AI505

Optimization

Derivative-Free Methods

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

Outline

1. Nelder-Mead Simplex Method

2. Divided Rectangles

Derivative-Free Methods

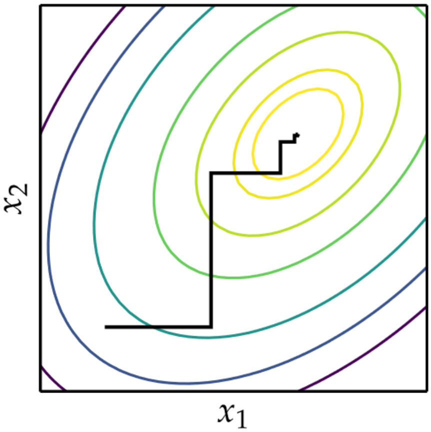
- Also called **direct search methods**, zero-order, black box, pattern search
- Direct method search using function evaluations only

Cyclic Coordinate Search

- Also known as coordinate descent, or taxicab search
- Performs line search in alternating coordinate directions

$$x_{1,1} = \text{argmin}_{x_1} f(x_1, x_{2,0}, x_{3,0}, \ldots, x_{n,0})$$

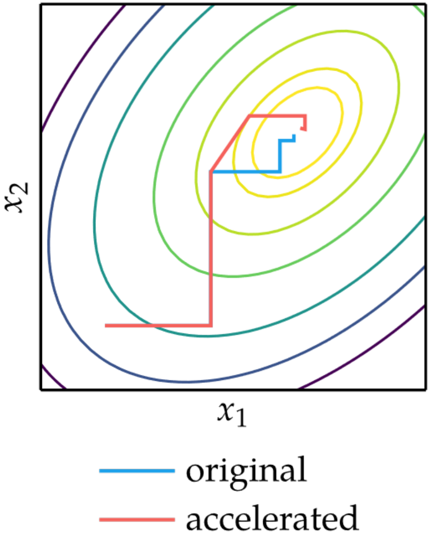
$$x_{2,1} = \text{argmin}_{x_2} f(x_{1,1}, x_2, x_{3,1}, \ldots, x_{n,1})$$



Cyclic Coordinate Search

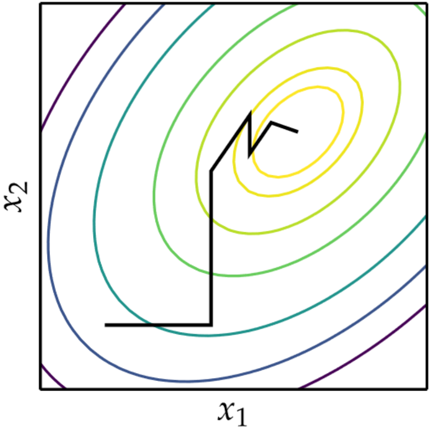
- Can be augmented to accelerate convergence

- For every full cycle starting with optimizing $\boldsymbol{x}_1$ along $[1, 0, \ldots, 0]$ and ending with $\boldsymbol{x}_{n+1}$ after optimizing along $[0, 0, \ldots, 1]$, an additional line search is conducted along the direction $\boldsymbol{x}_{n+1} - \boldsymbol{x}_1$.



Powell's Method

- Similar to Cyclic Coordinate Search, but can search in non-orthogonal directions
- Drops the oldest search direction in favor of the overall direction of progress
- It can lead the search directions to become linearly dependent and the search directions can no longer cover the full design space, and the method may not be able to find the minimum



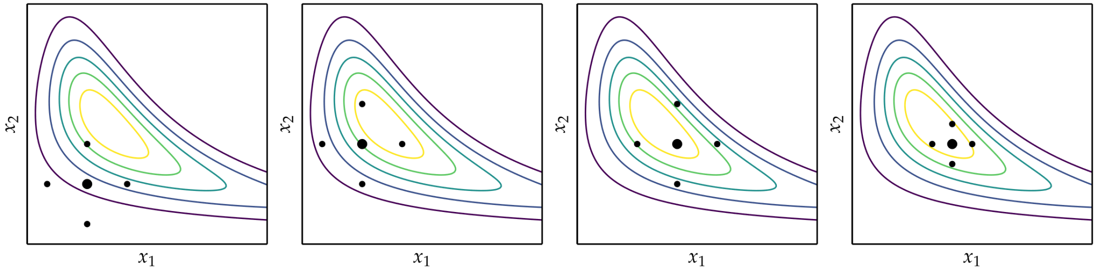
Powell's Method

Input: $f, \boldsymbol{x}_0$
Output: $\boldsymbol{x}^*$
search directions $\boldsymbol{u}_1 = \boldsymbol{e}_1, \ldots, \boldsymbol{u}_n = \boldsymbol{e}_i$;
while not converged **do**
    for i in $\{1, \ldots, n\}$ **do**
        $\boldsymbol{x}_{i+1} \leftarrow \texttt{line\_search}(f, \boldsymbol{x}_i, \boldsymbol{u}_i)$;
    for i in $\{1, \ldots, n-1\}$ **do**
        $\boldsymbol{u}_i \leftarrow \boldsymbol{u}_{i+1}$;
    $\boldsymbol{u}_n \leftarrow \boldsymbol{x}_{n+1} - \boldsymbol{x}_1$;

- search directions can become linearly dependent and no longer cover the full design space.
- peridocally reset the diections to the canonical basis.

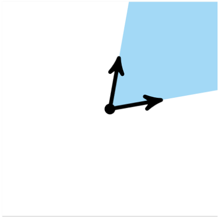
Hooke-Jeeves

- Evaluate $f(\boldsymbol{x})$ and $f(\boldsymbol{x} \pm \alpha \boldsymbol{e}_i)$ for a given step size α in every coordinate direction from an anchoring point $\boldsymbol{x}$.
- It accepts any improvement it may find.
- If no improvements are found, it decreases the step size.
- The process repeats until the step size is sufficiently small.
- $2n$ evaluations for an n-dimensional problem

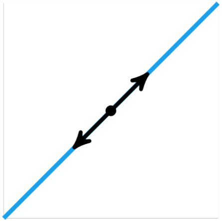


Generalized Pattern Search

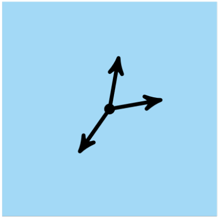
- Generalization of Hooke-Jeeves method
- A pattern P can be constructed from a set of directions D about an anchoring point $\boldsymbol{x}$ with a step size α according to: $P = \{\boldsymbol{x} + \alpha \boldsymbol{d} \text{ for each } \boldsymbol{d} \in D\}$
- Searches in set of directions that **positively spans** (nonnegative linear combination) search space. (if D has full row rank and if $D\boldsymbol{x} = -D1$ with $\boldsymbol{x} \geq 0$



only positively spans the cone

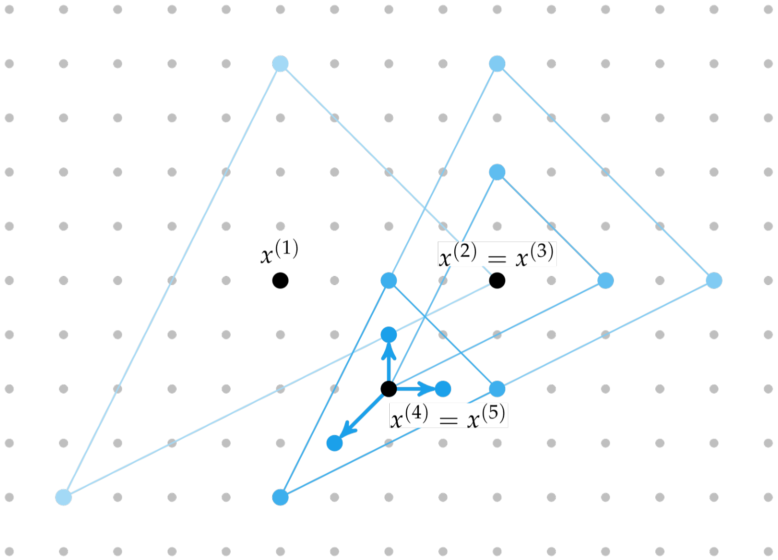


only positively spans 1d space



positively spans $\mathbb{R}^2$

Generalized Pattern Search



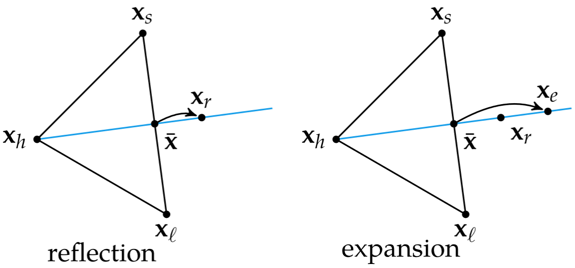
Outline

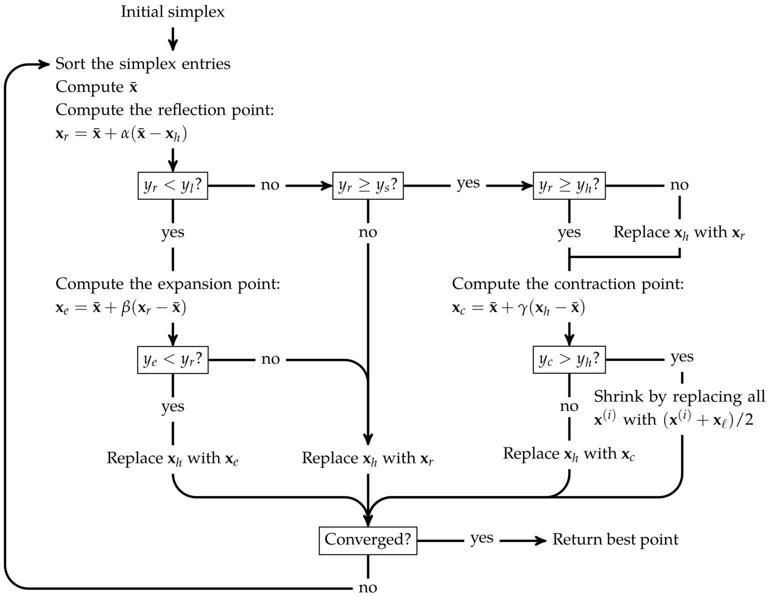
1. Nelder-Mead Simplex Method

2. Divided Rectangles

Nelder-Mead Simplex Method

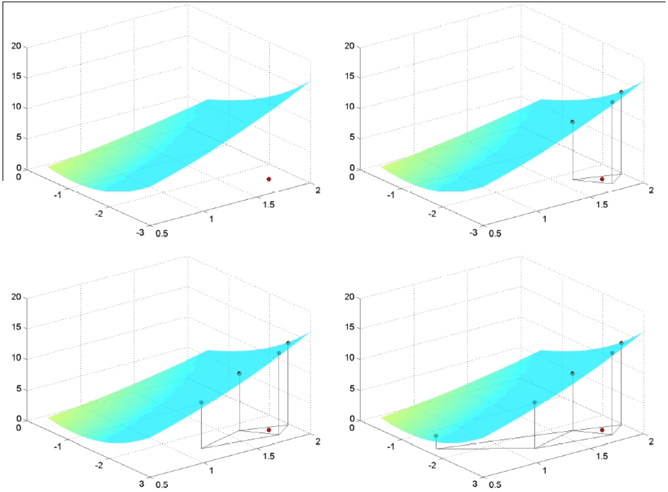
Uses simple algorithm to traverse search space using set of points defining a simplex



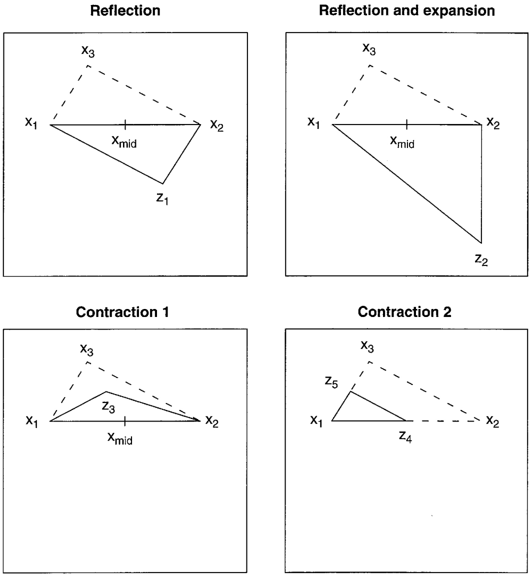

Initial simplex
Sort the simplex entries
Compute $\bar{\mathbf{x}}$
Compute the reflection point:
$\mathbf{x}_r = \bar{\mathbf{x}} + \alpha(\bar{\mathbf{x}} - \mathbf{x}_h)$
$y_r < y_l$?
no
$y_r \geq y_s$?
yes
$y_r \geq y_h$?
no
yes
no
yes
Replace $\mathbf{x}_h$ with $\mathbf{x}_r$
Compute the expansion point:
$\mathbf{x}_e = \bar{\mathbf{x}} + \beta(\mathbf{x}_r - \bar{\mathbf{x}})$
Compute the contraction point:
$\mathbf{x}_c = \bar{\mathbf{x}} + \gamma(\mathbf{x}_h - \bar{\mathbf{x}})$
$y_e < y_r$?
no
$y_c > y_h$?
yes
yes
no
Shrink by replacing all
$\mathbf{x}^{(i)}$ with $(\mathbf{x}^{(i)} + \mathbf{x}_\ell)/2$
Replace $\mathbf{x}_h$ with $\mathbf{x}_e$
Replace $\mathbf{x}_h$ with $\mathbf{x}_r$
Replace $\mathbf{x}_h$ with $\mathbf{x}_c$
Converged?
yes
Return best point
no

Nelder-Mead

Simplex based method [Spendley et al. (1962)]

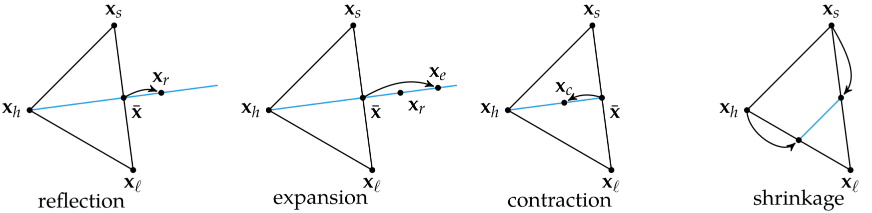


Nelder-Mead (cont.)



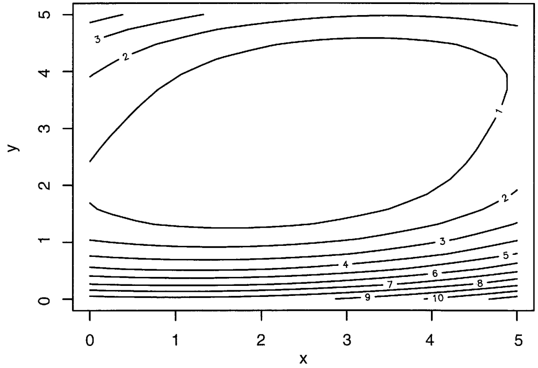
Nelder-Mead (cont.)

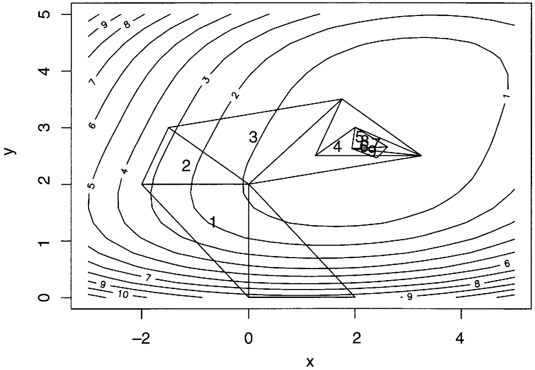
Nelder-Mead simplex method [Nelder and Mead, 1965]:

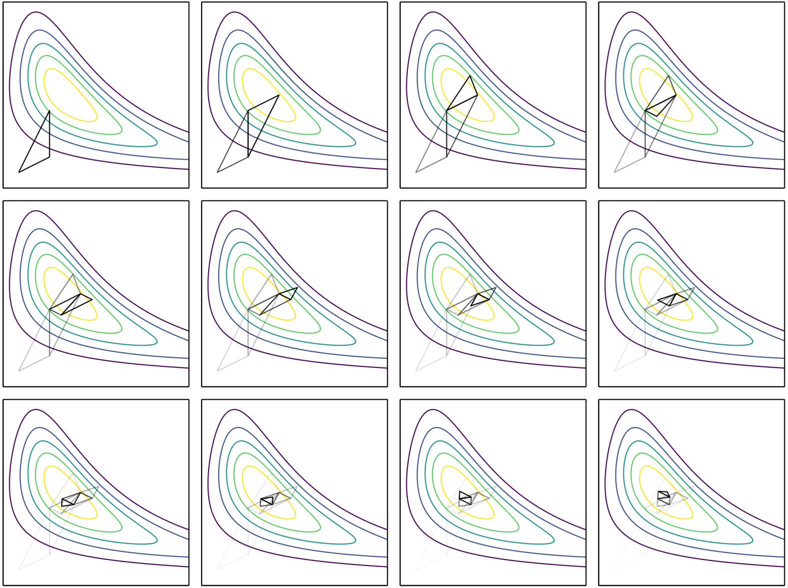


Nelder-Mead (cont.)

Example:







Nelder-Mead (cont.)

Algorithm: Simplex search

Let $x_1, \dots, x_{n+1}$ be vertices of a simplex
Let h: $y_h = \max_i y_i = \max f(x_i)$ and
l: $y_l = \min_i y_i = \min f(x_i)$
Let $\bar{x}$ be the centroid of points with $i \neq h$ and
$d(x_i, x_j)$ the distance between two points x_i and x_j
$(k \leftarrow 0)$

Reflect iter. k: $(k \leftarrow k+1)$ Generate the reflection x_R of x_h

Case 1 **if** $y_l \leq y_R < y_h$ then $x_h \leftarrow x_R$ and go to **Reflect**

Case 2 **else if** $y_R < y_l$ **then**
generate the expansion x_E of x_h
if $y_E < y_l$ **then** $x_h \leftarrow x_E$ and go to **Reflect**
else $x_h \leftarrow x_R$ and go to **Reflect**

Case 3 **else if** $y_R > y_i, \forall i \neq h$ **then** $x_h \leftarrow \min\{y_h, y_R\}$ and generate the contraction 1
if $y_C \leq \min\{y_h, y_R\}$ then $x_h \leftarrow x_C$ and go to **Reflect**
else contraction 2 $x_i \leftarrow (x_i + x_l)/2$

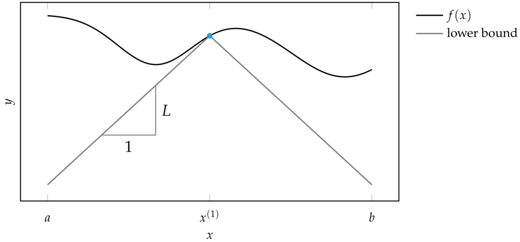
Outline

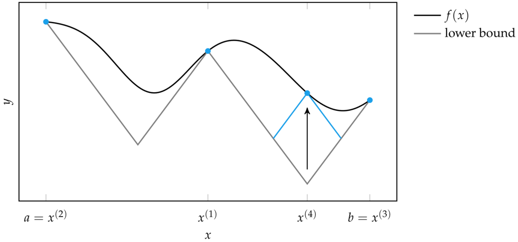
1. Nelder-Mead Simplex Method

2. Divided Rectangles

DIRECT – Divided Rectangles

- Also called DIRECT for Divided RECTangles
- Recall from Shubert-Piyavskii, a Lipshitz constant is used to provide a lower bound on the function, and a function evaluation is made where this bound is lowest





DIRECT – Divided Rectangles

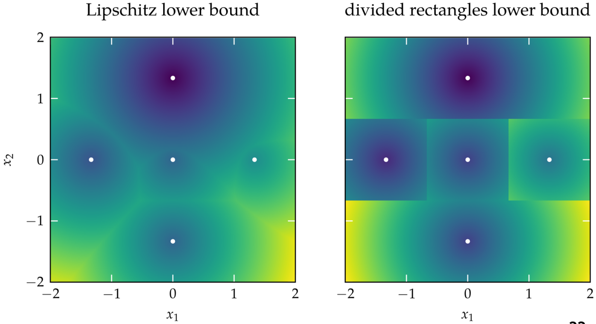
- The notion of Lipschitz continuity can be extended to multiple dimensions. If f is Lipschitz continuous over a domain X with Lipschitz constant $\ell > 0$, then for a given design $\mathbf{x}_0$ and $y = f(\mathbf{x}_0)$, the circular cone

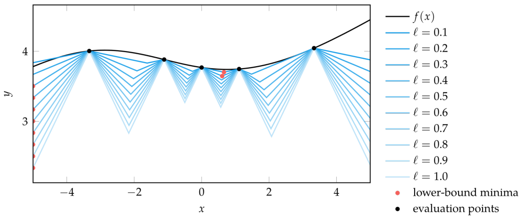
$$f(\mathbf{x}_0) - \ell \|\mathbf{x} - \mathbf{x}_0\|_2$$

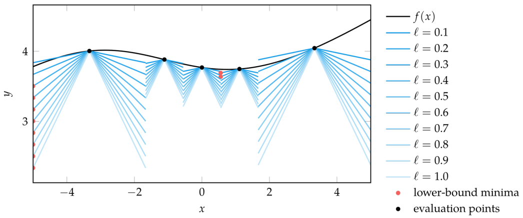
  forms a lower bound of f

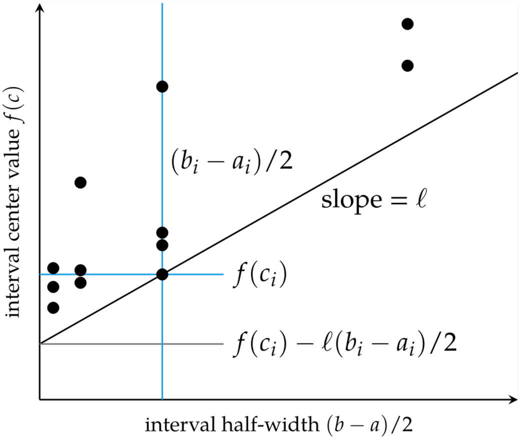
- Given m function evaluations with design points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$, we can construct a superposition of these lower bounds by taking their maximum:

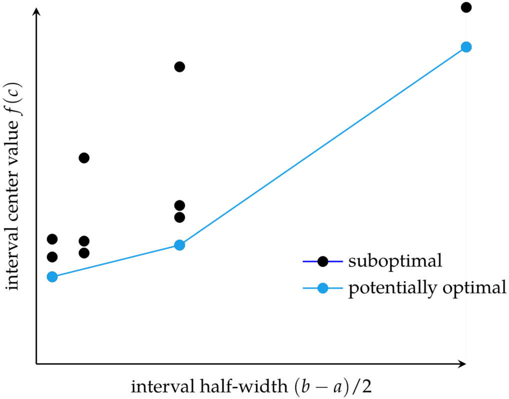
$$\max_i f(\mathbf{x}_i) - \ell \|\mathbf{x} - \mathbf{x}_i\|_2$$

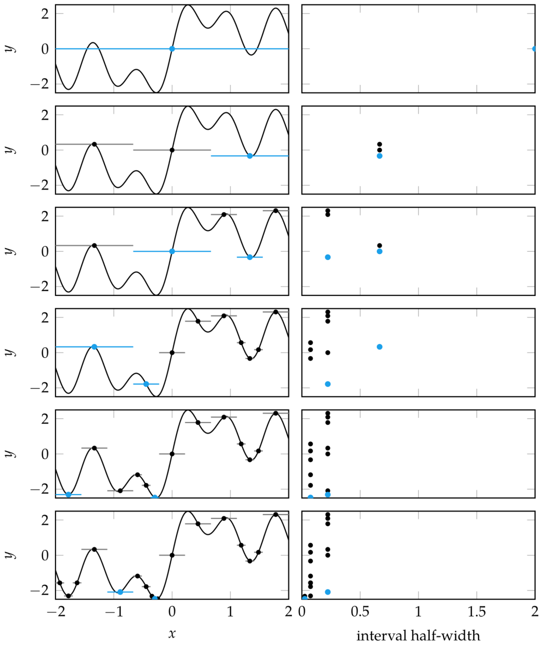



4
3
y
−4
−2
0
2
4
x
f(x)
ℓ = 0.1
ℓ = 0.2
ℓ = 0.3
ℓ = 0.4
ℓ = 0.5
ℓ = 0.6
ℓ = 0.7
ℓ = 0.8
ℓ = 0.9
ℓ = 1.0
lower-bound minima
evaluation points


4
3
y
−4
−2
0
2
4
x
f(x)
ℓ = 0.1
ℓ = 0.2
ℓ = 0.3
ℓ = 0.4
ℓ = 0.5
ℓ = 0.6
ℓ = 0.7
ℓ = 0.8
ℓ = 0.9
ℓ = 1.0
lower-bound minima
evaluation points


interval center value $f(c)$
interval half-width $(b-a)/2$
$(b_i - a_i)/2$
slope = ℓ
$f(c_i)$
$f(c_i) - \ell(b_i - a_i)/2$


interval center value $f(c)$
interval half-width $(b-a)/2$
suboptimal
potentially optimal

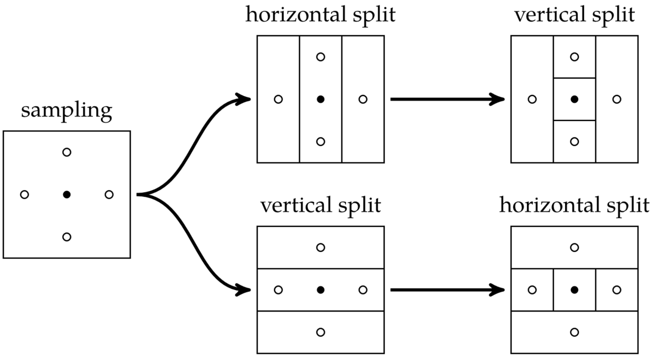


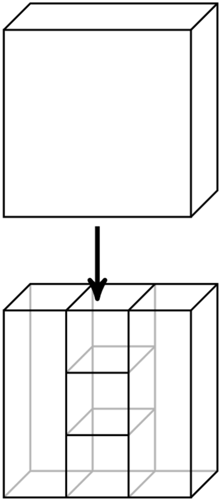
y
2
0
−2
−1
1
x
0.5
1.5
interval half-width

Multivariate DIRECT

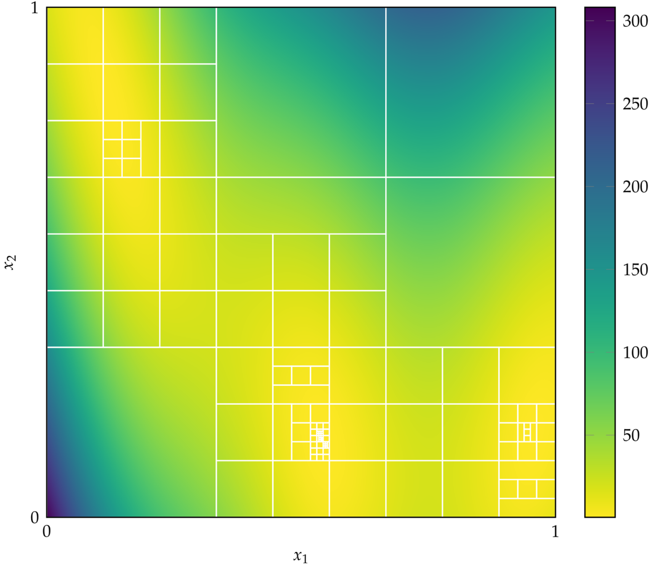
- intervals $\rightarrow$ hyper-rectangles
- normalizes the search space to be the unit hypercube
- divide the rectangles into thirds along the axis directions
- larger rectangles for the points with lower function evaluations
- larger rectangles are prioritized for additional splitting
- when splitting a region without equal side lengths, only the longest dimensions are split

Multivariate DIRECT





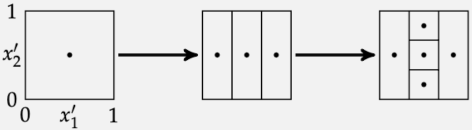
Multivariate DIRECT



Consider using DIRECT to optimize the flower function (appendix B.4) over $x_1 \in [-1, 3]$, $x_2 \in [-2, 1]$. The function is first normalized to the unit hypercube such that we optimize $x'_1, x'_2 \in [0, 1]$:

$$f(x'_1, x'_2) = \text{flower}(4x'_1 - 1, 3x'_2 - 2)$$

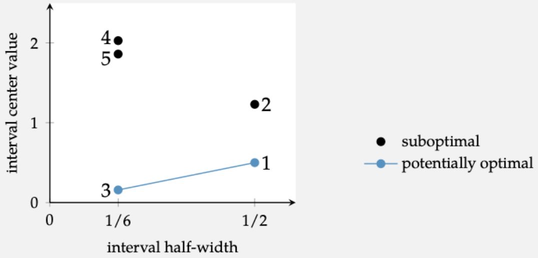
The objective function is sampled at $[0.5, 0.5]$ to obtain 0.158. We have a single interval with center $[0.5, 0.5]$ and side lengths $[1, 1]$. The interval is divided twice, first into thirds in x'_1 and then the center interval is divided into thirds in x'_2.



the interval width can only take on powers of one-third, hence the interval half-width is $\left\|\frac{a-b}{2}\right\|_2 = \left\|\frac{3^{-h}}{2}\right\|_2$ where h is the depth of the rectangle

We now have five intervals:

interval	center	side lengths	vertex distance	center value
1	$[1/6, 3/6]$	$[1/3, 1]$	0.527	0.500
2	$[5/6, 3/6]$	$[1/3, 1]$	0.527	1.231
3	$[3/6, 3/6]$	$[1/3, 1/3]$	0.236	0.158
4	$[3/6, 1/6]$	$[1/3, 1/3]$	0.236	2.029
5	$[3/6, 5/6]$	$[1/3, 1/3]$	0.236	1.861



We next split on the two intervals centered at $[1/6, 3/6]$ and $[3/6, 3/6]$.

Summary

- Direct methods rely solely on the objective function and do not use derivative information.
- Cyclic coordinate search optimizes one coordinate direction at a time.
- Powell's method adapts the set of search directions based on the direction of progress.
- Hooke-Jeeves searches in each coordinate direction from the current point using a step size that is adapted over time.
- Generalized pattern search is similar to Hooke-Jeeves, but it uses fewer search directions that positively span the design space.
- The Nelder-Mead simplex method uses a simplex to search the design space, adaptively expanding and contracting the size of the simplex in response to evaluations of the objective function.
- The divided rectangles algorithm extends the Shubert-Piyavskii approach to multiple dimensions and does not require specifying a valid Lipschitz constant.