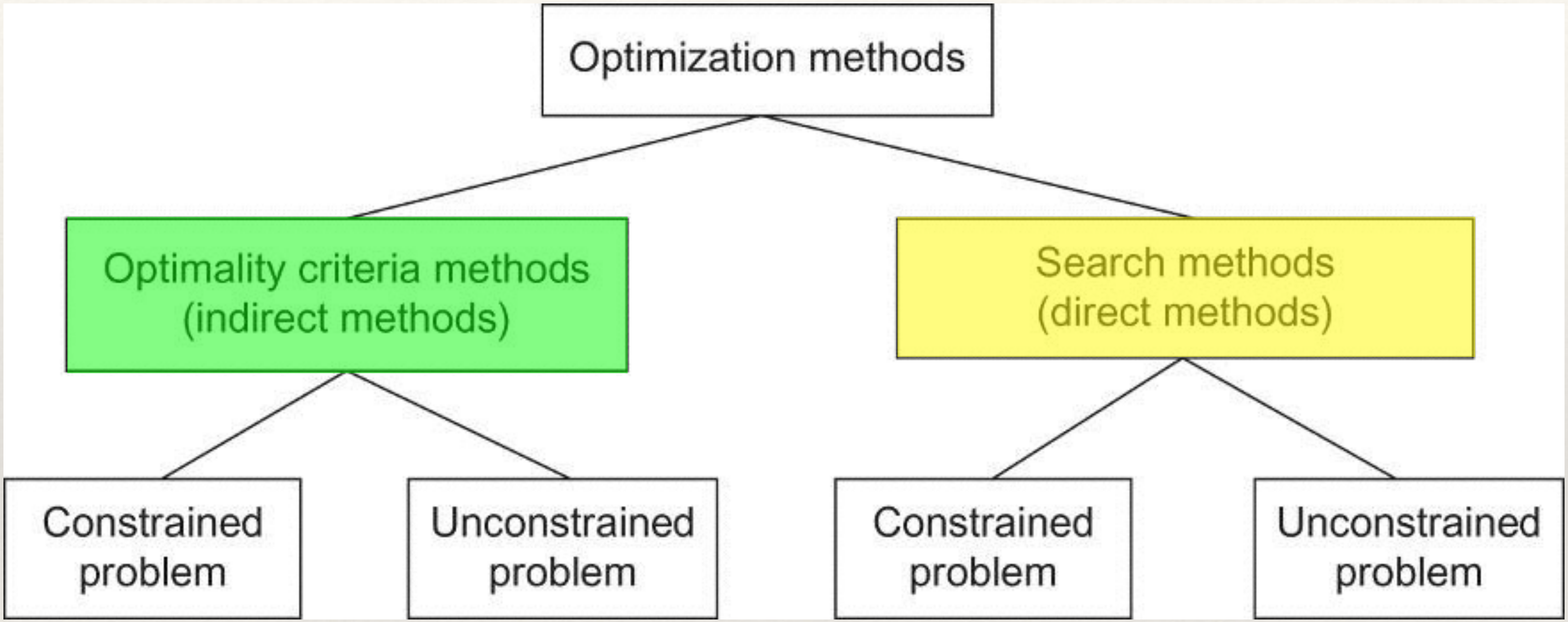
Chi-Lun Lin @ ME. NCKU

Optimum Design

Lecture 12
Numerical Methods for Unconstrained Problems

Spring 2016


Optimization methods
Optimality criteria methods
(indirect methods)
Search methods
(direct methods)
Constrained
problem
Unconstrained
problem
Constrained
problem
Unconstrained
problem

Numerical Methods

for non-linear programming problems

Gradient-Based Search

50s

1. Based on gradient of problem functions
2. Twice continuously differentiable

Direct Search

60s & 70s

1. Do not calculate derivatives
2. Simple and easy to use

Nature-Inspired Search

80s~

1. Use only function values
2. More general

Gradient-Based Search

- Iterative
- Estimate initial design and improve it until optimality conditions are satisfied

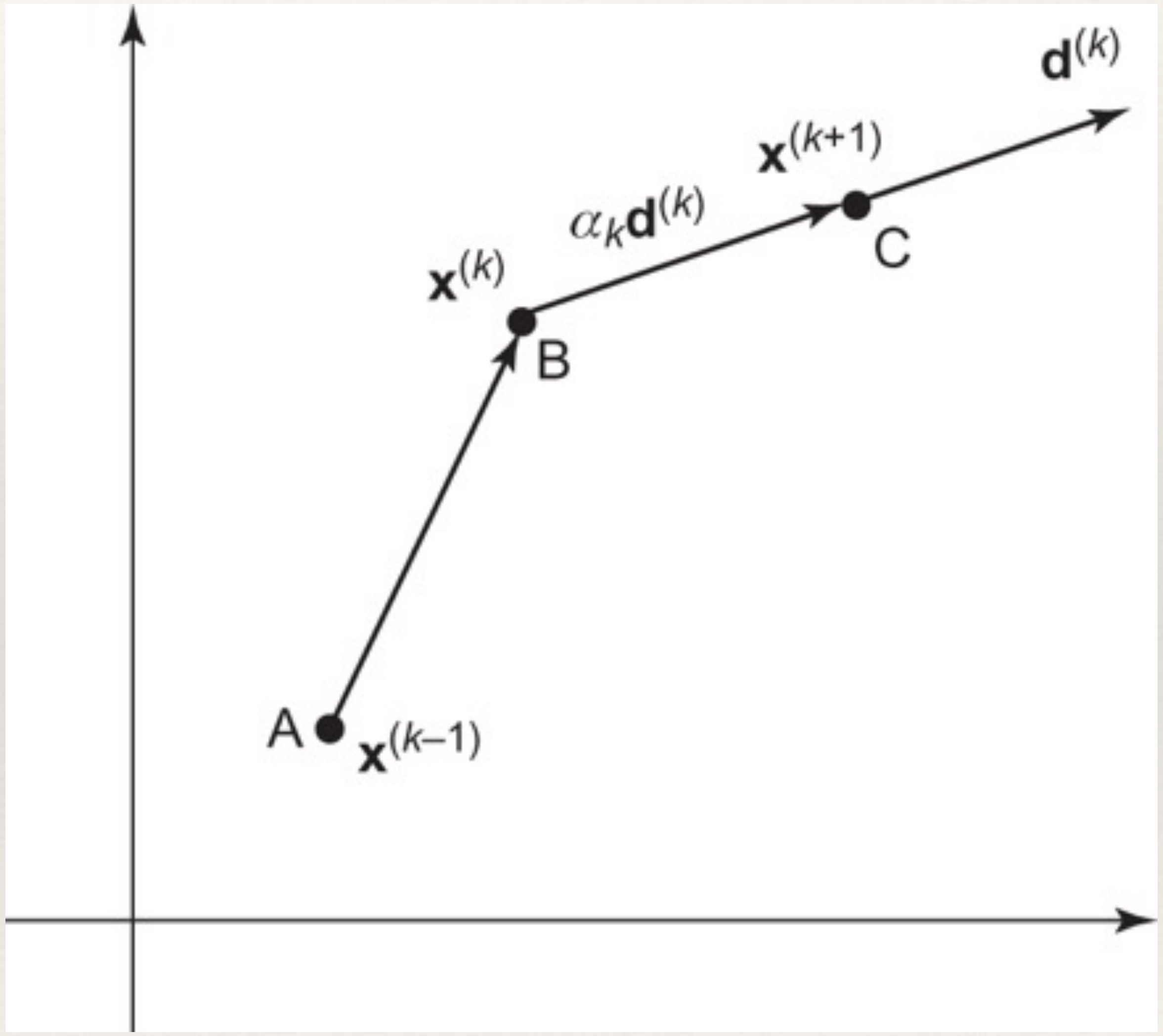
General Algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k)}; \quad k = 0,\ 1,\ 2, \ldots$$

$$\Delta\mathbf{x}^{(k)} = \underbrace{\alpha_k}_{\text{Step size}} \underbrace{\mathbf{d}^{(k)}}_{\text{desirable search direction}}$$

1. Estimate $\mathbf{x}^{(0)}$. Set $k = 0$.
2. Compute a search direction $\mathbf{d}^{(k)}$.
3. If it has converged, stop; otherwise, continue.
4. Calculate a positive step size α_k.
5. Update the design and set $k = k + 1$ and go to step 2.

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$$



Descent Condition

$$\mathbf{c}^{(k)} \cdot \mathbf{d}^{(k)} < 0$$

where $\mathbf{c}^{(k)} = \nabla f\left(\mathbf{x}^{(k)}\right)$

Step Size Determination - Analytical Approach

1. Reduce f($\mathbf{x}$) to a one variable function
2. Perform one-dimensional minimization

Step Size Determination

$$\nabla f(\mathbf{x}^{(k+1)}) \cdot \mathbf{d}^{(k)} = 0$$

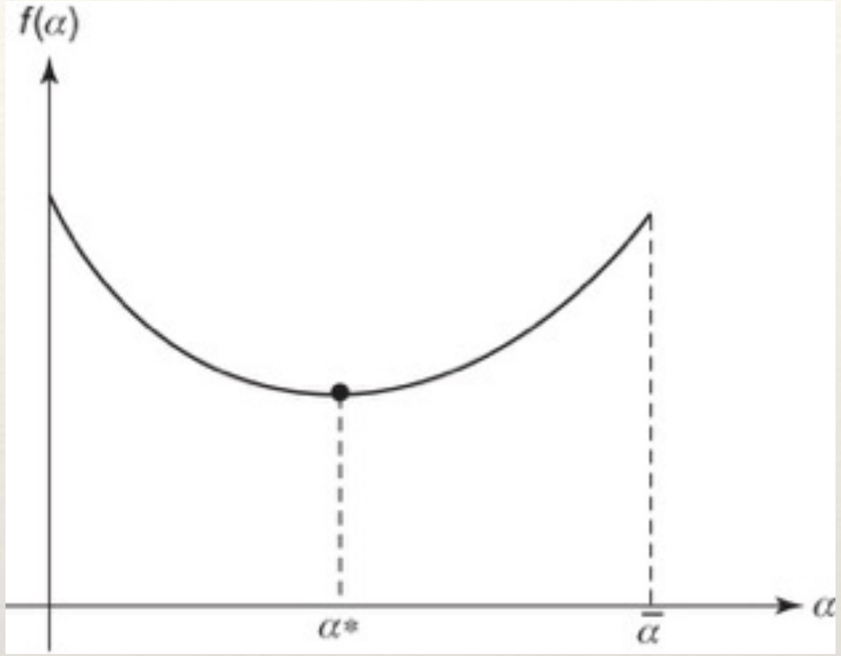
Note: f(α) is a single-variable function

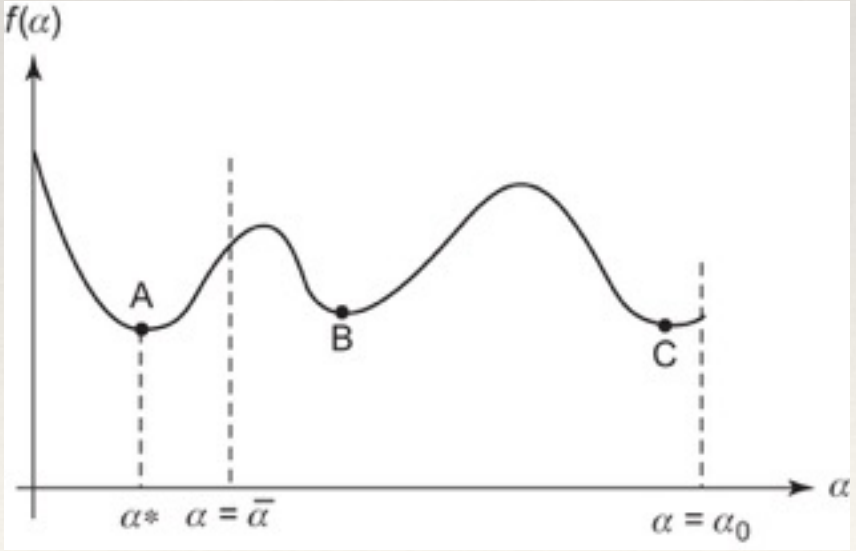
Step Size Determination - Numerical Approach

- For many problems:
  - it is not possible to obtain an explicit expression for $f(\alpha)$
  - or it is too complicated to solve for α

Unimodal Function

- A function decreases continuously until the minimum point is reached
- For functions that are not unimodal, we can locate only a local minimum point closest to the starting point.





Equal-Interval Search

Phase I: Initial Bracketing of Minimum

Check for:

$$f(q\delta) < f((q+1)\delta)$$

Then,

$$\alpha_l = (q-1)\delta, \; \alpha_u = (q+1)\delta$$

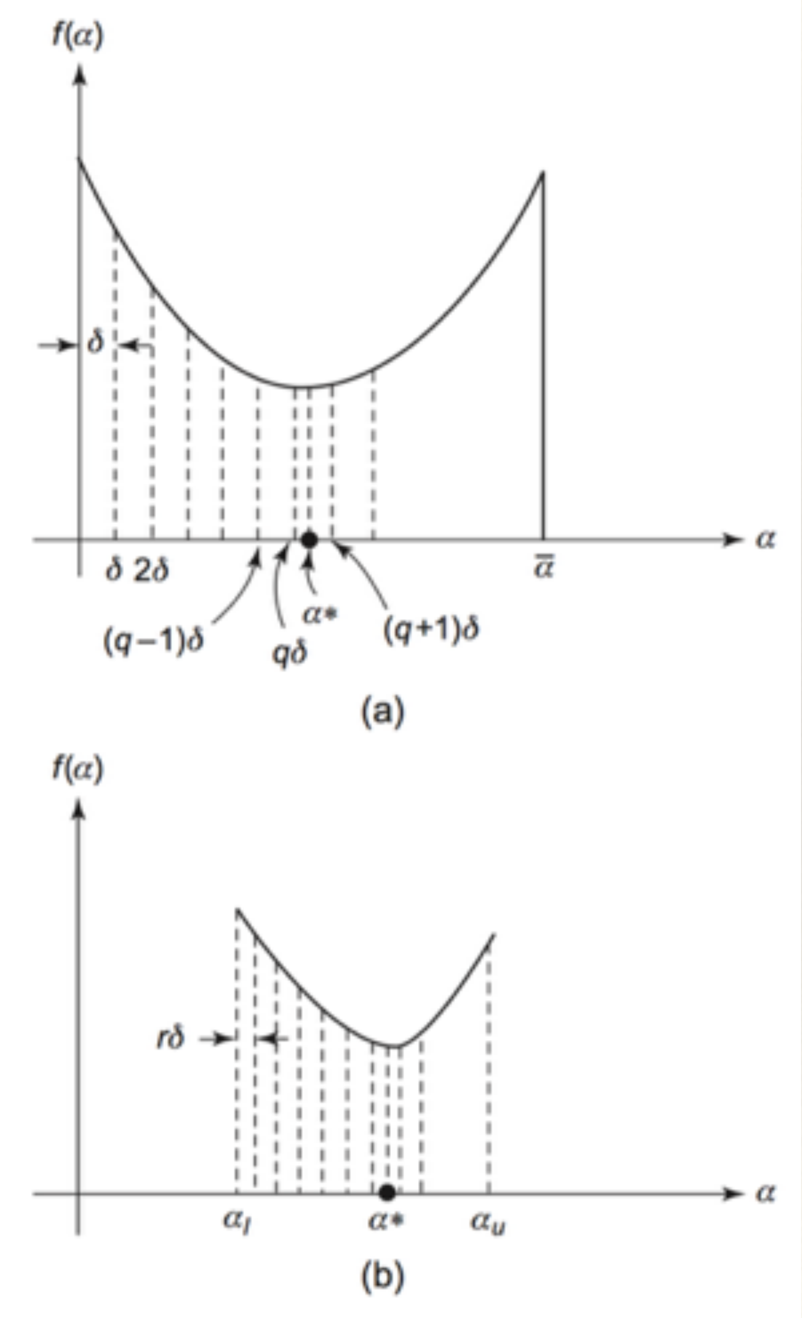
$$I = \alpha_u - \alpha_l = 2\delta$$

Phase II: Reducing the Interval

Start from α_l with a smaller increment: $r\delta, \; r \ll 1$

Obtain a new bracket: $2r\delta$

Repeat until the interval is less than a tolerance: ϵ



Alternate Equal-Interval Search

Phase I: Initial Bracketing of Minimum

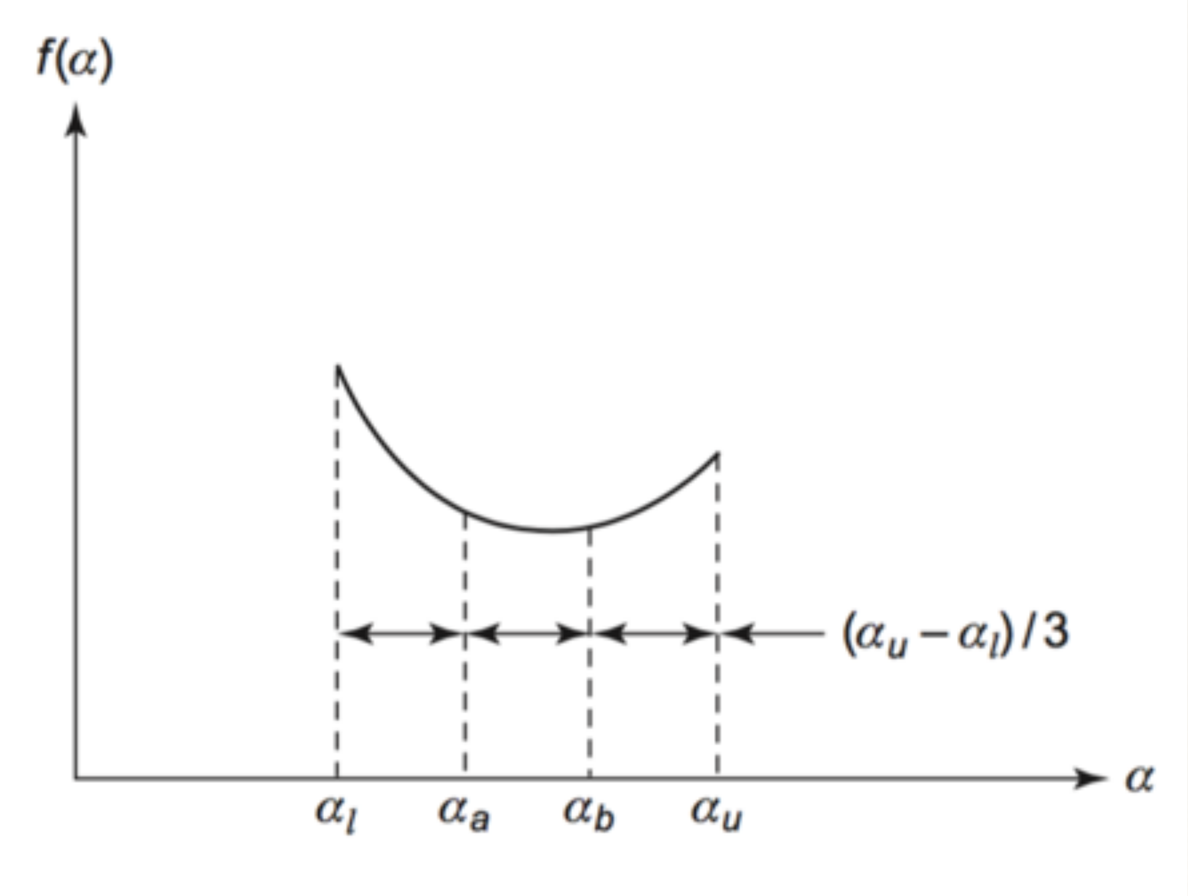
$$\alpha_l = (q-1)\delta,\ \alpha_u = (q+1)\delta$$

$$I = \alpha_u - \alpha_l = 2\delta$$

Phase II: Reducing the Interval

$$\alpha_a = \alpha_l + \frac{1}{3}I$$

$$\alpha_b = \alpha_l + \frac{2}{3}I = \alpha_u - \frac{1}{3}I$$



Reading

- Chapter 10.1 - 10.4