Lecture Note of Topics in Ship Design Automation

Optimum Design

Fall 2015

Myung-Il Roh

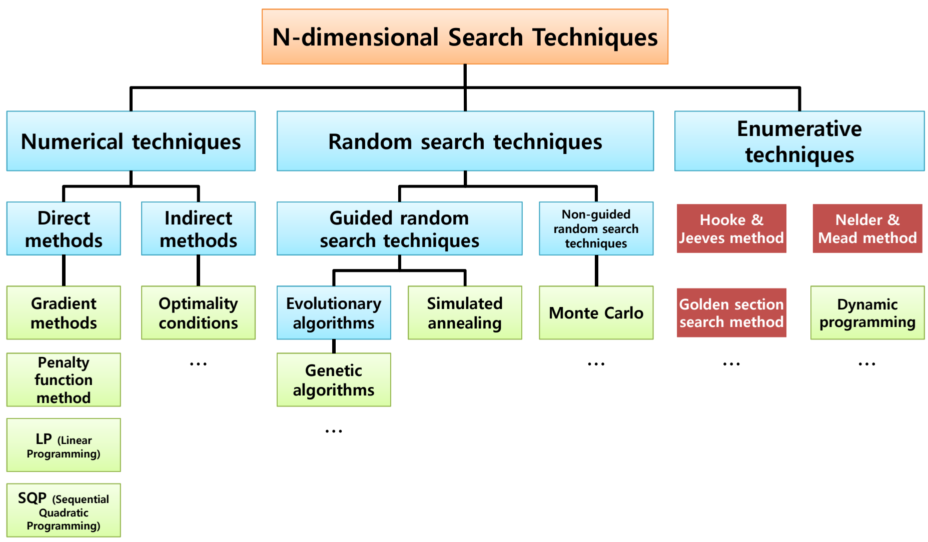
Department of Naval Architecture and Ocean Engineering
Seoul National University

Contents

Ch. 3 Unconstrained Optimization Method: Enumerative Method

3.1 Hooke & Jeeves Method
3.2 Nelder & Mead Simplex Method
3.3 Golden Section Search Method (One Dimensional Search Method)

Classes of Search Techniques

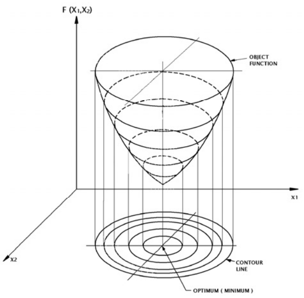


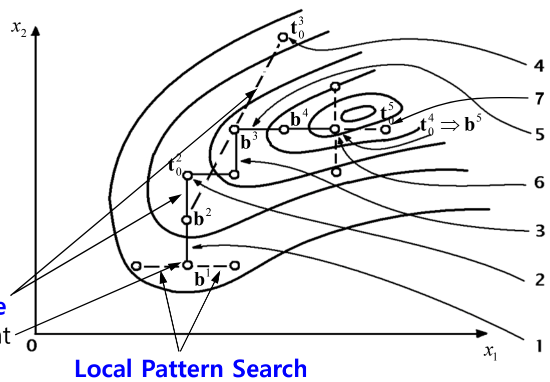
3.1 Hooke & Jeeves Method

Hooke & Jeeves Method (1/16)

1. Base Point
2. Global Pattern Move
3. Local Pattern Search

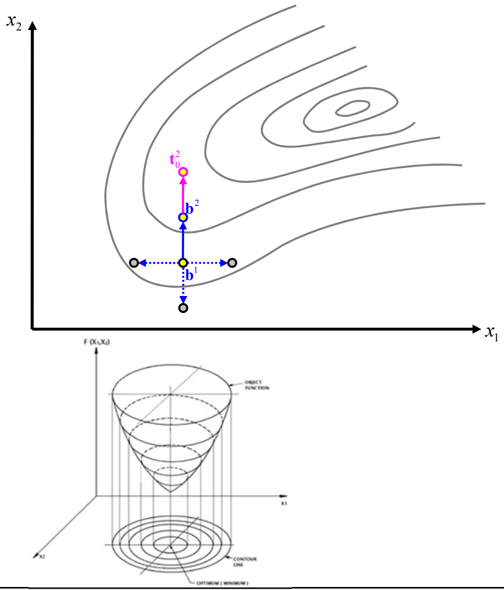
☑ **This method is a sequential technique, each step of which consists of two kinds of move, the 'Local Pattern Search' at a base point and 'Global Pattern Move' to the optimal point.**





Hooke & Jeeves Method (2/16)

1. Base Point
2. Global Pattern Move
3. Local Pattern Search



1. 'Local Pattern Search' at the base point b^1

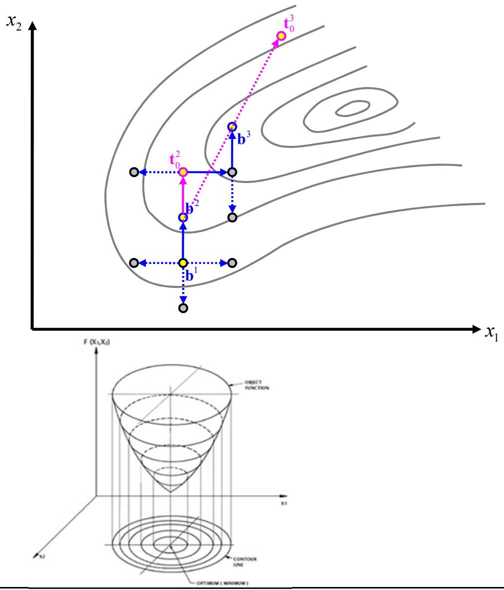
- Search in x_1 direction.
  - No improvement of the value of the objective function in x_1 direction
  ➡ No movement in x_1 direction
- Search in x_2 direction.
  - Improvement of the value of the objective function in x_2 direction
  ➡ Movement in the positive x_2 direction
- Move to and define the base point b^2.

2. 'Global Pattern Move' at the base point b^2

- Find a temporary base point t_0^2 by symmetrical displacement of b^1 to b^2.
- Because the value of the objective function at t_0^2 is better than that at b^2, perform the 'Local Pattern Search' at t_0^2.

Hooke & Jeeves Method (3/16)

1. Base Point
2. Global Pattern Move
3. Local Pattern Search



3. 'Local Pattern Search' at the temporary base point t_0^2

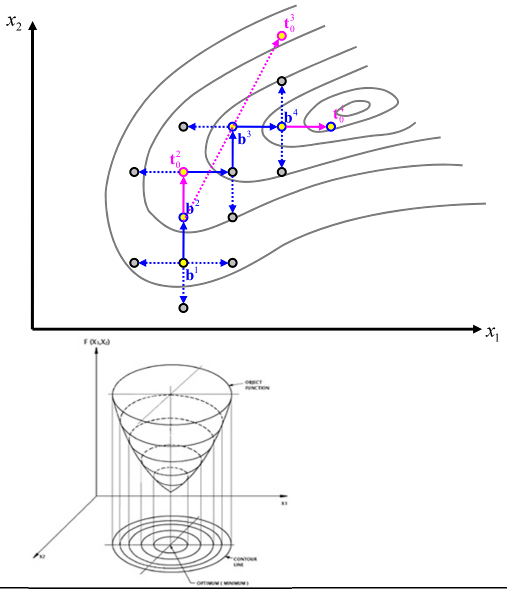
- Search in x_1 direction.
  - Improvement of the value of the objective function in x_1 direction
  ➡ Movement in the positive x_1 direction
- Search in x2 direction.
  - Improvement of the value of the objective function in x_2 direction
  ➡ Movement in the positive x_2 direction
- Move to and define the base point b^3.

4. 'Global Pattern Move' at the base point b^3

- Find a temporary base point t_0^3 by symmetrical displacement of b^2 to b^3.
- Because the value of the objective function at t_0^3 is not better than that at b^3, perform the 'Local Pattern Search' at b^3.

Hooke & Jeeves Method (4/16)

1. Base Point
2. Global Pattern Move
3. Local Pattern Search



5. 'Local Pattern Search' at the base point b^3

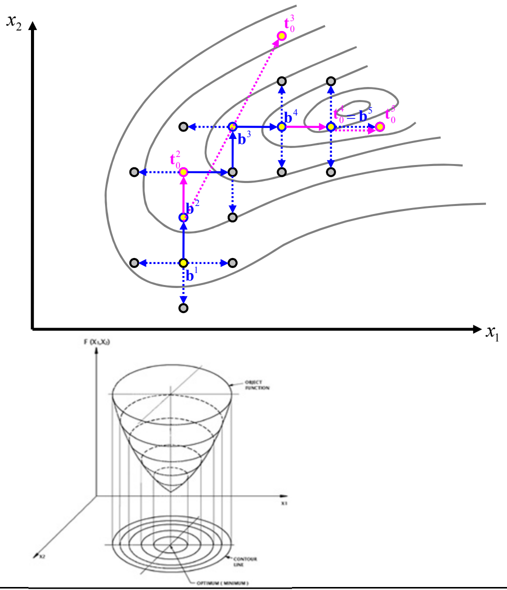
- Search in x_1 direction.
  - Improvement of the value of the objective function in x_1 direction
  ➡ Movement in the positive x_1 direction
- Search in x_2 direction.
  - No improvement of the value of the objective function in x_2 direction
  ➡ No movement in x_2 direction
- Move to and define the base point b^4.

6. 'Global Pattern Move' at the base point b^4

- Find a temporary base point t_0^4 by symmetrical displacement of b^3 to b^4.
- Because the value of the objective function at t_0^4 is better than that at b^4, perform the 'Local Pattern Search' at t_0^4.

Hooke & Jeeves Method (5/16)

1. Base Point
2. Global Pattern Move
3. Local Pattern Search



7. 'Local Pattern Search' at the temporary base point t_0^4

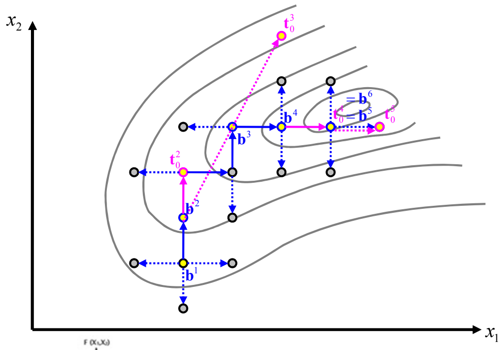
- Search in x_1 direction.
  - No improvement of the value of the objective function in x_1 direction
  ➡ No movement in x_1 direction
- Search in x_2 direction.
  - No improvement of the value of the objective function in x_2 direction
  ➡ No movement in x_2 direction
- Because there is no improvement of the value of the objective function in x_1 and x_2 direction, the current base point is defined as the base point b^5.

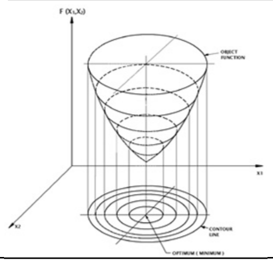
8. 'Global Pattern Move' at the base point b^5

- Find a temporary base point t_0^5 by symmetrical displacement of b^4 to b^5.
- Because the value of the objective function at t_0^5 is not better than at b^5, perform the 'Local Pattern Search' at b^5.

Hooke & Jeeves Method (6/16)

1. Base Point
2. Global Pattern Move
3. Local Pattern Search





9. 'Local Pattern Search' at the base point b^5

- Search in x_1 direction.
  - No improvement of the value of the objective function in x_1 direction

  ➡ No movement in x_1 direction
- Search in x_2 direction.
  - No improvement of the value of the objective function in x_2 direction

  ➡ No movement in x_2 in x_2 direction
- Because there is no improvement of the value of the objective function in x_1 and x_2 direction, the current base point defined as base point b^6.
- Because b^5 = b^6, reduce the step size by half and perform the 'Local Pattern Search' at b^6.

Hooke & Jeeves Method (7/16)
- Rule of the 'Local Pattern Search' (1/2)

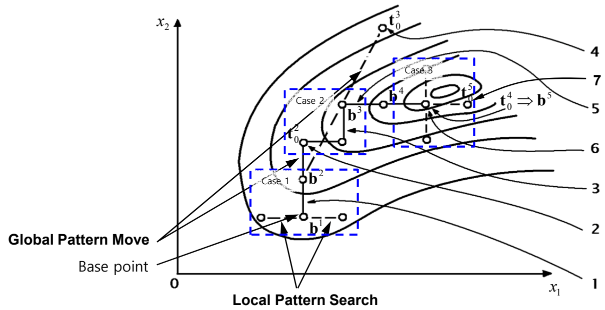
Rule of the 'Local Pattern Search'

(F: Fail, S: Success)

Rule ① Search in the positive x_i direction.		
- Move the exploratory point in the positive x_i direction and evaluate the value of the objective function at that point. b^k	- If the value of the objective function is increased (Fail)	- Come back to the previous point and search in the negative x_i direction. b^k F
	- If the value of the objective function is decreased (Success)	- Search in the x_{i+1} direction at the current point. b^k S
Rule ② Search in the negative x_i direction.		
- If the search in the positive x_i direction is failed, move the exploratory point in the negative x_i direction and evaluate the value of the objective function at that point. b^k F	- If the value of the objective function is increased (Fail)	- Come back to the previous point and search in x_{i+1} direction. F b^k F
	- If the value of the objective function is decreased (Success)	- Search in the x_{i+1} direction at the current point. S b^k F

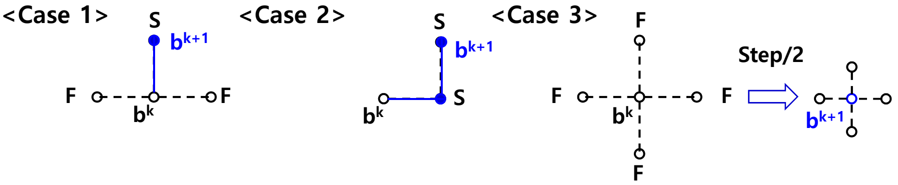
- This process of the 'Local Pattern Search' is continued for i = 1, ..., n.
- After searching in x_n direction, the current point is defined as new base point $\mathbf{b}^{k+1}$.

Hooke & Jeeves Method (8/16) - Rule of the 'Local Pattern Search' (2/2)



* Super script 'k' means the number of step.

▪ **Rule of the Local Pattern Search (F: Fail, S: Success)**



Hooke & Jeeves Method (9/16) - Algorithm Summary (1/4)

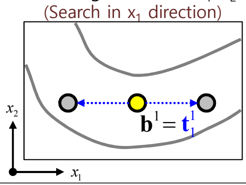
1) Local Pattern Search (Problem with n design variables)

1. Compute the value of the objective function at the starting base point $\mathbf{b}^1$.

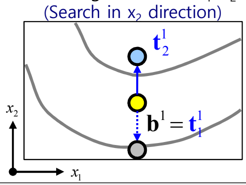
2. Compute the value of the objective function at $\mathbf{b}^1 \pm \boldsymbol{\delta}_1$, where $\boldsymbol{\delta}_1$ is input step size and a vector with n elements ($\boldsymbol{\delta}_1 = [\delta_1, 0, 0, \dots, 0]^T$). If the value of the objective function is decreased, $b^1 \pm \delta_1$ is adopted as $\mathbf{t}_1^{\,1}$ and the search is continued.

3. Compute the value of the objective function at $\mathbf{t}_1^{\,1} \pm \boldsymbol{\delta}_2$, where $\boldsymbol{\delta}_2$ is also input step size and a vector with n elements ($\boldsymbol{\delta}_2 = [0, \delta_2, 0, \dots, 0]^T$). If the value of the function is decreased, $t_1^{\,1} \pm \delta_2$ is adopted as $\mathbf{t}_2^{\,1}$.

Example of the 'Local Pattern Search' in the problem with two design variables (x_1, x_2) (Search in x_1 direction)



Example of the 'Local Pattern Search' in the problem with two design variables (x_1, x_2) (Search in x_2 direction)



Hooke & Jeeves Method (10/16)
- Algorithm Summary (2/4)

1) Local Pattern Search (Problem with n design variables)

4. After the 'Local Pattern Search' for all design variables, new base point is defined. (new base point $\mathbf{b}^2 = \mathbf{t}_n^1$)

5. Perform the 'Global Pattern Move' from the previous base point along the line from the previous to current base point.

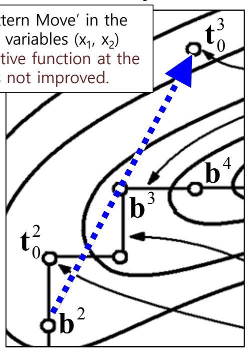
Hooke & Jeeves Method (11/16)
- Algorithm Summary (3/4)

2) Global Pattern Move

1. Define the temporary base point located the same distance between the previous and current base point (obtained from 'Local Pattern Search') from the current base point ('Global Pattern Move'), and calculate the value of the objective function at this point. The temporary base point is calculated by 'Global Pattern Move' as follows.

$$\mathbf{t}_0^{k+1} = \mathbf{b}^k + 2(\mathbf{b}^{k+1} - \mathbf{b}^k) = 2\mathbf{b}^{k+1} - \mathbf{b}^k$$

Example of the 'Global Pattern Move' in the problem with two design variables (x_1, x_2) when the value of the objective function at the temporary base point is not improved.



2. If the result of the temporary base point is a better point than the previous base point, perform the 'Local Pattern Search' at the temporary base point. Otherwise, come back to the previous base point and perform the 'Local Pattern Search'.

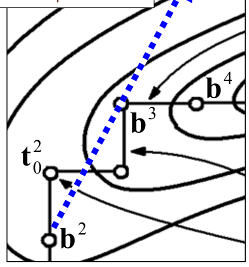
Hooke & Jeeves Method (12/16)
- Algorithm Summary (4/4)

3) Closing Condition (Stopping Criterion)

1. When even this 'Local Pattern Search' fails ($\mathbf{b}^{k+1} = \mathbf{b}^k$, there is no improvement), reduce the step sizes $\boldsymbol{\delta}_i$ by half, $\boldsymbol{\delta}_i/2$, and resume the 'Local Pattern Search'.

2. If the step size $\boldsymbol{\delta}_i$ is smaller than $\boldsymbol{\varepsilon}_i$, stop the iteration and current base point is the optimal point.

Example of the 'Global Pattern Move' in the problem with two design variables (x_1, x_2) when the value of the objective function at the temporary base point is not improved.



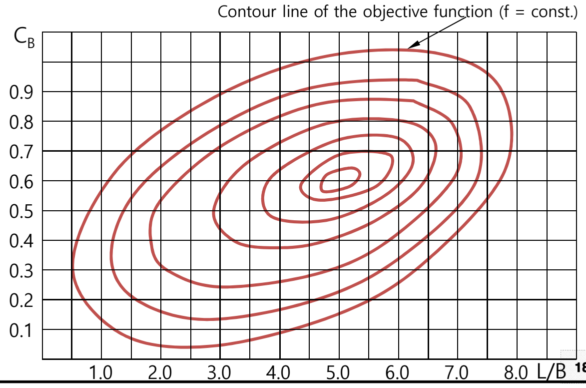
Hooke & Jeeves Method (13/16)
- Example (1/4)

☑ **If the contour line of the objective function of shipbuilding cost with two design variables, L/B and C_B, is given as shown in the Figure, find the optimal value of the L/B and C_B to minimize the shipbuilding cost by using the 'Hooke & Jeeves Direct Search Method' and plot the procedures in the graph.**

- **Hooke & Jeeves Direct Search Method**
  - **Starting design point: L/B = 7.0, C_B = 0.2**
  - **Step size at the starting design point: Δ(L/B) = 0.5, $\Delta(C_B)$ = 0.1**

Contour line of the objective function (f = const.)

Optimization problem with two unknown variables



Hooke & Jeeves Method (14/16) - Example (2/4)

$x_1 = L/B,\ x_2 = C_B$

- Iteration 1: Local Pattern Search 1

$\mathbf{b}^0 = (7,\ \ 0.2),\ \ \Delta x_1 = 0.5,\ \ \Delta x_2 = 0.1,$

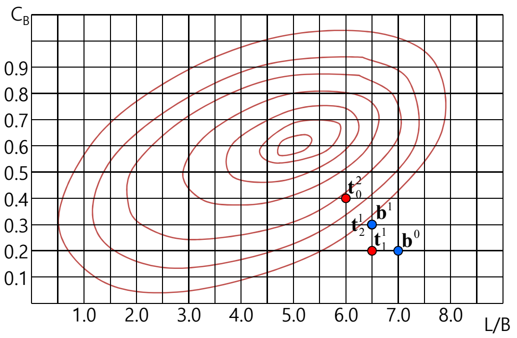
$\mathbf{t}_0^1 = \mathbf{b}^0$

Search from $\mathbf{t}_0^1$ in $-x_1$ direction $\rightarrow \mathbf{t}_1^1 = (6.5,\ 0.2)$

Search from $\mathbf{t}_1^1$ in $+x_2$ direction $\rightarrow \mathbf{t}_2^1 = (6.5,\ 0.3)$

Because the value of the objective function at $\mathbf{t}_2^1$ is improved, this point is adopted as a new base point.

$$\mathbf{b}^1 = \mathbf{t}_2^1$$



- Iteration 2: Global Pattern Move 1

Define the temporary base point by using $\mathbf{b}^0$ and $\mathbf{b}^1$

$\rightarrow \mathbf{t}_0^2 = (6,\ \ 0.4)$

Because the value of the objective function at $\mathbf{t}_0^2$ is improved, perform the 'Local Pattern Search' at this point.

Hooke & Jeeves Method (15/16) - Example (3/4)

- Iteration 3: Local Pattern Search 2

Search from $\mathbf{t}_0^2$ in $-x_1$ direction $\rightarrow \mathbf{t}_1^2 = (5.5,\ \ 0.4)$

Search from $\mathbf{t}_1^2$ in $+x_2$ direction $\rightarrow \mathbf{t}_2^2 = (5.5,\ \ 0.5)$

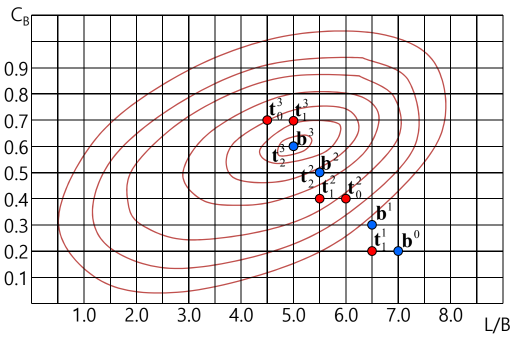
Because the value of the objective function at $\mathbf{t}_2^2$ is improved, this point is adopted as a new base point.

$$\mathbf{b}^2 = \mathbf{t}_2^2$$

- Iteration 4: Global Pattern Move 2

Define the temporary base point by using $\mathbf{b}^1$ and $\mathbf{b}^2$

$\rightarrow \mathbf{t}_0^3 = (4.5,\ \ 0.7)$



- Iteration 5: Local Pattern Search 3

Search from $\mathbf{t}_0^3$ in $+x_1$ direction $\rightarrow \mathbf{t}_1^0 = (5,\ \ 0.7)$

Search from $\mathbf{t}_1^3$ in $-x_2$ direction $\rightarrow \mathbf{t}_2^3 = (5,\ \ 0.6)$

Because the value of the objective function at $\mathbf{t}_2^3$ is improved, this point is adopted as a new base point.

$$\mathbf{b}^3 = \mathbf{t}_2^3$$

Hooke & Jeeves Method (16/16)
- Example (4/4)

- Iteration 6: Global Pattern Move 3

  Define the temporary base point by using $\mathbf{b}^2$ and $\mathbf{b}^3$

  $\rightarrow \mathbf{t}_0^4 = (4.5,\ \ 0.7)$

  Because the value of the objective function at $\mathbf{t}_0^4$ is not improved,

  $$\mathbf{t}_0^4 = \mathbf{b}^3$$

- Iteration 7: Local Pattern Search 4

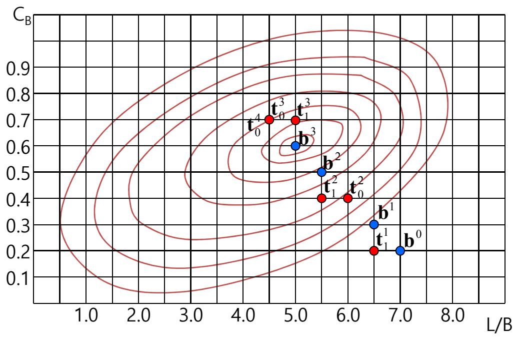
  Because there is no improvement of the value of the objective function from the temporary base design point $\mathbf{t}_0^4$ in x_1 direction and x_2 direction,

  $$\mathbf{t}_2^4 = \mathbf{t}_1^4 = \mathbf{t}_0^4$$

- Iteration 8: Global Pattern Move 4

  $\mathbf{b}^4 = \mathbf{b}^3 \rightarrow \ \ \Delta x_1 = 0.25,\ \ \Delta x_2 = 0.05,$

  $\mathbf{t}_0^5 = \mathbf{b}^4$



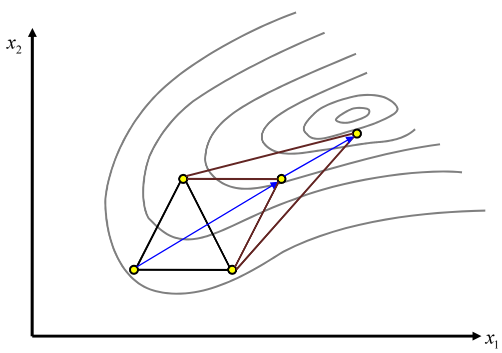
- Iteration 9: Stopping the Iteration of Search

  Because there is no improvement of the value of the objective function from base design point $(x_1, x_2) = (L/B,\ C_B) = (5.0,\ 0.6)$ in x_1 direction and x_2 direction by performing the 'Local Pattern Search' and 'Global Pattern Move', the optimal point is $L/B = 5.0,\ \ C_B = 0.6$.

3.2 Nelder & Mead Simplex Method

Nelder & Mead Simplex Method (1/14)

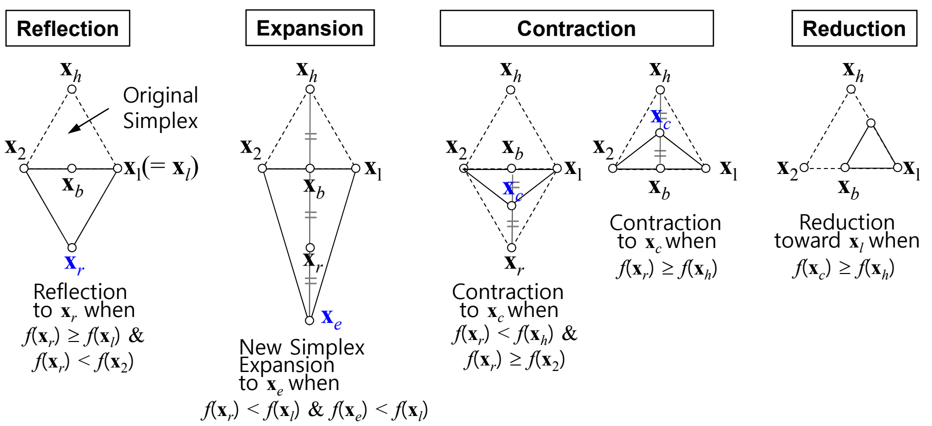
☑ **This method is used to find optimal point by successively reflecting, expanding, contracting, and reducing the simplex with (n+1) corners in the function of n design variables.**



1. This method uses n+1 points in the function of n design variables. Ex) If the number of the design variables is two, this method use three points, i.e., triangle.
2. The simplex is reflected in the direction where the value of the objective function is improved.
3. If the value of the objective function is improved, the simplex is expanded. Otherwise, the simplex is reduced.

Nelder & Mead Simplex Method (2/14)

☑ **The following figure shows various operations (Reflection, Expansion, Contraction, Reduction) for 2-dimensional case.**



$\mathbf{x}_h$: Simplex point having the largest value of the objective function
$\mathbf{x}_l$: Simplex point having the smallest value of the objective function
$\mathbf{x}_b$: Center point between $\mathbf{x}_1$ and $\mathbf{x}_2$

Nelder & Mead Simplex Method (3/14)

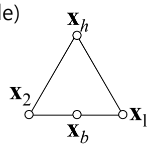
☑ **Step 1: Calculate the value of the objective function f at the $n+1$ corners of the simplex.**

☑ **Step 2: Establish the corners which yield the highest, $\mathbf{x}_h$, and lowest, $\mathbf{x}_l$, of $f(\mathbf{x})$ in the current simplex.**

☑ **Step 3: Calculate the value of the objective function f at the centroid ($\mathbf{x}_b$) of all $\mathbf{x}_i$ except $\mathbf{x}_h$, i.e.,**

$$\mathbf{x}_b = \frac{1}{n}\sum_{i=1}^{n+1}\mathbf{x}_i \text{(with } \mathbf{x}_h \text{ excluded)}$$

Example)

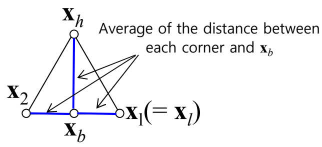


$$\mathbf{x}_b = \frac{\mathbf{x}_1 + \mathbf{x}_2}{2}$$

Nelder & Mead Simplex Method (4/14)

☑ **Step 4: Test stopping criterion:**

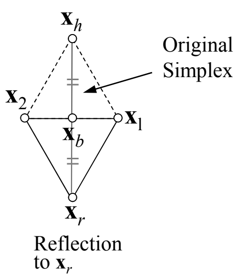
$$\left\{\frac{1}{n+1}\sum_{i=1}^{n+1}[f(\mathbf{x}_i)-f(\mathbf{x}_b)]^2\right\}^{1/2} \le \varepsilon$$



- **If the stopping criterion is satisfied, stop and return $f(\mathbf{x}_l)$ as minimum. Otherwise, continue.**

☑ **Step 5: Reflection**

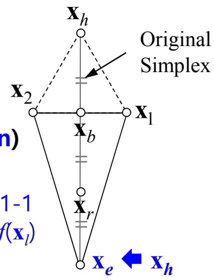
- **Reflect $\mathbf{x}_h$ through $\mathbf{x}_b$ to give $\mathbf{x}_r = 2\mathbf{x}_b - \mathbf{x}_h$. Calculate the value of the objective function f at $\mathbf{x}_r$ and change the simplex as following conditions.**



Nelder & Mead Simplex Method (5/14)

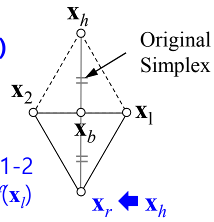
☑ Step 6: Expansion

- Step 6-1: If $f(\mathbf{x}_r) < f(\mathbf{x}_l)$, reflect $\mathbf{x}_b$ through $\mathbf{x}_r$ to give $\mathbf{x}_e = 2\mathbf{x}_r - \mathbf{x}_b$. And then, calculate $f(\mathbf{x}_e)$ and compare $f(\mathbf{x}_e)$ and $f(\mathbf{x}_l)$.
  - Step 6-1-1: If $f(\mathbf{x}_e) < f(\mathbf{x}_l)$, replace $\mathbf{x}_h$ by $\mathbf{x}_e$ (expansion) and return to Step 2.



➡ Step 6-1-1
$f(\mathbf{x}_e) < f(\mathbf{x}_l)$

  - Step 6-1-2: If $f(\mathbf{x}_e) \geq f(\mathbf{x}_l)$, replace $\mathbf{x}_h$ by $\mathbf{x}_r$ (reflection) and return to Step 2.

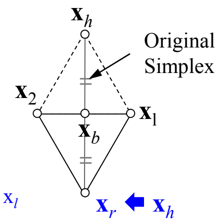


➡ Step 6-1-2
$f(\mathbf{x}_e) \geq f(\mathbf{x}_l)$

Nelder & Mead Simplex Method (6/14)

☑ Step 6: Expansion

- Step 6-2: If $f(\mathbf{x}_r) \geq f(\mathbf{x}_l)$,
  - Step 6-2-1: test $f(\mathbf{x}_r) < f(\mathbf{x}_i)$ for all $\mathbf{x}_i$ except $\mathbf{x}_l$. If true, replace $\mathbf{x}_h$ by $\mathbf{x}_r$ (reflection) and return to Step 2.



➡ Step 6-2-1
For all $\mathbf{x}_i$ except $\mathbf{x}_l$
$f(\mathbf{x}_r) < f(\mathbf{x}_i)$

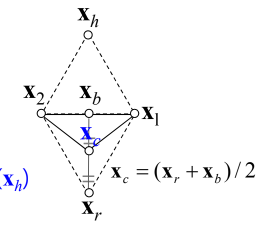
  - Step 6-2-2: If false, continue.

Nelder & Mead Simplex Method (7/14)

☑ **Step 7: Contraction**

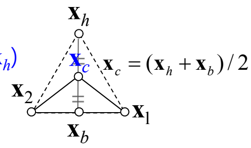
- **Step 7-1: If $f(\mathbf{x}_r) < f(\mathbf{x}_h)$,**
  calculate the value of the objective function f
  at $\mathbf{x}_c = (\mathbf{x}_r + \mathbf{x}_b)/2$.

➡ Step 7-1
$f(\mathbf{x}_r) < f(\mathbf{x}_h)$



- **Step 7-2: If $f(\mathbf{x}_r) \geq f(\mathbf{x}_h)$,**
  calculate the value of the objective function f
  at $\mathbf{x}_c = (\mathbf{x}_h + \mathbf{x}_b)/2$.

➡ Step 7-2
$f(\mathbf{x}_r) \geq f(\mathbf{x}_h)$

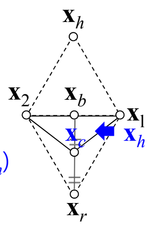


Nelder & Mead Simplex Method (8/14)

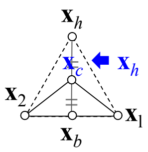
☑ **Step 8: Reduction**

- **Step 8-1: If $f(\mathbf{x}_c) < f(\mathbf{x}_h)$,**
  replace $\mathbf{x}_h$ by $\mathbf{x}_c$ (contraction)
  and return to Step 2.

➡ Step 8-1
$f(\mathbf{x}_c) < f(\mathbf{x}_h)$

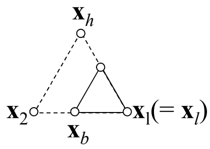


or



- **Step 8-2: If $f(\mathbf{x}_c) \geq f(\mathbf{x}_h)$,**
  reduce the simplex toward $\mathbf{x}_l$ using $\mathbf{x}_i = (\mathbf{x}_i + \mathbf{x}_l)/2$
  (reduction) and return to Step 2.

➡ Step 8-2
$f(\mathbf{x}_c) \geq f(\mathbf{x}_h)$



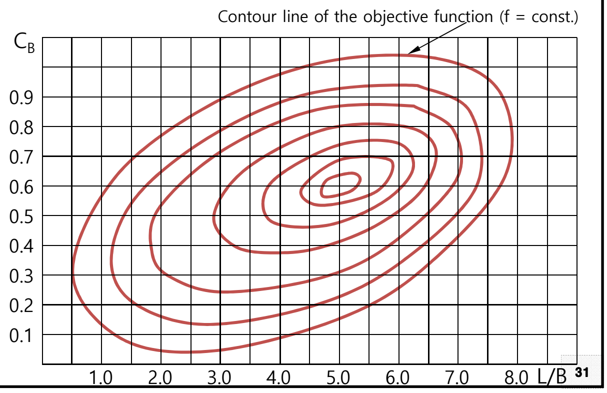
Reduction toward $\mathbf{x}_1$

Nelder & Mead Simplex Method (9/14) - Example (1/6)

☑ **If the contour line of the objective function of shipbuilding cost with two design variables, L/B and C_B, is given as shown in Fig, find the value of the L/B and C_B to minimize the shipbuilding cost by using the 'Nelder & Mead Simplex Method' and plot the procedures in the graph.**

- **Nelder & Mead Simplex Method**
  - **Starting corners of the simplex: (L/B, CB) = (7, 0.1), (7.5, 0.1), (7.5, 0.2)**
  - **Stopping criterion: 0.01**

Optimization problem with two unknown variables



Nelder & Mead Simplex Method (10/14) - Example (2/6)

$$x_1 = L/B,\ x_2 = C_B$$

Triangle 1: $x_1,\ x_2,\ x_3$

Iteration 1: Because x_2 is x_h, reflect x_2 through the center between x_1 and x_3. $\rightarrow x_r$

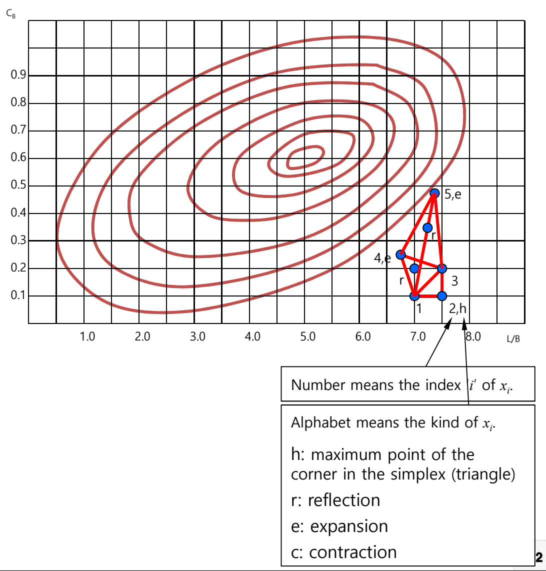
Because $f(x_r) < f(x_1)$ and $f(x_3)$, perform the expansion. $\rightarrow x_{4,e}$

$\rightarrow$ Triangle 2: $x_1,\ x_3,\ x_4$

Iteration 2: Because x_1 is x_h, reflect x_1 through the center between x_3 and x_4. $\rightarrow x_r$

Because $f(x_r) < f(x_3)$ and $f(x_4)$, perform the expansion. $\rightarrow x_{5,e}$

$\rightarrow$ Triangle 3: $x_3,\ x_4,\ x_5$

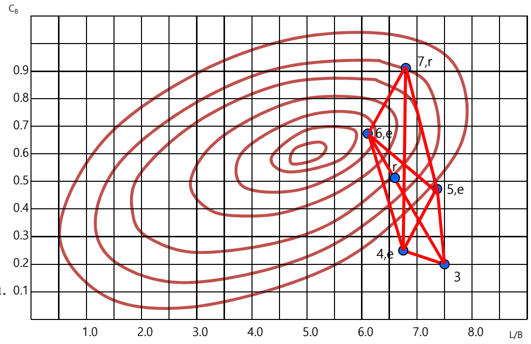


Nelder & Mead Simplex Method (11/14) - Example (3/6)

$x_1 = L/B,\ x_2 = C_B$

Iteration 3: Because x_3 is x_h, reflect x_3 through the center between x_4 and x_5. $\rightarrow x_r$
Because $f(x_r) < f(x_4)$ and $f(x_5)$, perform the expansion. $\rightarrow x_{6,e}$
$\rightarrow$ Triangle 4: $x_4,\ x_5,\ x_6$

Iteration 4: Because x_4 is x_h, reflect x_4 through the center between x_5 and x_6. $\rightarrow x_{7,r}$
Because $f(x_{7,r}) > f(x_6)$, go to the next iteration.
$\rightarrow$ Triangle 5: $x_5,\ x_6,\ x_7$

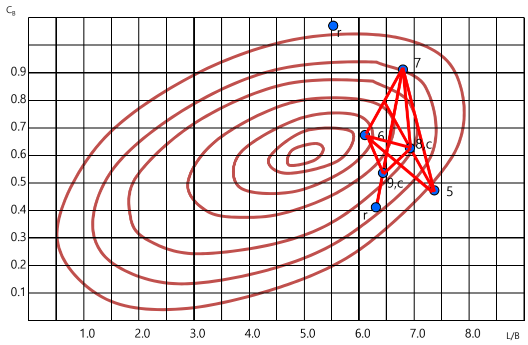


Nelder & Mead Simplex Method (12/14) - Example (4/6)

$x_1 = L/B,\ x_2 = C_B$

Iteration 5: Because x_5 is x_h, reflect x_5 through the center between x_6 and x_7. $\rightarrow x_r$
Because $f(x_r) > f(x_5),\ f(x_6)$, and $f(x_7)$, perform the constraction. $\rightarrow x_{8,c}$
$\rightarrow$ Triangle 6: $x_6,\ x_7,\ x_8$

Iteration 6: Because x_7 is x_h, reflect x_7 through the center between x_6 and x_8. $\rightarrow x_r$
Because $f(x_r) > f(x_6)$ and $f(x_8)$,
and $f(x_r) < f(x_7)$,
contract the simplex toward x_r. $\rightarrow x_{9,c}$
$\rightarrow$ Triangle 7: $x_6,\ x_8,\ x_9$

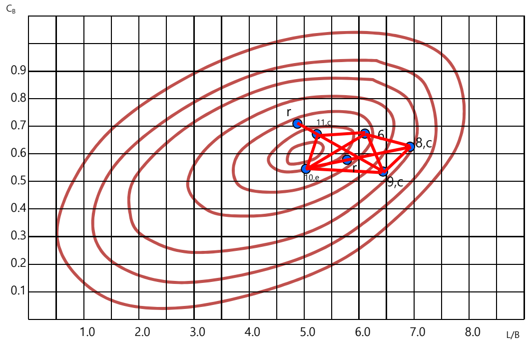


Nelder & Mead Simplex Method (13/14) - Example (5/6)

$x_1 = L/B,\ x_2 = C_B$

Iteration 7: Because x_8 is x_h, reflect x_8 through the center between x_6 and x_9. $\rightarrow x_r$
Because $f(x_r) < f(x_6)$ and $f(x_9)$, preforme the expansion. $\rightarrow$ $x_{10,c}$
$\rightarrow$ Triangle 8: $x_6,\ x_9,\ x_{10}$

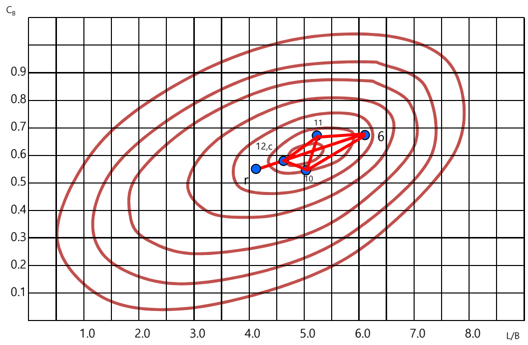
Iteration 8: Because $x_{9,c}$ is x_h, reflect $x_{9,c}$ through the center between x_6 and x_{10}. $\rightarrow x_r$
Because $f(x_r) > f(x_6)$ and $f(x_{10})$, and $f(x_r) < f(x_9)$, contract the simplex toward x_r. $\rightarrow$ $x_{11,c}$
$\rightarrow$ Triangle 9: $x_6,\ x_{10},\ x_{11}$



Nelder & Mead Simplex Method (14/14) - Example (6/6)

Iteration 9: Because x_6 is x_h, reflect x_6 through the center between x_{10} and x_{11}. $\rightarrow x_r$
Because $f(x_r) > f(x_{10})$ and $f(x_{11})$, and $f(x_r) < f(x_6)$, contract the simplex toward x_r. $\rightarrow$ $x_{12,c}$
$\rightarrow$ Triangle 10: $x_{10},\ x_{11},\ x_{12}$

$x_1(7,\ 0.1)$	$x_2(7.5,\ 0.1)$
$x_3(7.5,\ 0.2)$	$x_4(6.75,\ 0.25)$
$x_5(7.375,\ 0.475)$	$x_6(6.1875,\ 0.6875)$
$x_7(6.8125,\ 0.9125)$	$x_8(6.9375,\ 0.6375)$
$x_9(6.4375,\ 0.5375)$	$x_{10}(5.0625,\ 0.5625)$
$x_{11}(5.21875,\ 0.66875)$	$x_{12}(4.6171875,\ 0.5796875)$



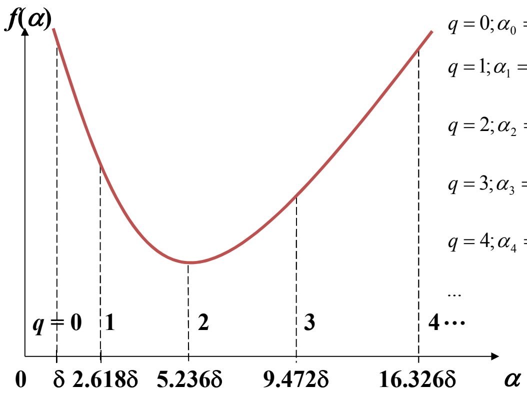
Performing 10 times iterations, we can recognize that the simplex (triangle) has the tendency to approach the result obtained by the 'Hooke & Jeeves method'.

3.3 Golden Section Search Method (One Dimensional Search Method)

Phase 1: Global Search (1/2)

☑ Searching for the interval in which the minimum lies

- In the figure, starting at $q = 0$, we evaluate $f(\alpha)$ at $\alpha = \delta$, where $\delta > 0$ is a small number. If the value $f(\delta)$ is smaller than the value $f(0)$, we then take an increment of 1.618δ in the step size (i.e., the increment is 1.618 times the previous increment δ). (See Fibonacci sequence)



$q = 0; \alpha_0 = \delta$

$q = 1; \alpha_1 = \delta + 1.618\delta = 2.618\delta = \sum_{j=0}^{1} \delta(1.618)^j$

$q = 2; \alpha_2 = 2.618\delta + 1.618(1.618\delta) = 5.236\delta = \sum_{j=0}^{2} \delta(1.618)^j$

$q = 3; \alpha_3 = 5.236\delta + (1.618)^3\delta = 9.472\delta = \sum_{j=0}^{3} \delta(1.618)^j$

$q = 4; \alpha_4 = 9.472\delta + (1.618)^4\delta = 16.326\delta = \sum_{j=0}^{4} \delta(1.618)^j$

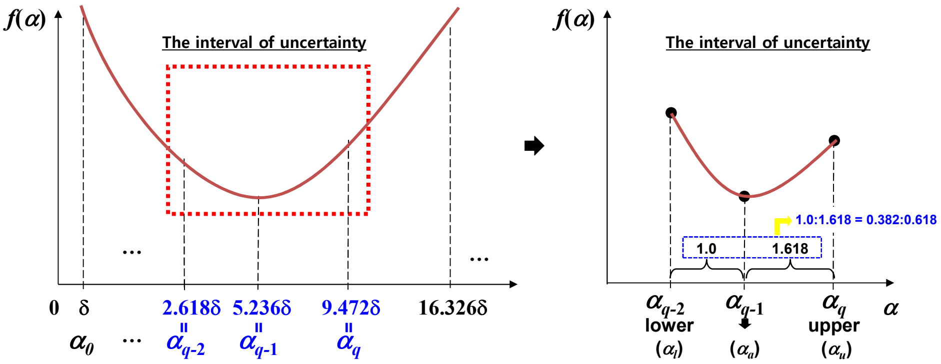
…

$$\boxed{\therefore \alpha_q = \sum_{j=0}^{q} \delta(1.618)^j, \ q = 0, 1, 2, \ldots}$$

Phase 1: Global Search (2/2)

- If the function at α_{q-1} is smaller than that at the previous point α_{q-2} and the next point α_q (i.e., $f(\alpha_{q-1}) < f(\alpha_{q-2}),\ f(\alpha_{q-1}) < f(\alpha_q)$), the minimum point lies between α_q and α_{q-2}.

(The interval in which the minimum lies is called the interval of uncertainty.)

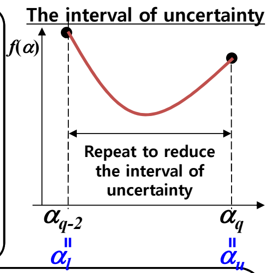


- Therefore, upper and lower limits on the interval of uncertainty are

$$\underset{\text{upper}}{\alpha_u} \equiv \alpha_q = \sum_{j=0}^{q}\delta(1.618)^j,\ \underset{\text{lower}}{\alpha_l} \equiv \alpha_{q-2} = \sum_{j=0}^{q-2}\delta(1.618)^j\quad ,\alpha_a \equiv \alpha_{q-1} = \sum_{j=0}^{q-1}\delta(1.618)^j$$

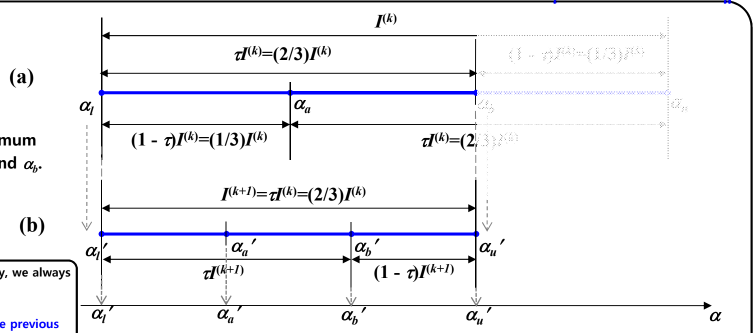
Phase 2: Local Search (1/3)

- **Reduction of the interval of uncertainty** by comparing function values at α_a and α_b
  - We consider two points symmetrically located from either end as shown in the figure – points α_a and α_b are located at a distance of $\tau I^{(k)}$ from either end of the interval.
  - Comparing function values at α_a and α_b, either the left (α_l, α_a) or the right (α_b, α_u) portion of the interval gets discarded because the minimum cannot lie there.



< If τ = 2/3 >

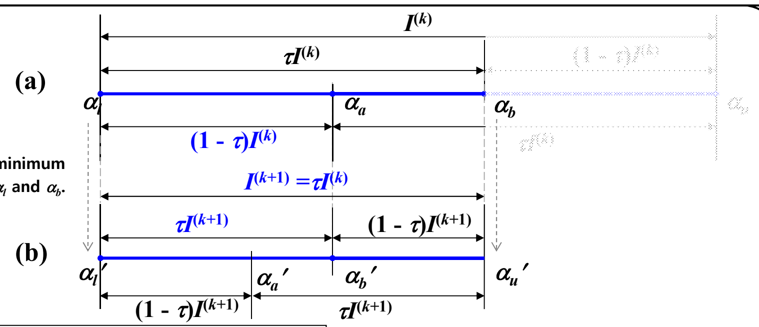
If $f(\alpha_a) < f(\alpha_b)$, then minimum point lies between α_l and α_b.



For new interval of uncertainty, we always have to compute $f(\alpha_a')$, $f(\alpha_b')$.
<Question>
Is there any method to use the previous function values?

Phase 2: Local Search (2/3)

- Reduction of the interval of uncertainty by comparing function values at α_a and α_b
  - We consider two points symmetrically located from either end as shown in the figure – points α_a and α_b are located at a distance of $\tau I^{(k)}$ from either end of the interval.



If $f(\alpha_a) < f(\alpha_b)$, then minimum point lies between α_l and α_b.

1. $f(\alpha_a)$ will be used for the next interval of uncertainty $I^{(k+1)}$.
2. α_a can be equal to α_a' or α_b' of the next interval of uncertainty $I^{(k+1)}$.

3-1. Assume that α_a is equal to α_a'.

$$\alpha_a = \alpha_a'$$
$$(1-\tau)I^{(k)} = (1-\tau)I^{(k+1)}$$
$$(1-\tau)I^{(k)} = (1-\tau)\tau I^{(k)}$$
$$I^{(k)} = \tau I^{(k)}$$

Because $\tau = 1$, this assumption is wrong.

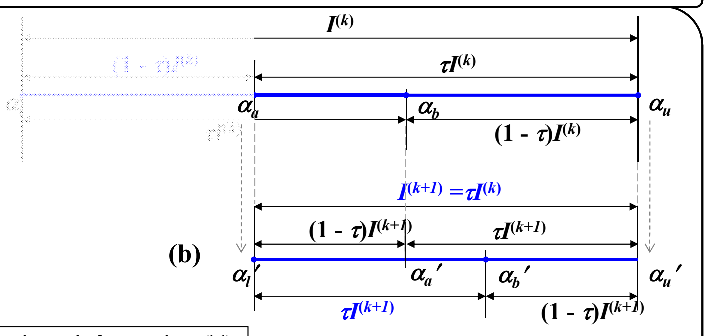
3-2. Assume that α_a is equal to α_b'.

$$\alpha_a = \alpha_b'$$
$$(1-\tau)I^{(k)} = \tau I^{(k+1)}$$
$$(1-\tau)I^{(k)} = \tau \cdot \tau \cdot I^{(k)}$$
$$\tau \cdot \tau I^{(k)} - (1-\tau)I^{(k)} = 0$$
$$\tau^2 + \tau - 1 = 0$$

$\Rightarrow \tau = 0.618, -1.618 \Rightarrow 0.618$

Phase 2: Local Search (3/3)

- Reduction of the interval of uncertainty by comparing function values at α_a and α_b
  - We consider two points symmetrically located from either end as shown in the figure – points α_a and α_b are located at a distance of $\tau I^{(k)}$ from either end of the interval.



If $f(\alpha_a) > f(\alpha_b)$, then minimum point lies between α_a and α_u.

1. $f(\alpha_b)$ will be used for the next interval of uncertainty $I^{(k+1)}$.
2. α_b can be equal to α_a' or α_b' of the next interval of uncertainty $I^{(k+1)}$.

3-1. Assume that α_b is equal to α_b'.

$$\alpha_b = \alpha_b'$$
$$(1-\tau)I^{(k)} = (1-\tau)I^{(k+1)}$$
$$(1-\tau)I^{(k)} = (1-\tau)\tau I^{(k)}$$
$$I^{(k)} = \tau I^{(k)}$$

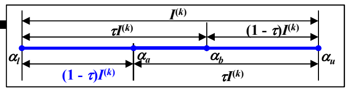
Because $\tau = 1$, this assumption is wrong.

3-2. Assume that α_b is equal to α_a'.

$$\alpha_b = \alpha_a'$$
$$(1-\tau)I^{(k)} = \tau I^{(k+1)}$$
$$(1-\tau)I^{(k)} = \tau \cdot \tau \cdot I^{(k)}$$
$$\tau \cdot \tau I^{(k)} - (1-\tau)I^{(k)} = 0$$
$$\tau^2 + \tau - 1 = 0$$

$\Rightarrow \tau = 0.618, -1.618 \Rightarrow 0.618$

Summary (1/3)



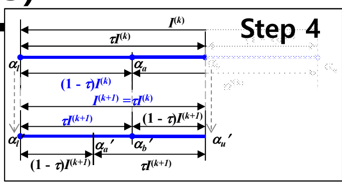
- Step 1: For a chosen small number δ, let q be the smallest integer to satisfy $f(\alpha_{q-1}) < f(\alpha_{q-2})$ and $f(\alpha_{q-1}) < f(\alpha_q)$, where α_q, α_{q-1}, and α_{q-2} are calculated from $\alpha_q = \sum_{j=0}^{q} \delta(1.618)^j, \ (q = 0,1,2,\ldots)$. The upper and lower bounds on α^* (the optimum value for α) are given as follows.

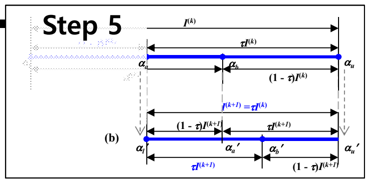
$$\alpha_u \equiv \alpha_q = \sum_{j=0}^{q} \delta(1.618)^j, \alpha_l \equiv \alpha_{q-2} = \sum_{j=0}^{q-2} \delta(1.618)^j$$

- Step 2: Compute $f(\alpha_a)$ and $f(\alpha_b)$ where $\alpha_a = \alpha_l + 0.382I$ and $\alpha_b = \alpha_l + 0.618I$ (interval of uncertainty $I = \alpha_u - \alpha_l$).

- Step 3: Compare $f(\alpha_a)$ and $f(\alpha_b)$, and go to Step 4, Step 5 or Step 6.

Summary (2/3)



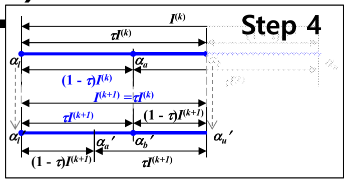


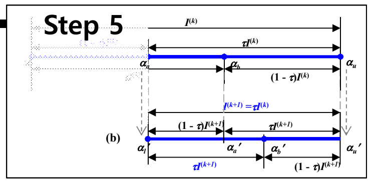
- Step 4: If $f(\alpha_a) < f(\alpha_b)$, then minimum point α^* lies between α_l and α_b, i.e., $\alpha_l \le \alpha^* \le \alpha_b$. The new limits for the reduced interval of uncertainty are $\alpha_l' = \alpha_l$ and $\alpha_u' = \alpha_b$. Also, $\alpha_b' = \alpha_a$. Compute $f(\alpha_a')$, where $\alpha_a' = \alpha_l' + 0.382(\alpha_u' - \alpha_l')$ and go to Step 7.

- Step 5: If $f(\alpha_a) > f(\alpha_b)$, then minimum point α^* lies between α_a and α_u, i.e., $\alpha_a \le \alpha^* \le \alpha_u$. Similar to the procedure in Step 4, let $\alpha_l' = \alpha_a$ and $\alpha_u' = \alpha_u$, so that $\alpha_a' = \alpha_b$. Compute $f(\alpha_b')$, where $\alpha_b' = \alpha_l' + 0.618(\alpha_u' - \alpha_l')$ and go to Step 7.

- Step 6: If $f(\alpha_a) = f(\alpha_b)$, let $\alpha_l = \alpha_a$ and $\alpha_u = \alpha_b$, and return to Step 2.

Summary (3/3)





- Step 7: If the new interval of uncertainty $I' = \alpha_u' - \alpha_l'$ is small enough to satisfy a stopping criterion (i.e., $I' < \varepsilon$), let $\alpha^* = (\alpha_u' - \alpha_l') / 2$ and stop. Otherwise, delete the primes (') on α_l', α_a', α_b' and α_u' and return to Step 3.