Computer Science

Jerzy Świątek

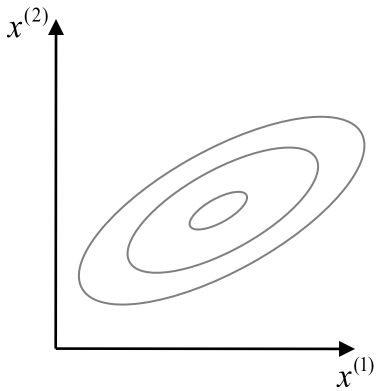
Systems Modelling and Analysis

Choose yourself and new technologies

L.17.a Numerical optimization methods – line search

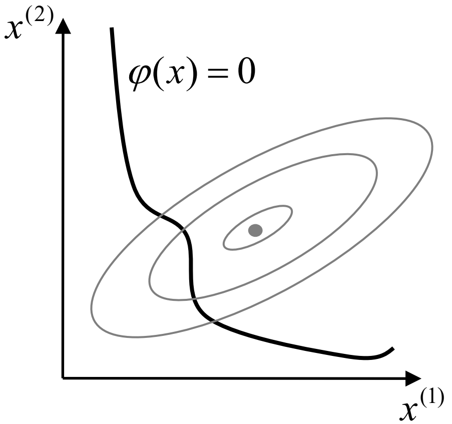
HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

General classification of optimization tasks



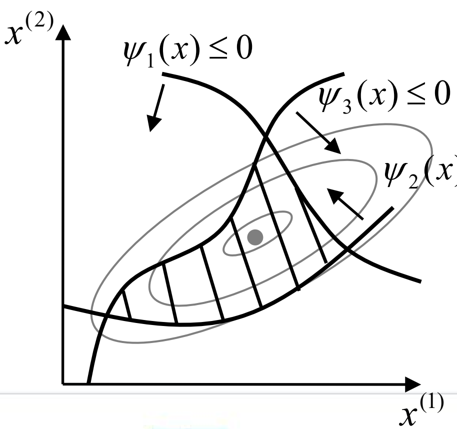
Unconstrained optimization:

$$\mathscr{D}_x = \mathscr{R}^S$$



Optimization under equality constraints:

$$\mathscr{D}_x = \left\{ x \in \mathscr{R}^S : \varphi_1(x) = 0, \varphi_2(x) = 0, \ldots, \varphi_L(x) = 0, L \le S \right\}$$



Optimization under inequality constraints:

$$\mathscr{D}_x = \left\{ x \in \mathscr{R}^S : \psi_1(x) \le 0, \psi_2(x) \le 0, \ldots, \psi_M(x) \le 0 \right\}$$

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Analytical methods

- Unconstrained optimization
- Lagrange multipliers method – equality constraints
- Kuhn-Tucker conditions – inequality constraints

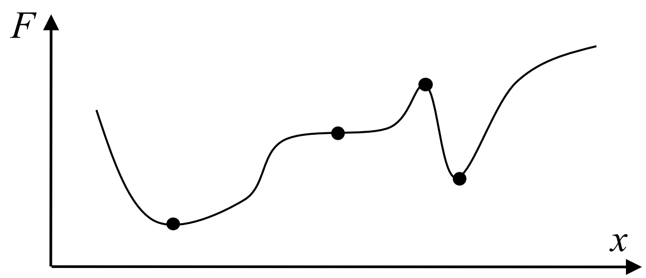
Unconstrained optimization

Optimization task: $x^* \rightarrow F(x^*) = \min_{x^* \in \mathscr{D}_x} F(x)$

Assumption: $F(x)$ is continuous and differentiable.

Necessary condition for x^* to be local minima: $\nabla_x F(x^*) = 0_S$

If $F(x)$ is convex function, then above equation is sufficient condition for x^* to be global minima.



Optimization under equality constraints

- The method of Lagrange multipliers

Lagrange function:

$$L(x,\lambda) = F(x) + \sum_{l=1}^{L} \lambda_l \varphi_l(x) = F(x) + \lambda^T \varphi(x)$$

$$\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_L \end{bmatrix}, \quad \varphi(x) = \begin{bmatrix} \varphi_1(x) \\ \varphi_2(x) \\ \vdots \\ \varphi_L(x) \end{bmatrix}$$

Necessary conditions of optimality:

$$\nabla_x L(x,\lambda)\Big|_{x^*,\lambda^*} = 0_S$$

$$\nabla_\lambda L(x,\lambda)\Big|_{x^*,\lambda^*} = 0_L$$

If and only if $\operatorname{rank} G(x) = \operatorname{rank} \left[G(x) \;\vdots\; -\nabla_x F(x) \right]$,

Where: $G(x) = \left[\nabla_x \varphi_1(x) \;\vdots\; \nabla_x \varphi_2(x) \;\vdots\; \cdots \;\vdots\; \nabla_x \varphi_L(x) \right]$

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Optimization under inequality constraints

Kuhn-Tucker conditions

Lagrange function:

$$L(x,\mu)=F(x)+\mu^T\psi(x) \quad \Leftrightarrow \quad L(x,\mu)=F(x)+\sum_{m=1}^{M}\mu_m\psi_m(x)$$

$$\mu=\begin{bmatrix}\mu_1\\ \mu_2\\ \vdots\\ \mu_M\end{bmatrix}$$

Necessary conditions of optimality:

$$\nabla_x L(x,\mu)\Big|_{x^*,\mu^*}=0_S$$

$$\mu^T\nabla_\mu L(x,\mu)\Big|_{x^*,\mu^*}=0$$

$$\nabla_\mu L(x,\mu)\Big|_{x^*,\mu^*}\le 0_M$$

$$\mu^*\ge 0_M$$

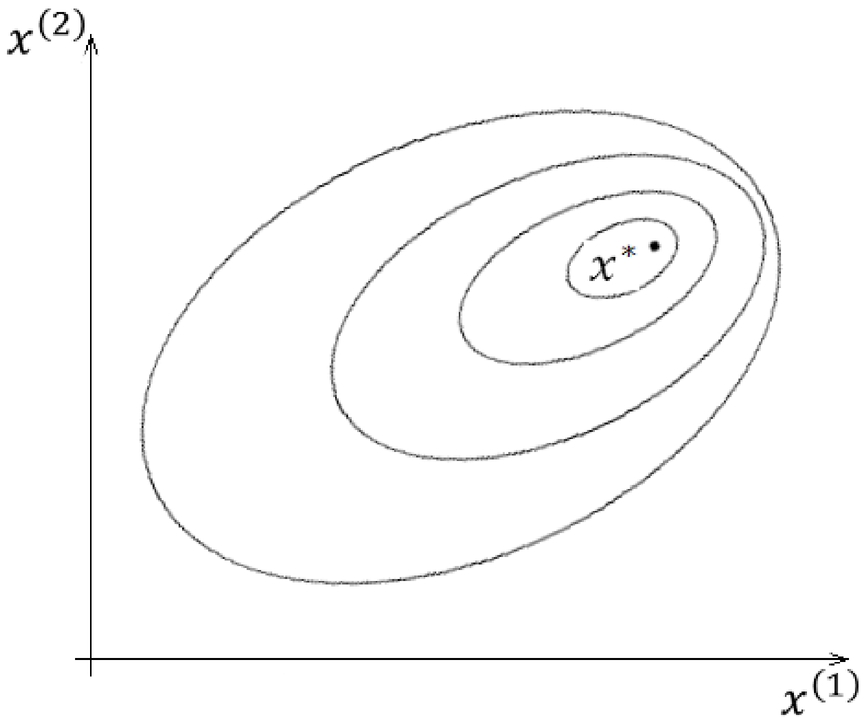
If solution is regular

$$\alpha=\begin{bmatrix}\alpha_1\\ \alpha_2\\ \vdots\\ \alpha_S\end{bmatrix} \quad \beta=\begin{bmatrix}\beta_1\\ \beta_2\\ \vdots\\ \beta_S\end{bmatrix}$$

$$\alpha\le\beta \Rightarrow \forall_{s=1,\dots,S}\,\alpha_s\le\beta_s$$

Numerical optimization methods

$$x^* \rightarrow F(x^*) = \min_{x \in D_x} F(x)$$

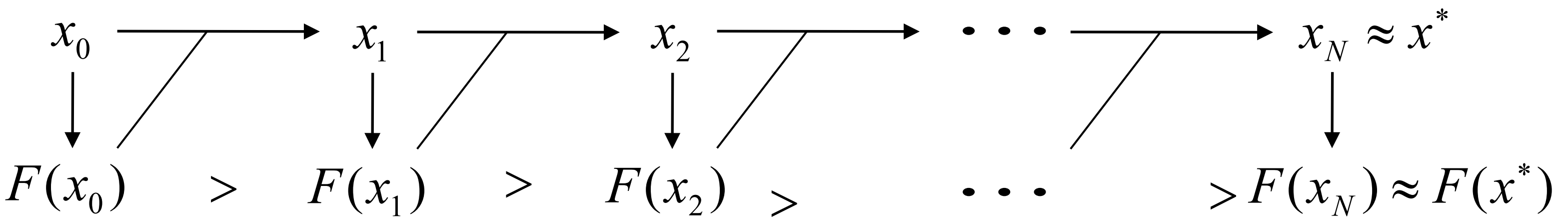


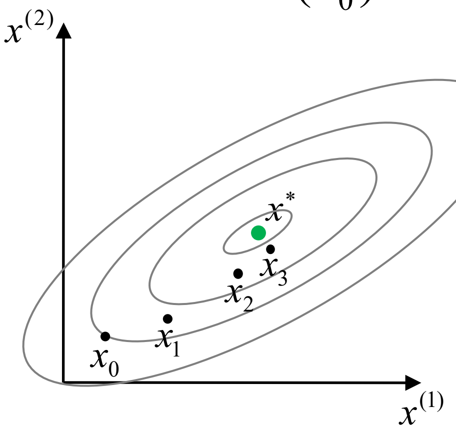
Analytical methods has drawbacks, when:

1. The goal function F and constraints φ, ψ are nonlinear.
2. Functions F, φ and ψ are non-differentiable
3. Mathematical formula describing functions F, φ and ψ is not available, it can only be „measured"
4. Large dimension of decision variables vector

Numerical methods

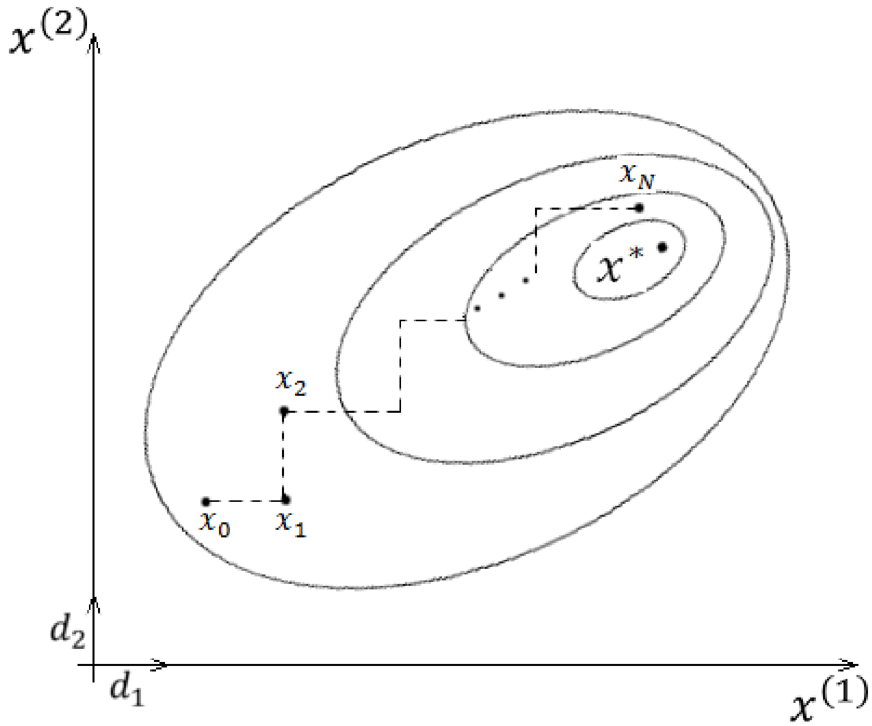
We only use information about values of objective function $F(x)$ for a given value of x.





The general idea behind numerical methods.

Numerical optimization methods

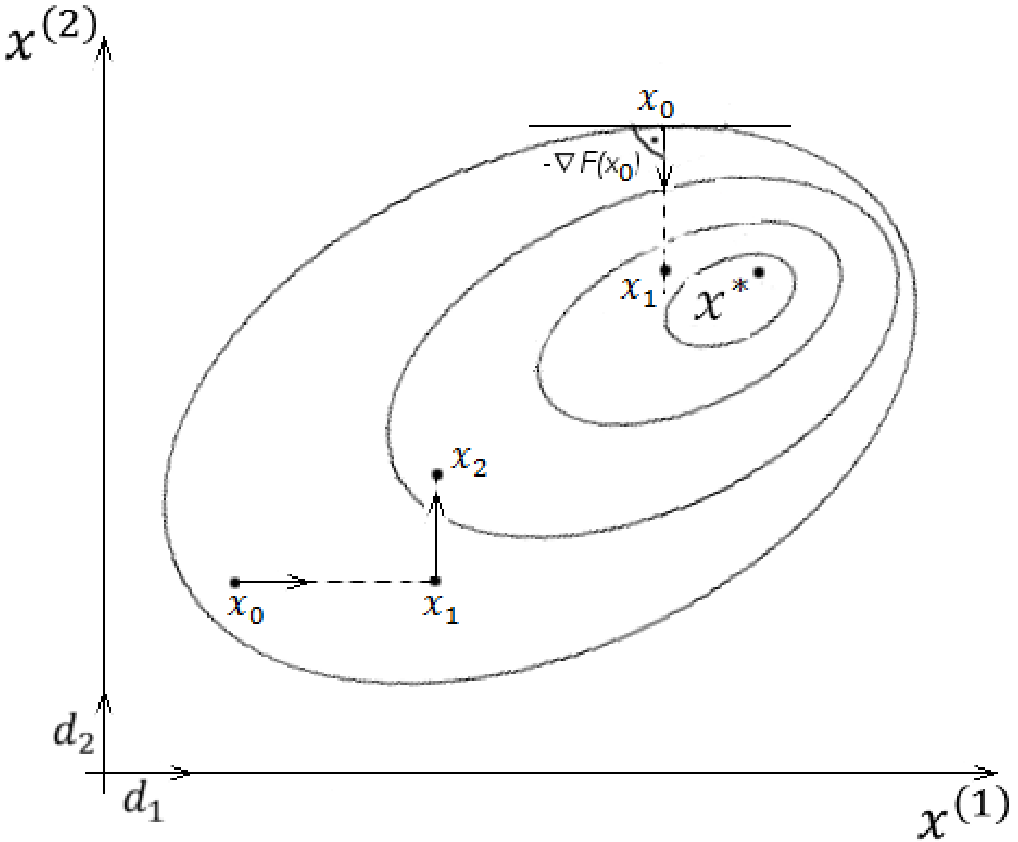


Algorithm
$x_{n+1} = \Psi(x_n), x_0$

- Choice of the search direction.
- Line search optimization.
- Stopping conditions.

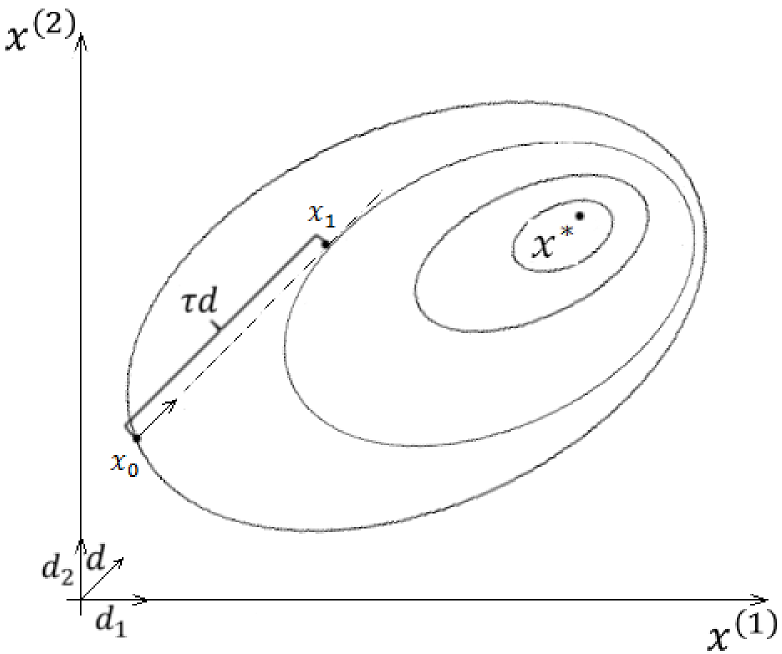
$$x_0, x_1, \dots, x_n, \dots, x_N \approx x^*$$
$$F(x_0) > F(x_1) > \dots > F(x_n) > \dots > F(x_N) \approx F(x^*)$$

Choice of the search direction



- Basis of search directions – non-gradient methods.
- Search directions based on gradient vectors – gradient-based methods.

Line search optimization



x_0 – initial solution
x_1 – next solution
d – search direction
τ – step size

$$\tau^* \rightarrow F(x_0 + \tau^* d) = \min_{\tau} F(x_0 + \tau d)$$

x_0, d – fixed

$$F(x_0 + \tau d) \triangleq f(\tau)$$

$f(\tau)$ – a single variable function
(of the step size τ)

$$\tau^* \rightarrow f(\tau^*) = \min_{\tau} f(\tau)$$

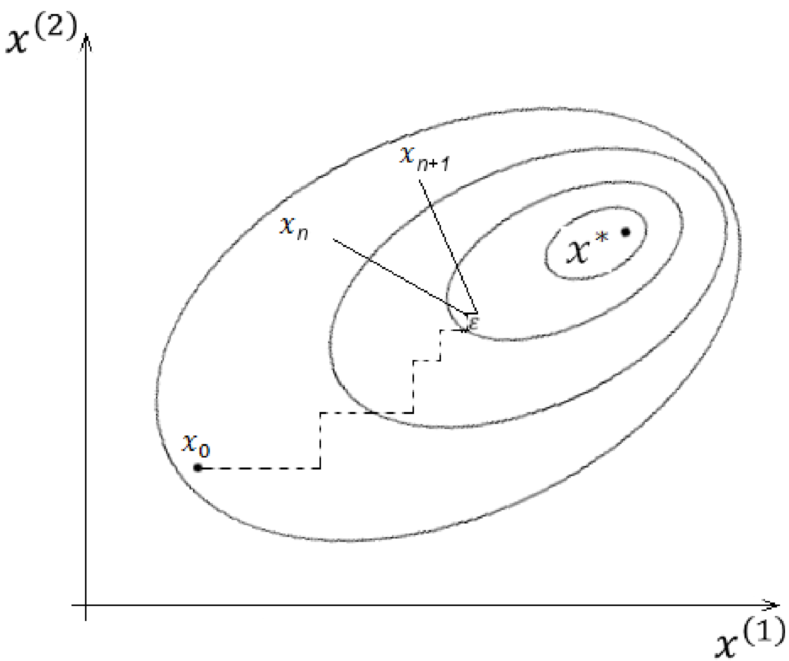
line search optimization ≡ optimization of a single variable function

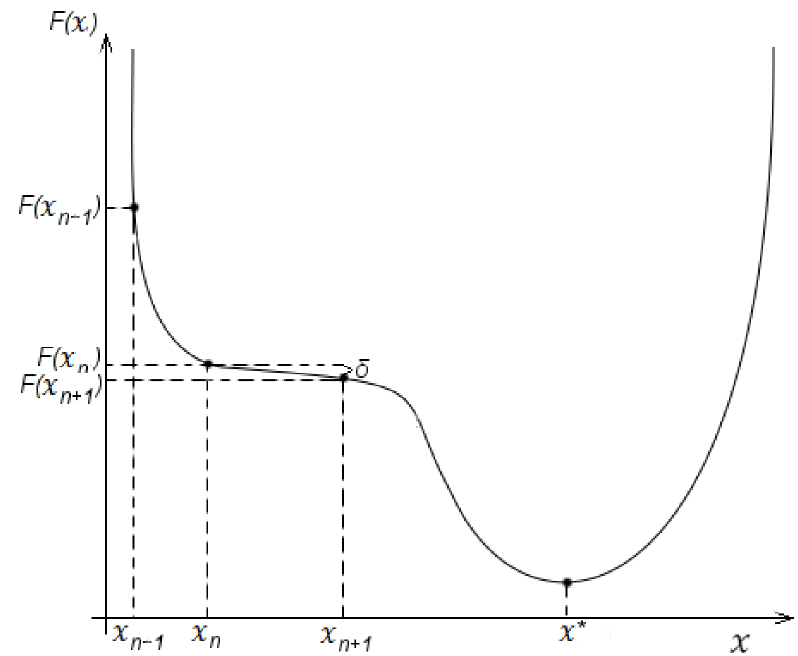
HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Stopping conditions

$$\|x_{n+1} - x_n\| < \varepsilon; \qquad |F(x_{n+1}) - F(x_n)| < \delta;$$

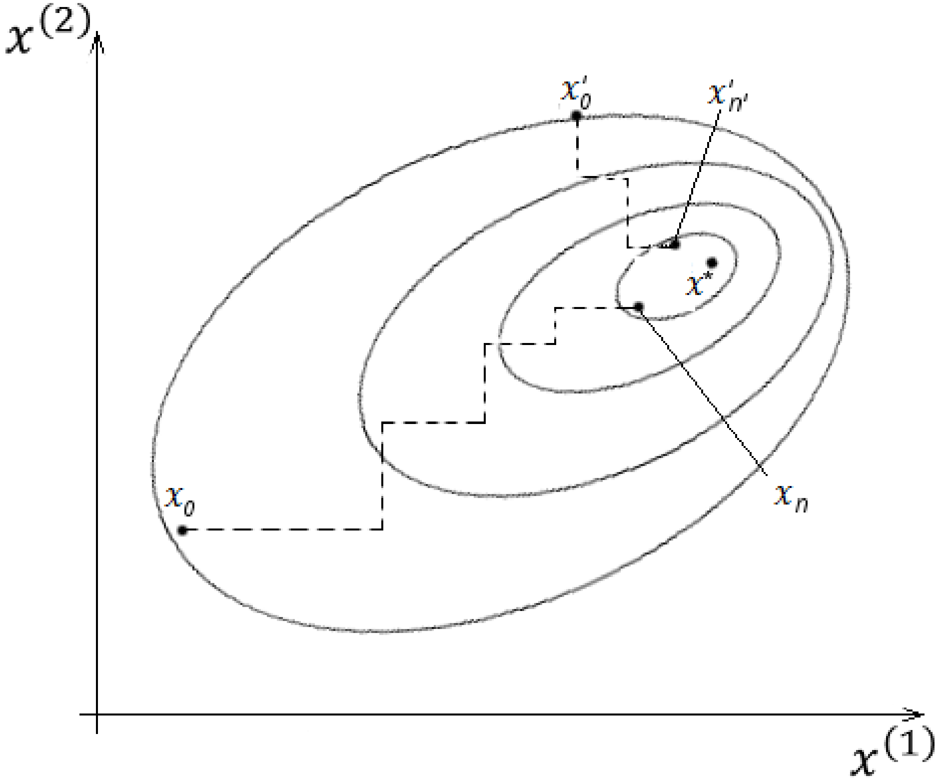
$$\frac{|F(x_{n+1}) - F(x_n)|}{\|x_{n+1} - x_n\|} < \varrho$$





Project co-financed from the EU European Social Fund

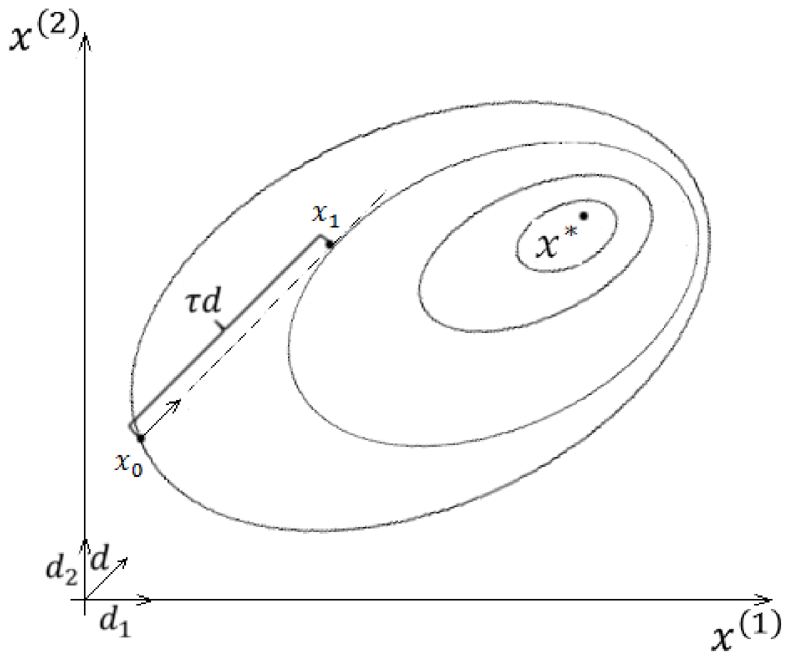
Remedy



$$\left\| x_n - x'_{n'} \right\| < \varepsilon$$

x_0, x'_0 – different initial solutions
$x_n, x'_{n'}$ – responding final solutions

Line search optimization



x_0 – initial solution
x_1 – next solution
d – search direction
τ – step size

$$\tau^* \rightarrow F(x_0 + \tau^* d) = \min_{\tau} F(x_0 + \tau d)$$

x_0, d – fixed

$$F(x_0 + \tau d) \triangleq f(\tau)$$

$f(\tau)$ – a single variable function (of the step size τ)

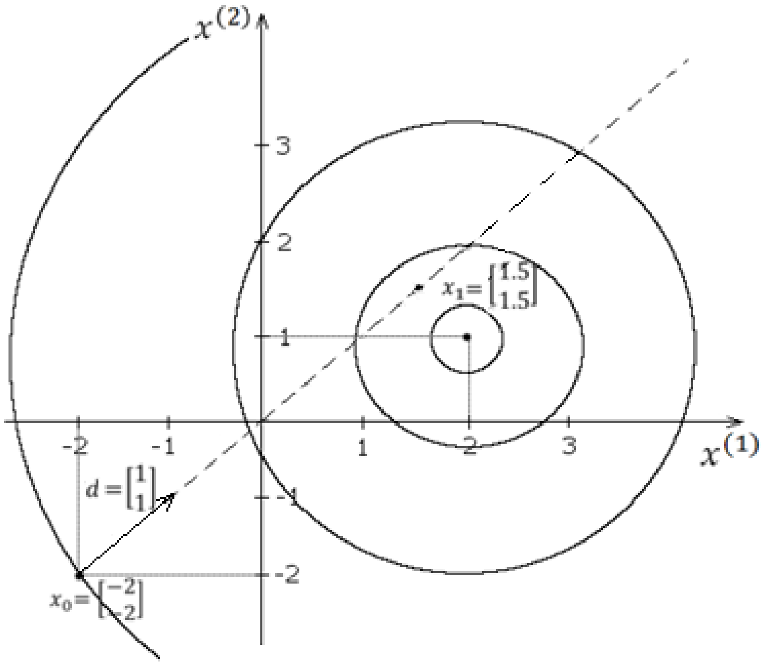
$$\tau^* \rightarrow f(\tau^*) = \min_{\tau} f(\tau)$$

line search optimization ≡ optimization of a single variable function

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Example

$$F(x) = \left(x^{(1)} - 2\right)^2 + \left(x^{(2)} - 1\right)^2, \; x_0 = \begin{bmatrix} -2 \\ -2 \end{bmatrix}, d = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



$$x_1 = x_0 + \tau d = \begin{bmatrix} -2 \\ -2 \end{bmatrix} + \tau \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 + 1 * \tau \\ -2 + 1 * \tau \end{bmatrix}$$

$$F(x_0 + \tau d) = (-2 + \tau - 2)^2 + (-2 + \tau - 1)^2 =$$
$$(\tau - 4)^2 + (\tau - 3)^2 = 2\tau^2 - 14\tau + 25 \triangleq f(\tau)$$

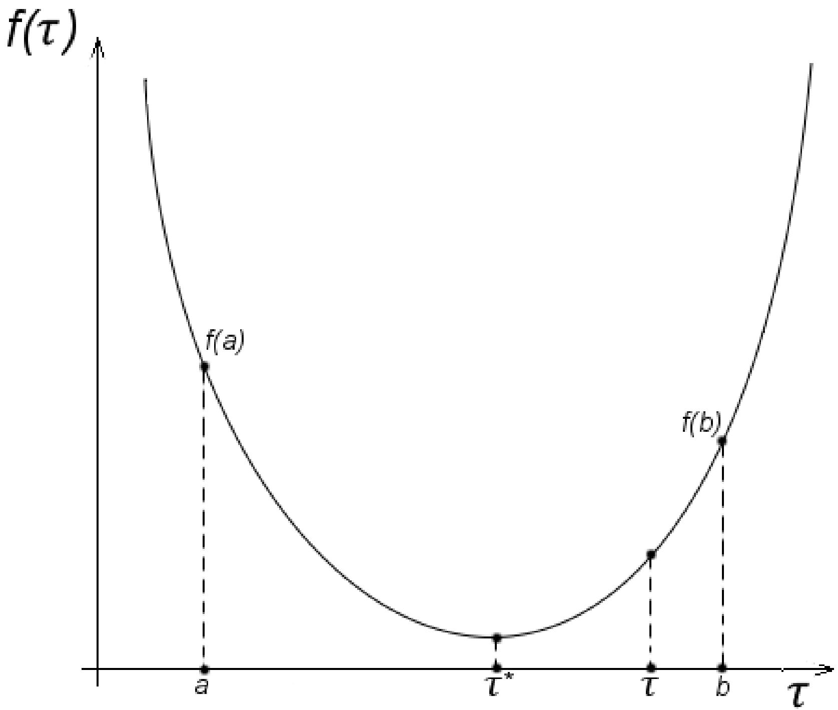
$$f'(\tau) = 4\tau^* - 14 = 0$$

$$\tau^* = 3.5$$

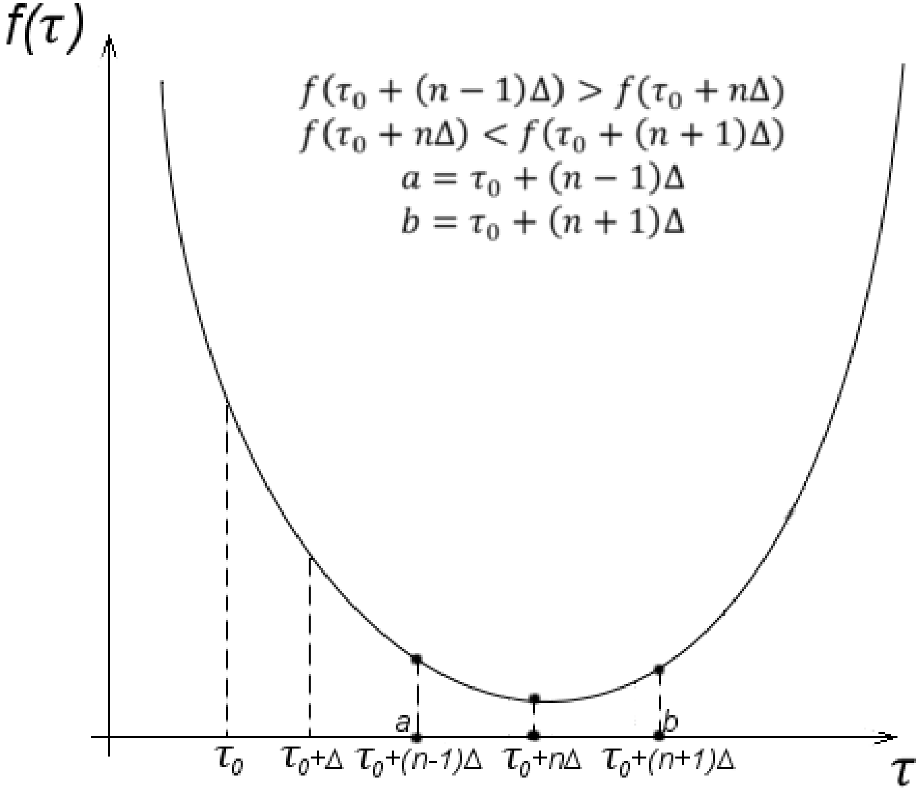
$$x_1 = x_0 + \tau^* d = \begin{bmatrix} -2 \\ -2 \end{bmatrix} + 3.5 * \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 1.5 \end{bmatrix}$$

Reducing the interval of uncertainty

Assumption: $\tau^* \in [a, b]$

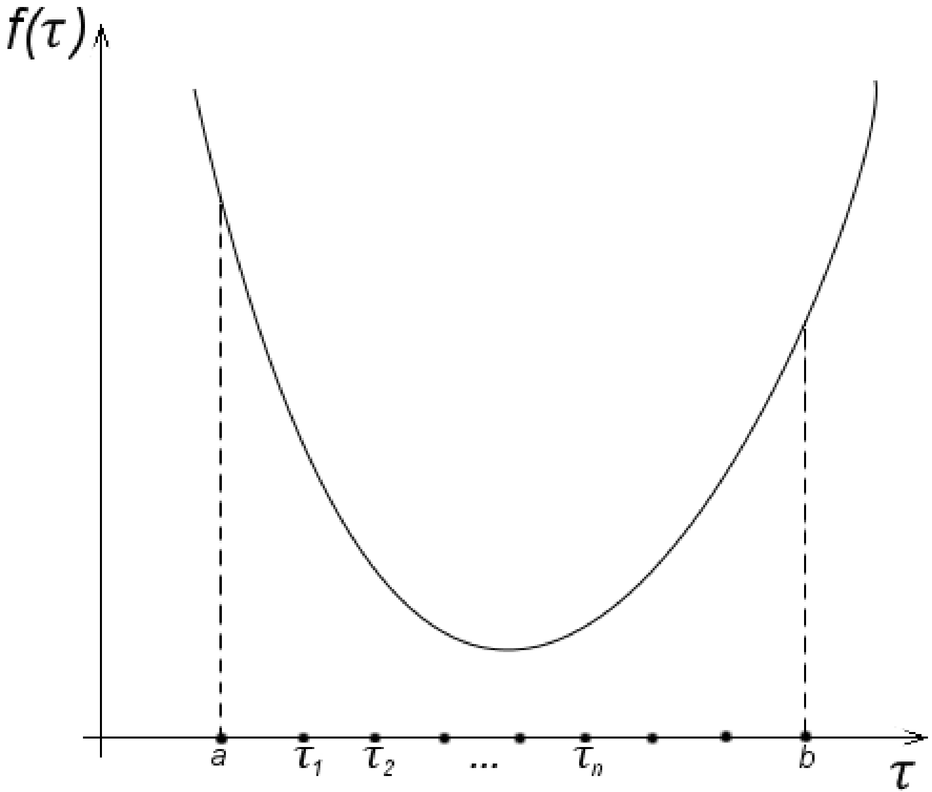


$$\boldsymbol{\tau} \in [\boldsymbol{a}, \boldsymbol{b}]$$



$$f(\tau_0 + (n-1)\Delta) > f(\tau_0 + n\Delta)$$
$$f(\tau_0 + n\Delta) < f(\tau_0 + (n+1)\Delta)$$
$$a = \tau_0 + (n-1)\Delta$$
$$b = \tau_0 + (n+1)\Delta$$

Uniform search method



$N = \left[\frac{b-a}{\varepsilon}\right]$ – the total number of the goal function evaluations

For example:

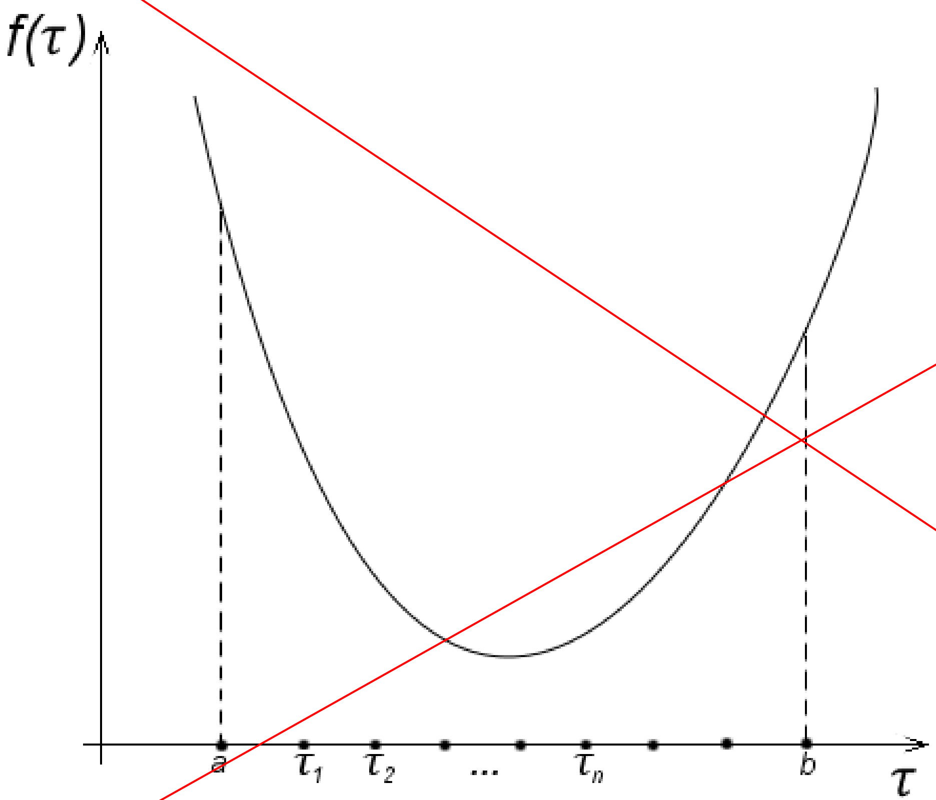
$$\Delta = b - a = 1, \varepsilon = 0.01$$

$$N = \left[\frac{b-a}{\varepsilon}\right] = \frac{1}{0.01} = 100$$

$$\tau_0 = a$$
$$\tau_n = \tau_0 + n\varepsilon$$
$$\tau^* \approx \tilde{\tau} \to f(\tilde{\tau}) = \min_{1 \le n \le N}\{f(\tau_n)\}$$

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Uniform search method



$N = \left[\frac{b-a}{\varepsilon}\right]$ – the total number of the goal function evaluations

For example:

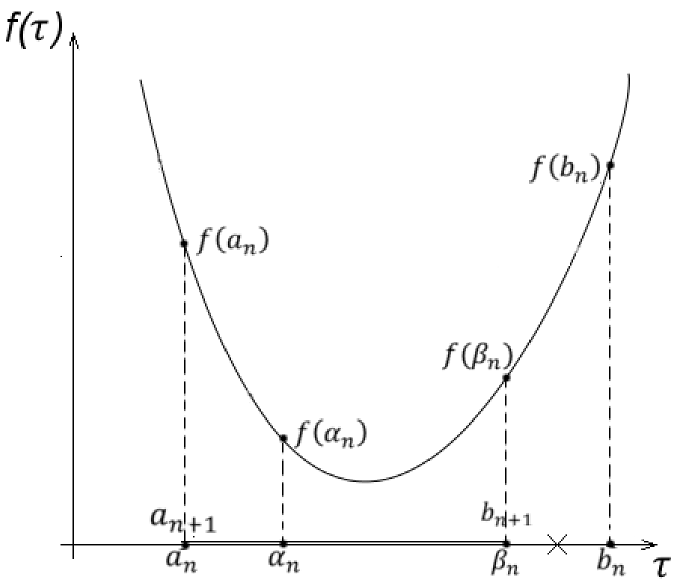
$$\Delta = b - a = 1, \varepsilon = 0.01$$

$$N = \left[\frac{b-a}{\varepsilon}\right] = \frac{1}{0.01} = 100$$

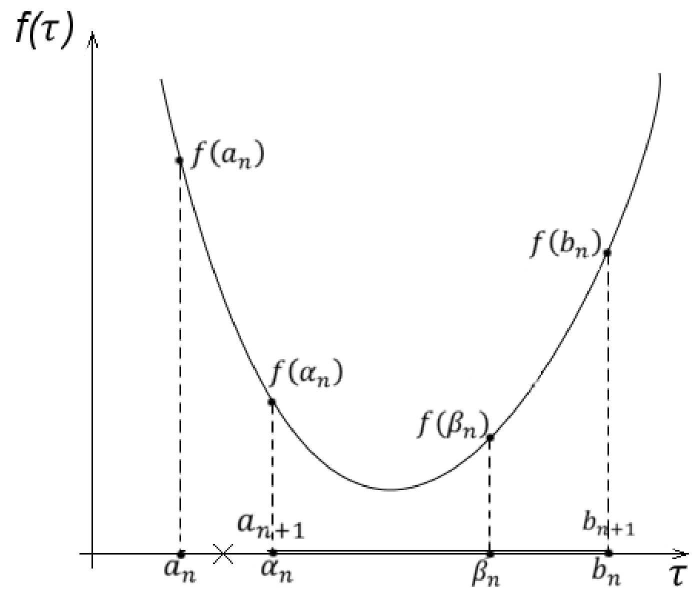
$$\tau_0 = a$$
$$\tau_n = \tau_0 + n\varepsilon$$
$$\tau^* \approx \tilde{\tau} \to f(\tilde{\tau}) = \min_{1 \le n \le N}\{f(\tau_n)\}$$

Splitting the section into two parts

$$f(\alpha_n)?\, f(\beta_n)$$



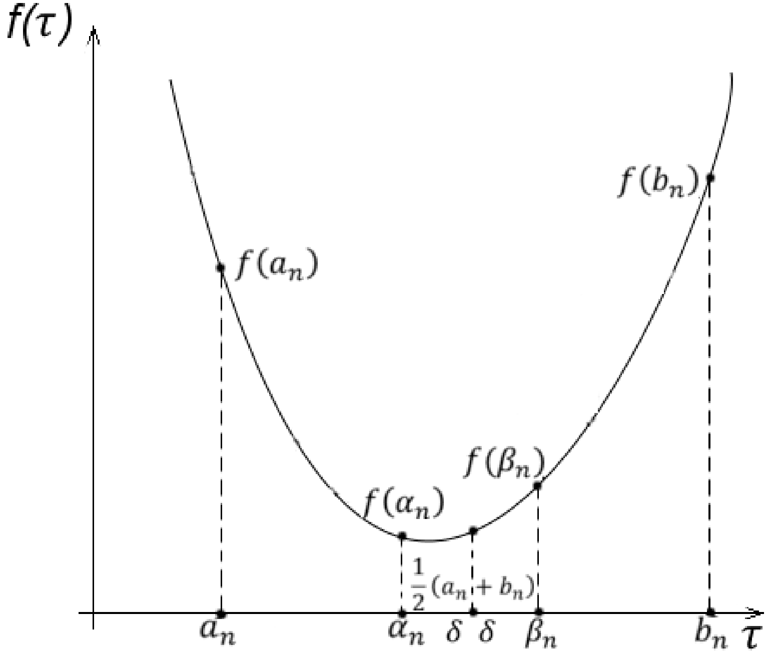
$$f(\alpha_n) \leq f(\beta_n)$$
$$a_{n+1} := a_n$$
$$b_{n+1} := \beta_n$$



$$f(\alpha_n) > f(\beta_n)$$
$$a_{n+1} := \alpha_n$$
$$b_{n+1} := b_n$$

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Dichotomous search method



$$\alpha_n = \frac{1}{2}(a_n + b_n) - \delta$$

$$\beta_n = \frac{1}{2}(a_n + b_n) + \delta$$

$N =?$ for $\varepsilon = 0.01$, $\Delta = b - a = 1$

Input data: $a_0, b_0, \varepsilon, \delta$

Step 0: $n = 0$

Step 1: $\alpha_n = \frac{1}{2}(a_n + b_n) - \delta$

$$\beta_n = \frac{1}{2}(a_n + b_n) + \delta$$

Step 2: If $f(\alpha_n) \le f(\beta_n)$ then
$a_{n+1} := a_n, b_{n+1} := \beta_n$,
otherwise
$a_{n+1} := \alpha_n, b_{n+1} := b_n$.

Step 3: If $|b_{n+1} - a_{n+1}| \ge \varepsilon$ then
$n := n + 1$, go to 1,
otherwise
$\tilde{\tau} = \frac{1}{2}(a_{n+1} + b_{n+1})$ (STOP)

Estimation of steps procedure number

- $\Delta_0 = b_0 - a_0 = 1$ initial length of interval
- $\Delta_1 = \frac{1}{2}\Delta_0$ interval length after one step
- $\Delta_2 = \frac{1}{2}\Delta_1 = \left(\frac{1}{2}\right)^2 \Delta_0$ interval length after two step
- $\vdots$
- $\Delta_N = \frac{1}{2}\Delta_{N-1} = \cdots = \left(\frac{1}{2}\right)^N \Delta_0$ interval length after N-th step
- We expect, that after *N* steps interval lenght will be less then ε. Then the number of steps must fulfil the following condition:

$$\Delta_N = \left(\frac{1}{2}\right)^N \Delta_0 \leq \varepsilon$$ we divide by Δ_0

Estimation of steps procedure number

- $\left(\frac{1}{2}\right)^N \leq \frac{\varepsilon}{\Delta_0}$ log both sides
- $ln\left(\frac{1}{2}\right)^N \leq ln\frac{\varepsilon}{\Delta_0}$ consequently
- $N\ ln\left(\frac{1}{2}\right) \leq ln\frac{\varepsilon}{\Delta_0}$ dividing both sides by $ln\left(\frac{1}{2}\right)$.

Because $ln\left(\frac{1}{2}\right)$ is a negative number we change sigh of inequality

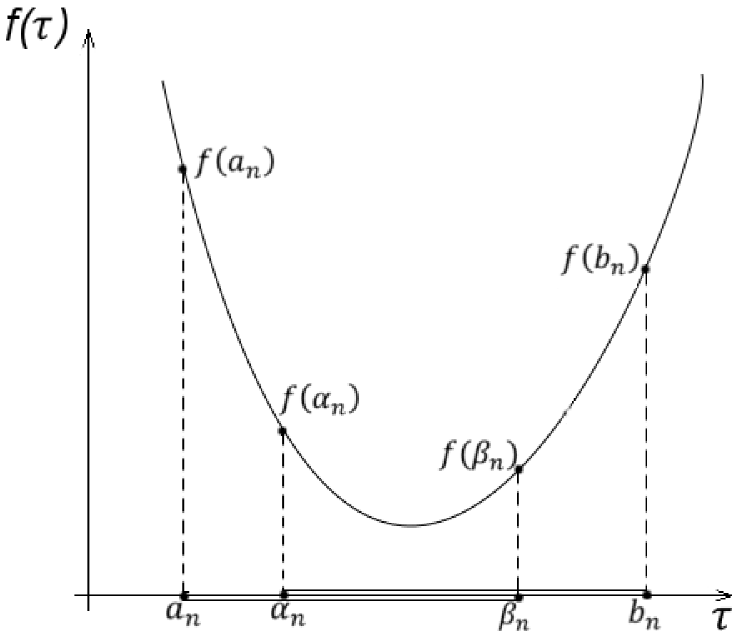
Then the minimum number of procedure steps must fulfill the following condition:

- $N \geq \frac{ln\frac{\varepsilon}{\Delta_0}}{ln\frac{1}{2}} = \frac{\ln 0.01}{\ln 0.5} = \frac{4.605}{0.693} \approx 7$

Because in each step we must calculate two Times value of function then necessary number of function calculation is:

- Number of function calculation is: = 2*N = 14*

The $\boldsymbol{\gamma}$ section method



$$\frac{b_n - \alpha_n}{b_n - a_n} = \frac{\beta_n - a_n}{b_n - a_n} = \gamma$$

$\alpha_n = b_n + \gamma(a_n - b_n)$

$\beta_n = a_n + \gamma(b_n - a_n)$ $\quad N = ?$ for $\varepsilon = 0.01, \Delta = b - a = 1$

Input data: $a_0, b_0, \varepsilon, \gamma$

Step 0: $n = 0$

Step 1: $\alpha_n = b_n + \gamma(a_n - b_n)$

$\beta_n = a_n + \gamma(b_n - a_n)$

Step 2: If $f(\alpha_n) \le f(\beta_n)$ then

$a_{n+1} := a_n, b_{n+1} := \beta_n,$

otherwise

$a_{n+1} := \alpha_n, b_{n+1} := b_n.$

Step 3: If $|b_{n+1} - a_{n+1}| \ge \varepsilon$ then

$n := n + 1$, go to 1,

otherwise

$\tilde{\tau} = \frac{1}{2}(a_{n+1} - b_{n+1})$ (STOP)

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Estimation of steps procedure number

- $\Delta_0 = b_0 - a_0 = 1$ initial length of interval , ε=0.01
- $\Delta_1 = \gamma\Delta_0$ interval length after one step
- $\Delta_2 = \gamma\Delta_1 = (\gamma)^2\Delta_0$ interval length after two step

  $\vdots$
- $\Delta_N = \gamma\Delta_{N-1} = \cdots = (\gamma)^N\Delta_0$ interval length after N-th step
- We expect, that after *N* steps interval lenght will be less then ε.

  Then the number of steps must fulfil the following condition:

$$\Delta_N = (\gamma)^N\Delta_0 \leq \varepsilon$$ we divide by Δ_0

Estimation of steps procedure number

- $(\gamma)^N \leq \frac{\varepsilon}{\Delta_0}$ log both sides
- $ln(\gamma)^N \leq ln\frac{\varepsilon}{\Delta_0}$ consequently
- $N\ ln(\gamma) \quad \leq ln\frac{\varepsilon}{\Delta_0}$ dividing both sides by $ln(\gamma)$.

Because $ln(\gamma)$ is a negative number we change sigh of nonequality

Then the minimum number of procedure steps must fulfill the following condition:

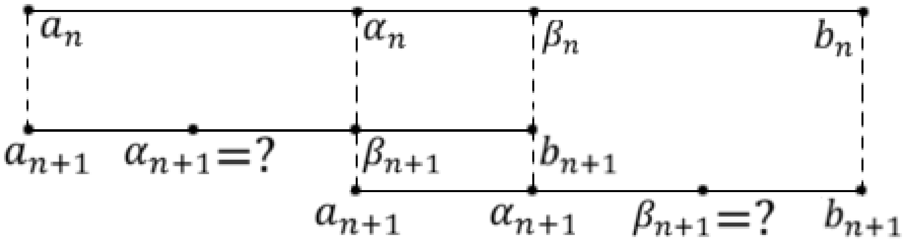
- $N \geq \frac{ln\frac{\varepsilon}{\Delta_0}}{ln\gamma}$

Because in each step we must calculate two Times value of function then necessary number of function calculation is:

Number of function calculation is = 2*N*

$$\gamma = ??$$

The golden section method



$$\gamma^2+\gamma-1=0$$
$$\gamma=\frac{\sqrt{5}-1}{2}\approx 0.618$$

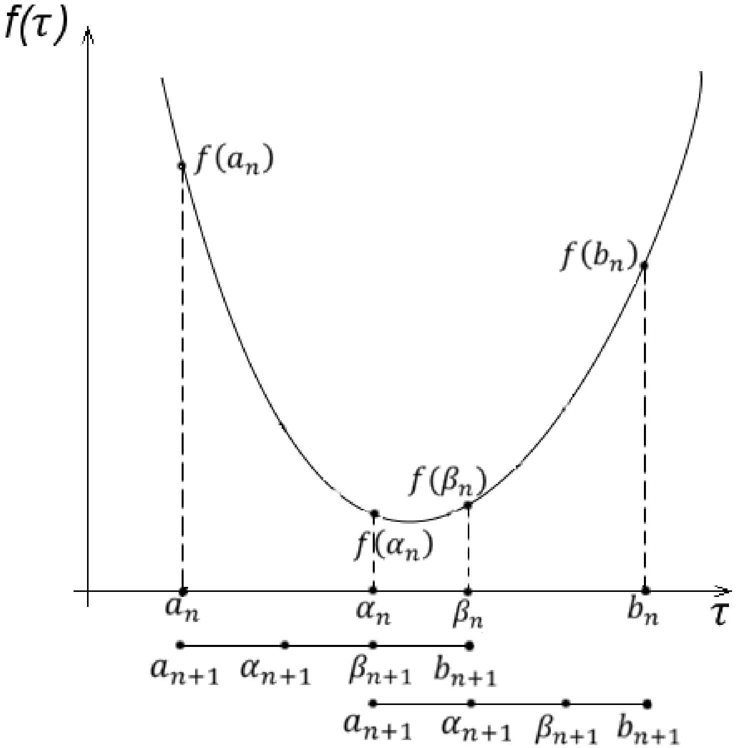
1. $\frac{\beta_{n+1}-a_{n+1}}{b_{n+1}-a_{n+1}}=\gamma$, which gives

$$\frac{\alpha_n-a_n}{\beta_n-a_n}=\frac{b_n+\gamma(a_n-b_n)-a_n}{a_n+\gamma(b_n-a_n)-a_n}=\frac{b_n-a_n+\gamma(a_n-b_n)}{\gamma(b_n-a_n)}=\frac{1}{\gamma}-1=\gamma$$

2. $\frac{b_{n+1}-\alpha_{n+1}}{b_{n+1}-a_{n+1}}=\gamma$, which gives

$$\frac{b_n-\beta_n}{b_n-a_n}=\frac{b_n-a_n+\gamma(b_n-a_n)}{b_n-b_n-\gamma(b_n-a_n)}=\frac{b_n-a_n+\gamma(b_n-a_n)}{\gamma(b_n-a_n)}=\frac{1}{\gamma}-1=\gamma$$

The golden section method



$\gamma^2 + \gamma - 1 = 0$

$\gamma = \frac{\sqrt{5}-1}{2} \approx 0.618$

$N =?$ for $\varepsilon = 0.01, \Delta = b - a = 1$

Input data: $a_0, b_0, \varepsilon, \gamma = \frac{\sqrt{5}-1}{2}$

Step 0: $n = 0$

$\alpha_0 = b_0 + \gamma(a_0 - b_0)$

$\beta_0 = a_0 + \gamma(b_0 - a_0)$

Step 1: If $|b_n - a_n| < \varepsilon$, then

$\tilde{\tau} = \frac{1}{2}(a_n + b_n)$(STOP)

otherwise go to 2

Step 2: If $f(\alpha_n) \leq f(\beta_n)$ then

$a_{n+1} := a_n, b_{n+1} := \beta_n,$

$\beta_{n+1} := \alpha_n, \quad \alpha_{n+1} := \beta_n + \gamma(a_n - b_n)$

$n := n + 1$, go to 1

otherwise

$a_{n+1} := \alpha_n, b_{n+1} := b_n,$

$\alpha_{n+1} := \beta_n, \quad \beta_{n+1} := \alpha_n + \gamma(b_n - \alpha_n)$

$n := n + 1$, go to 1

Estimation of steps procedure number

- $(\gamma)^N \leq \frac{\varepsilon}{\Delta_0}$ log both sides
- $ln(\gamma)^N \leq ln\frac{\varepsilon}{\Delta_0}$ consequently
- $N\, ln(\gamma) \quad \leq ln\frac{\varepsilon}{\Delta_0}$ dividing both sides by $ln(\gamma)$.

Because $ln(\gamma)$ is a negative number we change sigh of nonequality

Then the minimum number of procedure steps must fulfill the following condition

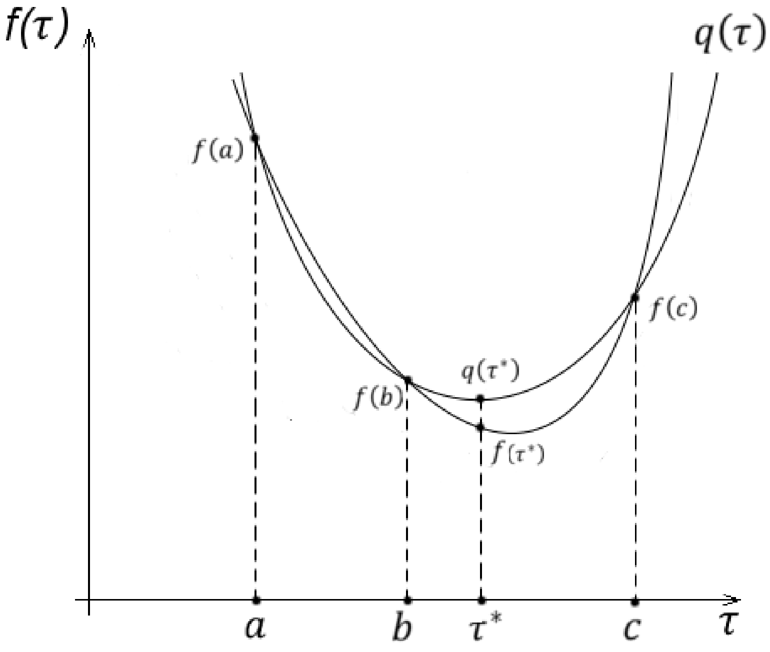
- $N \geq \frac{ln\frac{\varepsilon}{\Delta_0}}{ln\gamma}$ now $\gamma = \frac{\sqrt{5}-1}{2} \approx 0.618$

Because in each step we must calculate two Times value of function then necessary number of function calculation is:

- $ln\gamma$= $ln0.618$ =- 0.481
- $N \geq \frac{ln\frac{\varepsilon}{\Delta_0}}{ln\gamma} = \frac{\ln 0.01}{\ln 0.618} = \frac{4.605}{0.481} \approx 9{,}578 \approx 10$
- Number of function calculation is N+1 =10+1 =11

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Wrocław University of Technology
EUROPEAN SOCIAL FUND
Project co-financed from the EU European Social Fund

Quadratic-fit line search method



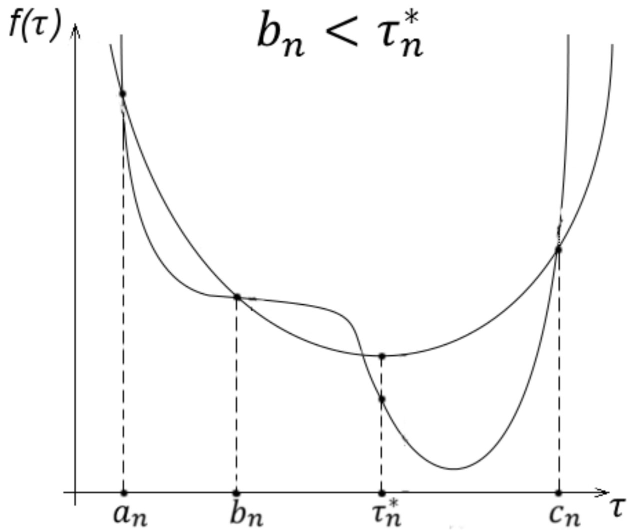
$$a < b < c$$
$$f(a) \geq f(b)$$
$$f(b) \leq f(c)$$

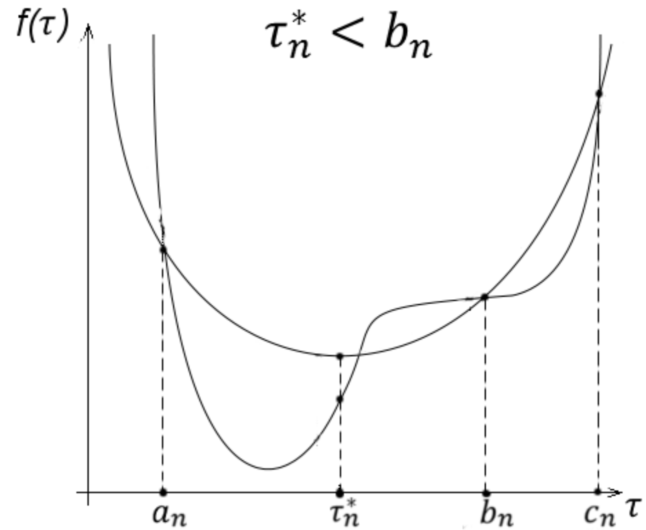
$q(\tau)$ – quadratic-fit function
τ^* - minimum of the function $q(\tau)$

$$q(\tau) = \frac{f(a)(\tau - b)(\tau - c)}{(a-b)(a-c)} + \frac{f(b)(\tau - a)(\tau - c)}{(b-a)(b-c)} + $$
$$+ \frac{f(c)(\tau - a)(\tau - b)}{(c-a)(b-c)}$$

$$\tau^* = \frac{1}{2}\frac{f(a)\left(b^2 - c^2\right) + f(b)\left(c^2 - a^2\right) + f(c)(a^2 - b^2)}{f(a)(b-c) + f(b)(c-a) + f(c)(a-b)}$$

Quadratic-fit line search method





$f(b_n) \geq f(\tau_n^*)$	$f(b_n) < f(\tau_n^*)$	$f(b_n) \geq f(\tau_n^*)$	$f(b_n) \geq f(\tau_n^*)$
$a_{n+1} := b_n$	$a_{n+1} := a_n$	$a_{n+1} := a_n$	$a_{n+1} := \tau_n^*$
$b_{n+1} := \tau_n^*$	$b_{n+1} := b_n$	$b_{n+1} := \tau_n^*$	$b_{n+1} := b_n$
$c_{n+1} := c_n$	$c_{n+1} := \tau_n^*$	$c_{n+1} := b_n$	$c_{n+1} := c_n$

$$|c_{n+1} - a_{n+1}| < \varepsilon \qquad \tilde{\tau} = \tau_{n+1}^*$$

Quadratic-fit line search method

Input data: $a_0, b_0, c_0, \varepsilon$

Step 0: $n = 0$

Step 1: $\tau_n = \frac{1}{2}\frac{f(a_n)(b_n^2-c_n^2)+f(b_n)(c_n^2-a_n^2)+f(c_n)(a_n^2-b_n^2)}{f(a_n)(b_n-c_n)+f(b_n)(c_n-a_n)+f(c_n)(a_n-b_n)}$

Step 2: If $b_n < \tau_n$ then go to 3

otherwise

If $f(b_n) \geq f(\tau_n)$ then $a_{n+1} \coloneqq a_n, b_{n+1} \coloneqq \tau_n, c_{n+1} \coloneqq b_n$ go to 4

otherwise $a_{n+1} \coloneqq \tau_n, b_{n+1} \coloneqq b_n, c_{n+1} \coloneqq c_n$ go to 4

Step 3: If $f(b_n) \geq f(\tau_n)$ to $a_{n+1} \coloneqq b_n, b_{n+1} \coloneqq \tau_n, c_{n+1} \coloneqq c_n$ go to 4

otherwise $a_{n+1} \coloneqq a_n, b_{n+1} \coloneqq b_n, c_{n+1} \coloneqq \tau_n$ go to 4

Step 4: If $|c_{n+1} - a_{n+1}| \geq \varepsilon$ then $n \coloneqq n + 1$ go to 1

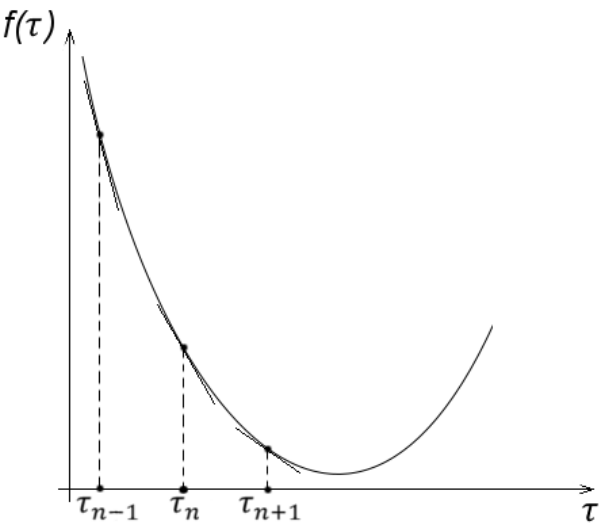
otherwise $\tilde{\tau} = \frac{1}{2}(a_{n+1} + c_{n+1})$ (STOP)

Line search using derivatives

$\tau_{n+1} = \tau_n - \gamma_n f'(t_n) \qquad \gamma_n > 0, \tau_0$

$\lim_{n\to\infty} \gamma_n = \gamma \qquad \sum_{n=0}^{\infty} \gamma_n = \infty$

e.g. $|\tau_{n+1} - \tau_n| < \varepsilon$ (STOP)



τ_0

$\tau_1 = \tau_0 - \gamma_0 f'(\tau_0)$

$\tau_2 = \tau_1 - \gamma_1 f'(\tau_1) = \tau_0 - \gamma_0 f'(\tau_0) - \gamma_1 f'(\tau_1)$

$\tau_{n+1} = \tau_n + \gamma_n f'(\tau_n) = \cdots = \tau_0 - \gamma_0 f'(\tau_0) - \gamma_1 f'(\tau_1) - \cdots - \gamma_n f'(\tau_n)$

$|\tau_{n+1} - \tau_0| = |\sum_{k=0}^{n} \gamma_k f'(\tau_k)| \leq \sum_{k=0}^{n} \gamma_k |f'(\tau_k)| \leq \max_{0<k<n} |f'(\tau_k)| \sum_{k=0}^{n} \gamma_k$

$|\tau_\infty - \tau_0| \leq \sum_{k=0}^{\infty} \gamma_k = \infty$

Line search using sign of derivatives

$\tau_{n+1} = \tau_n - \vartheta_n sign[f'(\tau_n)]$

$\gamma_n f'(\tau_n) = \gamma_n |f'(\tau_n)| * sign\, f'(\tau_n) = \vartheta_n\, sign[f'(\tau_n)]$, where $\vartheta_n = \gamma_n |f'(\tau_n)|$

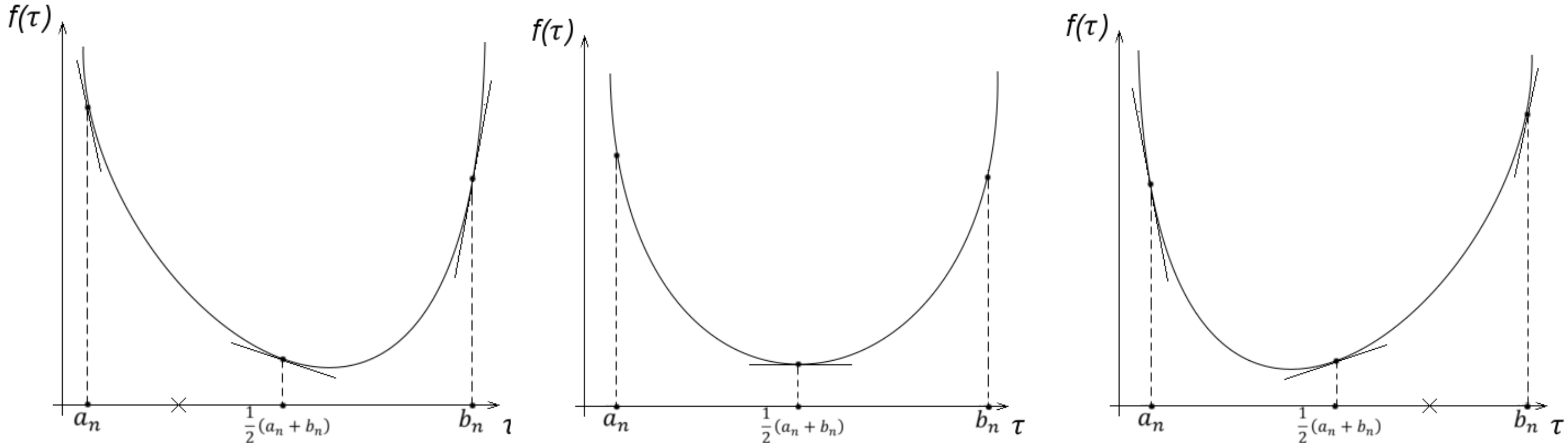
$\vartheta_n > 0$

$\lim_{n\to\infty} \vartheta_n = 0$, because $\lim_{n\to\infty} |f'(\tau_n)| = 0$, $\lim_{n\to\infty} \gamma_n = \gamma$

$$\sum_{n=0}^{\infty} \vartheta_n = \infty \qquad \lim_{n\to\infty} \vartheta_n = \lim_{n\to\infty} \gamma_n |f'(\tau_n)| = 0$$

Bolzano method

$$sign\ a_n \neq sign\ b_n$$



$$sign\ f'(a_n) = sign f'(\frac{1}{2}(a_n + b_n))$$

$$a_{n+1} := \frac{1}{2}(a_n + b_n)$$

$$b_{n+1} := b_n$$

$$f'\left(\frac{1}{2}(a_n + b_n)\right) = 0$$

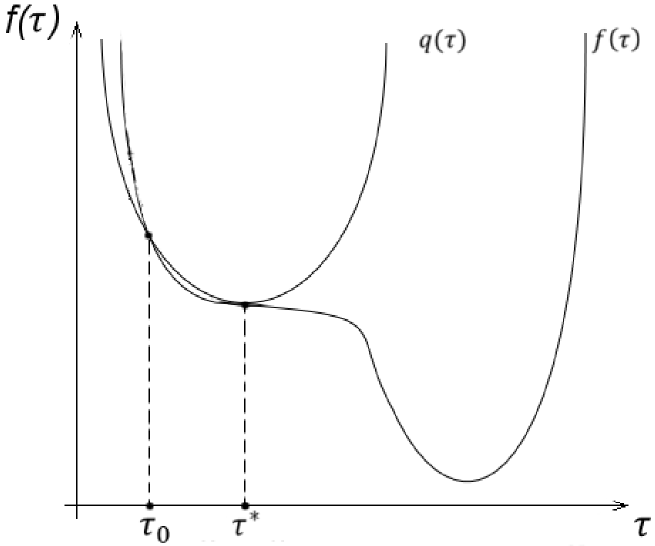
$$\tilde{\tau} := \frac{1}{2}(a_n + b_n)$$

$$sign\ f'(b_n) = sign f'(\frac{1}{2}(a_n + b_n))$$

$$a_{n+1} := a_n$$

$$b_{n+1} := \frac{1}{2}(a_n + b_n)$$

Newton's method



τ_0

$$\tau_{n+1} = \tau_n - \frac{f'(\tau_n)}{f''(\tau_n)}$$

$|\tau_{n+1} - \tau_n| < \varepsilon$ (STOP)

$$\underbrace{f(\tau) = f(\tau_0) + (\tau - \tau_0)f'(\tau_0) + \frac{1}{2}(\tau - \tau_0)^2 f''(\tau_0)}_{q(\tau)} + 0_3(|\tau - \tau_0|)$$

$$q'(\tau) = f'(\tau_0) + (\tau^* - \tau_0)f''(\tau_0) = 0 \qquad \tau^* = \tau_0 - \frac{f'(\tau_0)}{f''(\tau_0)}$$

Thank you for attention