



IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

OPTIMIZATION IN CHEMICAL ENGINEERING

Unconstrained Multivariable Optimization: Direct Search Methods

(WEEK – 6: LECTURE – 26)

Dr. Debasis Sarkar

**Department of Chemical Engineering
Indian Institute of Technology Kharagpur**

Unconstrained Multivariable Optimization: Direct Search Methods

Week 6:

- Simplex method
- Hooke-Jeeves pattern search method
- Powell's conjugate direction method

➤ Today's Topic:

- Optimality criteria
- Descent direction
- Classification

Optimality Criteria for Unconstrained Multi-Variable Functions: Necessary Conditions: Review

The necessary conditions for $\mathbf{x}^*$ to be a stationary point of a multivariable function $f(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}^*$:

1. The function $f(\mathbf{x})$ is twice differentiable
2. At stationary point, each partial derivative is equal to zero. Thus each element of the gradient vector of $f(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}^*$ must be equal to zero.
3. The Hessian matrix H is:
 - positive semi-definite when $\mathbf{x}^*$ is a local or relative minimum point
 - negative semi-definite when $\mathbf{x}^*$ is a local relative maximum point.

Optimality Criteria for Unconstrained Multi-Variable Functions: Sufficient Conditions: Review

For a stationary point $\mathbf{x}^*$ to be an extreme point (maximum or minimum), the Hessian matrix H of $f(\mathbf{x})$ evaluated at $\mathbf{x}^*$ must be:

- positive definite when $\mathbf{x}^*$ is a local or relative minimum point
- negative definite when $\mathbf{x}^*$ is a local relative maximum point.
- If the Hessian at $\mathbf{x}^*$ is neither positive semi-definite nor negative semi-definite, the stationary point, $\mathbf{x}^*$, is neither a local maximum nor a local minimum. It is a saddle point.

Optimality Criteria of Multivariable Functions: Summary

Consider a function of n variables, $f(x)$. At x^* :

$$\frac{\partial f}{\partial x_1} = 0, \quad \frac{\partial f}{\partial x_2} = 0, \quad \dots, \quad \frac{\partial f}{\partial x_n} = 0$$

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

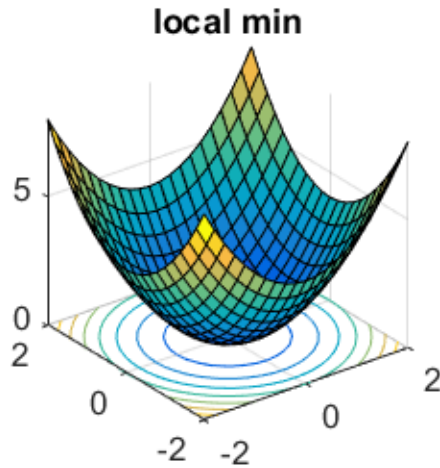
Necessary condition: $\nabla f(x^*) = 0$

Hessian Matrix, $\mathbf{H}$ at x^*	x^* is
Positive semi-definite	minimum
Negative semi-definite	maximum

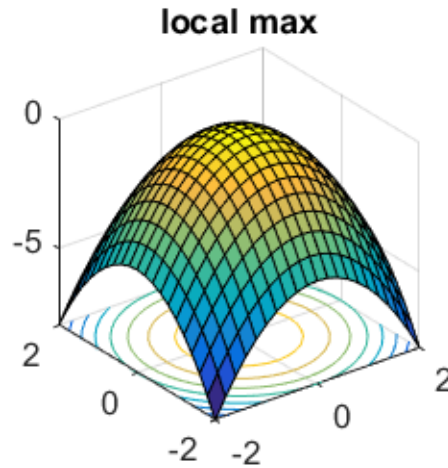
Sufficient condition:

Hessian Matrix, $\mathbf{H}$ at x^*	x^* is
Positive definite	minimum
Negative definite	maximum

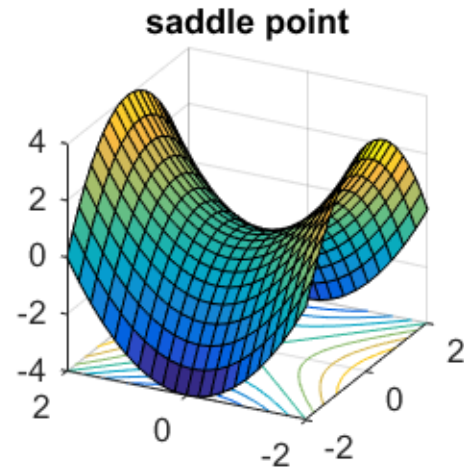
Example



H is positive definite
for minimum.



H is negative definite
for maximum.



H is indefinite for
saddle point.

Example - 1

Find all the minimum and maximum of:

$$f(x, y) = -2x^2 - y^2 + 20x + 16y - 2xy + 10$$

Answer:

Maximum at (2, 6)

Example - 2

Find all the minimum and maximum of:

$$f(x, y) = 4x^2 + 5y^2 - 5x - 9y + 8xy + 100$$

Answer:

Minimum at $(-11/8, 2)$

Example - 3

Find all the minimum and maximum of:

$$f(x, y) = -\frac{x^2}{2} + 2y^2 + 20x - 5y$$

Answer:

Saddle point at $(20, 5/4)$

Descent Direction

Consider the following unconstrained optimization problem: $\min_{x \in \mathcal{R}} f(x)$

At any k -th iteration the next point is given by: $x_{k+1} = x_k + \alpha_k d_k$

Here d_k is a search direction and α_k is the step length (positive).

If d_k is a descent direction: $f(x_{k+1}) < f(x_k) \Rightarrow f(x_k + \alpha_k d_k) < f(x_k)$

By Taylor series expansion:

$$f(x_k) + \alpha_k (\nabla f(x_k) \cdot d_k) < f(x_k)$$

$$\Rightarrow \nabla f(x_k) \cdot d_k < 0$$

Dot product is less than zero.
This is the condition for the
descent direction.

Check for Descent Direction: Example

Consider the two dimensional function: $f(\mathbf{x}) = (2x_1 + x_2^2)^2$

Show that $V = (-1, 0)^T$ is a descent direction at $\mathbf{X}_0 = (0, 1)^T$.

Solution:

$$\nabla f(x_1, x_2) = \begin{pmatrix} 4(2x_1^2 + x_2^2) \\ 4x_2(2x_1 + x_2^2) \end{pmatrix}$$

V is a descent direction if $\nabla f^T(0,1) V < 0$

$$\nabla f(0,1) = \begin{pmatrix} 4(2x_1^2 + x_2^2) \\ 4x_2(2x_1 + x_2^2) \end{pmatrix}_{0,1} = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

$$\text{Now, } \nabla f^T(0,1) V = \begin{bmatrix} 4 & 4 \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix} = -4 < 0$$

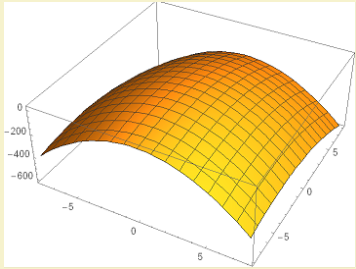
Thus V is a descent direction.

Unconstrained Multivariable Optimization: Classification

- **Direct Search Methods**
 - Nelder and Mead Simplex Search
 - Hook and Jeeves Pattern Search
 - Powell's Conjugated Direction Search
- **Gradient Based Methods**
 - Cauchy's Method (the Steepest Decent)
 - Newton's Method
 - Marquardt's Method

Optimization in Chemical Engineering

Thank You





IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

OPTIMIZATION IN CHEMICAL ENGINEERING

Unconstrained Multivariable Optimization: Direct Search Methods

(WEEK – 6: LECTURE – 27)

Dr. Debasis Sarkar

**Department of Chemical Engineering
Indian Institute of Technology Kharagpur**

Unconstrained Multivariable Optimization: Direct Search Methods

Week 6:

- Simplex search method
- Hooke-Jeeves pattern search method
- Powell's conjugate direction method

➤ Today's Topic:

- General search algorithm
- Unidirectional search
- Simplex search method

General Search Algorithm

$$\underset{x_1, x_2, \dots, x_n}{\text{Minimize}} \ f(x_1, x_2, \dots, x_n)$$

Most iterative procedure has the following general structure:

1. Choose a search direction $\mathbf{s}^k$
2. Minimize in the direction of $\mathbf{s}$ to get $\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta \mathbf{x}^k$
where $\Delta \mathbf{x}^k = \text{increment}$

We must specify: (1) Initial guess or starting vector, (2) Termination criteria

General Search Algorithm: Steps

Let $\mathbf{x}^k$ be the current estimate for $\mathbf{x}^*$

Step – 1: Test for convergence. If conditions are satisfied, *stop*.

The solution is $\mathbf{x}^k$

Step – 2: Compute a nonzero vector $\mathbf{d}^k \in \mathbb{R}^n$ as search direction.

Step – 3: Compute the step length $\alpha_k > 0$ such that

$$f(\mathbf{x}^k + \alpha_k \mathbf{d}^k) < f(\mathbf{x}^k)$$

Step – 4: Update the estimate for minimum.

Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$; $k = k + 1$; go to *Step – 1*

General Search Algorithm: Convergence Rate

These definitions apply to both single-variable and multivariable optimization problems.

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|^r} < p, \quad 0 \leq p \leq 1; r \geq 1$$

Note if $p > 1$, the sequence will diverge

Linearly convergent: $r = 1$ (Slow convergence. Here norm of error decreases by a constant factor at each iteration.)

Superlinear convergence (Fast): $r = 1, p = 0$.

Sublinear convergence: $r = 1, p = 1$.

If $r = 2$, the method is said to be quadratically convergent. Fast convergence. If the initial error is 10^{-1} and $r = 1$, then the sequence of errors will be $10^{-1}, 10^{-2}, 10^{-4}, 10^{-8}, 10^{-16}$, etc.

Unidirectional Search

We can solve a multivariable optimization problem by successively using unidirectional search technique along a specified search direction.

The unidirectional search refers to minimizing a multivariable function along a specified direction by one-dimensional search performed by function value only.

Consider minimization of a multivariable function $f(\mathbf{x})$ in the search direction S_i . if x_i is the initial starting point then we need to determine scalar α such that $f(\alpha) = x_i + \alpha S_i$ is minimized.

The optimal value of α^* gives the new point along the direction of S_i . Repeat this process in a new search direction.

Simplex Search: Introduction

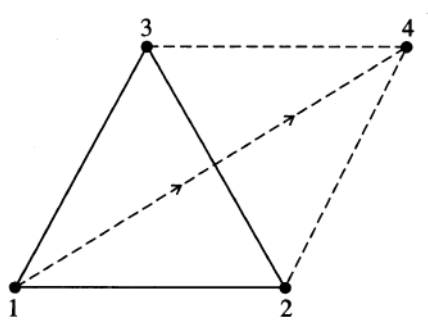
Original formulation: Regular Sequential Search: Spendley, Hext, and Himsworth (1962).

In n dimensions, a regular simplex is a polyhedron composed of $(n+1)$ equidistant points, which form its vertices. For example,

- (a) Equilateral triangle $\rightarrow$ Simplex in 2D
- (b) Tetrahedron $\rightarrow$ Simplex in 3D

A new simplex can be formed on any face of the old simplex by projecting any chosen vertex a suitable distance through the centroid of the remaining vertices of the old simplex. The old vertex is replaced by the newly generated projected point. Thus each new simplex is generated by a single evaluation of the objective function.

Simplex Search: Minimization of a Function



Let $f(x_1)$ is highest.

Reflection of point 1 to
a new point 4.

For minimization:

Best point $\rightarrow$ lowest f

Worst point $\rightarrow$ highest f

The function value is evaluated at each of three vertices. Consider the worst point. The direction of search is oriented away from this worst point (highest value for the function) through the centroid of the simplex. By making the search direction bisect the line between the other two points of the triangle, the direction goes through the centroid. A new point is selected in this reflected direction, preserving the geometric shape. The objective function is then evaluated at the new point, and a new search direction is determined. The method proceeds, rejecting one vertex at a time until the simplex straddles the optimum.

Problem: Excessive repetition of the same cycle or simplexes.

Simplex Search: How To Generate a Regular Simplex?

Requirement:

1. Generation of a regular simplex given a base point and an appropriate scale factor.
2. Calculation of reflected point.

Given an n-dimensional starting (or base point) x_0 and a scale factor (say α), the other n vertices of the simplex in n dimensions are given by:

$$x_j^{(i)} = \begin{cases} x_j^{(0)} + \delta_1, & \text{if } j = i \\ x_j^{(0)} + \delta_2, & \text{if } j \neq i \end{cases}$$

$$\delta_1 = \left[\frac{(n+1)^{1/2} + n - 1}{n\sqrt{2}} \right] \alpha$$

$$\delta_2 = \left[\frac{(n+1)^{1/2} - 1}{n\sqrt{2}} \right] \alpha$$

Simplex Search: How to Calculate Reflected Point?

- Sort points x_k according to objective function values:
$$f(x_1) < f(x_2) < \dots < f(x_{n+1})$$
- Consider the worst point x_{n+1} . Calculate the center of gravity x_c for remaining n points ($x_1, \dots, x_n$) as

$$x_c = \frac{1}{n} \sum_{k=1}^n x_k$$

- Replace worst point with a point reflected through the centroid x_c

$$x_r = x_{n+1} + r * (x_c - x_{n+1})$$

Note: $r = 0$ gives original point. $r = 1$ gives the centroid itself. $r = 2$ will retain the regularity of the simplex as the reflection will be symmetric. Thus, the new point is obtained as:

$$x_{\text{new}} = 2x_c - x_{\text{worst}}$$

Simplex Search: Algorithm

- Start with $(n+1)$ points that define the initial simplex. Evaluate the objective function value at each of these points (vertices).
- Sort points x_k according to objective function values: $f(x_1) < f(x_2) < \dots < f(x_{n+1})$
- Consider the worst point x_{n+1} . Calculate the center of gravity x_c for remaining n points $(x_1, \dots, x_n)$ as:

$$x_c = \frac{1}{n} \sum_{k=1}^n x_k$$

- Replace worst point with a point reflected through the centroid x_c and get the new point as: $x_{\text{new}} = 2x_c - x_{\text{worst}}$
- Evaluate the new point. If it is not an improvement, consider the next worst point. Repeat the process.

Simplex Search: Termination

When the optimum is approached, the last simplex straddles the optimum point or is very close to the optimum point. We may not get closer to the optimum and if we iterate further the simplex repeats itself.

Obviously, we need to change the size of the simplex to enforce convergence. We can consider the midpoint of all the sides of the last simplex and a new simplex composed of the midpoints may be constructed.

The search is terminated when the simplex becomes small enough or the function values at all the vertices are very close to each other.

Simplex Search: Example

Minimize $f(x) = (1 - x_1)^2 + (2 - x_2)^2$
 x_1, x_2

Consider an initial point $= [0, 0]^T$ and scale factor $\alpha = 2$.

$$\delta_1 = 1.9318, \delta_2 = 0.5176$$

$$x^{(0)} = [0, 0]^T; f(x^{(0)}) = 5$$

$$x^{(1)} = [0 + \delta_1, 0 + \delta_2]^T = [1.9318, 0.5176]^T;$$

$$f(x^{(1)}) = 3.0658$$

$$x^{(2)} = [0 + \delta_2, 0 + \delta_1]^T = [0.5176, 1.9318]^T;$$

$$f(x^{(2)}) = 0.2374$$

$x^{(0)}$ is worst. Reflect this point.

$$\text{Get } x^{(3)} = [2.4494, 2.4494]$$

$$f(x^{(3)}) = 2.3027. \text{ This is an improvement.}$$

The new simplex will consist of $x^{(1)}, x^{(2)}, x^{(3)}$

$$f(x^{(1)}) = 3.0658$$

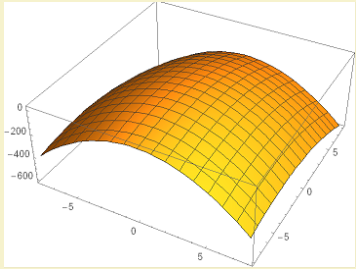
$$f(x^{(2)}) = 0.2374$$

$$f(x^{(3)}) = 2.3027$$

$x^{(1)}$ is the worst point. Reflect it and continue.

Optimization in Chemical Engineering

Thank You





IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

OPTIMIZATION IN CHEMICAL ENGINEERING

Unconstrained Multivariable Optimization: Direct Search Methods

(WEEK – 6: LECTURE – 28)

Dr. Debasis Sarkar

**Department of Chemical Engineering
Indian Institute of Technology Kharagpur**

Unconstrained Multivariable Optimization: Direct Search Methods

Week 6:



- Simplex method
- Hooke-Jeeves pattern search method
- Powell's conjugate direction method

➤ Today's Topic:

- Simplex search method (Cont'd)

Simplex Search: Review

When the optimum is approached, the last simplex straddles the optimum point or is very close to the optimum point. We may not get closer to the optimum and if we iterate further the simplex repeats itself.

Obviously, we need to change the size of the simplex to enforce convergence. We can consider the midpoint of all the sides of the last simplex and a new simplex composed of the midpoints may be constructed.

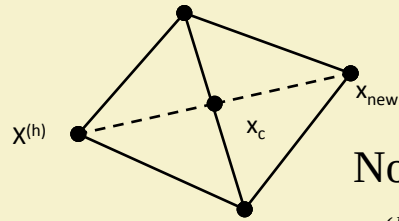
The search is terminated when the simplex becomes small enough or the function values at all the vertices are very close to each other.

Simplex Search: Improvement

- We need to change the size of the simplex to enforce convergence
- Also, want to adjust the step size
 - If new point is best point – increase step size
 - If new point is worse then $x_1 \dots x_n$ – decrease step size

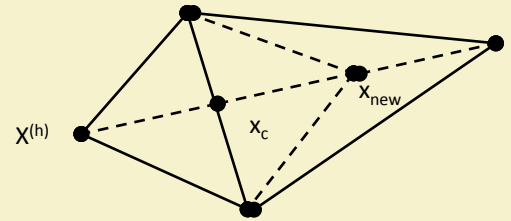
Nelder and Mead (1965) described a more efficient version of the simplex method that permitted the geometric figures to expand and contract continuously during the search. Nelder and Mead Simplex search method will minimize a function of n variables using $(n+1)$ vertices of a flexible polyhedron.

Nelder Mead Simplex Search



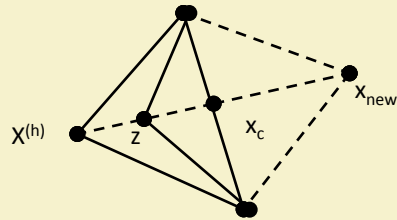
Normal Reflection

$$f^{(l)} < f(x_{new}) < f^{(h)}$$



Expansion ($\gamma > 1$)

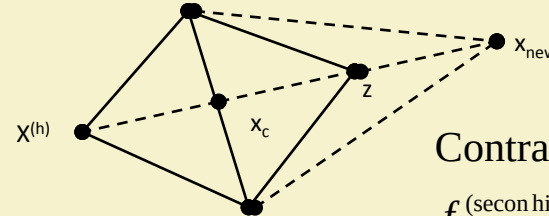
$$f(x_{new}) < f^{(l)}$$



Contraction ($\beta < 0$)

$$f(x_{new}) > f^{(h)}$$

Shrinkage



Contraction ($\beta > 0$)

$$f^{(\text{second highest})} < f(x_{new}) < f^{(h)}$$

Nelder Mead Simplex Search: Algorithm

Step – 1: Choose $\gamma > 1$; $\beta \in [0, 1]$; ε (stopping criteria)

Generate an initial simplex.

Step – 2: Let : x_h = worst point, x_l = best point, x_g = next to worst point.

$$\text{Calculate } x_c = \frac{1}{n} \sum_{i=1, i \neq h}^{n+1} x_i$$

Step – 3: Compute the reflected point: $x_r = 2x_c - x_h$. Set $x_{new} = x_r$.

If $f(x_r) < f(x_l)$, Perform expansion : set $x_{new} = (1 + \gamma)x_c - \gamma x_h$;

Else if $f(x_r) \geq f(x_h)$, Perform contraction : set $x_{new} = (1 - \beta)x_c + \beta x_h$;

Else if $f(x_g) < f(x_r) < f(x_h)$, Perform contraction : set $x_{new} = (1 + \beta)x_c - \beta x_h$;

Calculate $f(x_{new})$ and replace x_h by x_{new} .

Step – 4: If $T = \left[\sum_{i=1}^{n+1} \frac{(f(x_i) - f(x_c))^2}{n+1} \right] \leq \varepsilon$, Stop

Else go to Step – 2

Nelder Mead Simplex Search: Example

Let us consider the Himmelblau function:

$$\underset{x_1, x_2}{\text{Minimize}} \quad f(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$$

Nelder Mead Simplex Search: MATLAB

Write a function file as follows:

```
function f = HimmelblauFun(x)
f = (x(1)^2+x(2)-11)^2 + (x(1)+x(2)^2-7)^2;
```

Now write this on MATLAB command window (or write a script):

```
x0 = [0, 0];
options = optimset('Display','iter','TolFun',1e-8);
[x, fval, exitflag] = fminsearch(@HimmelblauFun,x0,options)
```

Nelder Mead Simplex Search: MATLAB Output

Iteration	Func-count	min $f(x)$	Procedure
0	1	170	
1	3	169.994	initial simplex
2	5	169.986	expand
3	7	169.978	expand
4	9	169.958	expand
5	11	169.932	expand
6	13	169.879	expand
7	15	169.8	expand
8	17	169.653	expand
9	19	169.419	expand
10	21	168.999	expand

Nelder Mead Simplex Search: MATLAB Output

75	145	4.00995e-007	contract inside
76	147	4.00995e-007	contract inside
77	149	1.53635e-007	contract inside
78	151	7.83007e-008	contract inside
79	153	3.59737e-008	contract inside
80	155	3.59737e-008	contract inside
81	157	1.43332e-008	contract inside
82	159	7.77599e-009	contract inside
83	161	3.29678e-009	contract inside
84	163	3.29678e-009	contract inside

Solution:

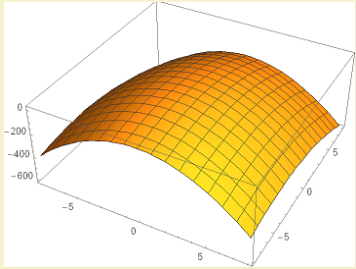
Optimization terminated:

the current x satisfies the termination criteria using OPTIONS.TolX of 1.000000e-04
and F(X) satisfies the convergence criteria using OPTIONS.TolFun of 1.000000e-08

x = 3.0000 2.0000, fval = 3.2968e-09, exitflag = 1

Optimization in Chemical Engineering

Thank You





IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

OPTIMIZATION IN CHEMICAL ENGINEERING

Unconstrained Multivariable Optimization: Direct Search Methods

(WEEK – 6: LECTURE – 29)

Dr. Debasis Sarkar

**Department of Chemical Engineering
Indian Institute of Technology Kharagpur**

Unconstrained Multivariable Optimization: Direct Search Methods

Week 6:

-  Simplex search method
- Hooke-Jeeves pattern search method
- Powell's conjugate direction method

➤ Today's Topic:

- Hooke-Jeeves pattern search method

Univariate Search

The univariate search refers to minimizing a multivariable function with respect to one variable at a time, keeping all other variables constant. We can use any direct search techniques for single variable optimization.

$$x_i^{(k+1)} = x_i^{(k)} + \alpha_i; \quad i = 1, 2, \dots, n$$

Here k is iteration number and the increment α_i is calculated to minimize the function in the coordinate direction x_i

$$\underset{\alpha}{\text{Minimize}} \quad f(x_i^{(k)} + \alpha)$$

If the problem as expresses by above two equations fails to improve function value, then an increment of $(-\alpha)$ is used. If that also fails, the decision variable $x_i^{(k)}$ is left unchanged and we continue search with next decision variable.

Univariate Search (Cont'd)

The univariate search may not converge always. However, univariate search is effective (converges quickly) for a quadratic function of the following form:

$$f(\mathbf{x}) = \sum_{i=1}^n c_i x_i^2$$

This is because here the search directions line up with the principle axes. However, this method does not work so well for more general quadratic function (takes more iterations to converge)

$$f(\mathbf{x}) = \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_i x_j$$

Pattern Search Methods

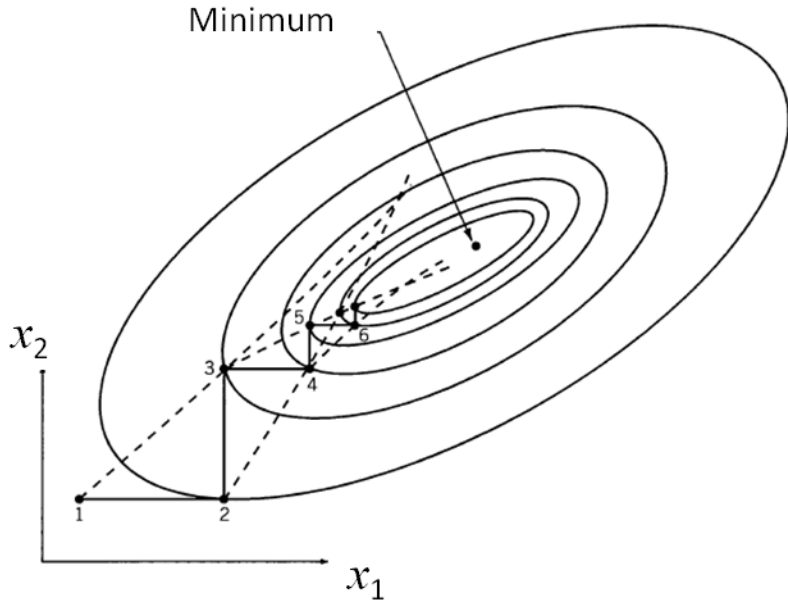
Pattern Directions:

In the univariate method, we search for the minimum along the directions parallel to the coordinate axes. This makes the convergence process slow. Sometimes, the method may not even converge.

These problems can be avoided by changing the directions of search in a favorable manner instead of retaining them always parallel to the coordinate axes. Favorable direction will be the direction toward optimum point. These directions are known as pattern directions.

The process of going from one point to another is called “move”. For minimization of a function, a move is a success if the function value decreases after the move, otherwise it is a failure.

Pattern Direction



Univariate search:

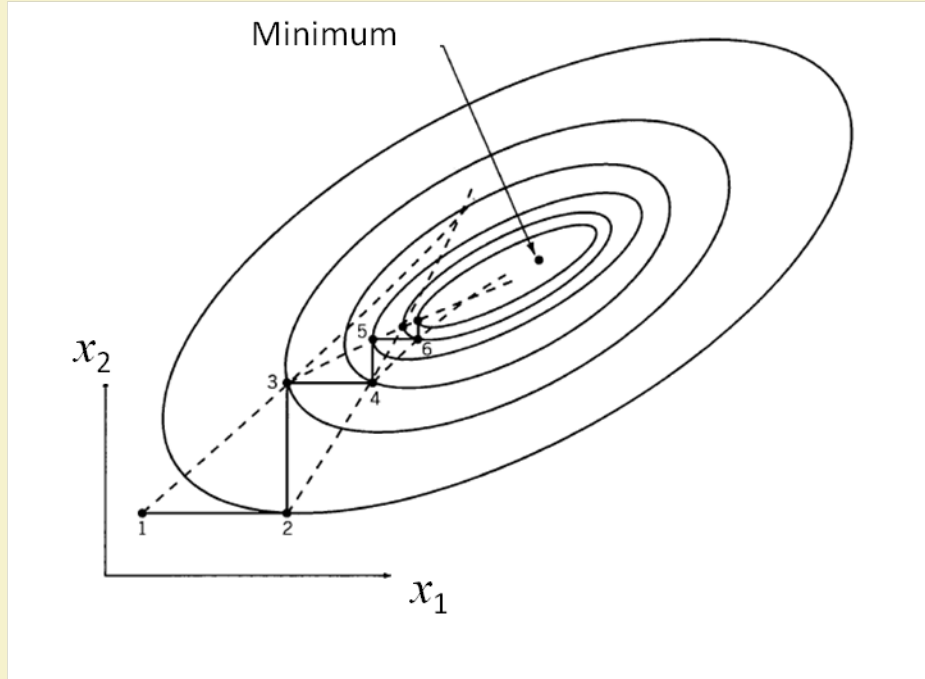
$$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$$

Pattern direction:

$$1 \rightarrow 3 \rightarrow 5$$

If the objective function is quadratic in two variables, pattern directions pass through the minimum. Although for general multivariable functions, this property is not retained, we can use this idea to achieve rapid convergence.

Pattern Direction



For a two variable function, at least two linearly independent search directions are required to go from one point to another.

For a n -variable function, we need n linearly independent search directions to go from one point to any other point.

There are many possible combinations of such directions. Some combinations will converge quickly compared to other combinations.

Pattern Search Methods

Methods that use pattern directions as search directions are known as **pattern search methods**. Two well known pattern search methods are:

- Hooke-Jeeves method
- Powell's method

In general, a pattern search method takes n univariate steps, where n denotes the number of design variables and then searches for the minimum along the pattern direction $\mathbf{S}_i$, defined by

$$\mathbf{S}_i = \mathbf{X}_i - \mathbf{X}_{i-n}$$

where $\mathbf{X}_i$ is the point obtained at the end of n univariate steps. $\mathbf{X}_{i-n}$ is the starting point before taking n univariate steps.

Hooke – Jeeves Pattern Search Methods

The pattern search method of Hooke and Jeeves is a sequential technique. Each step of this method consists of two kinds of moves:

- ☐ the exploratory move, and
- ☐ the pattern move.

The exploratory move is performed systematically in the vicinity of the current point using one-variable-at-a-time technique to explore the local behaviour of the objective function. The goal is to obtain the best point around the current point.

The pattern move is then performed using two best points (current and previous) to take advantage of the pattern direction.

Hooke – Jeeves Pattern Search Method

Exploratory Move:

The exploratory move is performed by one-variable-at-a-time technique. Given a specified step size, the current point (base point) is perturbed in positive and negative directions along each variable and the best point is recorded. The current point is changed to the best point at the end of each variable perturbation.

The exploratory move is a success, if after all the perturbations we get a point different from the original point; otherwise the exploratory move is a failure.

Note that the outcome of the exploratory move is always the best point and the best point is the new base point.

Hooke – Jeeves Pattern Search Method

Pattern Move:

The pattern move is performed along pattern direction by considering the current best point $x^{(k)}$ and the previous best point $x^{(k-1)}$. The pattern direction is obtained by connecting these two points as follows:

$$x_p^{(k+1)} = x^{(k)} + \left(x^{(k)} - x^{(k-1)} \right)$$

If the pattern move does not improve the solution, the move is not accepted and the exploratory move is performed with reduced step size.

The search is terminated when the step size becomes sufficiently small.

Hooke – Jeeves Pattern Search Method: Algorithm: Exploratory Search

Step – 1: Assume current solution (base point) = x^c ; increment $\Delta_i, i = 1, \dots, n$;

Set $i = 1$ and $x = x^c$

Step – 2: Calculate $f = f(x)$, $f^+ = f(x_i + \Delta_i)$, $f^- = f(x_i - \Delta_i)$;

Find $f_{\min} = \min(f, f^+, f^-)$. Set x corresponding to $f_{\min}$.

Step – 3: If $i \neq n$, set $i = i + 1$ and go to *Step – 2*

Else x is the result and go to *Step – 4*

Step – 4: If $x \neq x^c$, exploratory move is success, Else failure

Hooke – Jeeves Pattern Search Method: Overall Algorithm

Step – 1: Define: starting point $x^{(0)}$; increment $\Delta_i, i = 1, \dots, n$;

step reduction factor $\alpha > 1$; termination factor $\varepsilon > 0$. Set $k = 0$

Step – 2: Perform exploratory search with $x^{(k)}$ as base point.

Let x = result of exploratory search. If exploratory search is a success,

set $x^{(k+1)} = x$ and go to *Step – 4*; Else go to *Step – 3*.

Step – 3: If $\|\Delta_i\| < \varepsilon$, *Stop*; Current solution $\approx x^*$. Else set $\Delta_i = \Delta_i / \alpha$ and go to *Step – 2*

Step – 4: Set $k = k + 1$ and perform pattern move : $x_p^{(k+1)} = x^{(k)} + (x^{(k)} - x^{(k-1)})$

Step – 5: Perform exploratory search with $x_p^{(k+1)}$ as base point. Say, output = $x^{(k+1)}$

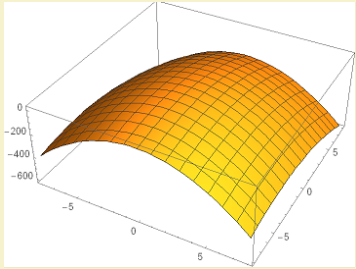
Step – 6: If $f(x^{(k+1)}) < f(x^{(k)})$, go to *Step – 4*; Else go to *Step – 3*.

Hooke – Jeeves Pattern Search Method: Example

Find the minimum of $f(x_1, x_2) = 8x_1^2 + 4x_1x_2 + 5x_2^2$, $x^{(0)} = [-4, 4]^T$

Optimization in Chemical Engineering

Thank You





IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

OPTIMIZATION IN CHEMICAL ENGINEERING

Unconstrained Multivariable Optimization: Direct Search Methods

(WEEK – 6: LECTURE – 30)

Dr. Debasis Sarkar

Department of Chemical Engineering
Indian Institute of Technology Kharagpur

Unconstrained Multivariable Optimization: Direct Search Methods

Week 6:

- Simplex search method
- Hooke-Jeeves pattern search method
- Powell's conjugate direction method

➤ Today's Topic:

- Powell's conjugate direction method

Powell's Conjugate Direction Method

Powell's method is an extension of the basic pattern search method. It is the one of the most successful and widely used direct search method.

Powell's method is a method of conjugate directions. A conjugate directions method will minimize a quadratic function in a finite number of steps.

A general nonlinear function can be approximated reasonably well by a quadratic function near its minimum. Thus, a conjugate directions method is expected to speed up the convergence of even general nonlinear objective functions.

Powell's Conjugate Direction Method

Powell's method first generates a set of n linearly independent search directions. Starting each time from the previous best point, unidirectional searches are performed along each of these search directions. The last search direction replaces the first one in the new iteration and the process is continued until the function value shows no improvement.

Previous search directions are stored in this method and they form a basis for the new search direction.

Powell's Conjugate Direction Method

This method guarantees to find the minimum of a quadratic function by one pass of n unidirectional searches along each search direction.

For general nonlinear unconstrained multivariable functions, more than one pass of n unidirectional searches are necessary.

What is a Conjugate Direction?

Conjugate directions are more effective as search directions than arbitrarily chosen search directions. Two directions s_1 and s_2 are said to be conjugate with respect to a positive-definite matrix Q if

$$s_1^T Q s_2 = 0$$

Let Q be an $(n \times n)$ square positive definite symmetric matrix. A set of n linearly independent search directions (i.e. vectors) $\{S_i\}$ is said to be conjugate with respect to matrix Q if

$$S_i^T Q S_j = 0 \quad \text{for all } i \neq j, \quad i = 1, 2, \dots, n, \quad j = 1, 2, \dots, n$$

For optimization, Q is the Hessian (H) of the objective function. Orthogonal (perpendicular) directions are a special case of conjugate directions. In this case $Q = I$.

Conjugate Direction

For a quadratic function of $f(x)$ of n variables, H is a constant matrix. For quadratic function we are guaranteed to reach the minimum of $f(x)$ in n stages if we minimize exactly on each stage.

In general, conjugate direction is not a unique direction.

In n dimensions (i.e. for n -variable functions), many different sets of conjugate directions exist for a given matrix Q (or H).

In two dimensions (i.e. for two-variable functions), if we choose an initial direction s_1 and Q , s_2 is fully specified because

$$s_1^T Q s_2 = 0$$

Conjugate Direction: Example

We want to minimize the following objective function starting at $\mathbf{x}_0 = [1 \ 1]^T$ with the initial direction $\mathbf{S}^0 = [-4 \ -2]^T$. Find a conjugate direction to the initial search direction $\mathbf{S}^0$.

$$f(x_1, x_2) = 2x_1^2 + x_2^2 + 6$$

Solution:

Let the conjugate direction be $\mathbf{s}^1 = \begin{bmatrix} s_1^1 \\ s_2^1 \end{bmatrix}$

$$\text{Given, } \mathbf{s}^0 = \begin{bmatrix} -4 \\ -2 \end{bmatrix}, \quad \mathbf{H}(\mathbf{x}) = \begin{bmatrix} 4 & 0 \\ 0 & 2 \end{bmatrix}$$

Now solve: $(\mathbf{s}^0)^T \mathbf{H} \mathbf{s}^1 = 0$

$$\Rightarrow \begin{bmatrix} -4 & -2 \end{bmatrix} \begin{bmatrix} 4 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} s_1^1 \\ s_2^1 \end{bmatrix} = 0$$

Since s_1^1 is not unique,

we can select $s_1^1 = 1$ and find s_2^1

$$\begin{bmatrix} -4 & -2 \end{bmatrix} \begin{bmatrix} 4 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ s_2^1 \end{bmatrix} = 0$$

$$\Rightarrow \begin{bmatrix} -16 & -4 \end{bmatrix} \begin{bmatrix} 1 \\ s_2^1 \end{bmatrix} = 0$$

$$\Rightarrow s_1 = \begin{bmatrix} 1 \\ -4 \end{bmatrix}$$

Parallel Subspace Property

Consider a quadratic function of two variables: $q(x) = a + b^T x + \frac{1}{2} x^T Q x$

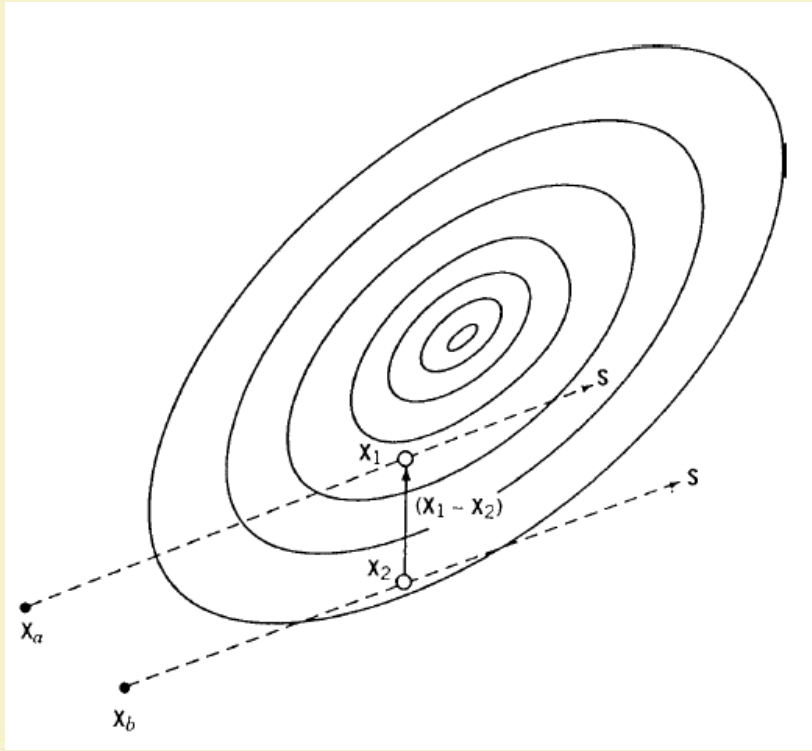
Here a is scalar, b is a vector, and Q is 2 X 2 matrix. Also consider two arbitrary but distinct points $x^{(1)}$ and $x^{(2)}$, and a direction d .

Let $y^{(1)}$ is solution to the problem: minimize $q(x^{(1)} + \lambda d)$

And $y^{(2)}$ is the solution to the problem: minimize $q(x^{(2)} + \lambda d)$

Then the direction $(y^{(2)} - y^{(1)})$ is conjugate to d . In other words: $(y^{(2)} - y^{(1)})^T Q d = 0$

Parallel Subspace Property



X_1 and X_2 are the minima of the quadratic function obtained by searching along the direction S from two different starting points X_a and X_b , respectively.

Then line $(X_1 - X_2)$ will be conjugate to the search direction S .

Extended Parallel Subspace Property

Let us consider:

$y^{(1)}$ is obtained after unidirectional searches along each of $m (\leq n)$ conjugate directions from a chosen point $x^{(1)}$

$y^{(2)}$ is obtained after unidirectional searches along each of $m (\leq n)$ conjugate directions from another point $x^{(2)}$

Then the direction $(y^{(2)} - y^{(1)})$ is conjugate to all m search directions.

Theorem

If a quadratic function

$$q(x) = a + b^T x + \frac{1}{2} x^T Q x$$

is minimized sequentially, once along each direction of a set of n mutually conjugate directions, the minimum of the function will be found at or before the n -th step irrespective of the starting point.

Powell's Conjugate Direction Method: Algorithm

Step – 1: Define : starting point $x^{(0)}$;

a set of linearly independent directions, $s^{(i)}$

Step – 2: Minimize along n unidirectional search directions
using the previous minimum point to begin the next search.

Start the search with $s^{(1)}$ and end with $s^{(n)}$.

Then perform another unidirectional search along $s^{(1)}$.

Step – 3: Use extended parallel subspace property to form new conjugate direction, d .

Step – 4: If $\|d\|$ is small or search directions are linearly dependent, Stop

Else replace $s^{(j)} = s^{(j-1)}$ for all $j = n, n-1, \dots, 2$. Set $s^{(1)} = \frac{d}{\|d\|}$ and go to *Step – 2*.

Powell's Conjugate Direction Method: Example

Find the minimum of $f(x_1, x_2) = 2x_1^3 + 4x_1x_2^3 - 10x_1x_2 + x_2^2$, $x^{(0)} = [5, 2]^T$

Optimization in Chemical Engineering

Thank You

