# Optimization

Hassan OMRAN

## Lecture 2: One-Dimensional Search Methods

Télécom Physique Strasbourg
Université de Strasbourg

École d'ingénieurs
Télécom Physique
Université de Strasbourg

# Outline of the talk

# Golden Section

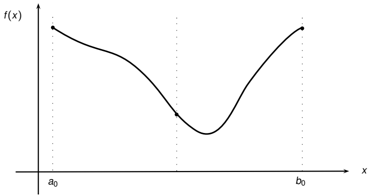
*In this part we study the problem*

$$\begin{aligned} \min_{x} \quad & f(x) \\ s.t. \quad & x \in [a_0, b_0] \end{aligned} \tag{1}$$

*Where the function* $f : \mathbb{R} \to \mathbb{R}$ *is supposed to be* ***unimodal****, thus it has only one local minimizer.*

# Golden Section

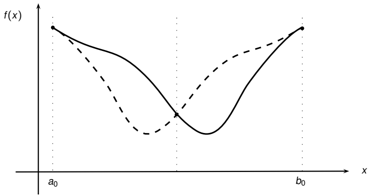
- *The idea is to narrow the search range*
- *Evaluating the function at one intermediate point of the interval is not sufficient*




Evaluating the function at one intermediate point of the interval is not sufficient.

# Golden Section

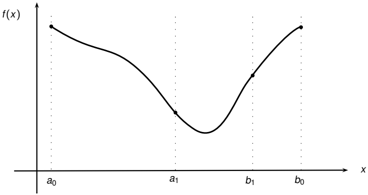
- *The idea is to narrow the search range*
- *Evaluating the function at one intermediate point of the interval is not sufficient*




Evaluating the function at one intermediate point of the interval is not sufficient.

# Golden Section

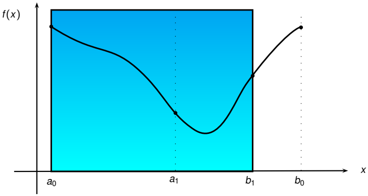
- *The idea is to narrow the search range*
- *Evaluating the function at one intermediate point of the interval is not sufficient*
- *Evaluating at two points permits to reduce the search range*

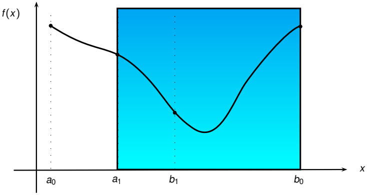



Evaluating the function at two intermediate points.

# Golden Section

- *The idea is to narrow the search range*
- *Evaluating the function at one intermediate point of the interval is not sufficient*
- *Evaluating at two points permits to reduce the search range*



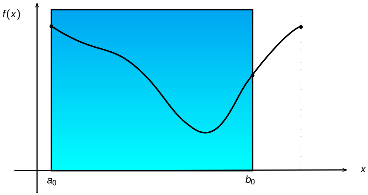




Reducing the search range.

*If* $f(b_1) > f(a_1)$, *then minimizer must be in* $[a_0, b_1]$
*If* $f(b_1) < f(a_1)$, *then minimizer must be in* $[a_1, b_0]$

# Golden Section

- *The idea is to narrow the search range*
- *Evaluating the function at one intermediate point of the interval is not sufficient*
- *Evaluating at two points permits to reduce the search range*




Reducing the search range.

*Then, we start a new search on the new domain.*

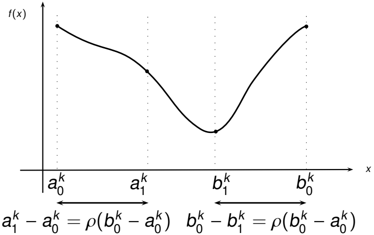
# Golden Section

- *At iteration $k$, consider the interval $[a_0^k, b_0^k]$. We choose the points in a way to have symmetry, that is*

$$a_1^k - a_0^k = b_0^k - b_1^k = \rho(b_0^k - a_0^k), \qquad \rho < \frac{1}{2} \tag{2}$$

- *In the next iteration, we want to use the already calculated value ($f(b_1^k)$ in the example below), to minimize the number of evaluations of $f(\cdot)$:*

$$\begin{aligned} a_1^{k+1} - a_0^{k+1} &= \rho(b_0^{k+1} - a_0^{k+1}) \\ b_1^k - a_1^k &= \rho(b_0^k - a_1^k) \end{aligned} \tag{3}$$



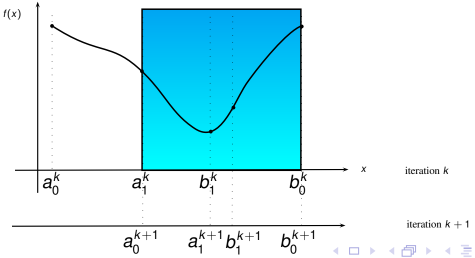
# Golden Section

◇ *At iteration $k$, consider the interval $[a_0^k, b_0^k]$. We choose the points in a way to have symmetry, that is*

$$a_1^k - a_0^k = b_0^k - b_1^k = \rho(b_0^k - a_0^k), \qquad \rho < \frac{1}{2} \tag{2}$$

◇ *In the next iteration, we want to use the already calculated value ($f(b_1^k)$ in the example below), to minimize the number of evaluations of $f(\cdot)$:*

$$\begin{aligned} a_1^{k+1} - a_0^{k+1} &= \rho(b_0^{k+1} - a_0^{k+1}) \\ b_1^k - a_1^k &= \rho(b_0^k - a_1^k) \end{aligned} \tag{3}$$



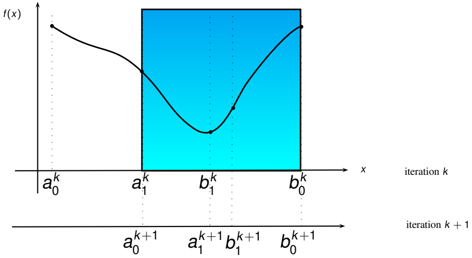
# Golden Section

*Consider* (3) *(add* $\pm\rho a_0^k$ *to the right side and* $\pm a_0^k \pm b_0^k$ *to the left side), then from* (2) *we find that*

$$\begin{aligned}(1-2\rho)(b_0^k - a_0^k) &= \rho(1-\rho)(b_0^k - a_0^k) \\ \rho^2 - 3\rho + 1 &= 0 \qquad \Rightarrow \rho = \frac{3 \pm \sqrt{5}}{2}\end{aligned}$$

*since* $\rho < \frac{1}{2}$ *then* $\rho = \frac{3-\sqrt{5}}{2} \approx 0.382$ *and* $1-\rho = 0.618$.

*Note that after N iteration the search range is reduced by* $(1-\rho)^N$.



# Golden Section

*Note that when dividing a segment* $\mathbf{s}$ *of length* $l$ *into two parts :*

- $\mathbf{s}_1$ *of length* $(1-\rho)l$
- $\mathbf{s}_2$ *of length* $\rho l$

*then we have that*

$$\frac{\text{length of } \mathbf{s}}{\text{length of } \mathbf{s}_1} = \frac{\text{length of } \mathbf{s}_1}{\text{length of } \mathbf{s}_2}$$

$$\frac{1}{(1-\rho)} = \frac{(1-\rho)}{\rho} = 1.618$$

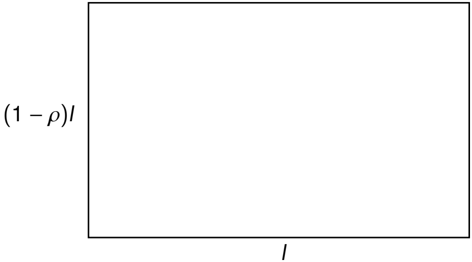
*which is called the* ***golden section***

# Golden Section

*A rectangle whose dimensions satisfy*

$$\frac{\textbf{\textit{length}}}{\textbf{\textit{width}}} = \frac{1}{1-\rho}$$

*is called the golden rectangle*

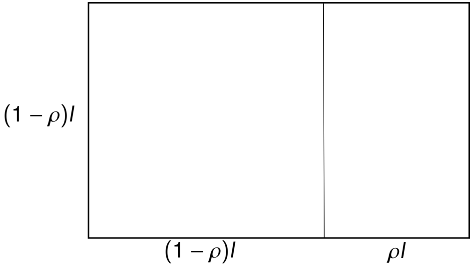


# Golden Section

*A rectangle whose dimensions satisfy*

$$\frac{\textbf{\textit{length}}}{\textbf{\textit{width}}} = \frac{1}{1-\rho}$$

*is called the golden rectangle*



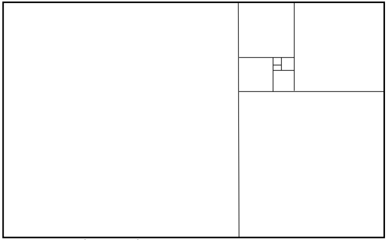
# Golden Section

*A rectangle whose dimensions satisfy*

$$\frac{\textbf{\textit{length}}}{\textbf{\textit{width}}} = \frac{1}{1-\rho}$$

*is called the golden rectangle*

$(1-\rho)l$



$(1-\rho)^2 l = \rho l$

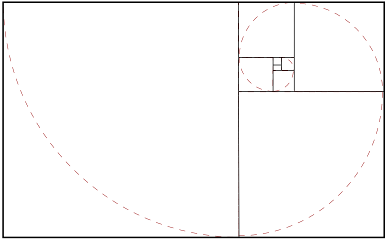
$(1-\rho)l$ $\rho l$

# Golden Section

*A rectangle whose dimensions satisfy*

$$\frac{\textit{\textbf{length}}}{\textit{\textbf{width}}} = \frac{1}{1-\rho}$$

*is called the golden rectangle*

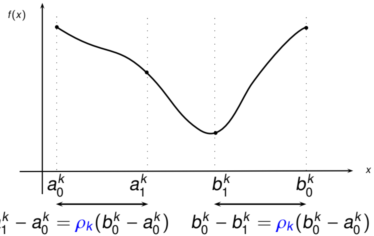


# Fibonacci search

*Consider again the problem* (1)

- *Fibonacci search is similar to the golden section*
- *The difference is that at each iteration we use a different value of* $\rho$
- *We still want one evaluation of the function each iteration*
- *At iteration k we have*

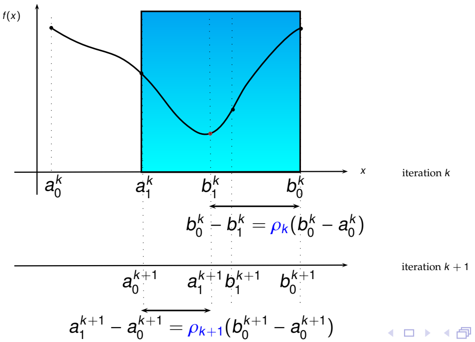
$$a_1^k - a_0^k = b_0^k - b_1^k = \rho_k(b_0^k - a_0^k), \qquad \rho < \frac{1}{2} \tag{4}$$



# Fibonacci search

◇ *At iteration $k+1$, we want to use the already calculated value ($f(b_1^k)$ in the example below), to minimize the number of evaluations of $f(\cdot)$. Note that*

$$\begin{aligned} a_1^{k+1} - a_0^{k+1} &= \rho_{k+1}(b_0^{k+1} - a_0^{k+1}) \\ b_1^k - a_1^k &= \rho_{k+1}(b_0^k - a_1^k) \end{aligned} \quad (5)$$



# Fibonacci search

*From* (4) *and* (5) *we find that*

$$\begin{aligned} (1-2\rho_k)(b_0^k - a_0^k) &= \rho_{k+1}(1-\rho_k)(b_0^k - a_0^k) \\ (1-2\rho_k) &= \rho_{k+1}(1-\rho_k) \end{aligned}$$

*which yields*

$$\rho_{k+1} = 1 - \frac{\rho_k}{1-\rho_k} \tag{6}$$

*There are several sequences that satisfy* (6) *(for example* $\rho_k = \frac{3-\sqrt{5}}{2} \rightarrow$ *Golden Section).*

*We are interested in the sequence which guarantees the maximum speed of convergence. That is, the sequence that guarantees the maximum reduction factor.*

*It is shown that this sequence is defined based on the* ***Fibonacci sequence****:*

$$F_{-1} = 0,\ F_0 = 1, \quad \textit{and} \quad F_{k+1} = F_k + F_{k-1},\ \forall k \geq 0$$

# Fibonacci search

*For N iterations we calculate the sequence*

| $F_1$ | $F_2$ | $F_3$ | $F_4$ | $F_5$ | $\cdots$ | $F_N$ | $F_{N+1}$ |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 5 | 8 | $\cdots$ | $F_{N-2} - F_{N-1}$ | $F_N + F_{N-1}$ |

*Then the reduction factors are*

$$\begin{cases} \rho_1 &= 1 - \frac{F_N}{F_{N+1}} \\ \rho_2 &= 1 - \frac{F_{N-1}}{F_N} \\ \vdots \\ \rho_N &= 1 - \frac{F_1}{F_2} \end{cases}$$

*Remarks:*

- *After N iteration the search range is reduced by*

$$(1-\rho_1)(1-\rho_2)\cdots(1-\rho_N) = \frac{F_N}{F_{N+1}}\frac{F_{N-1}}{F_N}\cdots\frac{F_1}{F_2} = \frac{1}{F_{N+1}}$$

- *Careful to the anomaly in the final iteration since* $\rho_N = 1 - \frac{F_1}{F_2} = 0.5$

# Newton's method

*Consider the twice continuously differentiable function $f : \mathbb{R} \to \mathbb{R}$, we study the following problem:*

$$\begin{aligned} \min_{x} \quad & f(x) \\ s.t. \quad & x \in \mathbb{R} \end{aligned} \tag{7}$$

*The idea is to minimize a second-order approximation of $f(\cdot)$ at each iteration. Consider the case where $\ddot{f}(x) > 0$:*

- *Start with an initial value $x_0$*
- *At each iteration minimize second-order approximation of $f(\cdot)$ at $x_k$:*

$$q_k(x) = f(x_k) + \dot{f}(x_k)\,(x - x_k) + \frac{1}{2}\ddot{f}(x_k)(x - x_k)^2$$

*in this case the minimum of $q_k(\cdot)$ is obtained by finding the point where $\dot{q}_k(x) = 0$*

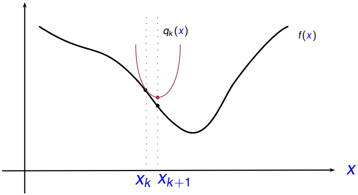
$$\dot{q}_k(x) = \dot{f}(x_k) + \ddot{f}(x_k)\,(x - x_k) = 0$$

# Newton's method

*which yields*

$$\dot{q}_k(x_{k+1}) = 0, \qquad x_{k+1} = x_k - \frac{\dot{f}(x_k)}{\ddot{f}(x_k)}$$

*We stop after N iterations when a certain accuracy is satisfied, i.e.* $|x_N - x_{N-1}| < \epsilon$ *for* $\epsilon > 0$.




Newton's method.

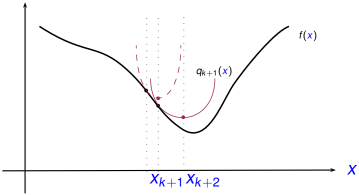
◇

# Newton's method

*which yields*

$$\dot{q}_k(x_{k+1}) = 0, \qquad x_{k+1} = x_k - \frac{\dot{f}(x_k)}{\ddot{f}(x_k)}$$

*We stop after N iterations when a certain accuracy is satisfied, i.e.* $|x_N - x_{N-1}| < \epsilon$ *for* $\epsilon > 0$.




Newton's method.

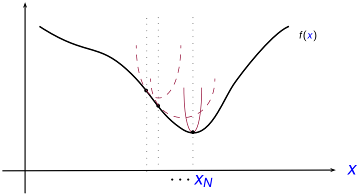
◇

# Newton's method

*which yields*

$$\dot{q}_k(x_{k+1}) = 0, \qquad x_{k+1} = x_k - \frac{\dot{f}(x_k)}{\ddot{f}(x_k)}$$

*We stop after N iterations when a certain accuracy is satisfied, i.e.* $|x_N - x_{N-1}| < \epsilon$ *for* $\epsilon > 0$.




Newton's method.

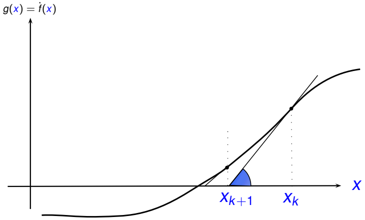
◇

# Newton's method

*A nice interpretation of the Newton's method is finding the zero of the derivative function* $\dot{f}(x) = g(x)$. *Indeed, the algorithm defined by the iterations*

$$x_{k+1} = x_k - \frac{\dot{f}(x_k)}{\ddot{f}(x_k)} = x_k - \frac{g(x_k)}{\dot{g}(x_k)}$$

*is a method for finding the solution to* $g(x) = 0$.



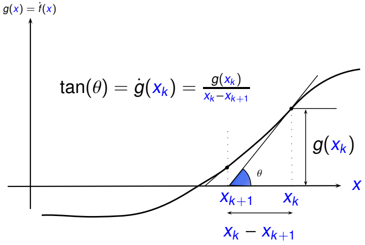

An interpretation of Newton's method.

# Newton's method

*A nice interpretation of the Newton's method is finding the zero of the derivative function $\dot{f}(x) = g(x)$. Indeed, the algorithm defined by the iterations*

$$x_{k+1} = x_k - \frac{\dot{f}(x_k)}{\ddot{f}(x_k)} = x_k - \frac{g(x_k)}{\dot{g}(x_k)}$$

*is a method for finding the solution to $g(x) = 0$.*

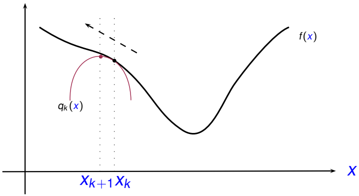



An interpretation of Newton's method.

# Newton's method

*Remarks:*

- *Convergence properties depend on the initial value*
- *In general the method converges locally*
- *For a quadratic function, the method converges in one step*
- *Requires calculating the first and second derivatives of* $f(\cdot)$
- *The method might not work if* $\ddot{f}(x) > 0$ *is not satisfied:*




An example where Newton's method diverges.

# Secant method

*Newton's method is an iterative method based on the use of the first and second derivative of* $f(\cdot)$

$$x_{k+1} = x_k - \frac{\dot{f}(x_k)}{\ddot{f}(x_k)}$$

*If the second derivative is not available, then we approximate it*

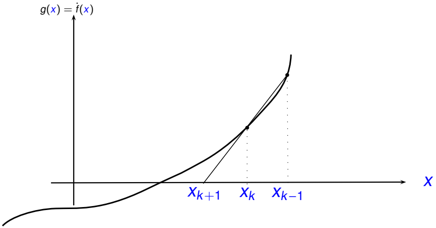
$$\ddot{f}(x_k) \approx \frac{\dot{f}(x_k) - \dot{f}(x_{k-1})}{x_k - x_{k-1}}$$

*and thus we have the following iterative method*

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{\dot{f}(x_k) - \dot{f}(x_{k-1})}\dot{f}(x_k)$$

# Secant method

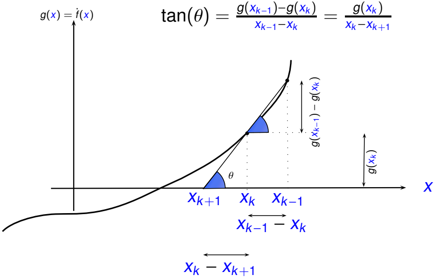
*Such as Newton's method, the secant method can be interpreted as finding the solution to* $g(x) = \dot{f}(x) = 0$.




An interpretation of the secant method.

# Secant method

*Such as Newton's method, the secant method can be interpreted as finding the solution to* $g(x) = \dot{f}(x) = 0$.




An interpretation of the secant method.

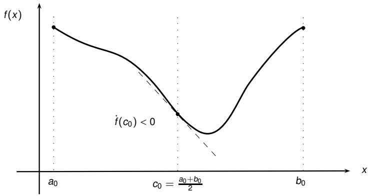
# Bisection method

*Consider the unimodal, continuously differentiable function $f : \mathbb{R} \to \mathbb{R}$, we study the following problem:*

$$\begin{array}{ll} \min\limits_{x} & f(x) \\ s.t. & x \in [a_0, b_0] \end{array}$$

*At each iteration we calculate $\dot{f}(c_0)$ (for $c_0 = \frac{a_0+b_0}{2}$) :*

- *if $\dot{f}(c_0) = 0$ then $c_0$ is a minimizer*
- *if $\dot{f}(c_0) < 0$ then the minimizer must be in $[c_0, b_0]$*
- *if $\dot{f}(c_0) > 0$ then the minimizer must be in $[a_0, c_0]$*
- *After N iteration the search range is reduced by $(0.5)^N$*



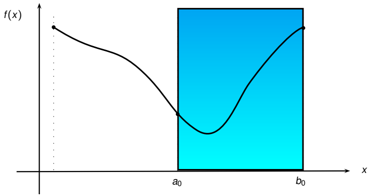
# Bisection method

*Consider the unimodal, continuously differentiable function $f : \mathbb{R} \to \mathbb{R}$, we study the following problem:*

$$\begin{array}{ll} \min\limits_{x} & f(x) \\ s.t. & x \in [a_0, b_0] \end{array}$$

*At each iteration we calculate $\dot{f}(c_0)$ (for $c_0 = \frac{a_0+b_0}{2}$) :*

- *if $\dot{f}(c_0) = 0$ then $c_0$ is a minimizer*
- *if $\dot{f}(c_0) < 0$ then the minimizer must be in $[c_0, b_0]$*
- *if $\dot{f}(c_0) > 0$ then the minimizer must be in $[a_0, c_0]$*
- *After N iteration the search range is reduced by $(0.5)^N$*



# Some Remarks

*One-dimensional search methods are an essential tool in multi-dimensional*

*We have studied some one-dimensional search methods that can be classified according to the available information*

- *Only $f(\cdot)$ is available : golden section and Fibonacci search*
- *$\dot{f}(\cdot)$ and $\ddot{f}(\cdot)$ are available : Newton's method*
- *Only $\dot{f}(\cdot)$ is available : secant method and bisection method*

*In general, using high order derivative information permits to have better performance*

# Some Remarks

*Notice that in one-dimensional search methods we either start with a domain of search and try to reduce its size, or consider the following algorithm :*

- *We start with one initial value* $\mathbf{x}_0$ *(or several ones)*
- *At each iteration we find the next candidate solution* $\mathbf{x}_{k+1}$ *based on* $\mathbf{x}_k$ *(or sometime based on* $\mathbf{x}_K, \mathbf{x}_{K-1}, ..$*)*
- *Based on a stopping criterion, the algorithm stops at iteration N, and we consider* $\mathbf{x}_N$ *to be a solution (local minimizer) with a certain precision*

***Iterative algorithms*** *are going to be considered also for multi-dimensional search methods (next lecture).*