Un extrem, màxim o mínim, pot ser global (el valor més gran o més petit de la funció en tot el seu domini) o local (el valor més gran o més petit en un entorn finit d'un punt). No es coneixen mètodes generals segurs que permetin trobar extrems globals de funcions de vàries variables. Hi ha, però, dues estratègies que tothom utilitza: (i) busqueu extrems locals començant per punts ampliament distribuïts per tot el domini i agafeu el més extrem dels que obtingueu, o (ii) pertorbeu un extrem local, allunyant-vos una mica d'ell, i vegeu si aneu a parar a un punt millor o torneu sempre al mateix lloc. Hi ha indicis de que els anomenats *mètodes termodinàmics* poden aportar un progrés important en aquest camp.

El tema que ens ocupa podria anomenar-se *optimització*, que és el nom usual per aquest ampli camp del càlcul numèric. La importància dels diversos tipus de problemes que es presenten depèn dels vostres interessos particulars. Els economistes i alguns enginyers estan especialment interessats en l'anomenada *optimització condicionada*, a on hi ha limitacions *a priori* sobre els valors de les variables independents. Per exemple, el nombre d'unitats d'un determinat producte que una factoria produeix en un dia no pot ser negatiu, i tampoc pot sobrepassar un cert valor, determinat per la capacitat dels proveïdors de matèries primeres que aquesta factoria té contractats, o per les quantitats que n'hi ha d'amagatzemades. Una àrea particularment ben coneguda de l'optimització condicionada és la *programació lineal*, a on tant la quantitat que s'ha d'optimitzar com les condicions que s'han de verificar són funcions lineals de les variables independents. Nosaltres no ens ocuparem aquí de l'optimització condicionada. El lector interessat pot consultar [3]. En els casos més senzills, hom pot resoldre les condicions, si aquestes són igualtats, i treballar amb un nombre menor de variables independents, o introduir multiplicadors de Lagrange, augmentant així el nombre de variables.

Per simplicitat, presentarem en forma explícita sols els mètodes de minimització. Si voleu maximitzar quelcom, apliqueu el mateix canviant el signe de la vostra funció. Hem organitzat el treball de la següent manera. A la secció 2 presentem els mètodes de minimització de funcions d'una variable, que s'utilitzaran

com a eines de treball pels mètodes de diverses variables. A la secció 3 discutim els mètodes del *descens màxim* i del gradient conjugat per funcions de diverses variables independents i els apliquem a un cas senzill de dues variables. Finalment, a la secció 4 presentem les idees fonamentals del mètode termodinàmic i l'apliquem a l'anomenat *problema del representant comercial*. A l'apèndix A presentem una breu exposició del significat geomètric del gradient i al B donem el codi en Pascal del mètode del gradient conjugat per a l'exemple de la secció 3.

La bibliografia que presentem al final permet ampliar els temes tractats i trobar informació sobre tot allò que no hem tocat.

2 Minimització unidimensional

2.1 El mètode de la *relació d'or*

Descriurem en primer lloc un mètode que utilitza sols la funció que volem minimitzar i no la derivada. El mètode requereix el coneixement d'una terna inicial de punts

$$a < b < c$$

tals que

$$f(a) > f(b) \quad \text{i} \quad f(b) < f(c)$$

tal com mostra, per exemple, la figura 1. Això assegura que, si la funció és contínua, hi ha un mínim entre a i c . Hom pot generar la terna inicial a partir de dos punts, anant en la "direcció de baixada". Per exemple si $f(x_1) > f(x_2)$ i $x_1 < x_2$, llavors hom pot anar a la dreta de x_2 fins a trobar un punt x_3 tal que $f(x_3) > f(x_2)$. Obviament això pot no acabar mai si la funció decreix indefinidament, o si assoleix un valor asimptòtic més petit que $f(x_2)$. Un cert coneixement de la funció és per tant imprescindible.

Una vegada tenim la terna inicial, agafem un punt d entre b i c (el podríem agafar entre a i b). Llavors, si $f(b) < f(d)$ seguim amb a, b, d (figura 2), mentre

que si $f(b) > f(d)$ seguim amb b , d i c (figura 3). Amb el nou interval podem seguir endavant generant noves ternes. A cada pas, el punt central de la terna és la nostra aproximació al mínim. Si el mínim es troba a $x = x^*$, hom pot esperar acotar-lo d'aquesta manera dins l'interval

$$(1 - \epsilon)x^* < x^* < (1 + \epsilon)x^*$$

on ϵ és la precisió de l'ordinador, és a dir, ϵ és el real positiu més petit que l'ordinador pot representar tal que $1 + \epsilon \neq 1$. En realitat hom no pot arribar tan lluny. En general, la forma de $f(x)$ vora el mínim serà

$$f(x) \approx f(x^*) + \frac{1}{2}f''(x^*)(x - x^*)^2$$

El segon terme serà, d'acord amb la precisió de l'ordinador, negligible comparat amb el primer quan

$$|x - x^*| \leq \sqrt{\epsilon} \sqrt{\frac{2|f(x^*)|}{|f''(x^*)|}} = \sqrt{\epsilon}|x^*| \sqrt{\frac{2}{(x^*)^2} \frac{|f(x^*)|}{|f''(x^*)|}}$$

Per a la majoria de funcions

$$\sqrt{\frac{2|f(x^*)|}{(x^*)^2|f''(x^*)|}} \sim 1$$

i per tant, per a aquells x tals que

$$\frac{|x - x^*|}{|x^*|} < \sqrt{\epsilon} \quad (1)$$

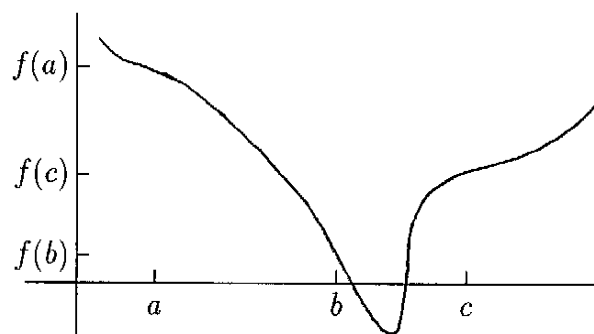


Figura 1: Terna inicial per al mètode de la *relació d'or*.

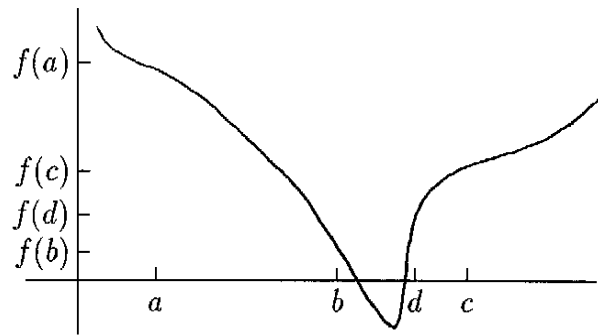


Figura 2: Segon pas amb $f(d) > f(b)$: continuem amb a, b, d .

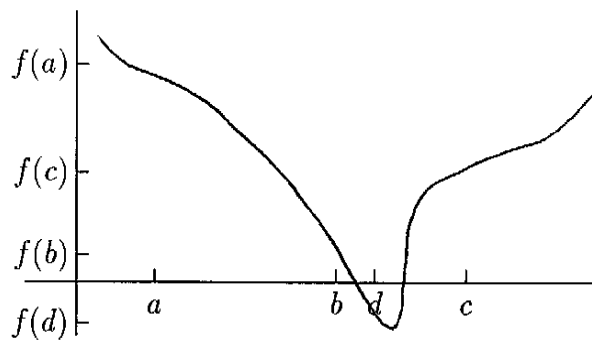
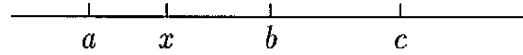


Figura 3: Segon pas amb $f(d) < f(b)$: continuem amb b, d, c .

Figura 4: x a l'esquerra de b .

tindrem que $f(x) = f(x^*)$. No té per tant sentit acostar-se a x^* una distància menor que $|x^*| \sqrt{\epsilon}$, a menys que tinguem alguna raó per a suposar que

$$\sqrt{\frac{2 |f(x^*)|}{(x^*)^2 |f''(x^*)|}} \ll 1$$

Cal ara decidir una estratègia per escollir el següent punt x a partir de a , b , c . Suposarem que b està a una fracció w de la distància entre a i c :

$$\frac{b-a}{c-a} = w, \quad \frac{c-b}{c-a} = 1-w \quad (2)$$

i suposem a més que el següent punt x està a una fracció addicional z més enllà de b

$$\frac{x-b}{c-a} = z \quad (3)$$

Com que no estem presuponant que x està a la dreta de b , z pot ser negatiu. Llavors la següent terna és a, b, x o b, x, c . Suposem que x es trobi a l'esquerra de b , es a dir, $z < 0$ com indica la figura 4. Si la terna es a, b, x , el nou segment té longitud w respecte al a, b, c , mentre que si és x, b, c , la longitud és $1-w-z$ (recordem que z és ara negatiu). Si volem que el nou segment sigui el més petit possible i volem minimitzar el pitjor cas, haurà de ser

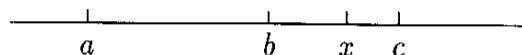
$$w = 1 - w - z$$

és a dir

$$z = 1 - 2w \quad (4)$$

Suposem ara que x es trobi a la dreta de b (figura 5). El mateix raonament ens porta a demanar

$$w + z = 1 - w$$

Figura 5: x a la dreta de b

i de nou anem a parar a la fórmula 4

$$z = 1 - 2w$$

Això ens indica, a més, si hem de buscar x a la dreta o a la esquerra de b . Si b està més aprop de c que de a , serà $w > 1/2$, $z > 0$ i x a la dreta de b . Per tant, sempre hem de buscar x dins el més gran dels subintervalls.

D'on ha vingut, però, el valor de w ? Hom pot demanar que hagi vingut d'un pas anterior a on haguem aplicat la mateixa estratègia, és a dir, voldrem que x sigui la mateixa fracció del camí de b a c que b ho era del camí de a a c . Si $x > b$ això vol dir

$$\frac{x - b}{c - b} = \frac{b - a}{c - a}$$

d'on

$$\frac{(x - b)/(c - a)}{(c - b)/(c - a)} = \frac{b - a}{c - a}$$

i finalment

$$\frac{z}{1 - w} = w \tag{5}$$

Si $x < b$, serà

$$\frac{b - x}{b - a} = \frac{c - b}{c - a}$$

i s'arriba a

$$\frac{-z}{1 - w} = w \tag{6}$$

Les equacions (5) i (6) poden resumir-se en

$$\frac{|z|}{1 - w} = w \tag{7}$$

Si ara combinem això amb (4) tenim

$$|1 - 2w| = w - w^2$$

Considerem de nou els dos casos possibles:

- cas 1. $1 - 2w \geq 0$ (b més a la vora de a que de c). Tindrem que

$$w^2 - 3w + 1 = 0$$

$$w = \frac{3 \pm \sqrt{5}}{2}$$

Com que $w \leq 1$, hem d'agafar

$$w = \frac{3 - \sqrt{5}}{2} \equiv R \approx 0.38197 \quad (8)$$

Aquest valor R s'anomena "la relació d'or". Tenim llavors

$$z = 1 - (3 - \sqrt{5}) = -2 + \sqrt{5} \approx 0.23607 \quad (9)$$

- cas 2. $1 - 2w < 0$ (b més aprop de c que de a) Serà ara

$$w^2 + w - 1 = 0$$

$$w = \frac{-1 \pm \sqrt{5}}{2}$$

Com que $w \geq 0$ ens quedem amb

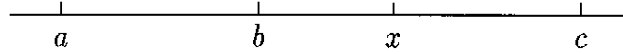
$$w = \frac{-1 + \sqrt{5}}{2} = 1 - R \approx 0.61803$$

i tenim

$$z = 1 - (-1 + \sqrt{5}) = 2 - \sqrt{5} \approx -0.23607$$

El signe de z confirma el que ja sabíem sobre quin dels dos subintervalls hem d'escollir per a provar un nou punt, es a dir, el subinterval més gran. Quina distància ens endinsem en aquest subinterval per posar-hi x ? En el cas 1, $1 - 2w \geq 0$ i $x > b$, tenim

$$\frac{x - b}{c - b} = \frac{b - a}{c - a} = w = R$$

Figura 6: Cas en que $x > b$.

de b cap a c . En el cas 2, $1 - 2w < 0$ i $x < b$, tenim

$$\frac{b-x}{b-a} = \frac{c-b}{c-a} = 1-w = 1-(1-R) = R$$

de b cap a a . Resumint, en un determinat moment del càlcul, quan tenim una terna de punts, el següent punt que s'ha de provar és el que es troba una fracció $R = 0.38197$ dins l'interval més gran, mesurant des del punt central de la terna.

Pot demostrar-se que si comencem amb tres punts que no estan en la proporció marcada per R , aquest algorisme tendeix a establir la proporció.

Quin és el grau de convergència de l'algorisme? Una vegada s'han establert les proporcions marcades per R , la longitud del nou interval és, en el cas $w \leq 1/2$, segons indica la figura 6. L'interval serà o el a, b, x amb longitud

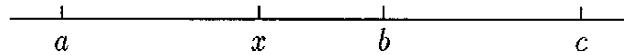
$$\begin{aligned} x-a &= (x-b) + (b-a) = z(c-a) + w(c-a) \\ &= (z+w)(c-a) = (1-w)(c-a) \\ &= (1-R)(c-a) = 0.61803(c-a) \end{aligned}$$

o el b, x, c amb longitud

$$\begin{aligned} c-b &= (1-w)(c-a) = (1-R)(c-a) \\ &= 0.61803(c-a) \end{aligned}$$

Si $w > 1/2$ (veure figura 7), si l'interval és a, x, b tindrem

$$b-a = (c-a)w = (c-a)(1-R) = 0.61803(c-a)$$

Figura 7: Cas en que $x < b$.

mentre que si és x, b, c serà

$$\begin{aligned}
 c - x &= (b - x) + (c - b) = (-z)(c - a) + (1 - w)(c - a) \\
 &= (-z + 1 - w)(c - a) = (-1 + 2w + 1 - w)(c - a) \\
 &= w(c - a) = (1 - R)(c - a) = 0.61803(c - a)
 \end{aligned}$$

Per tant, en qualsevol cas la convergència és lineal, amb un factor $1 - R = 0.61803$.

2.2 Mètodes que utilitzen la derivada

Volem ara aconseguir la localització del mínim dins la terna $a < b < c$ però utilitzant la capacitat de calcular la primera derivada de la funció a més de la pròpia funció. En principi, podríem simplement buscar un zero de la derivada, ignorant la informació dels valors de la funció, utilitzant algun mètode de càlcul d'arrels. Això pot portar, però, problemes. En primer lloc, l'anul·lació de la primera derivada no garanteix que tinguem un mínim: pot tractar-se d'un màxim o d'un punt d'inflexió. A més, els mètodes de càlcul d'arrels poden ser molt lents (bisecció i *regula falsi*) o no convergir (secant i Newton). Si bé això pot no ser important en una minimització d'una funció d'una variable, en cas que hom utilitzi la minimització unidimensional com una subrutina d'un mètode de minimització de vèries variables, aquestes consideracions poden ser crítiques.

No volem abandonar l'estratègia de mantenir el mínim encerclat tota l'estona. L'única manera d'aconseguir això és millorar el mínim utilitzant sols la informació aportada per la funció, no la seva derivada, fent que el punt central de la terna

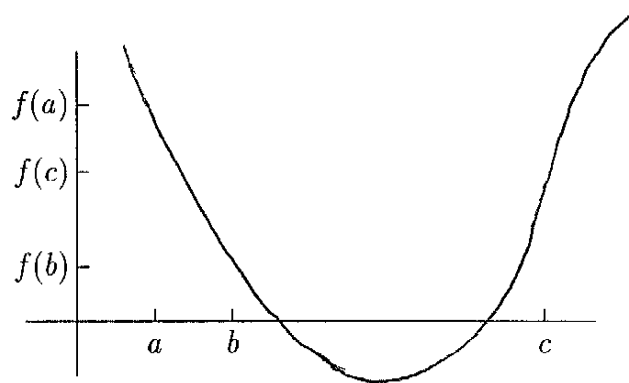


Figura 8: Terna inicial. El signe de $f'(b)$ (negatiu) indica que el següent punt estarà entre b i c .

sigui sempre el que tingui el valor més baix. Per tant, el paper de la derivada ha de ser sols ajudar a triar nous punts dins la terna.

Entre les moltes estratègies possibles, en presentarem una de caire conservador. En primer lloc, el signe de la derivada en el punt central b de la terna $a < b < c$ indica de manera única si el següent punt ha de ser agafat dins l'interval (a, b) o dins el (b, c) . Llavors agafem els valors de la derivada en el punt b i en el segon millor punt fins el moment, és a dir, c si $f(c) < f(a)$ i a en cas contrari i extrapolem la posició del mínim mitjançant el mètode de la secant, que, particularitzat a aquesta situació, més endavant descriurem. Si aquesta posició no cau dins l'interval que volem, simplement dividim l'interval per la meitat i seguim endavant amb la nova terna. Anem a descriure això amb més detall. Sigui una situació com la de la figura 8. El següent punt haurà d'estar dins (b, c) , ja que $f'(b) < 0$. El valor actual del mínim és b , i el segon millor valor és c , ja que $f(c) < f(a)$. La gràfica de la funció derivada és quelcom del tipus indicat a la figura 9. Calculem aleshores una secant que passi per $(b, f'(b))$ i $(c, f'(c))$:

$$y = f'(b) + \frac{f'(c) - f'(b)}{c - b}(x - b)$$

i en busquem el zero, que anomenem d ,

$$0 = f'(b) + \frac{f'(c) - f'(b)}{c - b}(d - b)$$

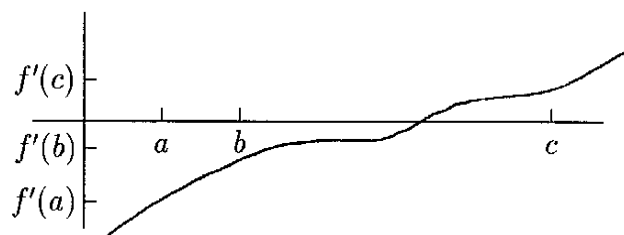


Figura 9: La derivada de la funció de la figura anterior té un zero entre b i c .

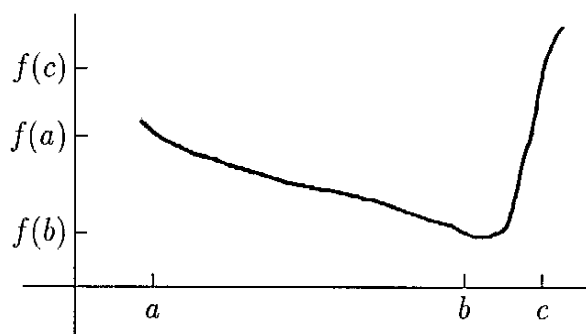


Figura 10: Una altra terna inicial.

d'on

$$d = b - f'(b) \frac{c - b}{f'(c) - f'(b)}$$

i aquest és el nou punt. En el cas dibuixat, d es troba segur dins (b, c) . Imaginem però que tenim el cas de la figura 10, amb $f(a) < f(c)$ i que la derivada és la donada per la figura 11. En aquest cas, d podria caure fora de (b, c) i el que fariem llavors seria simplement biseccionar:

$$d = \frac{c + b}{2}$$

Si el mètode de la secant funciona a cada pas, la convergència és, en el límit de molts passos, superlineal amb exponent 1.618... Si ϵ_i és la distància al mínim exacte en el pas i , es verifica, en el cas esmentat,

$$\epsilon_{n+1} = \text{constant} \cdot (\epsilon_n)^{1.618\dots}$$

Si és necessari posar pel mig biseccions, obviament tindrem quelcom més lent que això.

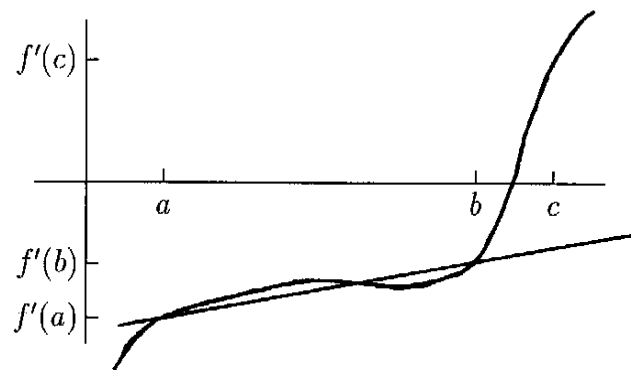


Figura 11: El zero de la derivada de la funció de la figura anterior, calculat amb el mètode de la secant, cau fora de l'interval (b, c) .

2.3 Condicions de finalització

Finalment, un comentari sobre la condició de finalització dels algorismes que hem presentat en aquesta secció. Tal com hem dit, no té sentit buscar una aproximació x al mínim x^* tal que

$$|x - x^*| < |x^*| \sqrt{\epsilon}$$

Si a, b, c és un terna i d és el següent punt, podem suposar que

$$|x - x^*| \sim \frac{|c - a|}{2}$$

i que

$$|x^*| \sim \frac{|b| + |d|}{2}$$

Per tant és convenient donar l'algorisme per acabat si

$$|c - a| \leq \sqrt{\epsilon} (|b| + |d|)$$

3 Minimització multidimensional

Sabem ja com minimitzar una funció d'una variable. Si comencem en un punt \mathbf{P} en un espai N -dimensional, i ens movem des d'aquest punt en una direcció \vec{n} , llavors qualsevol funció de N variables $f(\vec{x})$ pot ser minimitzada al llarg de la línia $\mathbf{P} + \lambda \vec{n}$ mitjançant un dels nostres mètodes unidimensionals. Podem llavors

intentar elaborar un mètode de minimització multidimensional que consisteixi en una seqüència de minimitzacions sobre línies. Els diferents mètodes diferiran sols en com, a cada pas, escollim la següent direcció. Podem classificar-los segons utilitzin o no el gradient de la funció. Nosaltres sols estudiarem alguns dels primers, concretament el mètode del descens màxim i el mètode del gradient conjugat. En qualsevol cas, no cal especificar si el mètode unidimensional utilitza o no el gradient, es a dir, la derivada, per fer la seva feina.

Més informació sobre aquests mètodes, així com sobre altres mètodes que utilitzen la matriu hessiana ,formada per les derivades de segon ordre, o que intenten construir-la, pot trobar-se a [1] i a [2].

3.1 El mètode del *descens màxim*

Suposem que sou capaços de calcular, en un punt \mathbf{P} , no tan sols el valor de la funció $f(\mathbf{P})$, sinó també el gradient $\vec{\nabla}f(\mathbf{P})$.

Suposem que el punt que tenim està prou a la vora del mínim com per a considerar que la funció pot ser aproximada per una forma quadràtica

$$f(\vec{x}) \approx c - \vec{b}\vec{x} + \frac{1}{2}\vec{x}\hat{A}\vec{x}$$

a on

$$\begin{aligned} c &= f(\vec{0}) \\ b_i &= -\frac{\partial f}{\partial x^i}(\vec{0}) \\ \hat{A}_{i,j} &= \frac{\partial^2 f}{\partial x^i \partial x^j}(\vec{0}) \end{aligned}$$

Obviament, no coneixem c , \vec{b} , \hat{A} . Com que suposem que f es porta prou bé, les derivades creuades són iguals i \hat{A} és per tant una matriu simètrica¹. Tenim així que el nombre de paràmetres que determinen la situació del mínim (donada

¹Si es tracte d'un mínim, \hat{A} haurà de ser a més definida positiva, és a dir, tots els seus valors propis seran estrictament positius.

per $\hat{A}^{-1}\vec{b}$) és:

$$\begin{aligned} \vec{b} &\rightarrow N \\ \hat{A} &\rightarrow \frac{N^2 - N}{2} + N = \frac{N^2 + N}{2} \\ \text{total} &\rightarrow \frac{N^2 + 3N}{2} \end{aligned}$$

Això és de l'ordre de N^2 . Per tant, la localització d'un mínim implica, al menys, la determinació de N^2 nombres reals i hom pot esperar que haurà de calcular f al menys en N^2 punts. Si per tal de calcular la direcció al llarg de la qual s'ha de minimitzar s'utilitza el gradient, al ser aquest un vector amb N components reals, hom esperarà que, si té sort, amb N minimitzacions unidimensionals haurà acabat. De fet, donat que la informació continguda en el gradient és altament redundant (podem generalment utilitzar una part dels càlculs fets per a una component per a calcular les altres), cal doncs esperar que sigui necessari bastant menys que N^2 evaluacions de la funció per a trobar el mínim. Aquest argument qualitatiu dona una idea de la millora que pot suposar utilitzar el gradient de la funció.

La manera més senzilla (i ineficaç) d'incorporar l'informació aportada pel gradient ens porta a l'anomenat mètode del descens màxim, basat en el fet que $\vec{\nabla}f$ indica la direcció del màxim increment local de la funció i, per tant, $-\vec{\nabla}f$ indicarà en quina direcció la funció disminueix més depressa.

D'una manera gràfica es podria comparar aquest algorisme al retorn d'un esquiator des d'una montanya fins al fons d'una vall amb una boira espessa. Incapaç de veure la seva meta comença a baixar en la direcció de la màxima pendent i continua baixant fins que el camí comença a pujar de nou. Aleshores escull una segona direcció de pendent més pronunciada i torna a fer un segon recorregut del mateix tipus.

L'algorisme d'aquest mètode es podria resumir de la següent manera. Comencem en un punt \mathbf{P}_1 . Calculem $\vec{\nabla}f(\mathbf{P}_1)$ i minimitzem al llarg de $\mathbf{P}_1 - \lambda\vec{\nabla}f(\mathbf{P}_1)$. Arribem a \mathbf{P}_2 . Calculem $\vec{\nabla}f(\mathbf{P}_2)$ i minimitzem al llarg de $\mathbf{P}_2 - \lambda\vec{\nabla}f(\mathbf{P}_2)$.

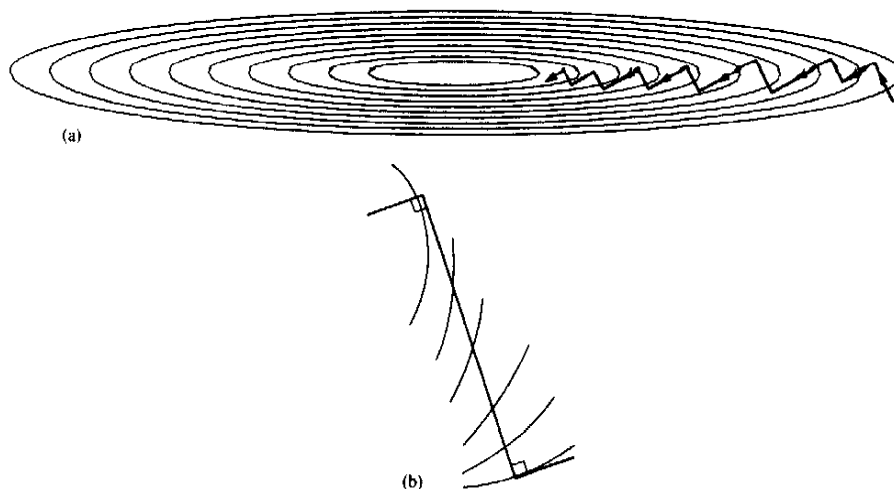


Figura 12: (a) El mètode del *descens màxim* és altament ineficaç. (b) A cada pas hi ha un gir en angle recte.

Procedim així fins que arribem a un punt on $\vec{\nabla}f$ s'anulla i aquest serà el nostre mínim.

Aquest mètode no és particularment bo degut a que les “valls” estretes fan donar-li molts pasos, fins i tot si la vall és una forma quadràtica perfecta. Hom podria esperar que, en dues dimensions, el primer pas ens portés al centre de la vall i el segon directament al mínim al llarg de l'eix gran; però el nou gradient en el punt on hem arribat per una minimització unidimensional és perpendicular a la direcció que hem seguit. Per tant, cal donar un gir en angle recte i això no ens porta, en general, al mínim, tal com mostra la figura 12.

3.2 El mètode del *gradient conjugat*

Hem vist que el mètode del descens màxim pot ser molt ineficient en certes situacions i, de fet, a la pràctica hom pot contar amb trobar-s'hi. Com podem evitar aquest malversament del esforç de càlcul? La resposta ve donada pel que s'anomena un conjunt de direccions conjugades.

Amb l'aproximació quadràtica de la subsecció anterior, el gradient és:

$$\vec{\nabla} f(\vec{x}) = \hat{A}\vec{x} - \vec{b}.$$

Com canvia el gradient de f si ens movem al llarg d'alguna direcció? Obviament

$$\vec{\nabla} f(\vec{x} + \delta\vec{x}) = \hat{A}(\vec{x} + \delta\vec{x}) - \vec{b} = \vec{\nabla} f(\vec{x}) + \hat{A}\delta\vec{x}.$$

Imaginem que ens hem mogut al llarg d'una direcció \vec{u} fins un mínim i ens proposem ara moure'ns al llarg d'una altra direcció \vec{v} . La condició de que el moviment sobre \vec{v} no espatlli la minimització al llarg de \vec{u} és que el gradient resti perpendicular a \vec{u} :

$$0 = \vec{u} \cdot \vec{\nabla} f(\vec{x} + \delta\vec{x}) = \vec{u} \cdot \vec{\nabla} f(\vec{x}) + \vec{u} \cdot \hat{A}\delta\vec{x} = \vec{u} \cdot \hat{A}\delta\vec{x}.$$

Com que $\delta\vec{x}$ està en la direcció \vec{v} , això implica

$$\vec{u} \cdot \hat{A}\vec{v} = 0.$$

Es diu llavors que les direccions (o els vectors) \vec{u} i \vec{v} són conjugades.

Si f és exactament una forma quadràtica en n dimensions, llavors hom pot esperar (i demostrar) que minimitzant al llarg de n direccions mutuament conjugades dos a dos s'arriba al mínim de la funció.

Anem en primer lloc a donar un algorisme que, a partir del coneixement de la matriu \hat{A} , permet construir un conjunt de direccions conjugades.

Sigui E un espai vectorial real de dimensió n i sigui \hat{A} una matriu simètrica qualsevol definida positiva (és a dir, amb tots els valors propis positius). Sigui \vec{g}_1 un vector qualsevol de E i $\vec{h}_1 = \vec{g}_1$. Construïm ara la seqüència

$$\vec{g}_{i+1} = \vec{g}_i - \lambda_i \hat{A}\vec{h}_i \tag{1}$$

$$\vec{h}_{i+1} = \vec{g}_{i+1} + \gamma_i \vec{h}_i \tag{2}$$

a on λ_i i γ_i estan triats de manera que:

$$\vec{g}_{i+1} \cdot \vec{h}_i = 0 \quad , \quad \vec{h}_{i+1} \cdot \hat{A}\vec{h}_i = 0 \tag{3}$$

La primera condició ens proporciona $0 = \vec{g}_{i+1} \cdot \vec{h}_i = \vec{g}_i \cdot \vec{h}_i - \lambda_i \vec{h}_i \cdot \hat{A} \vec{h}_i$, la qual cosa implica

$$\lambda_i = \frac{\vec{g}_i \cdot \vec{g}_i}{\vec{h}_i \cdot \hat{A} \vec{h}_i} \quad (4)$$

Amb la segona condició obtenim

$$0 = \vec{h}_{i+1} \cdot \hat{A} \vec{h}_i = (\vec{g}_{i+1} + \gamma_i \vec{h}_i) \cdot \hat{A} \vec{h}_i$$

i, per tant,

$$\gamma_i = -\frac{\vec{g}_{i+1} \cdot \hat{A} \vec{h}_i}{\vec{h}_i \cdot \hat{A} \vec{h}_i} \quad (5)$$

Com que \hat{A} és definida positiva, $\vec{h}_i \cdot \hat{A} \vec{h}_i > 0$ i el procediment està ben definit. Tenim llavors el següent

Teorema 3.1 *L'algorithm descrit finalitza per a $m \leq n$ (és a dir, existeix un m tal que $\vec{g}_{m+1} = \vec{h}_{m+1} = 0$) i es verifica que, per a tot $i \leq m$, $j = 1, 2, \dots, i-1$*

$$\vec{h}_i \cdot \hat{A} \vec{h}_j = 0 \quad (6)$$

$$\vec{g}_i \cdot \vec{g}_j = 0 \quad (7)$$

Demostració

Ho farem per inducció. Per a $i = 1$ (6) i (7) no diuen res. Per a $i = 2$ tenim que

$$\begin{aligned} \vec{h}_2 \cdot \hat{A} \vec{h}_1 &= (\vec{g}_2 + \gamma_1 \vec{h}_1) \cdot \hat{A} \vec{h}_1 = \\ &= \left(\vec{g}_2 - \frac{\vec{g}_2 \cdot \hat{A} \vec{h}_1}{\vec{h}_1 \cdot \hat{A} \vec{h}_1} \vec{h}_1 \right) \cdot \hat{A} \vec{h}_1 = 0 \end{aligned}$$

$$\begin{aligned} \vec{g}_2 \cdot \vec{g}_1 &= (\vec{g}_1 - \lambda_1 \hat{A} \vec{h}_1) \cdot \vec{g}_1 = \\ &= \vec{g}_1 \cdot \vec{g}_1 - \frac{\vec{g}_1 \cdot \vec{g}_1}{\vec{h}_1 \cdot \hat{A} \vec{h}_1} \vec{h}_1 \cdot \hat{A} \vec{h}_1 = 0 \end{aligned}$$

Suposem ara que (6) i (7) són vàlides per a $i < m$. Llavors, per a $j = 1, 2, \dots, i$

$$\begin{aligned} \vec{g}_{i+1} \cdot \vec{g}_j &= \vec{g}_i \cdot \vec{g}_j - \lambda_i \vec{g}_j \cdot \hat{A} \vec{h}_i \\ &= \vec{g}_i \cdot \vec{g}_j - \lambda_i (\vec{h}_j - \gamma_{j-1} \vec{h}_{j-1}) \cdot \hat{A} \vec{h}_i \\ &= \vec{g}_i \cdot \vec{g}_j - \lambda_i \vec{h}_j \cdot \hat{A} \vec{h}_i \end{aligned}$$

donat que $\vec{h}_{j-1} \cdot \hat{A}\vec{h}_i = 0$ per hipòtesi d'inducció. Si $j = i$, això és zero per (4). Si $j < i$, és zero per hipòtesi d'inducció.

D'altra banda,

$$\vec{h}_{i+1} \cdot \hat{A}\vec{h}_j = \vec{g}_{i+1} \cdot \hat{A}\vec{h}_j + \gamma_i \vec{h}_i \cdot \hat{A}\vec{h}_j$$

si $j = i$ és zero per (5). Si $j < i$, tenim

$$\vec{h}_{i+1} \cdot \hat{A}\vec{h}_j = \vec{g}_{i+1} \cdot \frac{\vec{g}_j - \vec{g}_{j+1}}{\lambda_j} + \gamma_i \vec{h}_i \cdot \hat{A}\vec{h}_j = 0,$$

per hipòtesi d'inducció.

Que l'algorisme acaba amb $m \leq n$ és evident, ja que si $m > n$, llavors tindriem

$$\vec{g}_m \cdot \vec{g}_j = 0 \quad j = 1, \dots, n$$

i, per tant, com que estem en un espai vectorial de dimensió n i tots els \vec{g}_i , $i = 1, \dots, n$ són perpendiculars dos a dos,

$$\vec{g}_m = 0$$

i, a més,

$$\vec{h}_m = \vec{g}_m + \gamma_{m-1} \vec{h}_{m-1} = \vec{g}_m - \frac{\vec{g}_m \cdot \hat{A}\vec{h}_{m-1}}{\vec{h}_m \cdot \hat{A}\vec{h}_m} = 0. \square$$

Podem reescriure λ_i i γ_i d'altres maneres. Per exemple:

$$\begin{aligned} \lambda_i &= \frac{\vec{g}_i \cdot \vec{g}_i}{\vec{h}_i \cdot \hat{A}\vec{h}_i} = \frac{\vec{g}_i \cdot (\vec{h}_i - \gamma_{i-1} \vec{h}_{i-1})}{\vec{h}_i \cdot \hat{A}\vec{h}_i} = \\ &= \frac{\vec{g}_i \cdot \vec{h}_i}{\vec{h}_i \cdot \hat{A}\vec{h}_i} \end{aligned} \quad (8)$$

$$\begin{aligned} \gamma_i &= -\frac{\vec{g}_{i+1} \cdot \hat{A}\vec{h}_i}{\vec{h}_i \cdot \hat{A}\vec{h}_i} = \frac{\vec{g}_{i+1} \cdot (\vec{g}_{i+1} - \vec{g}_i)}{\lambda_i \vec{h}_i \cdot \hat{A}\vec{h}_i} = \\ &= \frac{\vec{g}_{i+1} \cdot \vec{g}_{i+1}}{\vec{g}_i \cdot \vec{g}_i} \end{aligned} \quad (9)$$

$$\gamma_i = \frac{\vec{g}_{i+1} \cdot \vec{g}_{i+1}}{\vec{g}_i \cdot \vec{g}_i} = \frac{(\vec{g}_{i+1} - \vec{g}_i) \cdot \vec{g}_{i+1}}{\vec{g}_i \cdot \vec{g}_i} \quad (10)$$

Les expressions (9) i (10) són especialment importants, ja que permeten calcular γ_i sense conèixer \hat{A} , sempre que coneixem \vec{g}_{i+1} i \vec{g}_i .

Podem ara intentar aplicar aquest formalisme al problema de minimitzar una funció aproximada per una forma quadràtica

$$f(\vec{x}) \approx \vec{c} - \vec{b} \cdot \vec{x} + \frac{1}{2} \vec{x} \cdot \hat{A} \vec{x}.$$

Siguin \vec{g}_i i \vec{h}_i la mateixa seqüència de vectors d'abans. Imaginem que agafem $\vec{g}_i = -\vec{\nabla} f(\mathbf{P}_i)$ per a un cert punt \mathbf{P}_i . Anem ara desde \mathbf{P}_i en la direcció marcada per \vec{h}_i fins arribar al mínim sobre aquesta línia, \mathbf{P}_{i+1} . Serà

$$\mathbf{P}_{i+1} = \mathbf{P}_i + \alpha_i \vec{h}_i$$

amb α_i escollit de tal manera que:

$$\frac{df(\mathbf{P}_i + \alpha_i \vec{h}_i)}{d\alpha_i} = 0$$

és a dir,

$$\begin{aligned} 0 &= \frac{d(\vec{c} - \vec{b} \cdot (\mathbf{P}_i + \alpha_i \vec{h}_i) + \frac{1}{2} (\mathbf{P}_i + \alpha_i \vec{h}_i) \cdot \hat{A} (\mathbf{P}_i + \alpha_i \vec{h}_i))}{d\alpha_i} = \\ &= -\vec{b} \cdot \vec{h}_i + (\mathbf{P}_i + \alpha_i \vec{h}_i) \cdot \hat{A} \vec{h}_i \end{aligned} \quad (11)$$

Cal notar que

$$\vec{g}_i = \vec{b} - \hat{A} \mathbf{P}_i$$

i, per tant, això es pot reescriure com

$$0 = -\vec{g}_i \cdot \vec{h}_i + \alpha_i \vec{h}_i \cdot \hat{A} \vec{h}_i$$

d'on

$$\alpha_i = \frac{\vec{g}_i \cdot \vec{h}_i}{\vec{h}_i \cdot \hat{A} \vec{h}_i}$$

que és l'expressió (8) per a λ_i . Si ara calculem

$$\begin{aligned} \vec{g}_{i+1} &= -\vec{\nabla} f(\mathbf{P}_{i+1}) = \\ &= \vec{b} - \hat{A}(\mathbf{P}_i + \alpha_i \vec{h}_i) = \\ &= \vec{g}_i - \alpha_i \hat{A} \vec{h}_i \end{aligned}$$

obtenim exactament l'expressió (1) per a \vec{g}_{i+1} . Per tant, podem calcular \vec{g}_{i+1} simplement com menys el gradient de f en el punt \mathbf{P}_{i+1} , que s'obté de \mathbf{P}_i minimitzant al llarg d'una línia. Llavors \vec{h}_{i+1} es calcula a partir de (2) i (9) o (10) i podem seguir endavant, tot això sense necessitat de conèixer \hat{A} .

Tornem, però, a tocar de peus a terra. En primer lloc, això funciona exactament per a formes quadràtiques. A la pràctica, encara que el nostre punt inicial sigui vora el mínim exacte, la no quadraticitat de la nostra funció farà que després d'un cicle complet de minimitzacions al llarg de direccions conjugades no anem a parar exactament al mínim. A més, els errors en les minimitzacions unidimensionals no faran més que empitjorar la situació. En general seran necessàries varies tandes de minimitzacions per tal d'acostar-nos al mínim amb la precisió prèviament fixada.

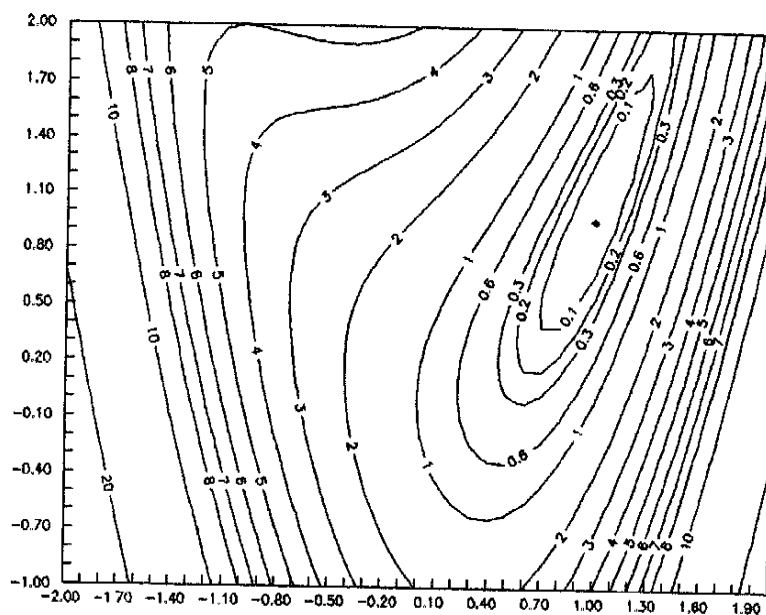


Figura 13: Corbes de nivell de la funció de Rosenbrook al voltant de l'origen de coordenades.

3.3 Un exemple senzill

Anem tot seguit a donar una implementació específica en Pascal de l'algorisme del gradient conjugat, aplicat a l'anomenada funció de Rosenbrook

$$f(x, y) = (y - x^2)^2 + (1 - x)^2$$

Les corbes de nivell d'aquesta funció estan representades a la figura 13. Aquesta funció presenta un mínim relatiu (i absolut) en el punt (1,1) i s'utilitza habitualment com a test senzill per a comprovar l'eficiència dels mètodes numèrics de minimització, ja que el mínim està situat en una vall estreta i allargada. El codi en Pascal del mètode del gradient conjugat per a aquesta funció es troba a l'apèndix B. Modificant-lo de manera trivial podeu obtenir el codi del mètode del descens màxim.

Presentem tot seguit els resultats per a diversos punts inicials, comparant el

nombre d'iteracions pel mètode del gradient conjugat amb el mètode del descens màxim.

punt inicial	gradient conjugat	descens màxim
(0.9, 1.1)	7	7
(2, 3)	13	25
(1, -1)	13	11
(2, -4)	12	27
(0, 0)	15	95
(-1, 1)	4	4
(-1, 7)	22	443
(-11, 14)	20	414
(-8, -9)	14	119
(23, 11)	17	186
(68, -19)	21	91
(168, -95)	14	133
(3, 19)	33	319
(2, 30)	21	1056
(1.01, 0.99)	6	7

Hom observa immediatament la superioritat del mètode del gradient conjugat per pràcticament qualsevol punt inicial. De fet, si teniu mala sort, com en el cas del punt (2, 30), el mètode del descens màxim us farà esperar més temps del que esteu disposats a soportar. En un espai de dimensió superior, aquests punts tant poc cooperatius amb el descens màxim són de fet molt fàcils de trobar i, per tant, el mètode del gradient conjugat és, de fet, el mètode d'elecció per a la majoria de problemes de minimització amb un gran nombre de variables. El punt (-1, 1) és especial degut a que $-\vec{\nabla}f(-1, 1)$ apunta cap a la dreta en la direcció

de l'eix de les X i, per tant, la recta que defineix passa pel mínim $(1, 1)$. De fet, si la minimització unidimensional que utilitzem fos exacta, en una sola iteració hauriem acabat. Casualitats semblants, encara que més complicades, expliquen també el bon comportament del descens màxim pel punt $(1, -1)$.

4 El mètode de la cristallització simulada

El mètode de la cristallització simulada és una tècnica que recentment ha rebut una atenció considerable com a mètode apropiat per l'optimització de problemes de grans dimensions. A efectes pràctics, ha solucionat el famós “problema del viatjant de comerç”, consistent en trobar l'itinerari més curt per un representant comercial que ha de visitar N ciutats. El mètode també ha estat utilitzat amb èxit per dissenyar circuits integrats complexos: la disposició de varis centenars de milers d'elements en un chip de silici és optimitzada per tal de minimitzar les interferències entre les seves connexions.

Les dues aplicacions esmentades són exemples de minimització combinatòria. Com sempre, hi ha una funció per minimitzar, però l'espai sobre el que està definida (espai de configuració) no és un subconjunt de \mathbf{R}^n , sino un espai discret, numerable, però molt gran, com per exemple el conjunt de totes les possibles ordenacions de les ciutats, o el conjunt de totes les possibles adjudicacions dels “solars” d'un chip als diferents elements del circuit. En qualsevol cas, el nombre d'elements de l'espai de configuració és factorialment gran, de manera que no poden ser explorats de manera exhaustiva. A més, com que el conjunt és discret, estem, en principi, privats de la idea de direcció i no podem aplicar els mètodes descrits a les seccions anteriors.

La idea fonamental del mètode de la cristallització simulada té una estreta relació amb la termodinàmica, especialment amb la manera en que els metalls es refreden i cristallitzen. A altes temperatures, les mol·lècules d'un líquid es mouen unes respecte les altres amb una certa llibertat. Si el líquid és refredat lenta-

ment, aquesta mobilitat tèrmica es va perdent poc a poc. Els àtoms o mol·lècules són capaços d'ordenar-se i formar un cristall amb una estructura regular sobre distàncies molts ordres de magnitud superiors a l'abast de les seves interaccions. El cristall així format és l'estat de mínima energia per aquest sistema. El fet sorprenent és que, amb un refredament prou lent, la Natura és capaç, per sí mateixa, de trobar aquest mínim absolut. De fet, si el refredament no és prou lent, l'estat final és policristallí o amorf, amb energies més grans. Per tant la característica essencial del procediment és el refredament lent, de manera que els àtoms tinguin temps de sobres per redistribuir-se mentre perden mobilitat.

Si bé l'analogia no és perfecta, en un cert sentit tots els mètodes de les seccions anteriors corresponen a refredaments ràpids. Sempre hem anat, del punt inicial, cap avall en busca del mínim més proper. Això porta a un mínim local però no necessàriament global. El mètode de minimització que utilitza la Natura és de caire ben diferent. La distribució de Boltzmann

$$P(E) \sim \exp\left(-\frac{E}{kT}\right)$$

indica que un sistema en equilibri tèrmic a temperatura T pot tenir totes les energies possibles, amb una distribució de probabilitat exponencial. Fins i tot a baixa temperatura hi ha una probabilitat, si bé molt petita, de que el sistema estigui en un estat d'alta energia. Per tant hi ha la possibilitat de que el sistema surti d'un estat de mínima energia local, passi a un estat d'energia superior, i torni a caure a un altre mínim local, potser globalment millor que el primer. En altres paraules, el sistema va algunes vegades cap amunt i altres cap avall, però com més baixa és la temperatura menys probable és l'excursió cap amunt. La constant k , anomenada de Boltzmann, és una constant fonamental de la Natura que relaciona temperatura i energia. Si el seu valor fos zero, sols l'estat de mínima energia seria possible i la vida seria impossible perquè no estarien permesos els canvis d'estat. Inversament, si valguès infinit, totes les energies serien igualment probables i de nou la vida seria impossible per la impossibilitat de formar estructures amb un

cert nivell d'estabilitat. Afortunadament,

$$k = 1.380662 \cdot 10^{-23} J/K$$

i això sembla que ens està prou bé.

L'any 1953, Metropolis i els seus col·laboradors [4] van incorporar per primera vegada aquestes idees als càlculs numèrics. Donades diverses opcions, un sistema termodinàmic simulat podia canviar la seva configuració amb energia E_1 a una altra configuració amb energia E_2 amb probabilitat

$$P(E_1 \rightarrow E_2) = \begin{cases} \exp[-(E_2 - E_1)/kT] & \text{si } E_2 > E_1 \\ 1 & \text{si } E_2 < E_1 \end{cases}$$

és a dir, la transició cap un estat de menor energia es produeix segur mentre que cap un estat d'energia més gran segueix la llei de Boltzmann. Aquest esquema general, d'anar sempre avall i de vegades amunt, es coneix com algorisme de Metropolis.

Per tal d'utilitzar l'algorisme de Metropolis en casos diferents dels sistemes termodinàmics, és necessari proveir-se dels següents elements:

- una descripció de les possibles configuracions del sistema
- un generador aleatori de canvis del sistema
- una funció E (l'anàlog de l'energia) la minimització de la qual és l'objectiu del problema
- un paràmetre de control $\alpha = kT$ i un procediment de *refredament* que indiqui com α ha de ser baixat, és a dir, després de quants canvis aleatoris hem de disminuir α i en quant l'hem de disminuir. Això requereix una mica d'experimentació amb el sistema que tinguem a ma.

El sistema que ens ocuparà és l'anomenat *problema del viatjant de comerç*, consistent en trobar el menor recorregut possible que ens porta per una sèrie de ciutats

distribuides en el pla i retornar al punt d'origen. Si (x_i, y_i) són les coordenades de la ciutat i -ésima i hi ha N ciutats, la funció a minimitzar, que fa el paper de l'energia, és

$$E(\{(x_i, y_i)\}) = \sum_{i=1}^N \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$$

amb $x_{N+1} = x_1$, $y_{N+1} = y_1$. Una configuració del sistema consisteix en un recorregut particular, no important per quina ciutat es comenci, ja que tornem al punt de sortida. S'ha demostrat que, per a aquest problema en concret, el generador aleatori de canvis de configuració en té prou amb les següents possibilitats:

1. inversió d'un troç del camí. Es trien dues ciutats i, j i s'inverteix el camí entre elles, passant del recorregut

$$1, 2, \dots, i-1, i, i+1, \dots, j-1, j, j+1, \dots, N$$

al recorregut

$$1, 2, \dots, i-1, j, j-1, \dots, i-1, i, j+1, \dots, N$$

2. transport d'un troç de camí. Es trien dues ciutats i, j i una de tercera k , que no estigui entre les dues primeres, i el troç de camí entre i i j es porta darrera k , passant-se de

$$1, 2, \dots, i-1, i, i+1, \dots, j-1, j, j+1, \dots, k-1, k, k+1, \dots, N$$

a

$$1, 2, \dots, i-1, j+1, \dots, k-1, k, i, i+1, \dots, j-1, j, k+1, \dots, N$$

L'algorisme funciona llavors de la següent manera. Hom parteix d'una determinada configuració i decideix aleatoriament si s'efectua una inversió o un transport. En cada cas es trien aleatoriament les ciutats que defineixen l'operació i es calcula la diferència de camí entre la configuració original i la proposada. Si el camí disminueix, es realitza el canvi. Si augmenta, es decideix si es realitza o no d'acord amb la llei de Boltzmann amb un cert valor α inicial. Es repeteix l'operació

fins que s'aconsegueix un cert nombre de canvis i es procedeix a disminuir α en un cert factor, tornant-se a generar canvis de configuració. El procediment es repeteix fins que s'arriba a un α pel que el nombre de canvis realitzats després d'un cert nombre d'intents està per sota d'un cert valor crític; en altres paraules, l'algorisme s'acaba quan el sistema està ja tan *fred* que li és molt difícil moure's de l'estat en que es troba.

Hem realitzat una implementació d'aquest algorisme en C i a continuació presentarem els resultats. Per detalls del programa podeu consultar [1] o bé dirigiu-vos a C.B.

Tenim $N = 20$ ciutats distribuïdes sobre un quadrat de costat la unitat. Per a cada valor del paràmetre α intentem $100 \cdot N = 2000$ canvis de configuració, però parem si es realitzen $10 \cdot N = 200$ canvis. Hem comprovat que hom pot començar l'algorisme amb un valor d' α del mateix ordre de magnitud que l'*energia* de la configuració inicial, és a dir, la longitud del camí inicial, i que una bona estratègia consisteix en disminuir α cada vegada en un factor 0.5. Per tal de mostrar la flexibilitat del mètode, hem afegit una filigrana al problema. Imagineu que el territori està dividit de nord a sud en dues parts iguals per una barrera natural o política, i que creuar-la suposa alguna mena de cost per al viatjant, que pot ser positiu (per exemple un riu que dificulta el creuament) o negatiu (per exemple una frontera que el viatjant té molt d'interès en creuar el més sovint possible ja que es dedica al contrabàndol). Llavors la funció que voldrem minimitzar serà

$$E(\{(x_i, y_i)\}) = \sum_{i=1}^N \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} + \lambda/2 \sum_{i=1}^N |\mu_i - \mu_{i+1}|$$

on $\mu_i = -1$ si la ciutat i es troba a l'oest de la divisió i $\mu_i = +1$ si es troba a l'est. D'aquesta manera, s'afegeix una penalització de $+\lambda$ cada vegada que es creua la barrera. El paràmetre λ , que pot ser positiu o negatiu, controla el caràcter de la penalització, així com la importància que se li vol donar respecte a la longitud del camí. Presentem els resultats per a $\lambda = 0, +1, -1$ i -0.1 . El recorregut inicial està representat a la figura 14 i els recorreguts finals per als diferents valors de λ a les figures 15, 16, 17 i 18.

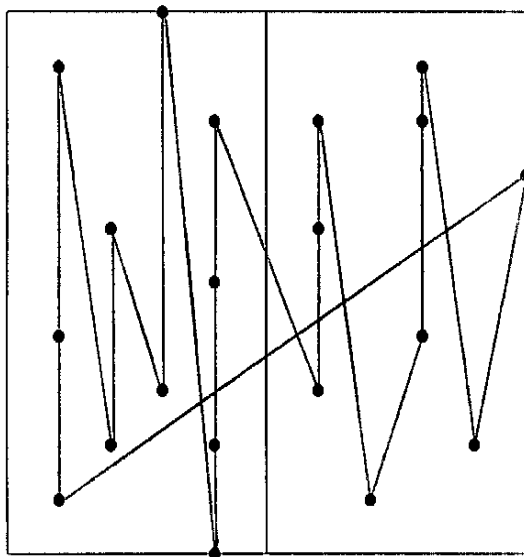


Figura 14: Recorregut inicial. $E_i = 9.588847$ amb $\lambda = 0$.

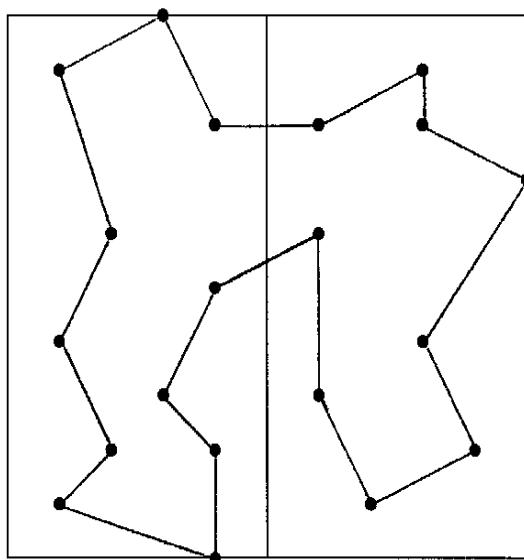


Figura 15: Recorregut final amb $\lambda = 0$. $E_f = 4.535528$.

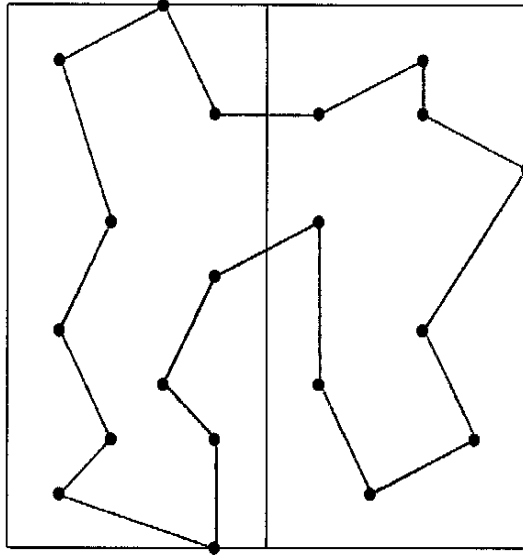


Figura 16: Recorregut final amb $\lambda = +1$. $E_f = 6.535528$.

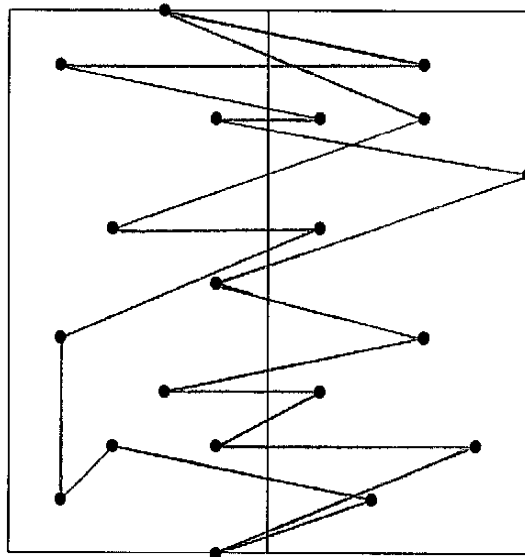


Figura 17: Recorregut final amb $\lambda = -1$. $E_f = -8.978089$.

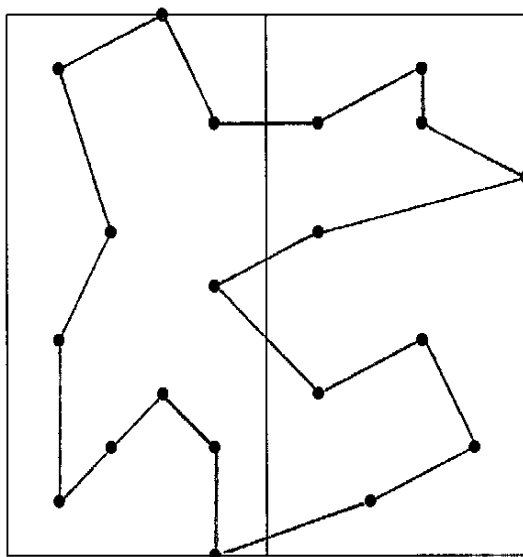


Figura 18: Recorregut final amb $\lambda = -0.1$. $E_f = 4.164334$.

Cal observar en primer lloc que els recorreguts per a $\lambda = 0$ i $\lambda = +1$ són el mateix, ja que en ambdós casos es creua la divisió el menor nombre de vegades possible, i la penalització no té cap poder coercitiu. Per a $\lambda = -1$ la minimització de la longitud del recorregut juga un paper molt petit ja que el sistema està incentivat per a creuar la frontera com més vegades millor. Finalment, per a $\lambda = -0.1$, els dos efectes estan més equilibrats i la configuració que assoleix el mínim és una petita modificació del cas $\lambda = 0$.

A Significat geomètric del gradient

Sovint es parla del gradient d'un camp escalar com un "camp vectorial gradient" de manera que a cada punt del domini de la funció li associa un vector, el vector gradient.

El camp vectorial gradient té un important significat geomètric. Ens diu la direcció en que f creix més ràpidament i, a més, la direcció que és ortogonal a les superfícies de nivell de f . Veiem més concretament aquests dos fets, suposant sempre que les funcions considerades són totes infinitament diferenciables amb

continuitat.

Teorema A.1 *Sigui $\vec{\nabla}f(x) \neq \vec{0}$. Aleshores $\vec{\nabla}f(x)$ apunta en la direcció al llarg de la qual f està creixent més ràpidament.*

Demostració

La derivada direccional de f en la direcció \vec{v} ve donada per

$$\vec{\nabla}f(x) \cdot \vec{v} = \|\vec{\nabla}f(x)\| \|\vec{v}\| \cos \theta$$

on θ és l'angle entre $\vec{\nabla}f(x)$ i \vec{v} .

Si \vec{v} és unitari,

$$\vec{\nabla}f(x) \cdot \vec{v} = \|\vec{\nabla}f(x)\| \cos \theta$$

i serà màxim quan $\theta = 0$, és a dir, quan \vec{v} i $\vec{\nabla}f$ són paral·lels. \square

Observeu que si $\vec{\nabla}f = 0$ la raó del canvi és zero per a tot \vec{v} .

Considerem $f: \mathbf{R}^3 \rightarrow \mathbf{R}$ diferenciable amb continuïtat, i una superfície de nivell S en \mathbf{R}^3 definida per $f(x, y, z) = k$, $k \in \mathbf{R}$. Aleshores,

Teorema A.2 *Sigui (x_0, y_0, z_0) un punt en la superfície de S . Aleshores $\vec{\nabla}f(x_0, y_0, z_0)$ és normal a la superfície en el següent sentit: si \vec{v} és el vector tangent en $t = 0$ d'una trajectòria $c(t)$ en S amb $c(0) = (x_0, y_0, z_0)$, es té $\vec{\nabla}f(x_0, y_0, z_0) \cdot \vec{v} = 0$.*

Demostració

Sigui $c(t)$ en S , per tant $f(c(t)) = k$, amb vector tangent \vec{v} en (x_0, y_0, z_0) , és a dir, $c'(0) = \vec{v}$.

D'una banda,

$$f(c(t)) = k \rightarrow \left[\frac{d}{dt} f(c(t)) \right]_{t=0} = 0$$

i, per la regla de la cadena,

$$\begin{aligned} \left[\frac{d}{dt} f(c(t)) \right]_{t=0} &= Df(x_0, y_0, z_0) \cdot c'(0) \\ &= \vec{\nabla}f(x_0, y_0, z_0) \cdot \vec{v} = 0. \square \end{aligned}$$

B Codi en Pascal per a la funció de Rosenbrook

```
program gradconj;
type
  vector = array [1..4] of real;
const
  r=0.38197;
  e=0.00001;
  gold=1.618034;
var
  cont:integer;
  y,p1,p2:real;
  h,g:vector;
function fun(x,y:real):real; (* funcio de Rosenbrook *)
  begin
    fun:=sqr(y-sqr(x))+sqr(1-x)
  end;
function F(p1,p2,h1,h2,x:real):real; (* funcio definida sobre
  begin
    una linea *)
    F:=fun(p1+x*h1,p2+x*h2)
  end;
function pesc(a1,a2:real):real; (* norma al quadrat *)
  begin
    pesc:=sqr(a1)+sqr(a2)
  end;
function dir(a,b,c,d,e:real):real; (* calcul de h *)
  begin
    dir:=a+(pesc(a,e)/pesc(c,d))*b
```

```

        end;
function grad1f(x,y:real):real;          (* primera component
begin                                     del gradient *)
    grad1f:=-4*x*y+4*x*sqr(x)+2*x-2
end;
function grad2f(x,y:real):real;          (* segona component
begin                                     del gradient *)
    grad2f:=2*(y-sqr(x))
end;
procedure minim(p1,p2,h1,h2:real; var y:real);
    (* donat un punt (p1,p2) i una direccio (h1,h2),
       retorna el valor del parametre de la recta, y,
       per al qual s'assoleix el minim *)
    var
        a,b,b1,c:real;
        x1,x2,x3:real;
        x,z,m:real;
procedure terna(x1,x2,x3,p1,p2,h1,h2:real; var x,y,z,m:real);
    (* donada una terna x1, x2, x3 al llarg de la recta
       que passa per (p1,p2) en la direccio (h1,h2), es
       retorna la terna seguent x,y,z i el punt m de la
       terna inicial que ha quedat exclos *)
    var
        p,q,t:real;
    begin (*inici procediment terna*)
        p:=x1-x2;
        q:=x2-x3;
        if p>=q then begin
            t:=x2+r*p;

```

```
if F(p1,p2,h1,h2,x2)>F(p1,p2,h1,h2,t) then
begin
x:=x2;
y:=t;
z:=x1;
m:=x3
end

else

begin
x:=x3;
y:=x2;
z:=t;
m:=x1
end

end
else begin
t:=x2-r*q;
if F(p1,p2,h1,h2,x2)>F(p1,p2,h1,h2,t) then
begin
x:=x3;
y:=t;
z:=x2;
m:=x1
end

else

begin
x:=t;
y:=x2;
z:=x1;
```

```
        m:=x3
        end
    end
end; (*fi procediment terna*)
begin (*inici procediment minim*)
a:=0;
b:=1;
if F(p1,p2,h1,h2,a)<F(p1,p2,h1,h2,b) then
    begin
        a:=1;
        b:=0
    end;
b1:=b;
repeat
c:=b1+gold*(b-a);
b1:=c
until F(p1,p2,h1,h2,c)>F(p1,p2,h1,h2,b);
if c>a then
    begin
        x1:=c;
        x2:=b;
        x3:=a
    end
    else
    begin
        x1:=a;
        x2:=b;
        x3:=c
    end;
end;
```

```
repeat
terna(x1,x2,x3,p1,p2,h1,h2,x,y,z,m);
x3:=x;
x2:=y;
x1:=z
until abs(x-z)<sqrt(e)*(abs(y)+abs(m)) (* condicio per
acabar la
minimitzacio
de linea *)

end; (*fi procediment minim*)
begin (*cos programa principal*)
writeln('introduir un punt (2 coordenades)');
readln(p1,p2);
g[1]:=-grad1f(p1,p2);
g[2]:=-grad2f(p1,p2);
h[1]:=g[1];
h[2]:=g[2];
cont:=0;
repeat
minim(p1,p2,h[1],h[2],y);
p1:=p1+y*h[1];
p2:=p2+y*h[2];
g[3]:=-grad1f(p1,p2);
g[4]:=-grad2f(p1,p2);
h[1]:=dir(g[3],h[1],g[1],g[2],g[4]);
h[2]:=dir(g[4],h[2],g[1],g[2],g[4]);
g[1]:=g[3];
g[2]:=g[4];
cont:=cont+1
```

```
until sqrt(pesc(g[3],g[4]))<e; (* s'acaba si en un punt
                                el gradient es massa
                                petit *)
writeln('minim en el punt: (',p1:4:1,',',p2:4:1,')')
writeln('nombre iteracions:',cont)
end. (*fi programa principal*)
```

Bibliografia

- [1] PRESS, WILLIAM H., ET AL., Numerical Recipes in C, Cambridge University Press, New York, 1988.
- [2] FLETCHER, R., Practical Methods of Optimization, vol. 1, John Wiley & Sons, New York, 1980.
- [3] GILL, PHILIP E., ET AL., Practical Optimization, Academic Press, London, 1981.
- [4] METROPOLIS, N., ET AL., J. Chem. Phys., **21**, 1087 (1953)