Optimización y Control

Ignacio Delgado, Ignacio Gómez,
José M. Perales, José M. Vega

Escuela de Ingeniería Aeronáutica y del Espacio
Programa: **Grado en Ingeniería Aeronáutica**
Curso 2017-2018
Tema #4: Métodos de búsqueda directa

Index

MÉTODOS DE BÚSQUEDA DIRECTA (I)

- El nombre de *búsqueda directa* (direct search) se debe a Hooke y Jeeves [1]
- Fueron los primeros métodos, desarrollados al principio de la era de optimización mediante ordenador, mediante sencillos argumentos heurísticos [2]. Se hicieron muy populares y se desarrollaron rápidamente, sobre todo a partir de los años 90 [3].
- Renacido interés, muy reciente, en el contexto de cálculo paralelo.
- También conocidos (de modo un tanto impreciso) como métodos de no-gradiente (non-gradient), o métodos sin derivadas (derivative-free), o métodos de orden cero.

---

[1] R. Hooke and T.A. Jeeves. Direct search solution of numerical and statistical problems. J. ACM, 8:212–229, 1961.

[2] H.H. Rosenbrock. An automatic method for finding the greatest or least value of a function. Computer Journal, 3:175–184, 1960.

[3] V. Torczon. On the convergence of the multidirectional search algorithm. SIAM J. Optim., 1: 123–145, 1991.

MÉTODOS DE BÚSQUEDA DIRECTA (II)

- Los métodos de búsqueda directa son necesarios en casos en que el cálculo del gradiente es difícil, imposible, o no conveniente. Por ejemplo, cuando los datos son discretos (dados en tablas o mallas), o cuando la función objetivo, o alguno de sus ingredientes, no es suficientemente regular.
- Como los métodos de tipo gradiente, los métodos de búsqueda directa pueden combinarse con ideas de regiones de confianza o condiciones de tipo Wolfe para mejorar su robustez, o con ideas de tipo conjunto activo para tratar restricciones. Pero estas combinaciones se salen del objetivo de este curso.

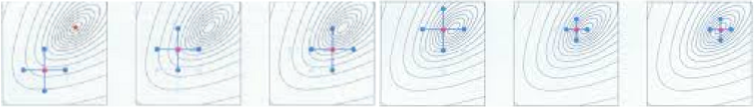
MÉTODOS DE BÚSQUEDA DIRECTA (III)

- El método más sencillo es el de búsqueda en malla (grid search), en que se calcula la función objetivo, F, en una malla y se toma como óptimo aquel punto de la malla en que F tiene un valor menor. Presentan la maldición de la dimensionalidad (curse of dimensionality).
- Resto de los métodos (gran variedad de ellos [4]) buscan evitar la dificultad. Algunos de ellos son estadísticos (no considerados en lo que sigue). Otros, conocidos como de búsqueda de patrones (pattern search) son la contrapartida discreta de los métodos de tipo gradiente: son métodos iterativos que, en cada paso, hacen una *búsqueda local* (local search) de la mejor 'dirección de descenso'.
- En lo que sigue se describen solamente algunos métodos, sencillos, de búsqueda directa, a modo de introducción al tema.

[4] T.G. Kolda, R.M. Lewis, and V. Torczon. Optimization by direct search: new perspectives on some classical and modern methods. SIAM Review, 45:385–482, 2003.

MÉTODOS DE LA BRÚJULA (I)

- El método de la brújula (compass method) es el método más antiguo, inventado por Fermi y Metropolis [5], en Los Alamos, y desarrolado más tarde por Davidson [6].
- En cada paso de iteración, se calculan los valores de F en los $2N$ puntos, distantes una cantidad δ del punto de partida, a lo largo de los ejes. Se toma como siguiente punto aquél en que F tiene un valor menor. Si F es mayor que en el punto de partida en todos los puntos, se divide δ por dos.



[5] E. Fermi and N. Metropolis. Los Alamos Unclassified Report LS–1492. Los Alamos National Laboratory, 1952.

[6] W.C. Davidson. Variable Metric Method for Minimization. Argonne National Laboratory, Tech. Rep. 5990, 1959.

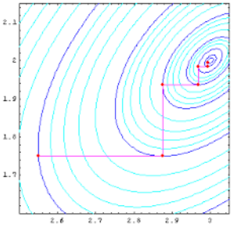
MÉTODOS DE LA BRÚJULA (II)

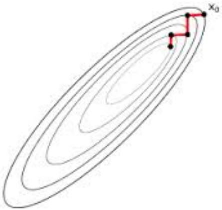
- En cada paso, el método require $2N$ evaluaciones de la función objetivo.
- De algún modo, es la variante discreta del método del descenso más empinado.
- Se trata de un método simple y fácil de implementar pero, como en el descenso más empinado, la convergencia es lenta y presenta un comportamiento de tipo zig-zag si las hipersuperficies de nivel son alargadas.
- La convergencia sólo puede detectarse cuando δ es demasiado pequeño.
- Hay variantes del método más eficaces computacionalmente [7].

[7] V. Torczon. On the convergence of pattern search algorithms. SIAM J. Optim., 7:1–25, 1997.

MÉTODO UNO-CADA-VEZ

- En cada paso, se minimiza la función objetivo (p.e., mediante el método de la secante, que no require utilizar derivadas) a lo largo de uno de los ejes coordenados, $x^1, \ldots, x^N$.
- De ser necesario, se repite el ciclo $x^1, \ldots, x^N$.
- Como en el método de la brújula, pueden darse comportamientos de tipo zig-zag.





MÉTODOS DE ROSENBROCK Y POWELL

- Inventado por Rosenbrock[8] (que lo probó en su función 'banana') para acelerar la convergencia del método uno-cada-vez.
- Para cada ciclo (a lo largo de los ejes $x^1, \ldots, x^N$) en el método uno-cada-vez, se minimiza la función a lo largo de la recta que une los puntos minimizantes primero y último del ciclo. Tal como se ve en la figura de la trasparencia anterior, la idea efectivamente acelera la convergencia.
- El método de Powell[9] es una versión discreta del método de gradiente conjugado que supone que, en las cercanías del mínimo, el comportamiento de la función objetivo es cuadrático.

---

[8] H.H. Rosenbrock. An automatic method for finding the greatest or least value of a function. Computer Journal, 3:175–184, 1960.

[9] M.J.D. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. The Computer Journal, 7:155-162, 1964.

MÉTODO SIMPLEX (NELDER-MEAD)

- Es, seguramente, el método de búsqueda directa más popular.
- Utiliza un *simplex local* que se va moviendo y adaptando (mediante reflexiones, contracciones y expansiones) para irse acercando al mínimo.
- Un simplex de un espacio de diseño de N dimensiones es el hiper-poliedro irregular que tiene un número menor de vértices $(N+1)$. Se trata de un triángulo en dimensión dos, de un tetraedro en dimensión tres, etc.
- Los vértices del simplex, $\boldsymbol{u}^0, \ldots, \boldsymbol{u}^N$, se ordenan de mejor a peor, es decir, de modo que $f(\boldsymbol{u}^0) < f(\boldsymbol{u}^1) < \ldots < f(\boldsymbol{u}^N)$.
- En cada paso de de la iteración se procede como se indica en la siguiente transparecia.

MÉTODO SIMPLEX (ITERACIÓN)

El peor punto, $\boldsymbol{u}^N$, se refleja respecto del centroide del simplex, $\boldsymbol{c} = \sum \boldsymbol{u}^j/(N+1)$, obteniendo $\boldsymbol{u}_r = 2\boldsymbol{c} - \boldsymbol{u}^N$. Si $f(\boldsymbol{u}_r) \leq f(\boldsymbol{u}^0)$, se calcula (expansión) un nuevo punto $\boldsymbol{u}_e = \boldsymbol{u}_c + \beta(\boldsymbol{u}_r - \boldsymbol{c})$ para $\beta > 1$ (calibrable). Si $f(\boldsymbol{u}_e) < f(\boldsymbol{u}^r)$, entonces se toma $\boldsymbol{u}_e$ como nuevo mejor vértice. En otro caso, el nuevo mejor vértice es $\boldsymbol{u}_r$. Los demás vértices se mantienen.

Si, en cambio, $f(\boldsymbol{u}_r) > f(\boldsymbol{u}^0)$, el simplex no es suficientemente bueno y se contrae, considerando el nuevo punto $\boldsymbol{u}_c = \boldsymbol{c} + \gamma(\boldsymbol{u}^N - \boldsymbol{c})$ si $f(\boldsymbol{x}^N) < f(\boldsymbol{x}_r)$, o $\boldsymbol{u}_c = \boldsymbol{c} + \gamma(\boldsymbol{u}_r - \boldsymbol{c})$ si $f(\boldsymbol{x}^N) \geq f(\boldsymbol{x}_r)$, con $0 < \gamma < 1$ (calibrable). Si $f(\boldsymbol{u}_c) < \min\{f(\boldsymbol{u}_r), f(\boldsymbol{u}^N)\}$, entonces el punto $\boldsymbol{u}_c$ sustituye al peor vértice, manteniento los demás puntos, y termina la iteración. En otro caso, se contrae el simplex entero alrededor del mejor punto, manteniendo éste y dividiendo por 2 las distancias a los demás.