Reducing Trials by Thinning-out in Skill Discovery

Hayato Kobayashi[1], Kohei Hatano[2],
Akira Ishino[1], and Ayumi Shinohara[1]
[1]Tohoku University, Sendai, Japan
[2]Kyushu University, Japan

DS 2007 @ Sendai, Japan

Contents

- Background
- Thinning-out for reducing trials
  - Max Gradient (MG) method
  - Gathering Differences (GD) method
- Performance evaluation by test functions
- Discovery of strong shots on virtual robots
- Conclusions and future work

Background

- For robots to function in the real world, they need abilities to acquire basic skills
  - e.g., walking, running, jumping, catching, throwing, **shooting**, hitting, kicking, swimming, flying, …

Procedure of each trial

1. Pick up a candidate
   (think up a new motion)
2. Try the candidate
   (perform the motion)
3. Evaluate the score
   (estimate the distance)
4. Restore the state
   (return the ball)

Acquisition of passing (shooting) skills on real robots
Passing is to exactly move a ball to the target position

Background

- For robots to function in the real world,
  they ne
  – e.g., w
    shoot , flying, …

Motion (Sequence of joint angles)
[-5.0, 5.0, -10.0, 5.0, 45.0, 60.0, 0.0, 10.0, 75.0, -50.0, 15.0, 105.0, -60.0, 15.0, 90.0, -7, -15.0, 5.0, 25.0, 40.0, 60.0, -10.0, 0.0,……….]

Score is 800

Procedure of each trial

1. Pick up a candidate
   (think up a new motion)
2. Try the candidate
   (perform the motion)
3. Evaluate the score
   (estimate the distance)
4. Restore the state
   (return the ball)

Acquisition of passing (shooting) skills on real robots
Passing is to exactly move a ball to the target position

Main Difficulty in Skill Discovery

- Each trial consumes much time and costs
  - Each trial needs more than 30 seconds (average)
  - Robots can be broken easily for many trials

Example: if we use 1,000 generations by GA on real robots

Trials = 1,000 (generations)

× 10 (if we use 10 population size)

× 5 (if we use the average of 5 trials for noise reduction)

= **50,000** (trials)

Time = 50,000 (trials) × 30 (seconds)

= 1,500,000 (seconds)

= 416 (hours)

= **17** (days) ＞ **11** (days), Sleepless World Record

(Robot's battery is dead in only 30 minutes)

Contents

- Background
- Thinning-out for reducing trials
  - Max Gradient (MG) method
  - Gathering Differences (GD) method
- Performance evaluation by test functions
- Discovery of strong shots on virtual robots
- Conclusions and future work

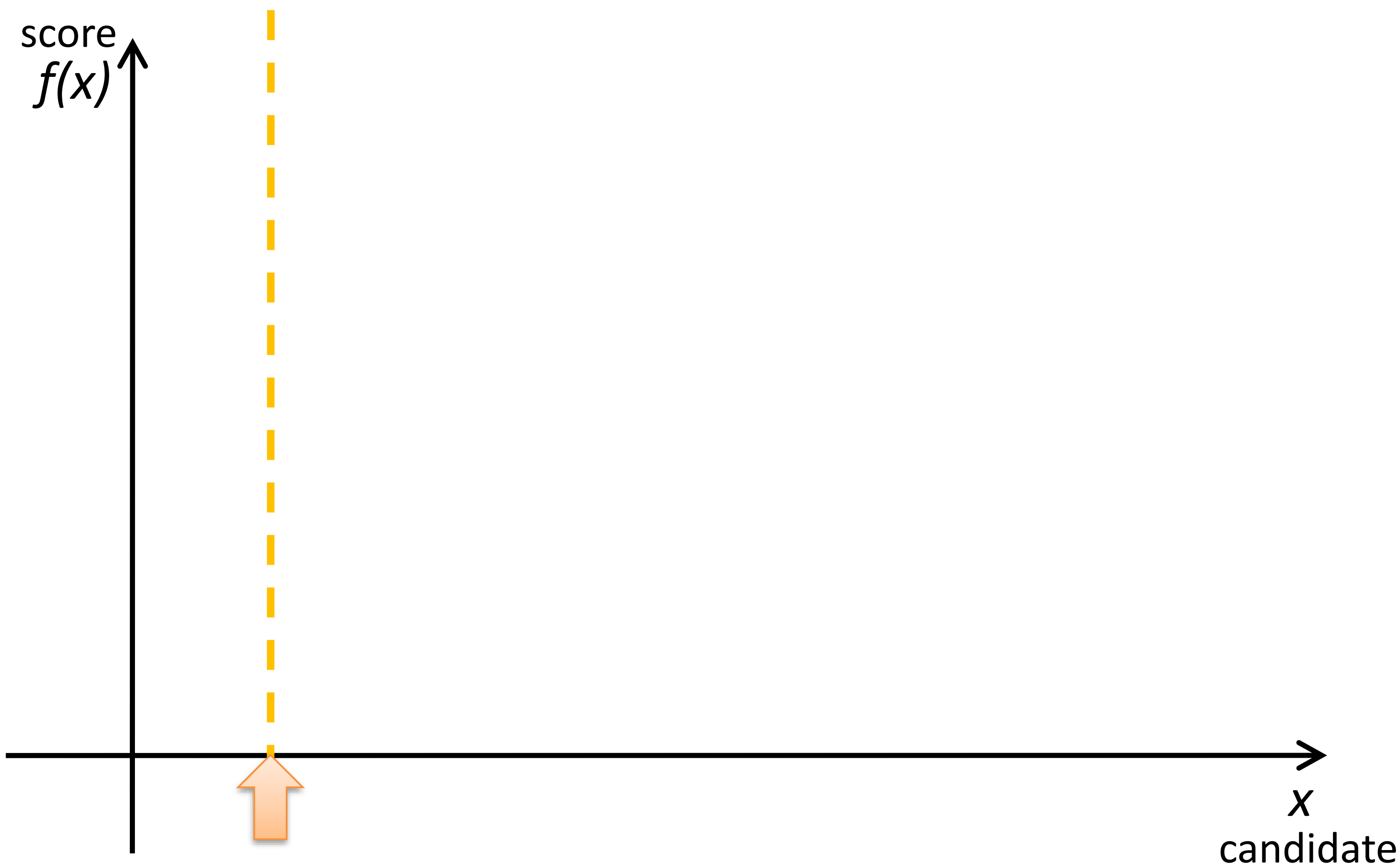
Thinning-out

- To skip over unnecessary trials
  - The same concept as “pruning” in search trees
- By Inferring the upperbound of each score
  - We utilize the smoothness of score functions

- Related work
  - Acceleration of search methods
  - Memory based learning
    - Memory-based fitness evaluation GA [Sano et al., 2000]
    - Locally weighted regression [Schaal and Atkeson, 1994]
  - Typical methods by inferring scores directly
    - Approximation of scores using kriging interpolation [Ratle, 1998]

Basic idea of thinning-out

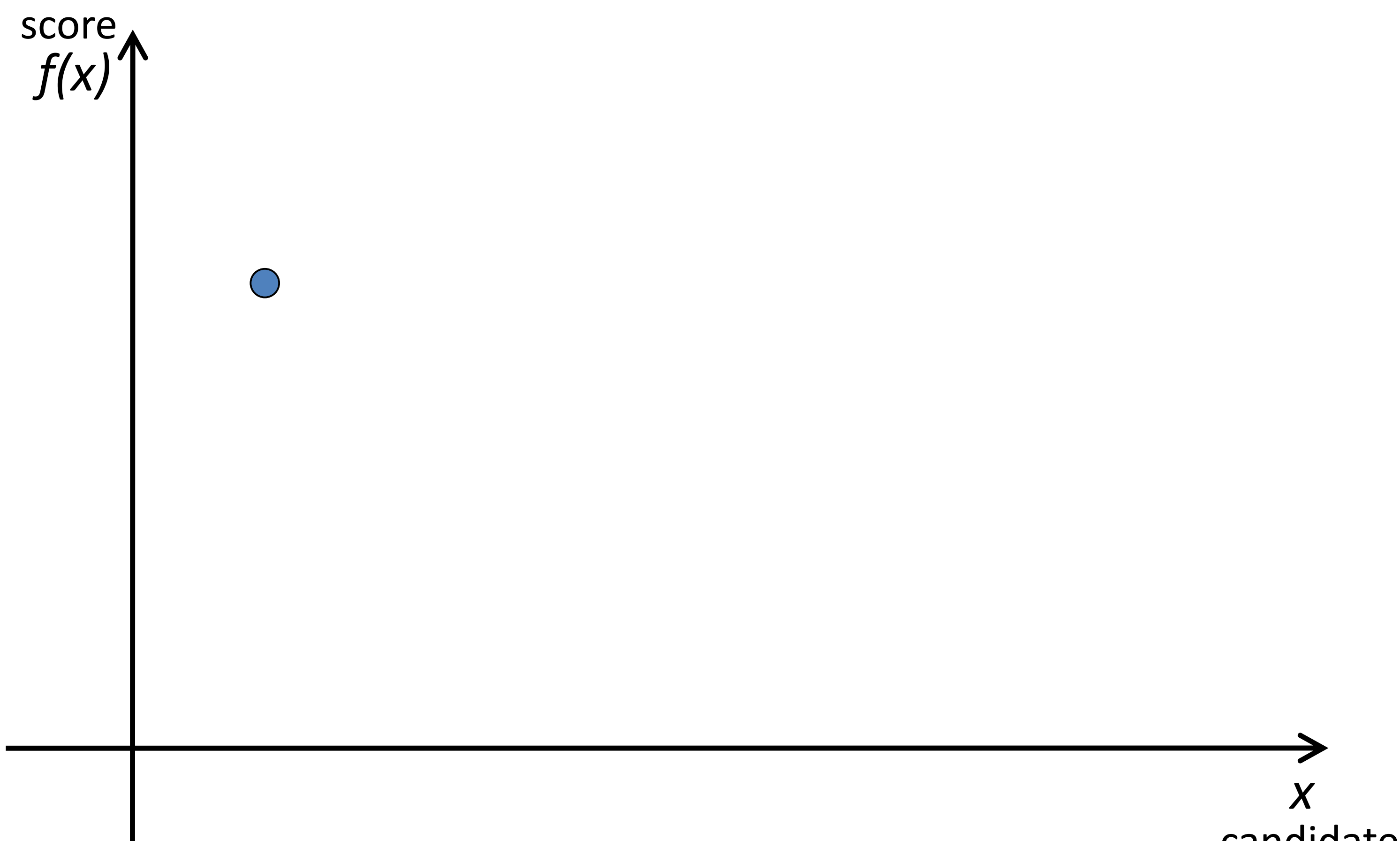
Unnecessary trials = duplicated candidates and unpromising candidates



Example: Maximization of unknown score function in one dimension

Basic idea of thinning-out

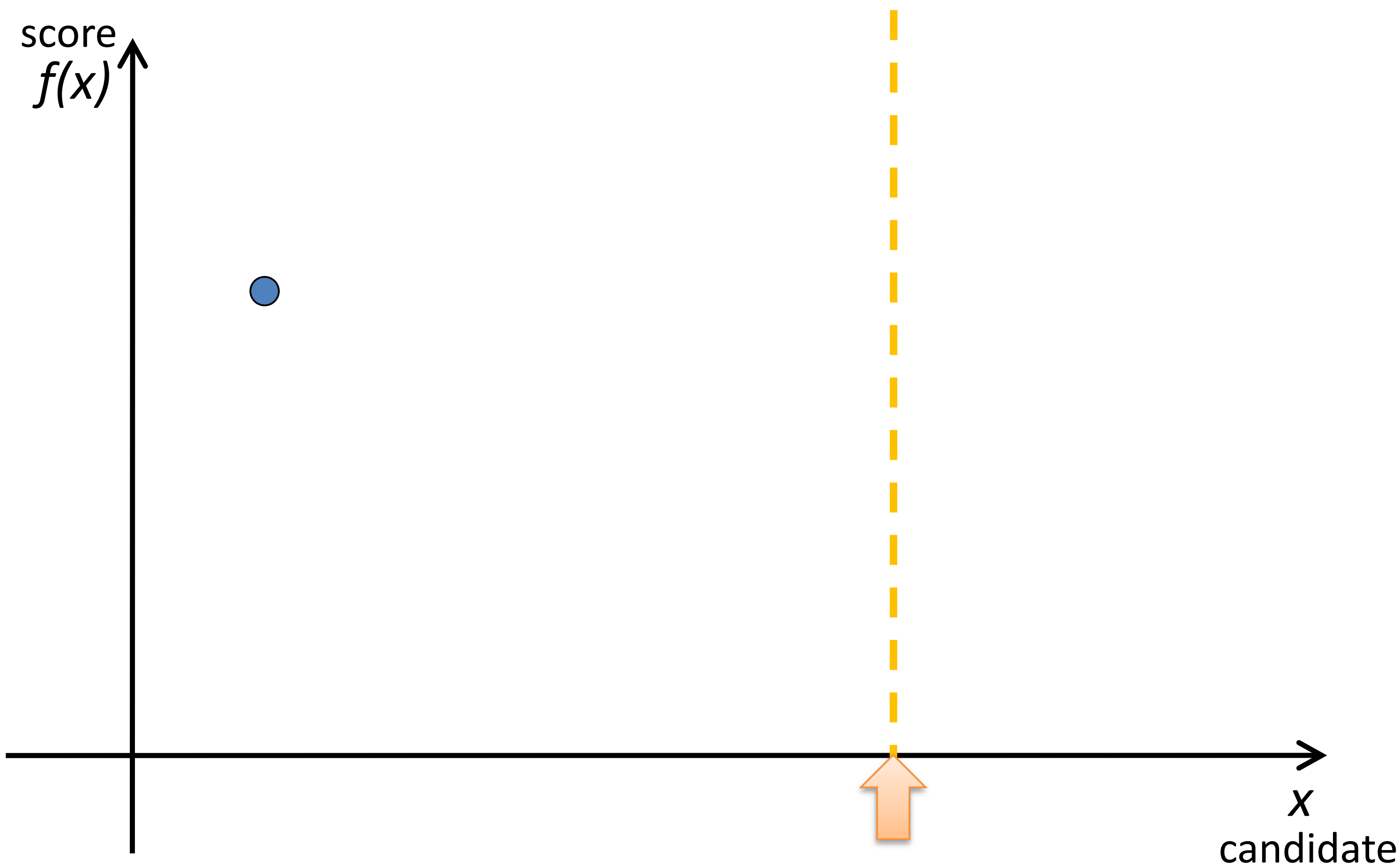
Unnecessary trials = duplicated candidates and unpromising candidates



Example: Maximization of unknown score function in one dimension

Basic idea of thinning-out

Unnecessary trials = duplicated candidates and unpromising candidates



Example: Maximization of unknown score function in one dimension

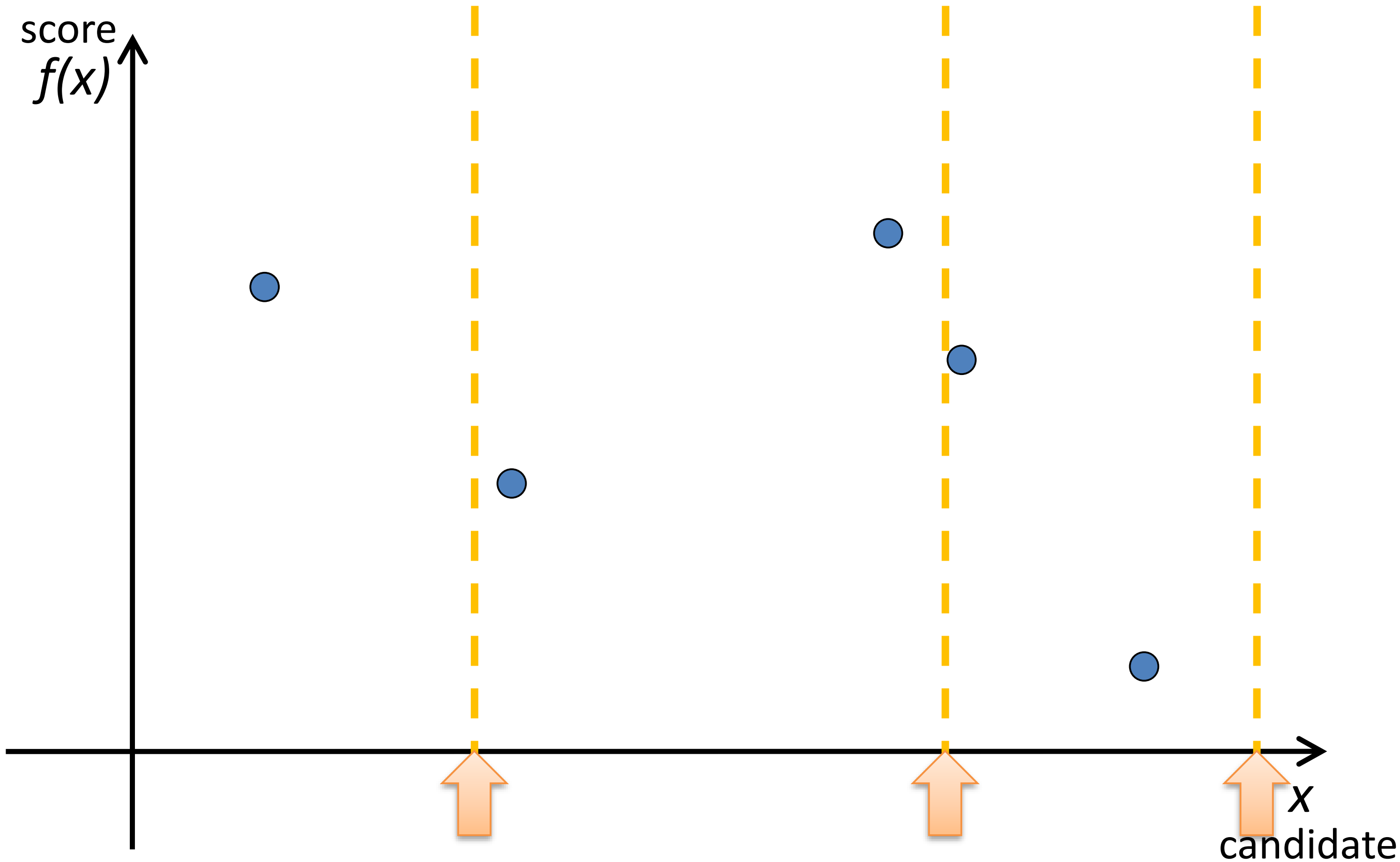
Basic idea of thinning-out

Unnecessary trials = duplicated candidates and unpromising candidates

Example: Maximization of unknown score function in one dimension

Basic idea of thinning-out

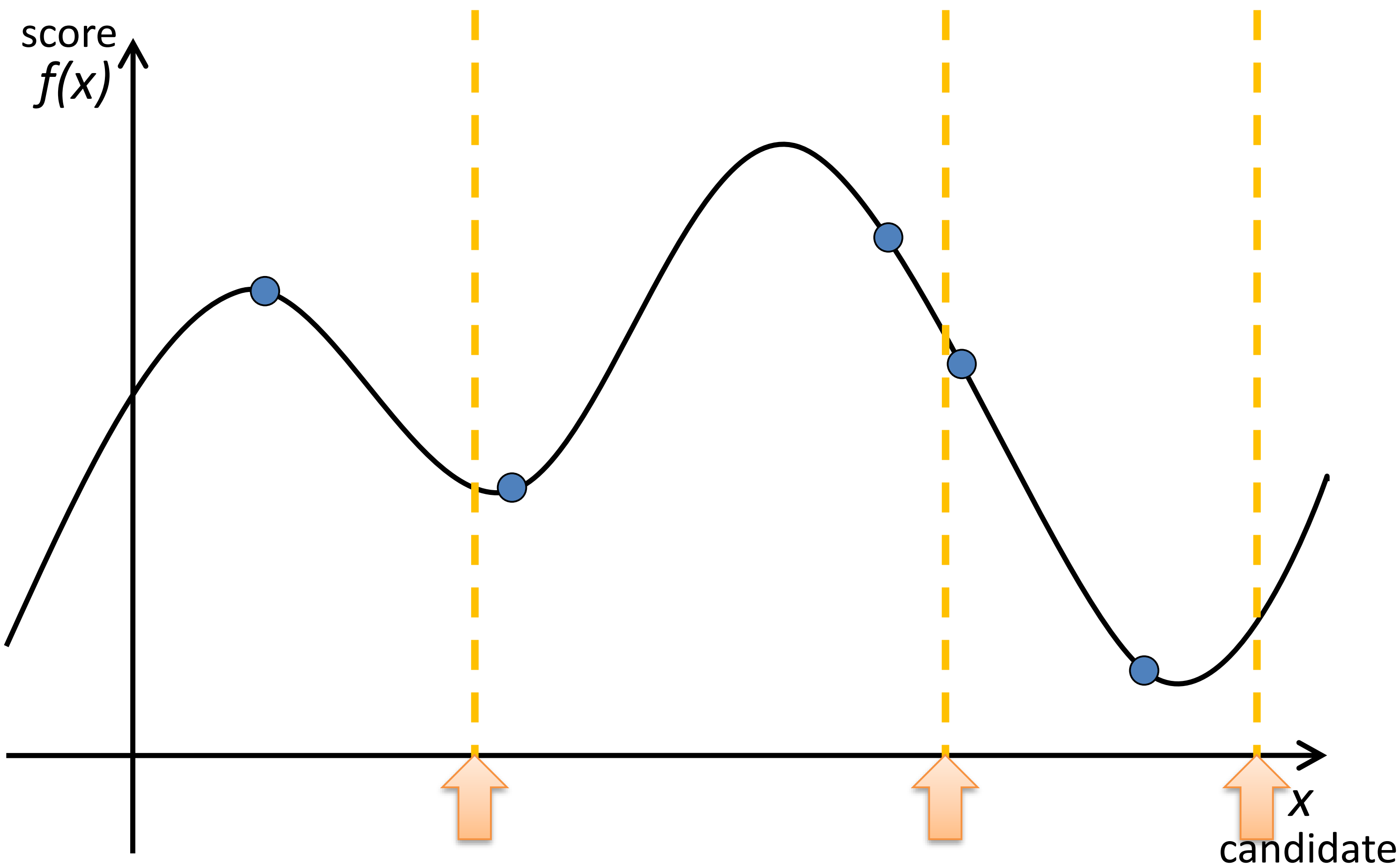
Unnecessary trials = duplicated candidates and unpromising candidates



Example: Maximization of unknown score function in one dimension

Basic idea of thinning-out

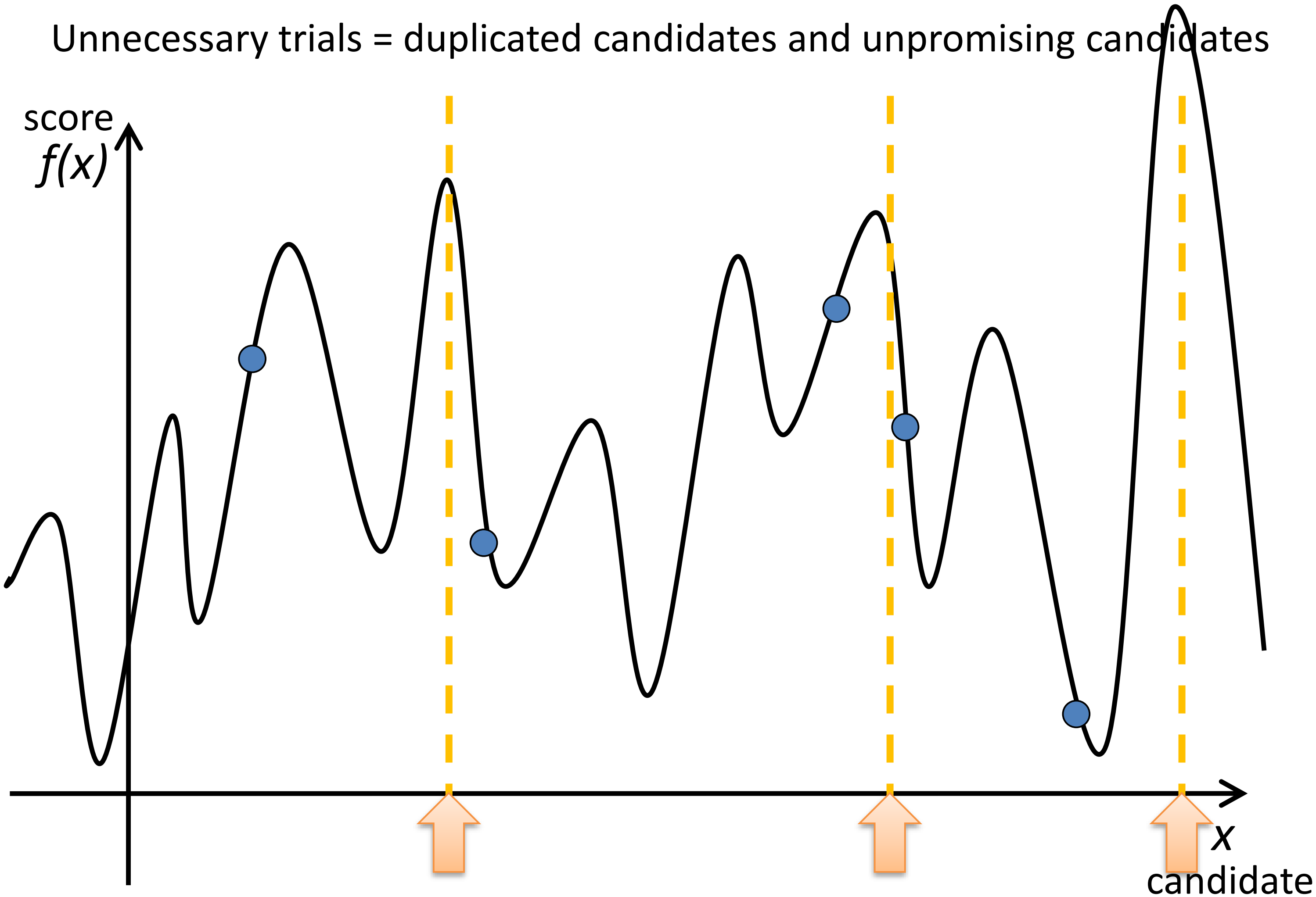
Unnecessary trials = duplicated candidates and unpromising candidates



Example: Maximization of unknown score function in one dimension

Basic idea of thinning-out

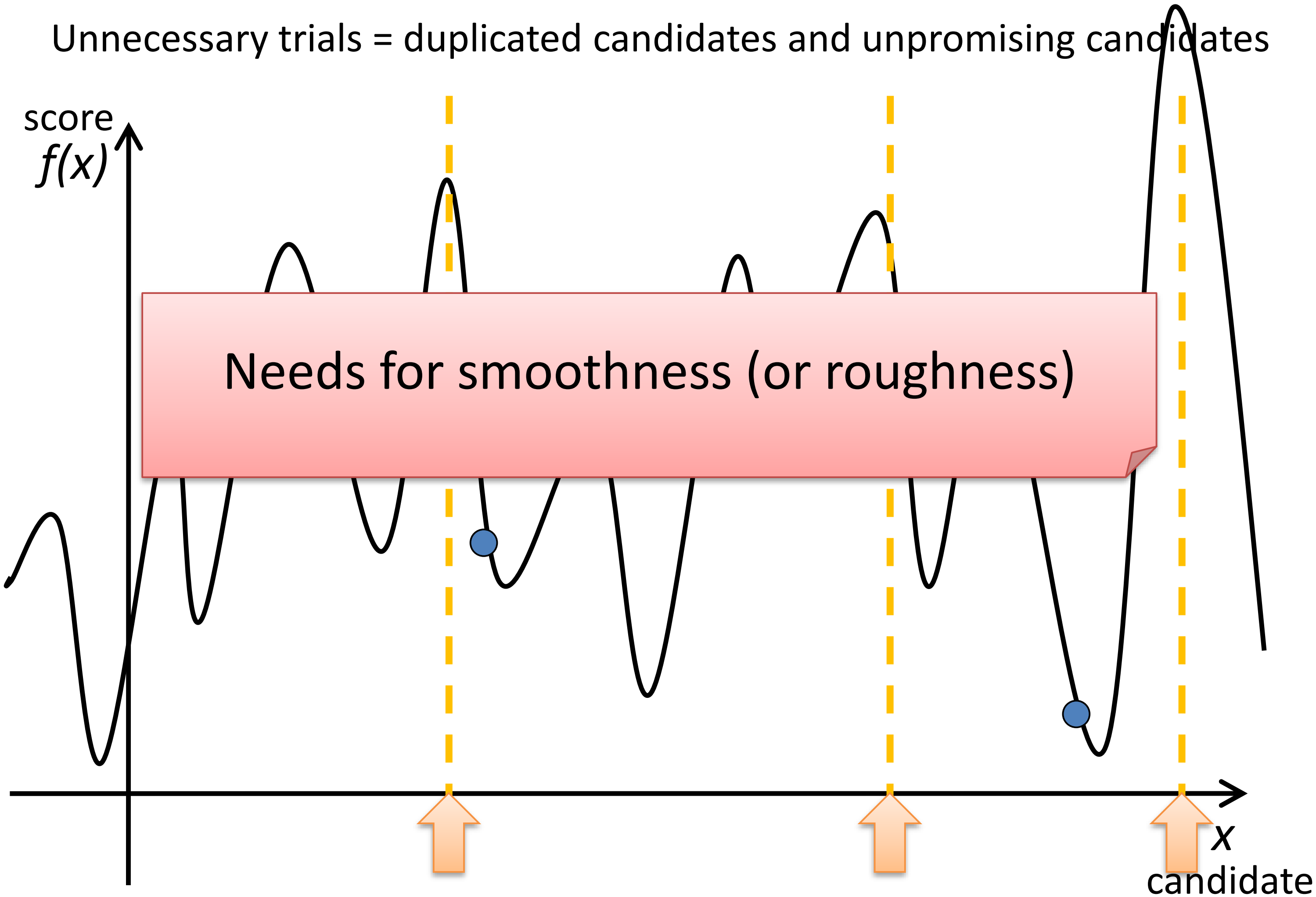
Unnecessary trials = duplicated candidates and unpromising candidates



Example: Maximization of unknown score function in one dimension

Basic idea of thinning-out

Unnecessary trials = duplicated candidates and unpromising candidates



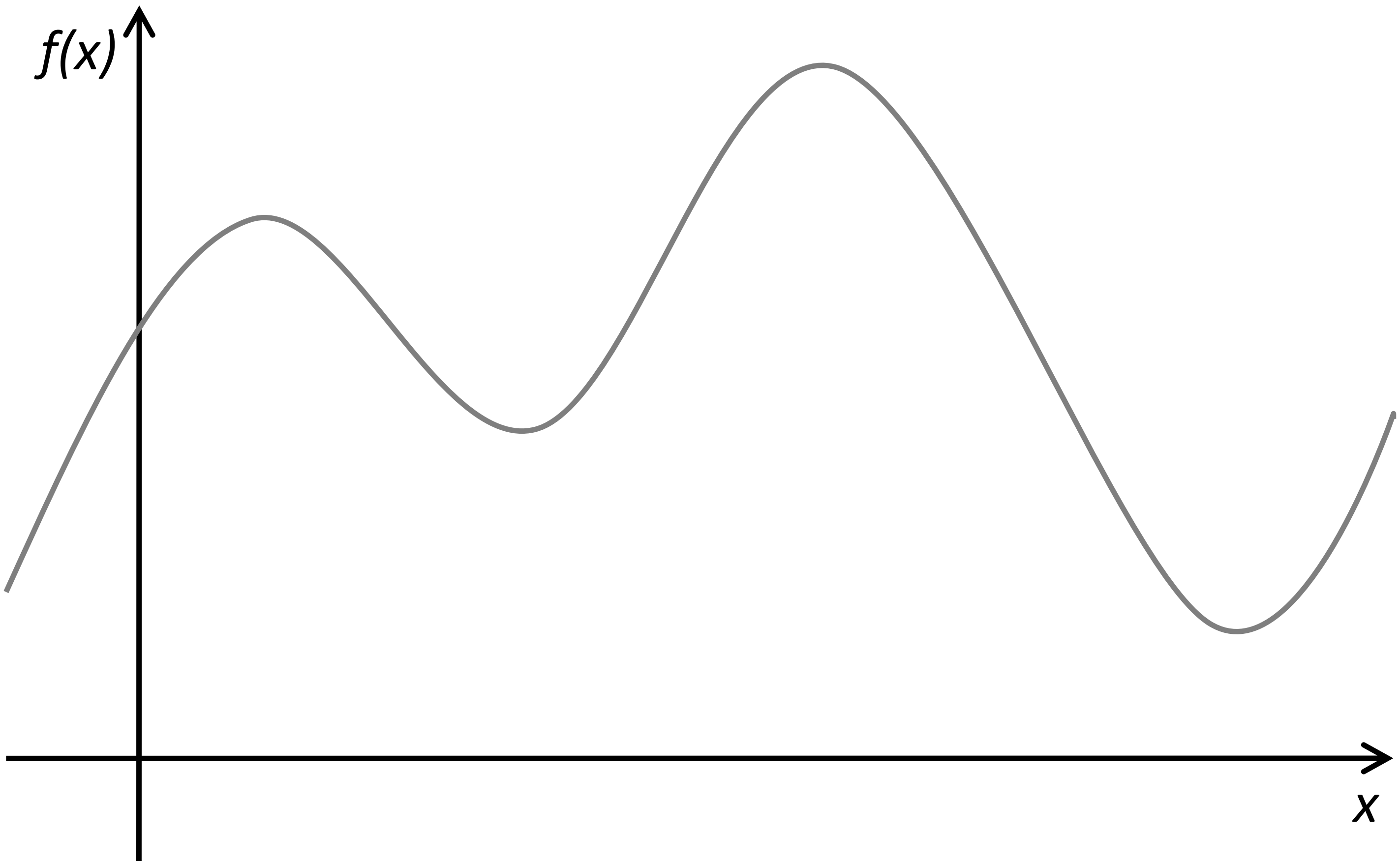
Example: Maximization of unknown score function in one dimension

Lipschitz condition

Lipschitz condition

$$\exists c \in \boldsymbol{R}\ \forall x_1, x_2 \in X\ \left| f(x_1) - f(x_2) \right| \le c \cdot d(x_1, x_2)$$

X: search space
f: score function
d: metric of *X*

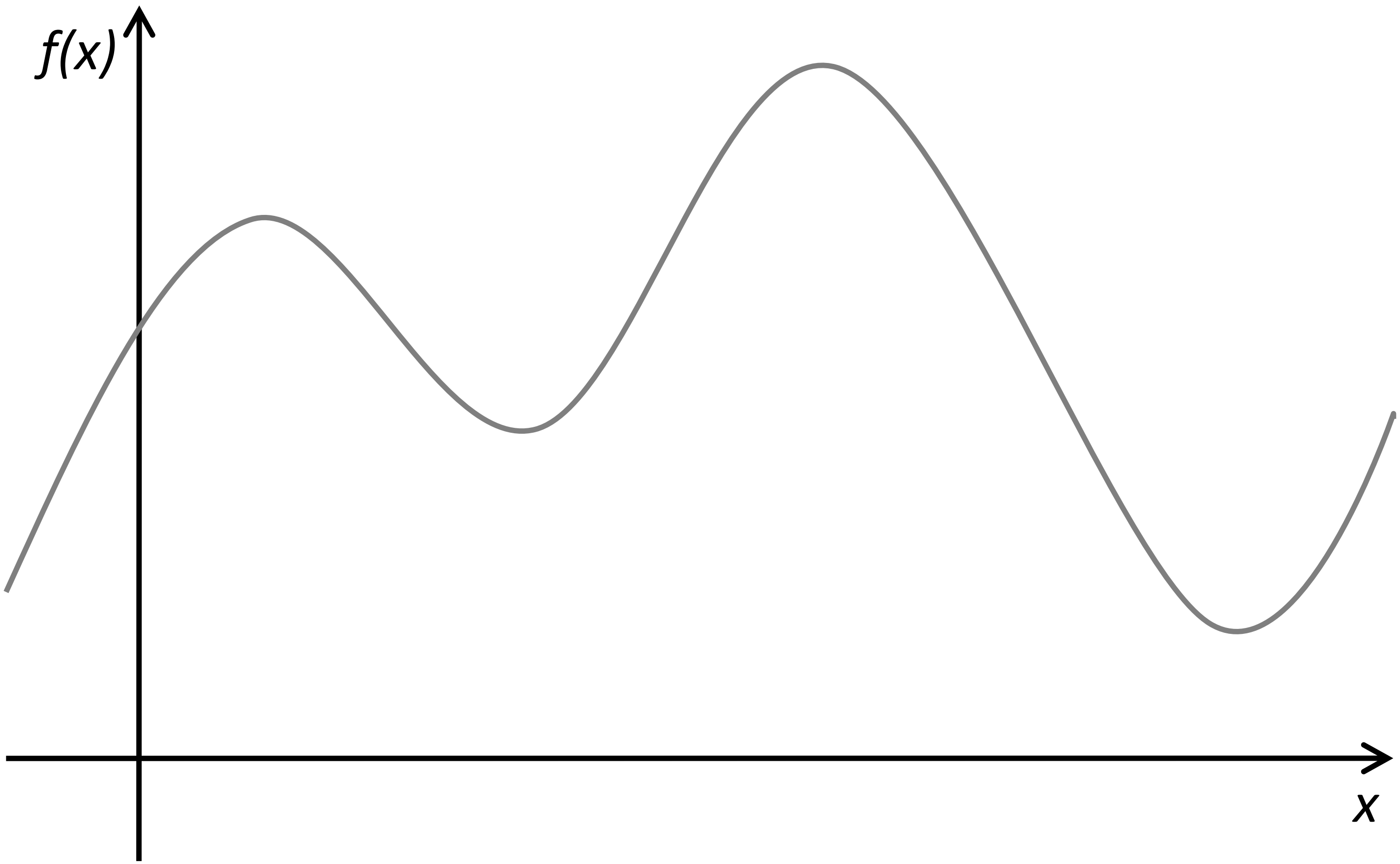


$$f(x_1) - c \cdot d(x_1, x_2) \le f(x_2) \le f(x_1) + c \cdot d(x_1, x_2)$$

Lipschitz condition

$$\exists c \in \boldsymbol{R}\ \forall x_1, x_2 \in X\ \left| f(x_1) - f(x_2) \right| \le c \cdot d(x_1, x_2)$$

X: search space
f: score function
d: metric of *X*

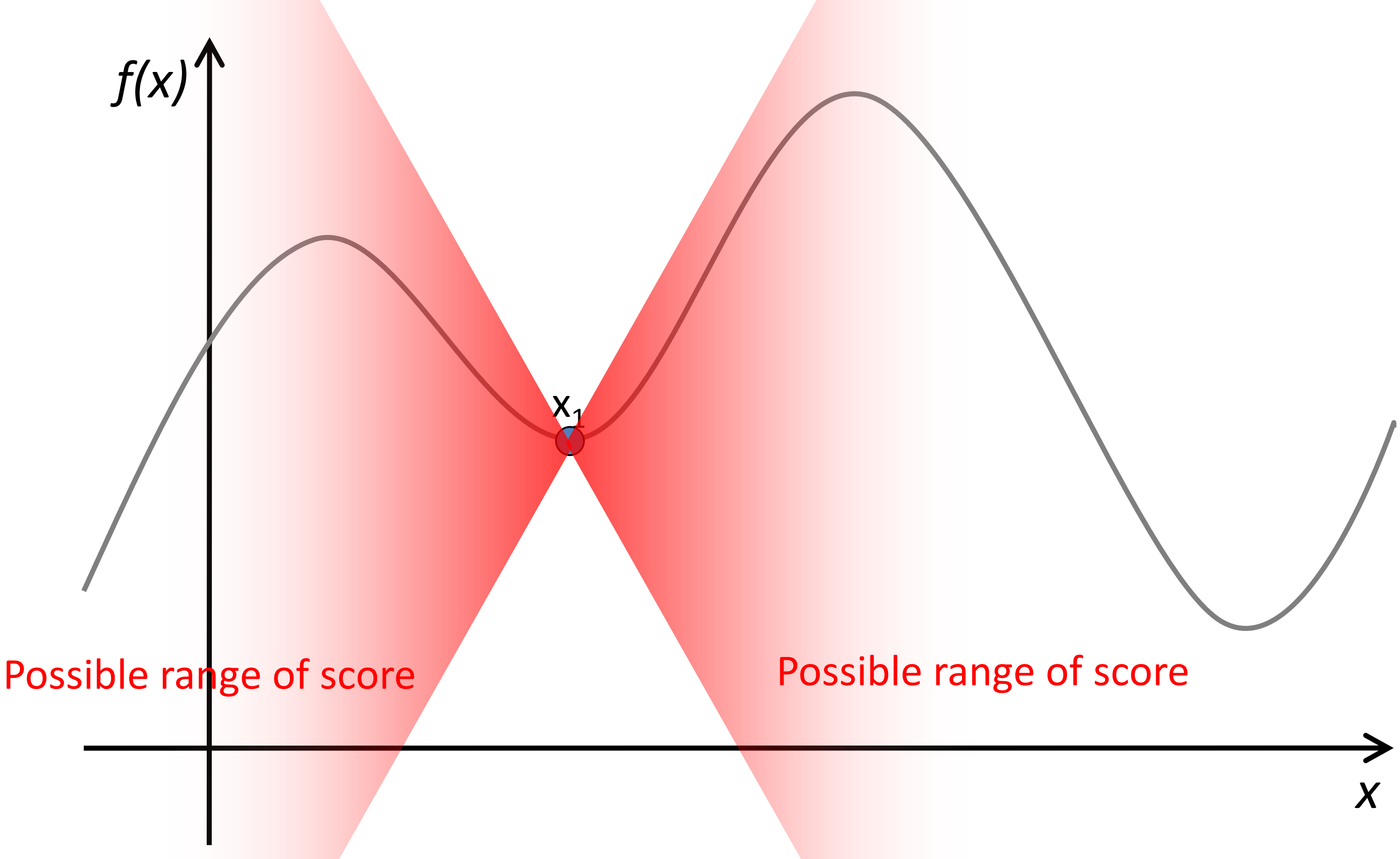


$$f(x_1) - c \cdot d(x_1, x_2) \le f(x_2) \le f(x_1) + c \cdot d(x_1, x_2)$$

Lipschitz condition

$$\exists c \in \boldsymbol{R}\ \forall x_1, x_2 \in X\ \left| f(x_1) - f(x_2) \right| \le c \cdot d(x_1, x_2)$$

X: search space
f: score function
d: metric of *X*

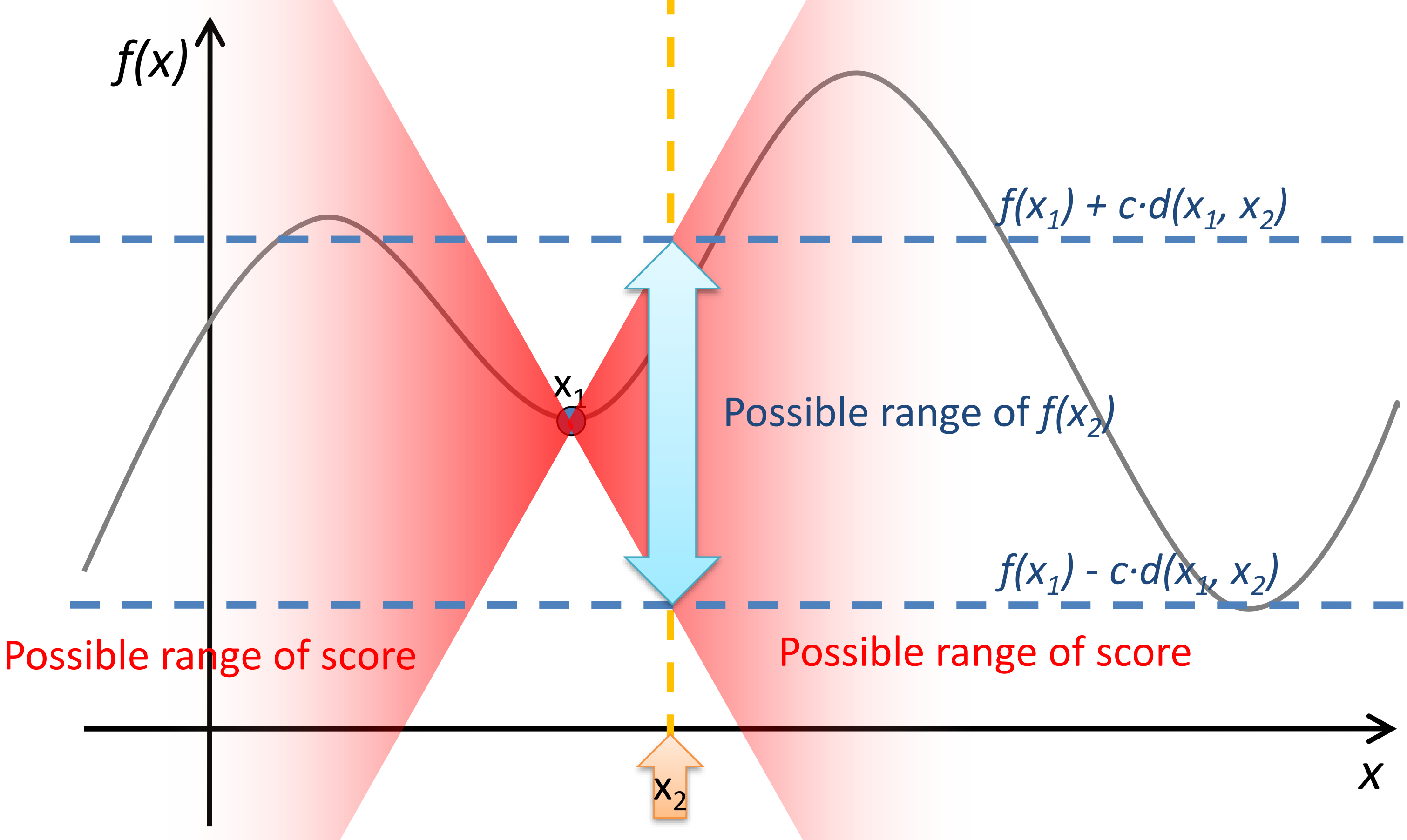


$$f(x_1) - c \cdot d(x_1, x_2) \leq f(x_2) \leq f(x_1) + c \cdot d(x_1, x_2)$$

Lipschitz condition

$$\exists c \in \boldsymbol{R}\ \forall x_1, x_2 \in X\ \left| f(x_1) - f(x_2) \right| \leq c \cdot d(x_1, x_2)$$

X: search space
f: score function
d: metric of *X*



Extension of Lipschitz condition

Lipschitz condition

$$\exists c \in \boldsymbol{R}\ \forall x_1, x_2 \in X\ \left| f(x_1) - f(x_2) \right| \le c \cdot d(x_1, x_2)$$

f is said to be c-Lipschitz continuous

c is said to be a Lipschitz constant

extend

Lipschitz condition

$$\exists g : \boldsymbol{R} \to \boldsymbol{R}\ \forall x_1, x_2 \in X\ \left| f(x_1) - f(x_2) \right| \le g(d(x_1, x_2))$$

f is said to be g-Lipschitz continuous

g is said to be a Lipschitz function

Thinning-out condition

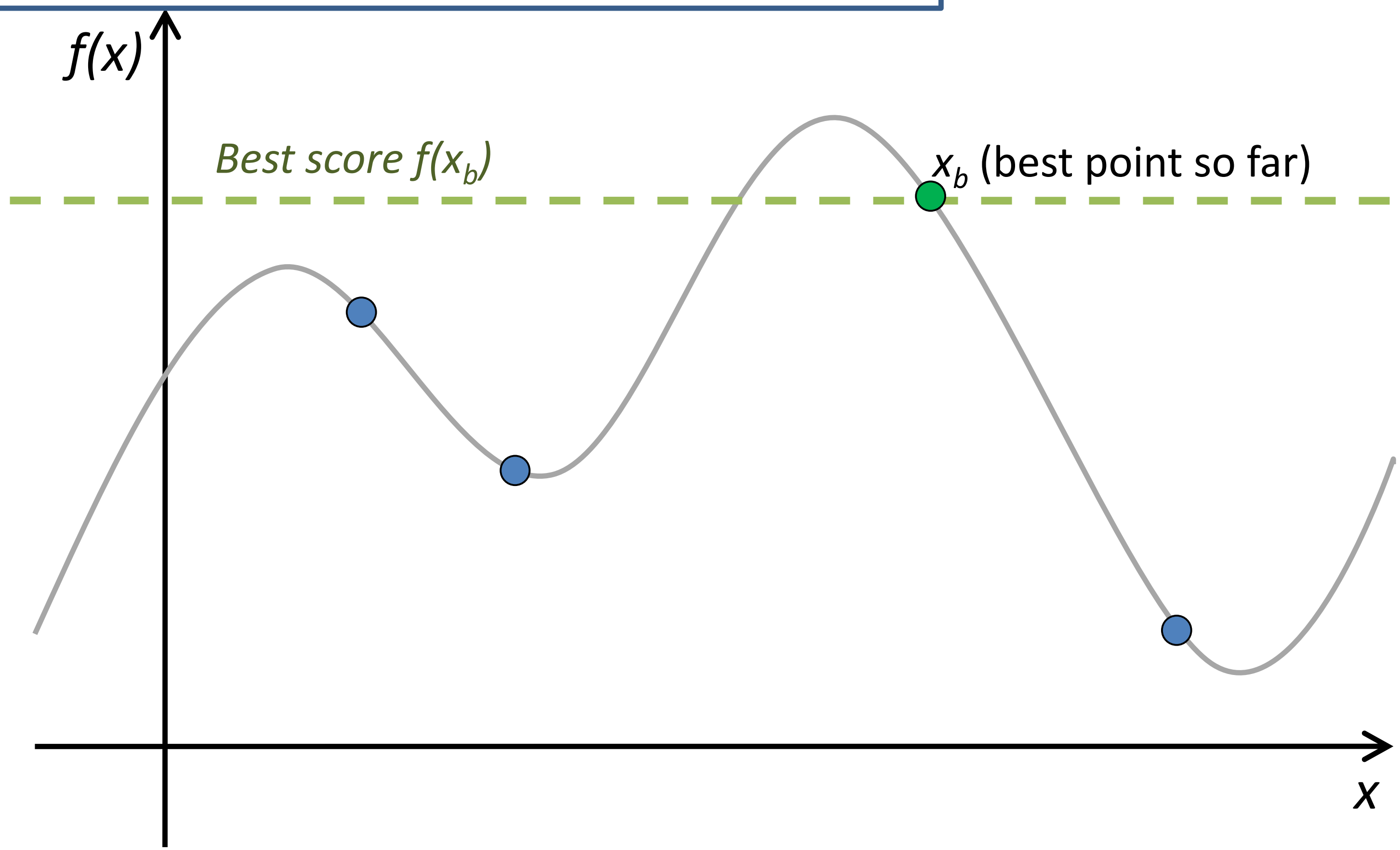
Thinning-out condition

$$\underline{f(x_n) + g(d(x_c, x_n))} \leq f(x_b)$$

The upperbound of the score range of x_c

$$\left(\Rightarrow f(x_c) \leq f(x_b) \right)$$

X: search space
f: score function,
g-Lipschitz continuous
d: metric of *X*



Thinning-out condition

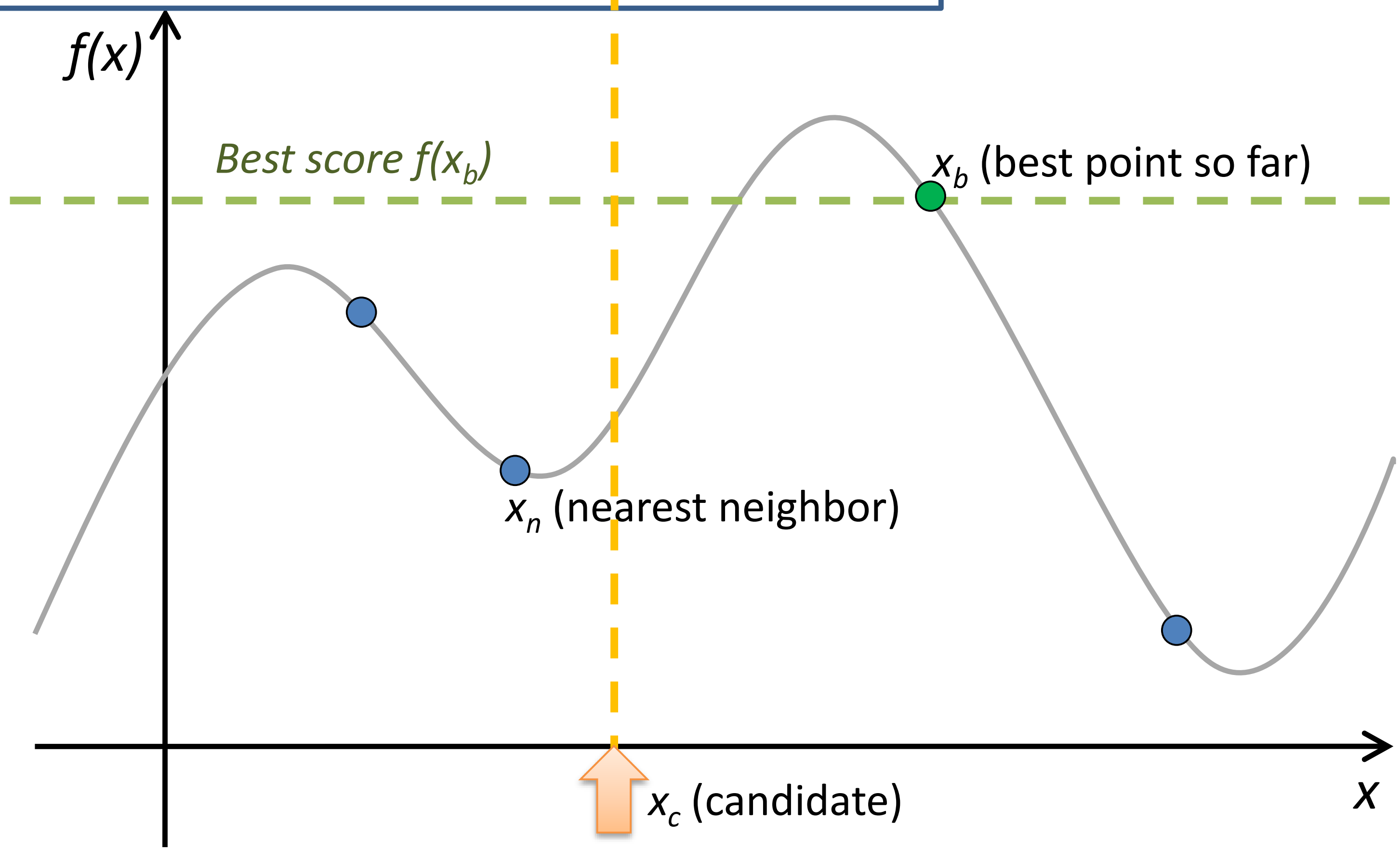
Thinning-out condition

$$\underline{f(x_n) + g(d(x_c, x_n))} \leq f(x_b)$$

The upperbound of the score range of x_c

$$\left(\Rightarrow f(x_c) \leq f(x_b)\right)$$

X: search space
f: score function, *g*-Lipschitz continuous
d: metric of *X*



Thinning-out condition

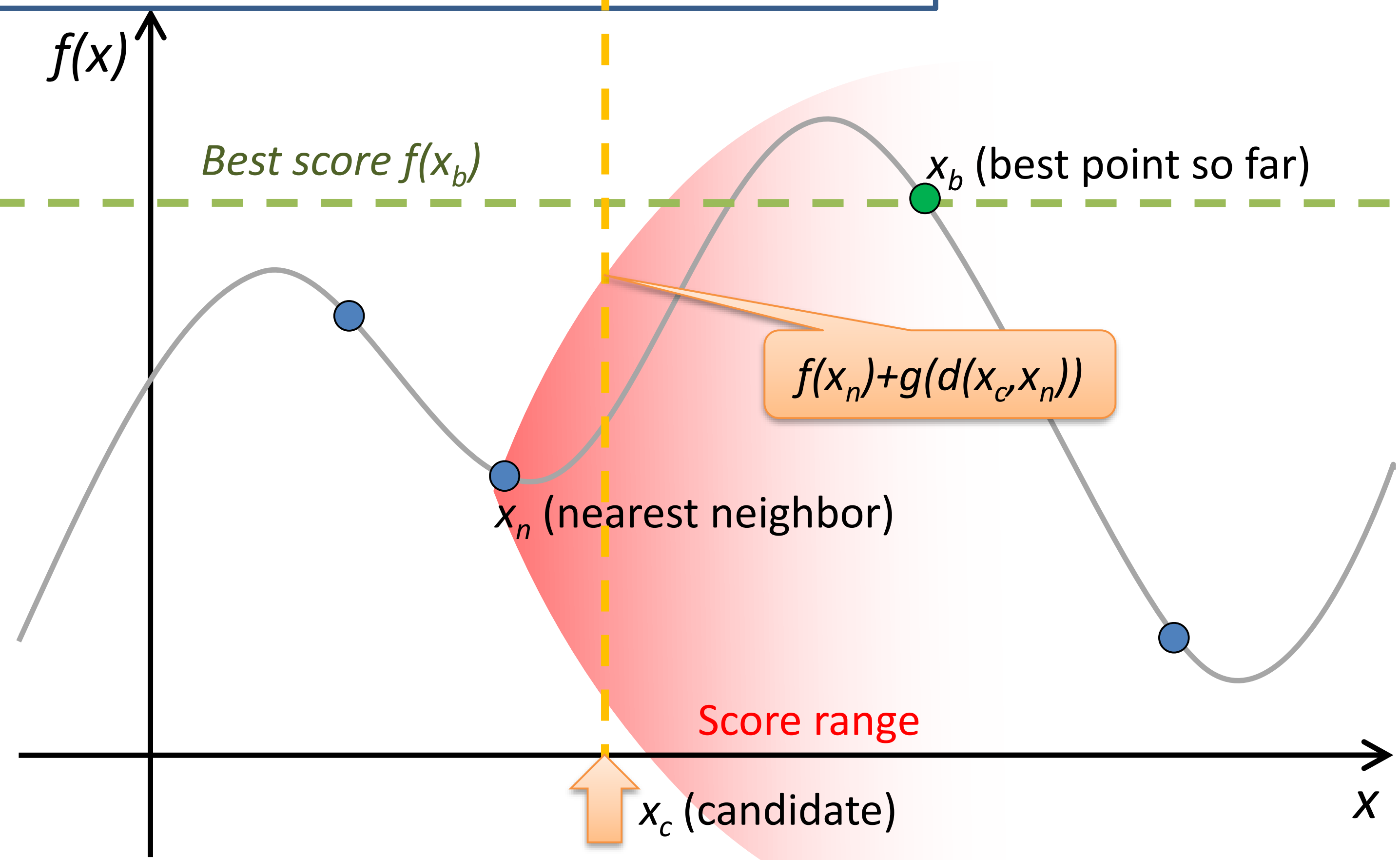
Thinning-out condition

$$\underline{f(x_n) + g(d(x_c, x_n))} \leq f(x_b)$$

The upperbound of the score range of x_c

$$\left(\Rightarrow f(x_c) \leq f(x_b)\right)$$

X: search space
f: score function,
g-Lipschitz continuous
d: metric of *X*



Thinning-out condition

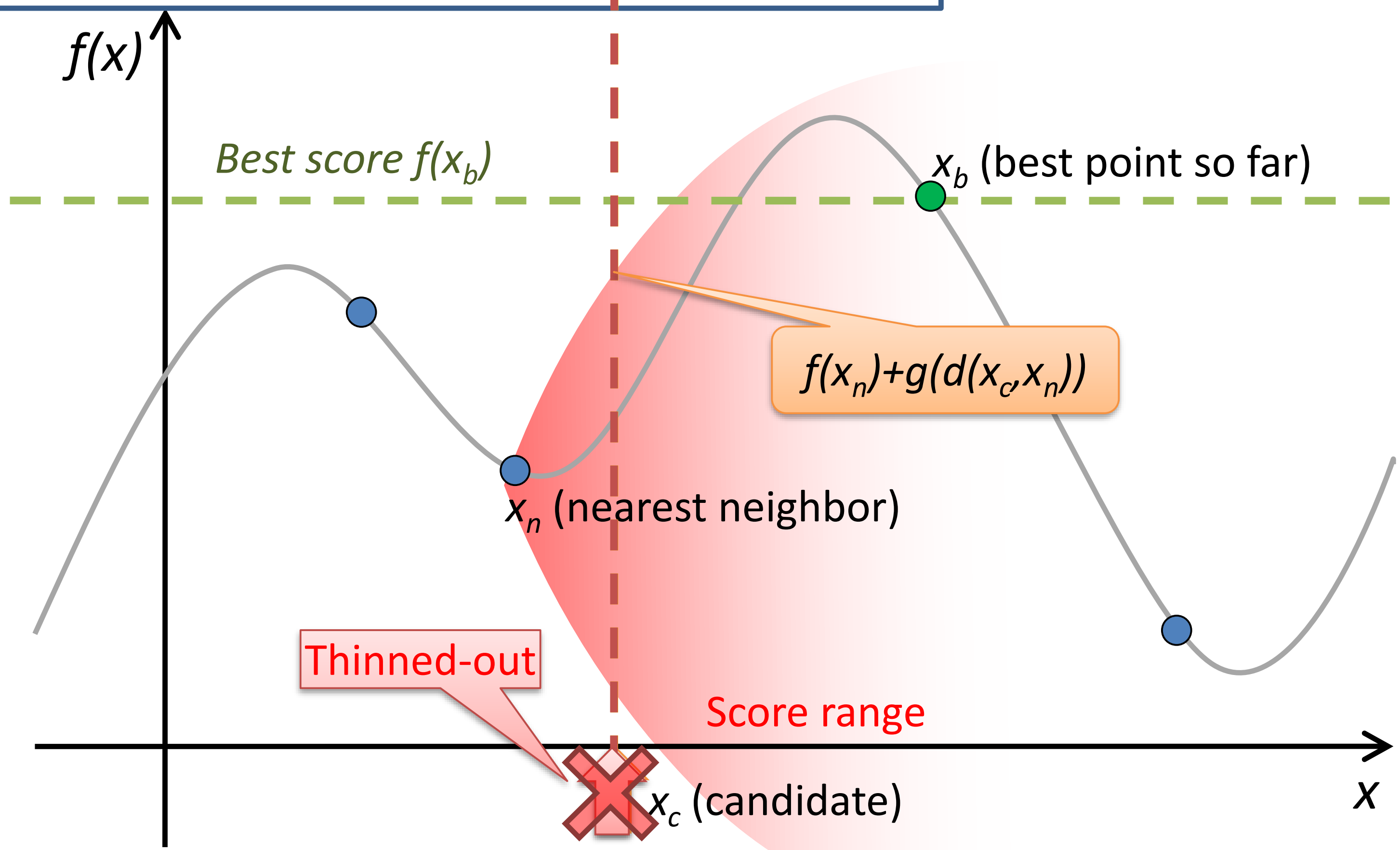
Thinning-out condition

$$f(x_n) + g(d(x_c, x_n)) \leq f(x_b)$$

The upperbound of the score range of x_c

$$\left(\Rightarrow f(x_c) \leq f(x_b)\right)$$

X: search space
f: score function,
g-Lipschitz continuous
d: metric of *X*



Thinning-out condition

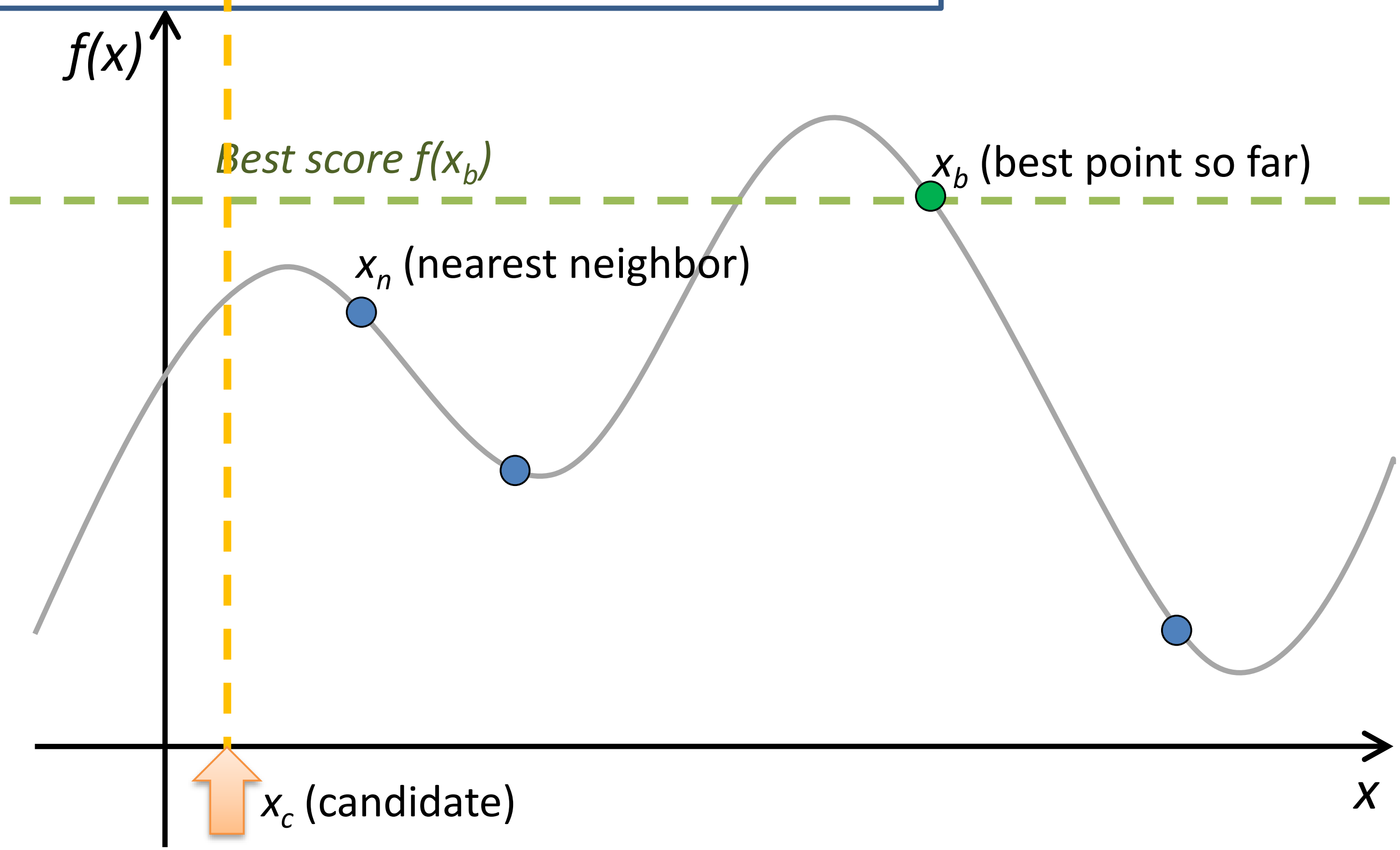
Thinning-out condition

$$f(x_n) + g(d(x_c, x_n)) \leq f(x_b)$$

The upperbound of the score range of x_c

$$\left(\Rightarrow f(x_c) \leq f(x_b)\right)$$

X: search space
f: score function, g-Lipschitz continuous
d: metric of X



Thinning-out condition

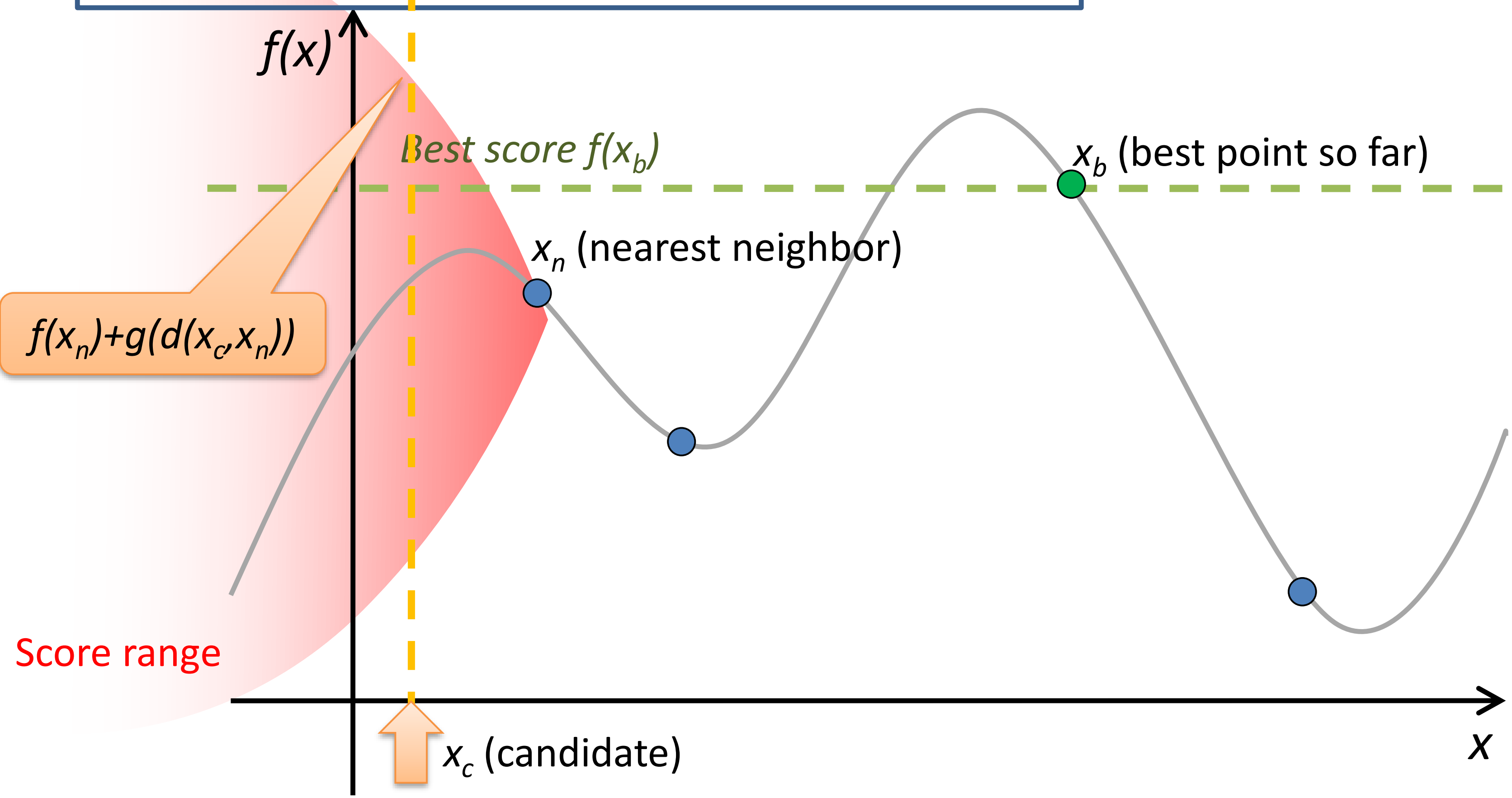
Thinning-out condition

$$\underline{f(x_n) + g(d(x_c, x_n))} \le f(x_b)$$

The upperbound of the score range of x_c

$$\left(\Rightarrow f(x_c) \le f(x_b)\right)$$

X: search space
f: score function, g-Lipschitz continuous
d: metric of X



Thinning-out condition

Thinning-out condition

$$f(x_n) + g(d(x_c, x_n)) \leq f(x_b)$$

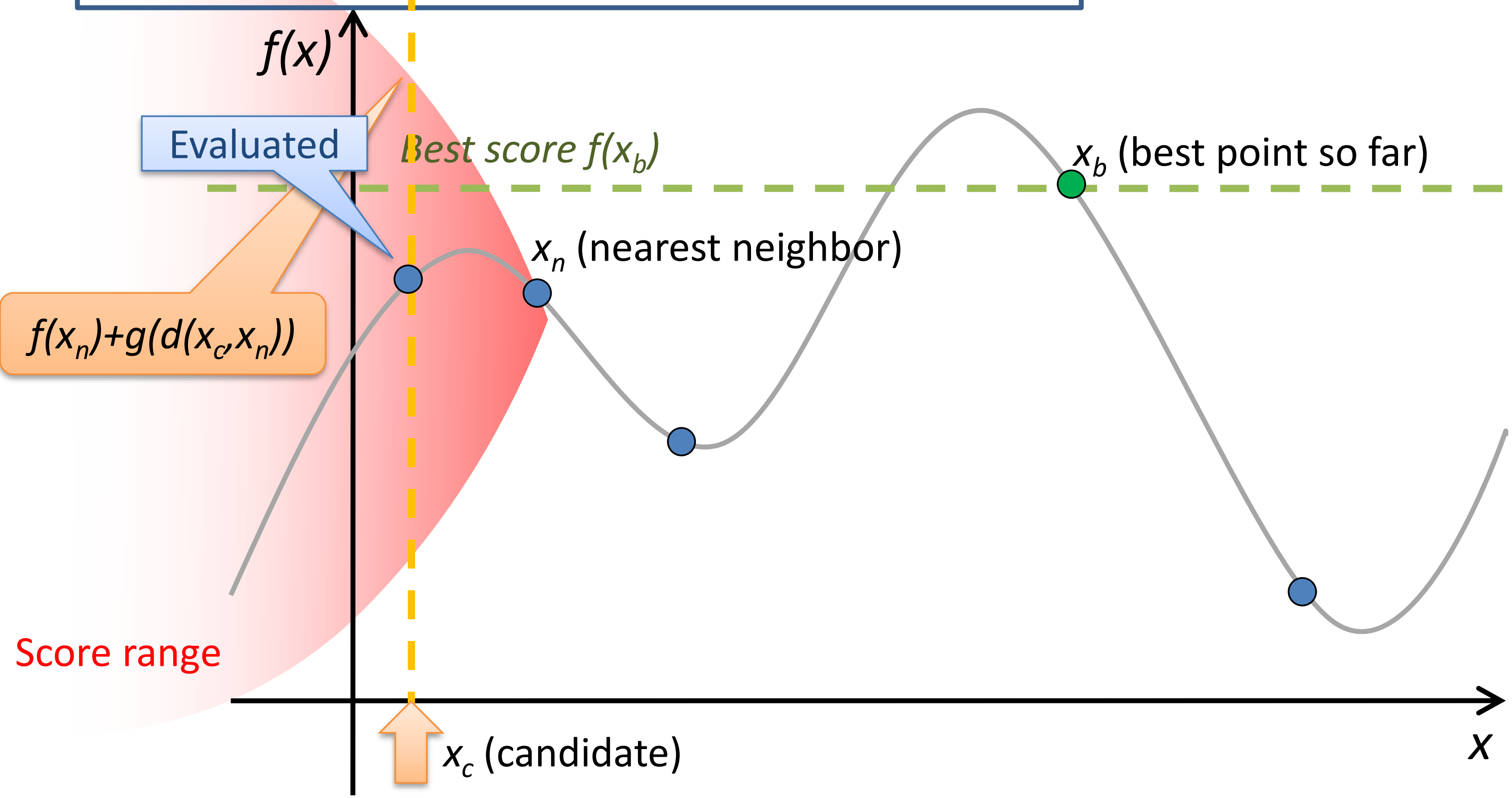
The upperbound of the score range of x_c

$$\left(\Rightarrow f(x_c) \leq f(x_b)\right)$$

X: search space

f: score function, *g*-Lipschitz continuous

d: metric of *X*

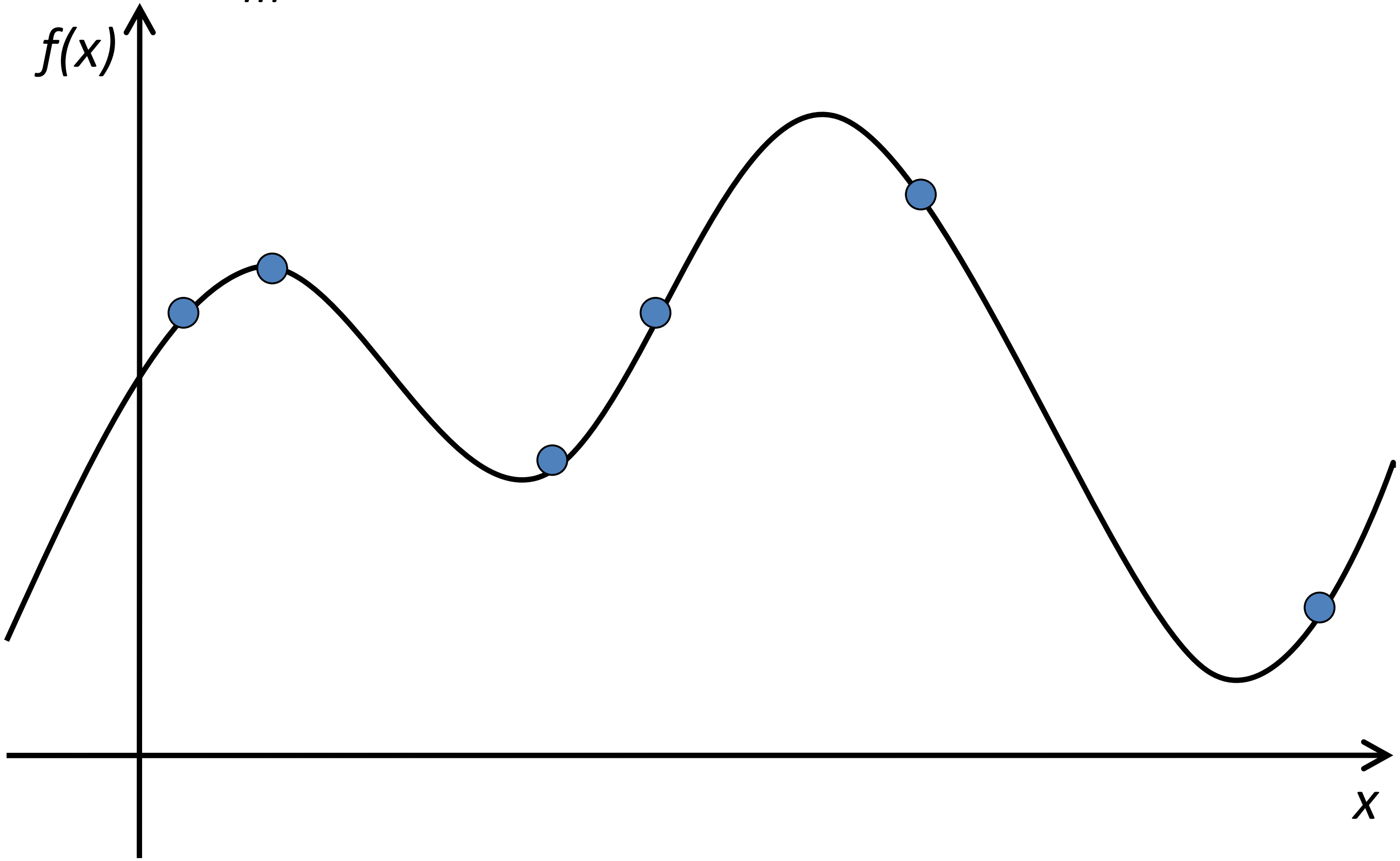


Inferring methods of Lipschitz function

- Max Gradient (MG)
  - Naïve method
  - Thin-out correctly
- Gathering Differences (GD)
  - Heuristics method
  - Thin-out a lot

Max Gradient (MG)

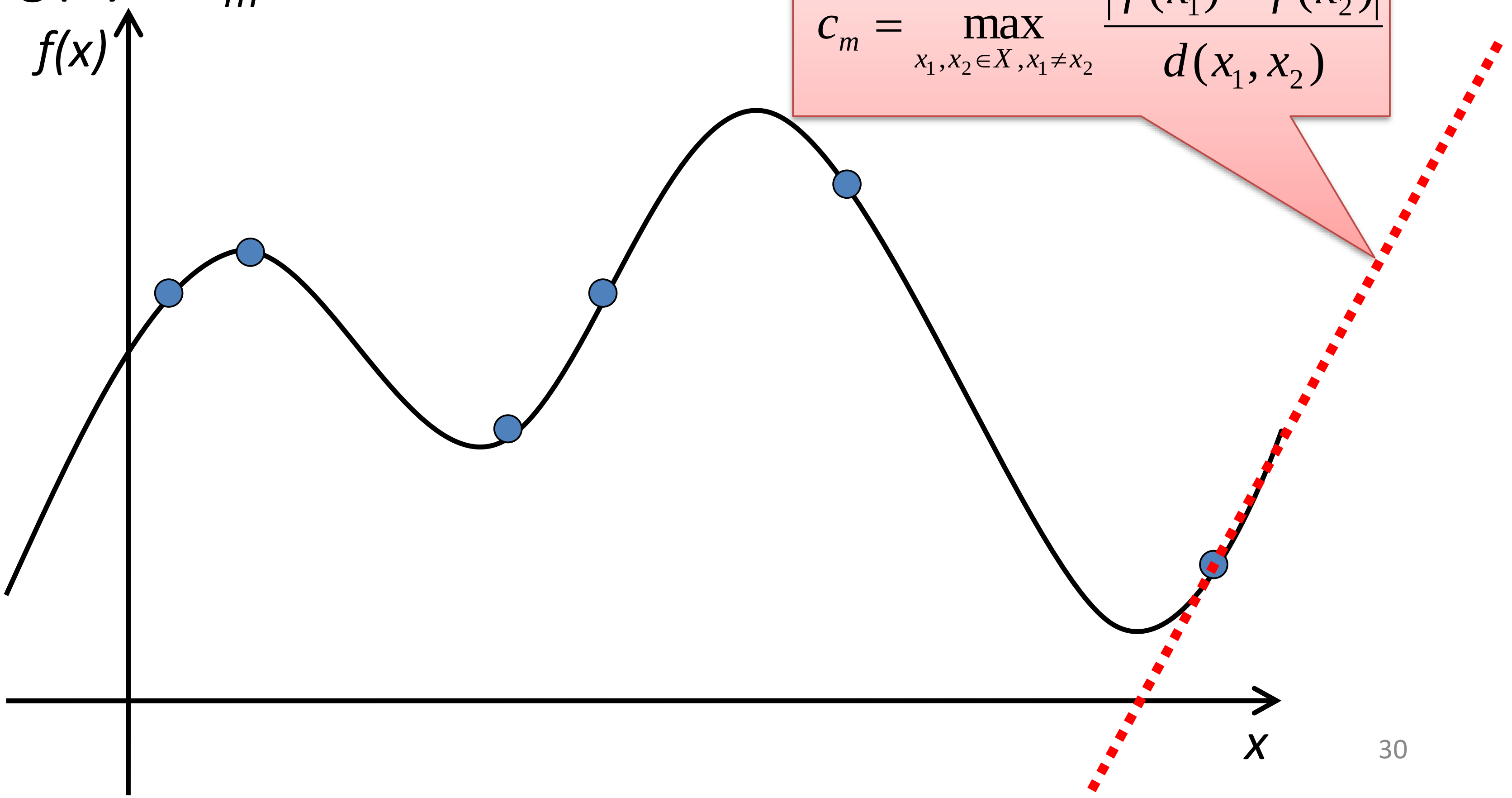
- Utilize the maximum gradient
  - $g(d) = c_m \cdot d$



Max Gradient (MG)

- Utilize the maximum gradient
  - $g(d) = c_m \cdot d$

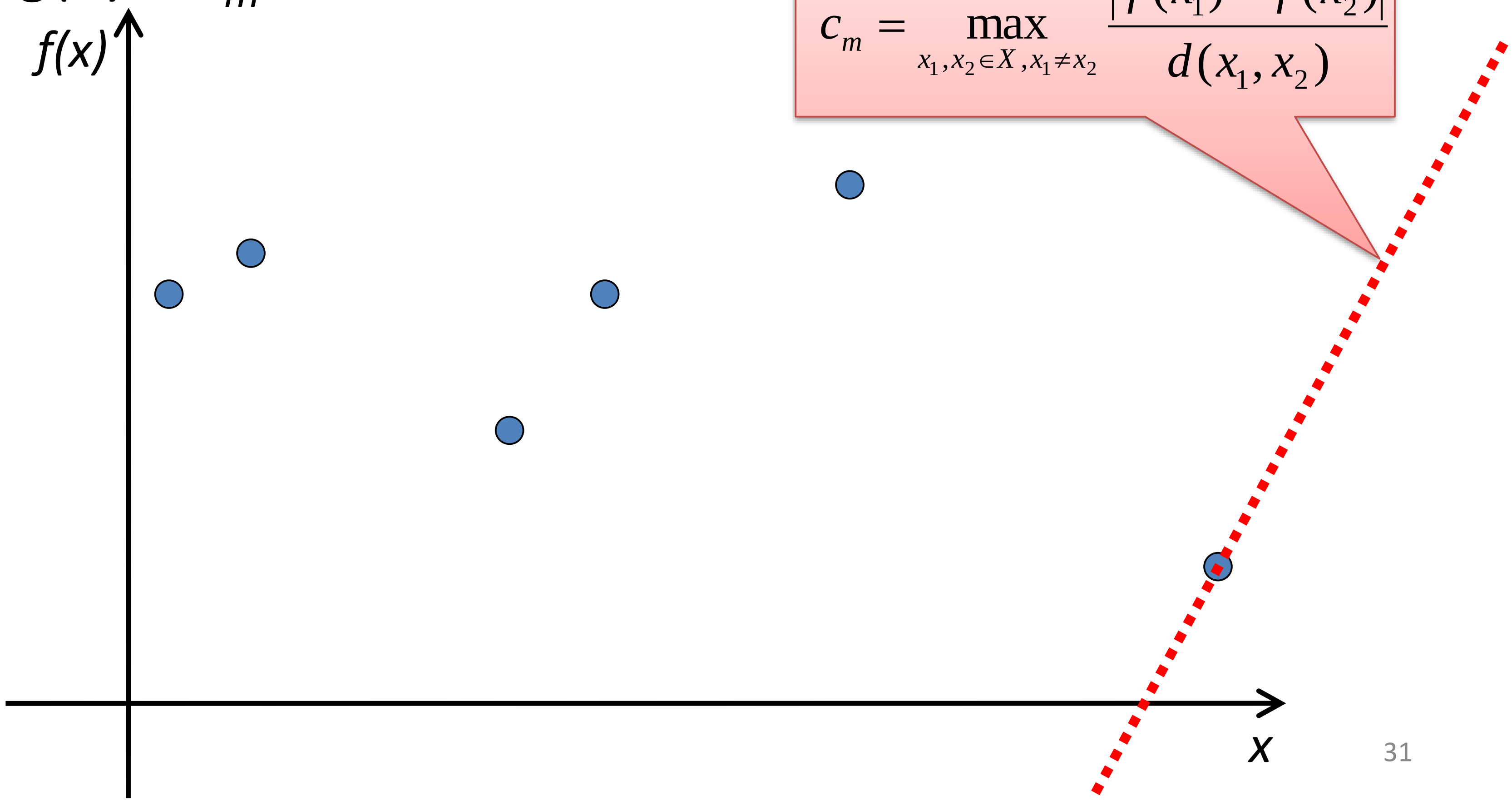
$$c_m = \max_{x_1, x_2 \in X, x_1 \neq x_2} \frac{|f(x_1) - f(x_2)|}{d(x_1, x_2)}$$



Max Gradient (MG)

- Utilize the maximum gradient
  - $g(d) = c_m \cdot d$

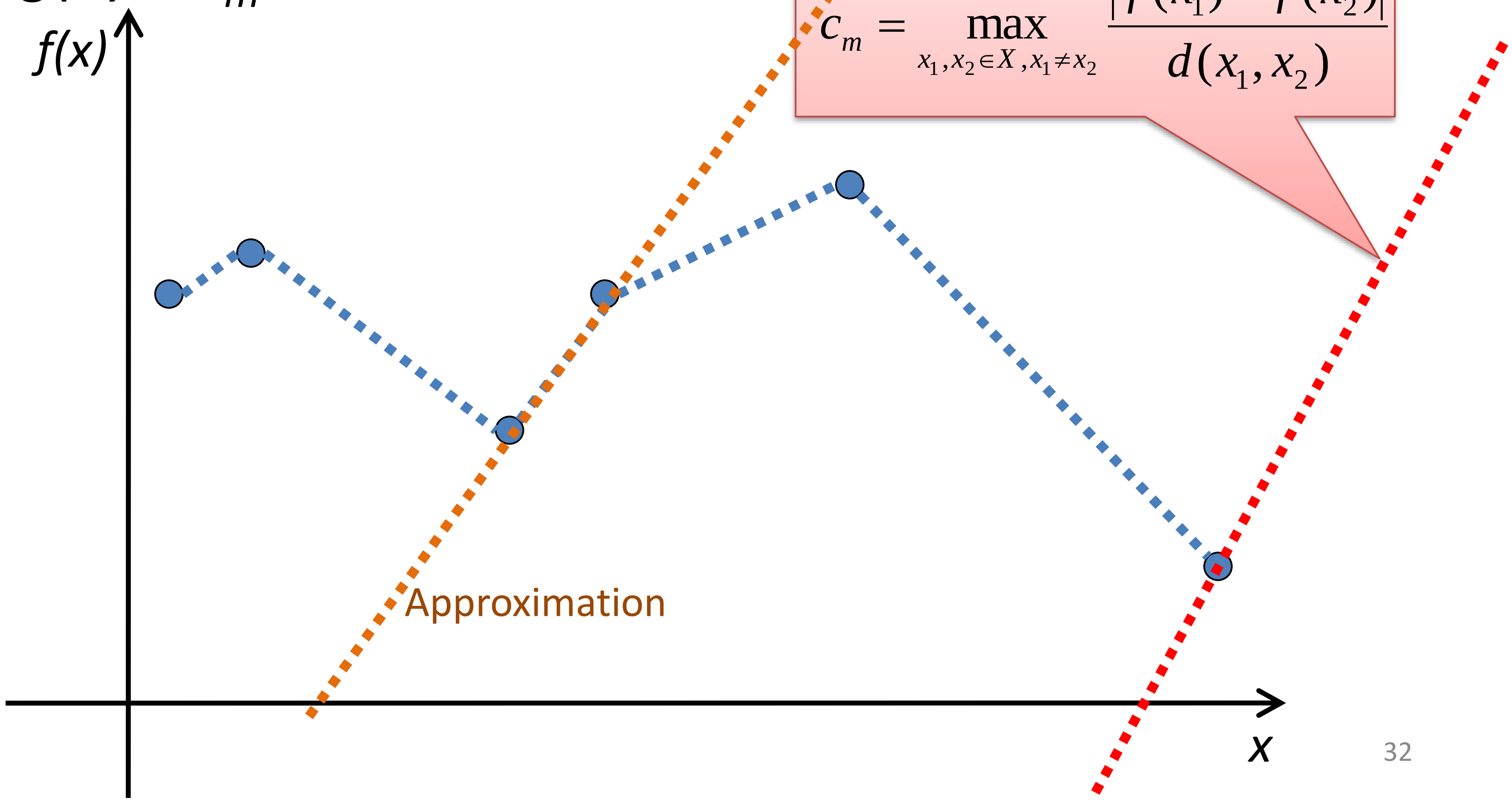
$$c_m = \max_{x_1, x_2 \in X, x_1 \neq x_2} \frac{|f(x_1) - f(x_2)|}{d(x_1, x_2)}$$



Max Gradient (MG)

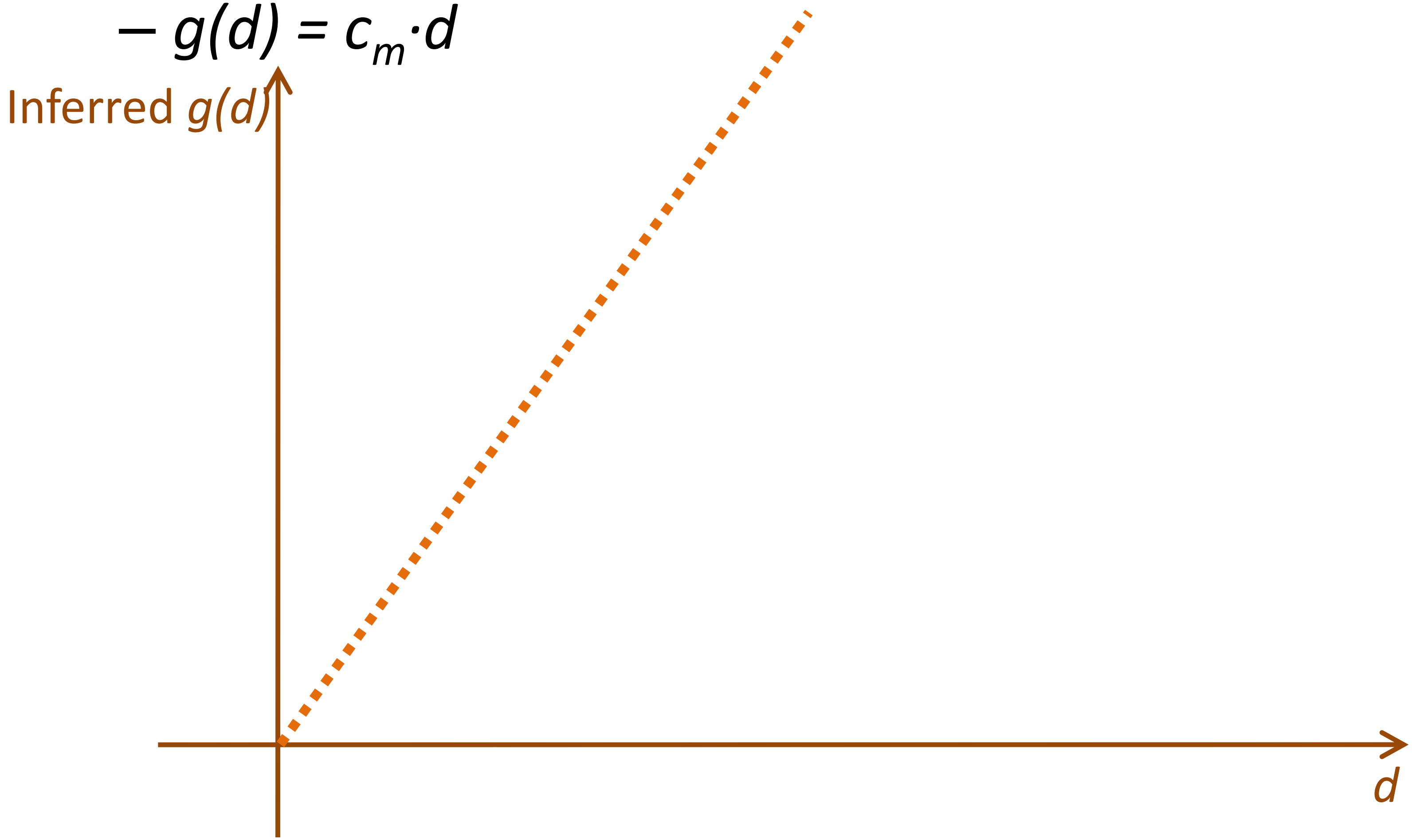
- Utilize the maximum gradient
  - $g(d) = c_m \cdot d$

$$c_m = \max_{x_1, x_2 \in X, x_1 \neq x_2} \frac{|f(x_1) - f(x_2)|}{d(x_1, x_2)}$$



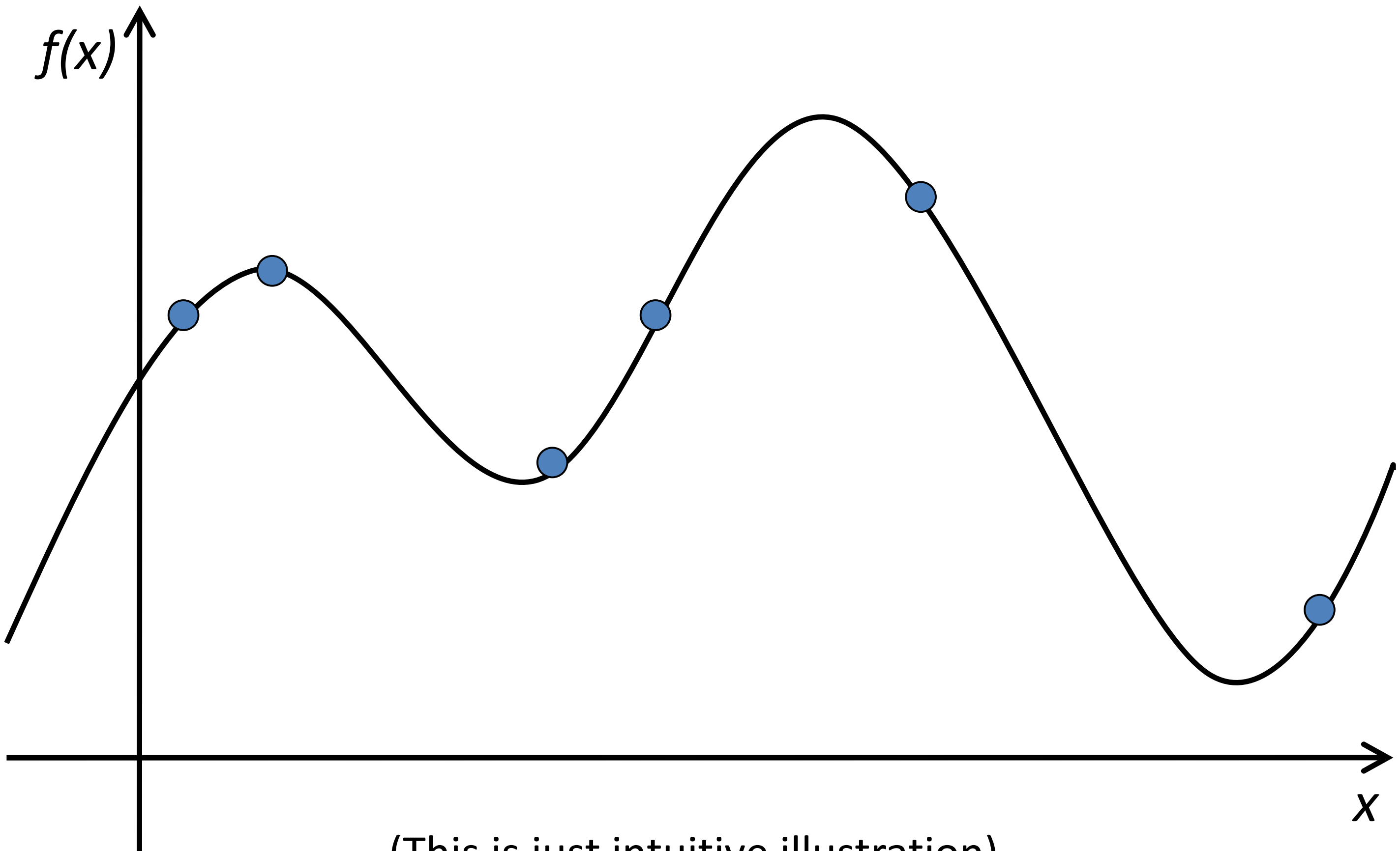
Max Gradient (MG)

- Utilize the maximum gradient
  - $g(d) = c_m \cdot d$



Gathering Differences (GD)

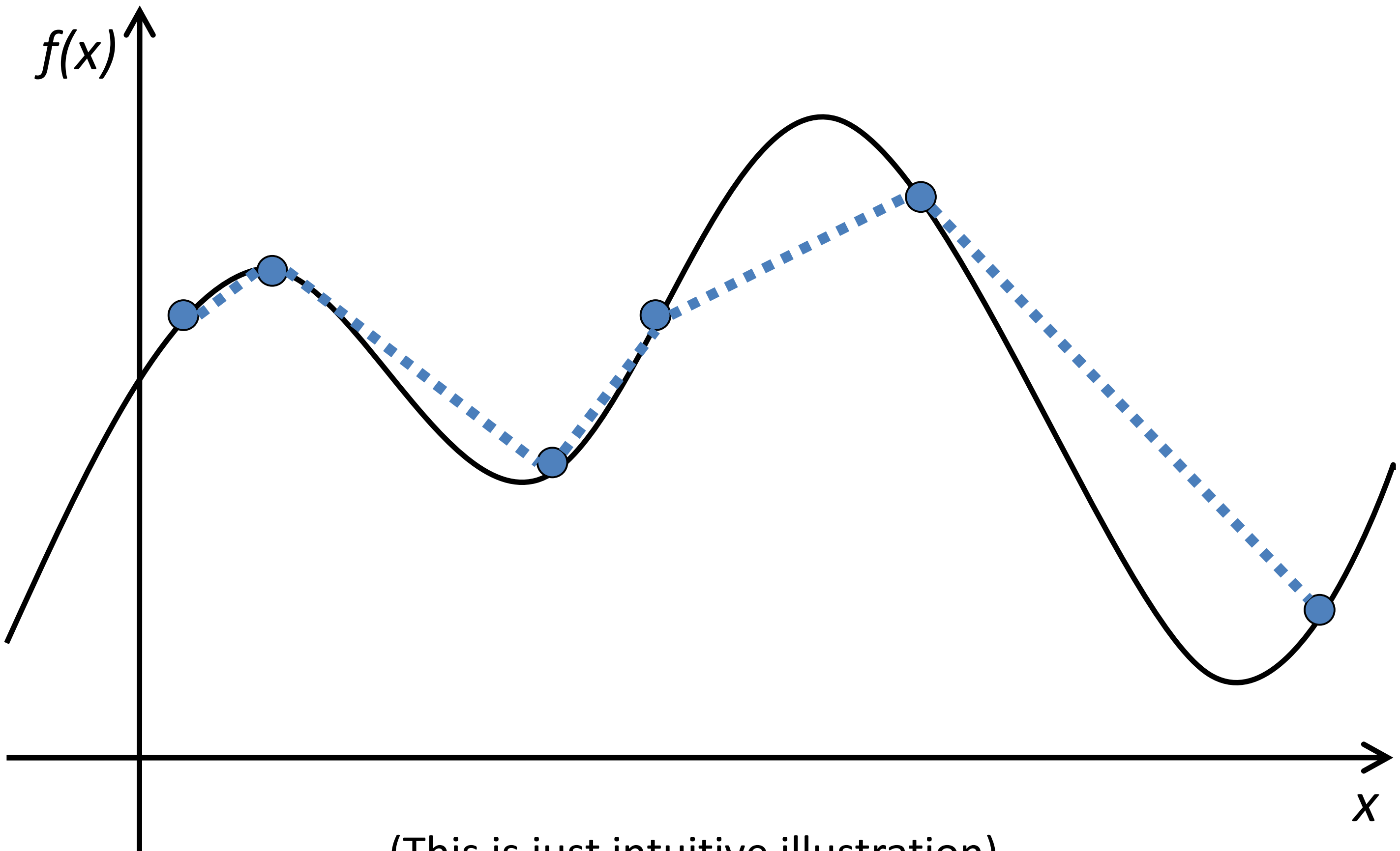
- Utilize gradients with smaller distance in first
  - Better approximation of the landscape



(This is just intuitive illustration)

Gathering Differences (GD)

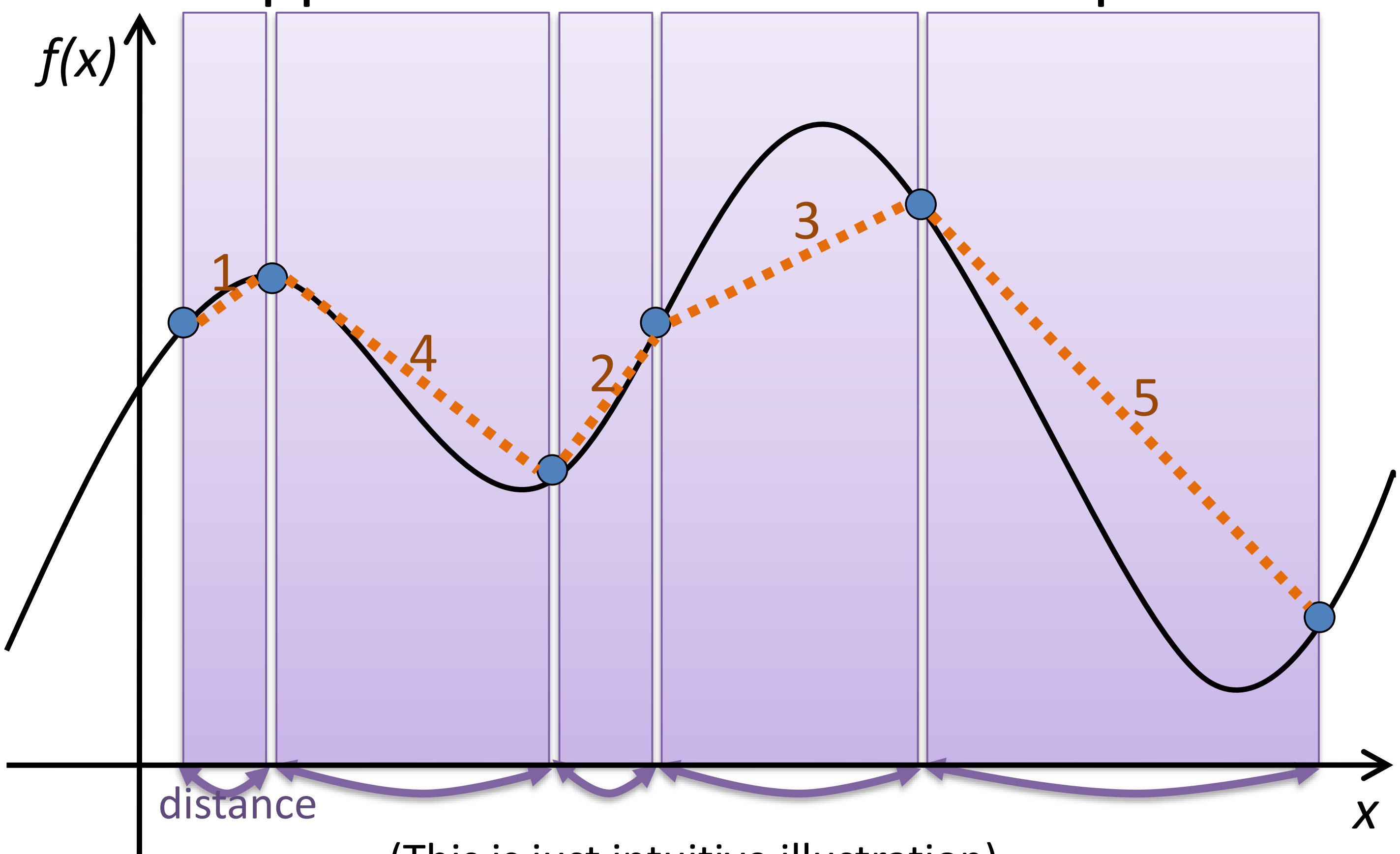
- Utilize gradients with smaller distance in first
  - Better approximation of the landscape



(This is just intuitive illustration)

Gathering Differences (GD)

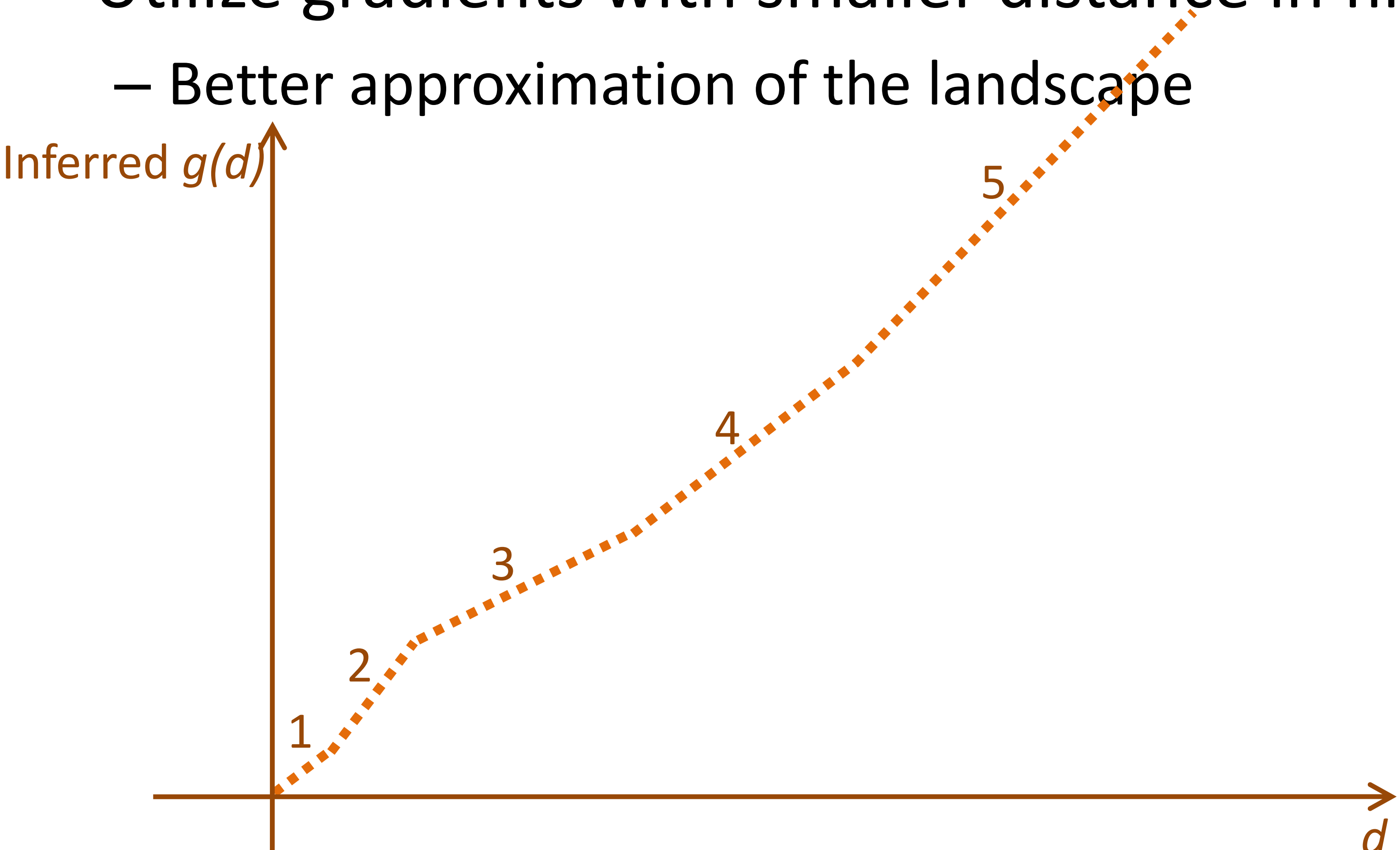
- Utilize gradients with smaller distance in first
  - Better approximation of the landscape



(This is just intuitive illustration)

Gathering Differences (GD)

- Utilize gradients with smaller distance in first
  - Better approximation of the landscape

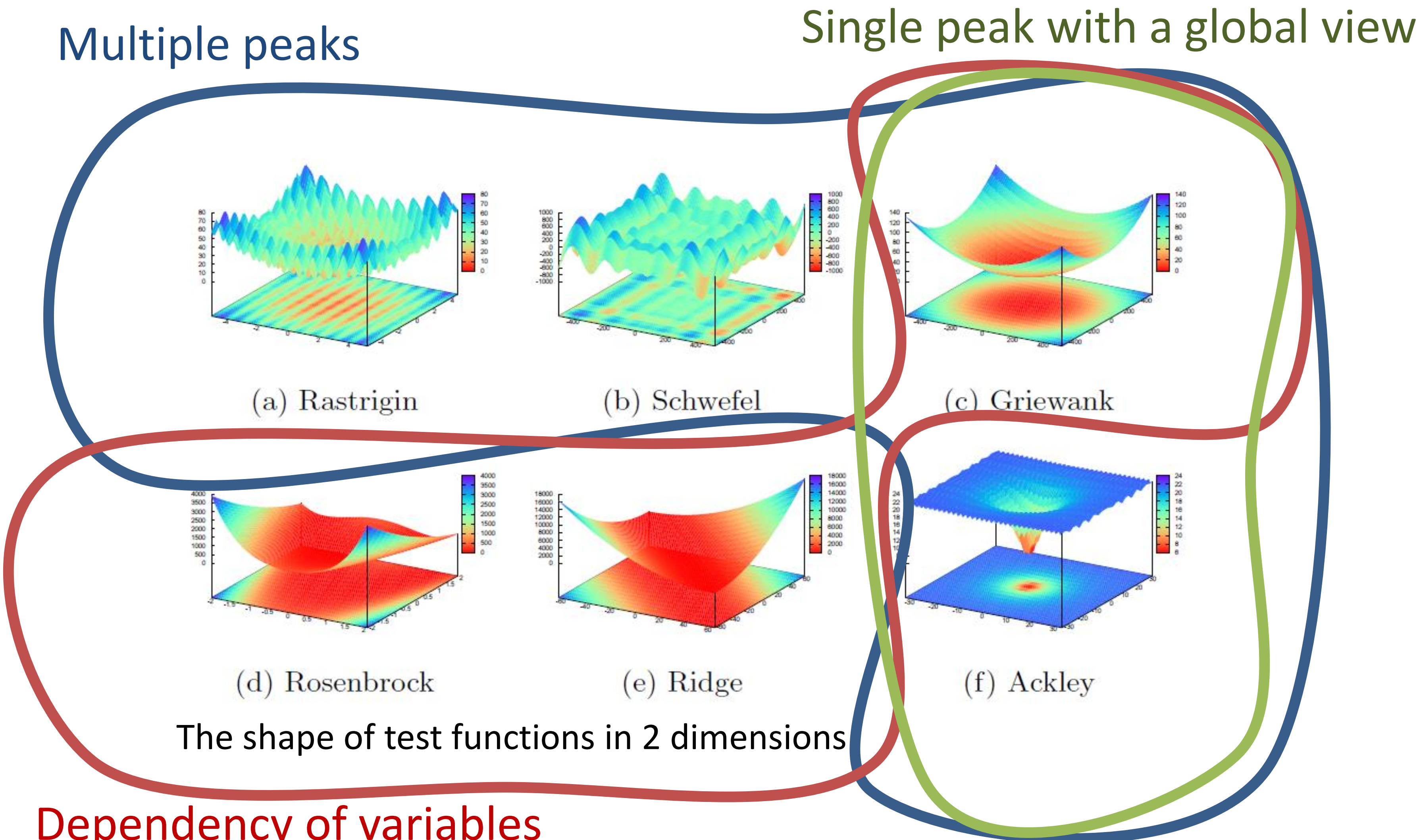


(This is just intuitive illustration)

Contents

- Background
- Thinning-out for reducing trials
  - Max Gradient (MG) method
  - Gathering Differences (GD) method
- Performance evaluation by test functions
- Discovery of strong shots on virtual robots
- Conclusions and future work

Test functions



The shape of test functions in 2 dimensions

(Sano et al. (2000) also utilized these test functions for evaluating distributed GA)

Sampling method

- Need a sampling method
  - Our method only skips over candidates

- Meta-heuristics
  - **Genetic Algorithm (GA)**
  - Simulated Annealing
  - Hill Climbing
  - Policy Gradient
  - (Random Sampling)

(We can combine any meta-heuristics and our method)

Performance evaluation by the kind of test functions

Trial rate and error rate of GA+MG and GA+GD

Function	GA+MG		GA+GD	
	Trial rate (%)	Error rate (%)	Trial rate (%)	Error rate (%)
Rastrigin	19.53	0.23	21.40	1.99
Schwefel	17.02	0.18	21.03	1.42
Griewank	17.18	0.24	21.02	1.01
Rosenbrock	17.77	0.03	21.29	0.69
Ridge	18.11	0.00	19.60	0.75
Ackley	20.09	2.77	29.52	2.94

(The average over 100 experiments using 100 candidates in 2 dimensions)

Trial rate = #trials / #candidates × 100

Error rate = #(wrongly thinned-out candidates) / #(thinned-out candidates) × 100

Performance evaluation by the kind of test functions

Our method can reduce **many trials** with **a few errors**

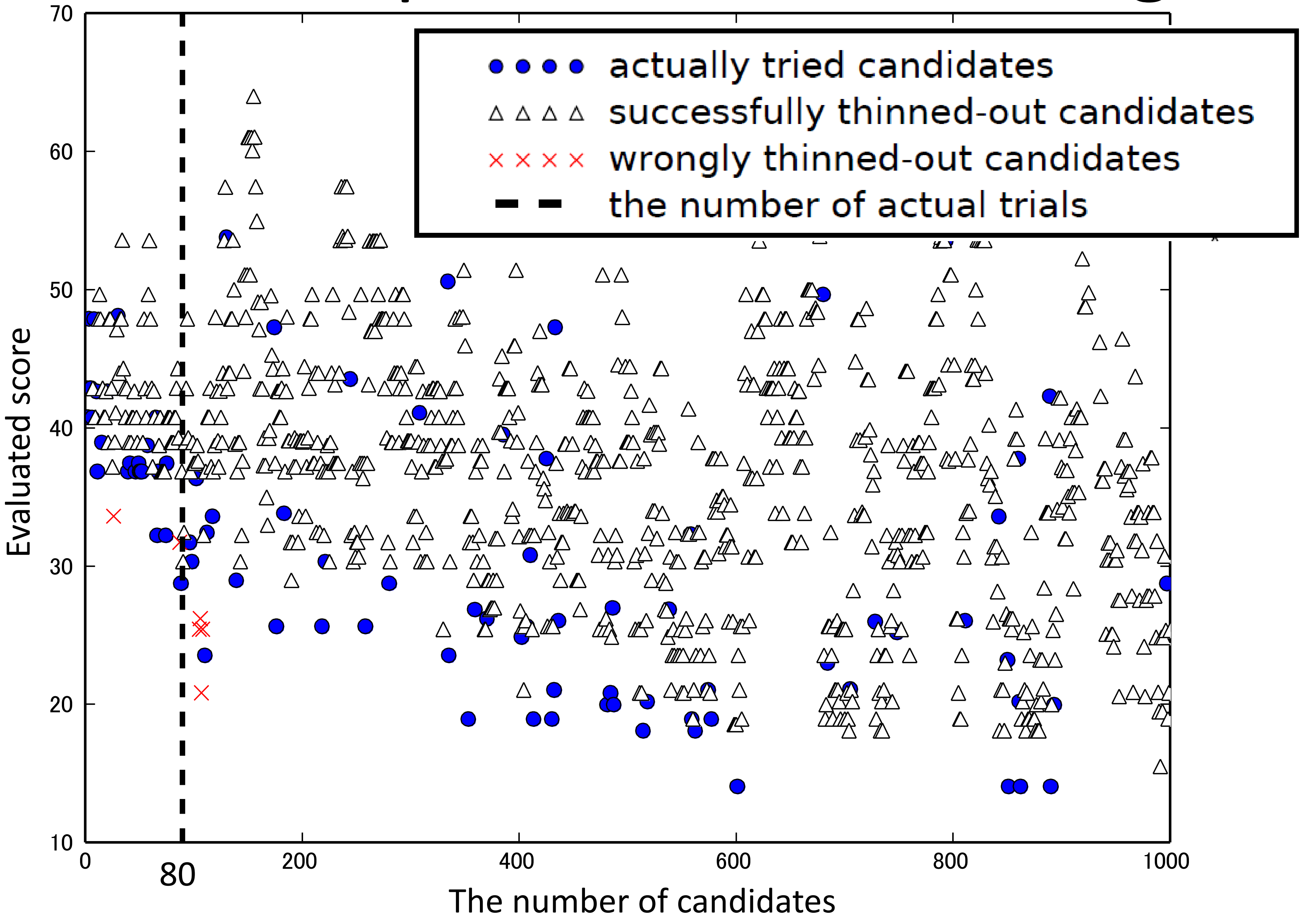
Trial

Function	GA+MG		GA+GD	
	Trial rate (%)	Error rate (%)	Trial rate (%)	Error rate (%)
Rastrigin	19.53	0.23	21.40	1.99
Schwefel	17.02	0.18	21.03	1.42
Griewank	17.18	0.24	21.[illegible]2	1.01
Rosenbrock	17.77	0.05	21.[illegible]9	0.69
Ridge	18.11	0.00	19.60	0.75
Ackley	20.09	2.77	29.52	2.94

(The average over 100 experiments using 100 candidates in 2 dimensions)

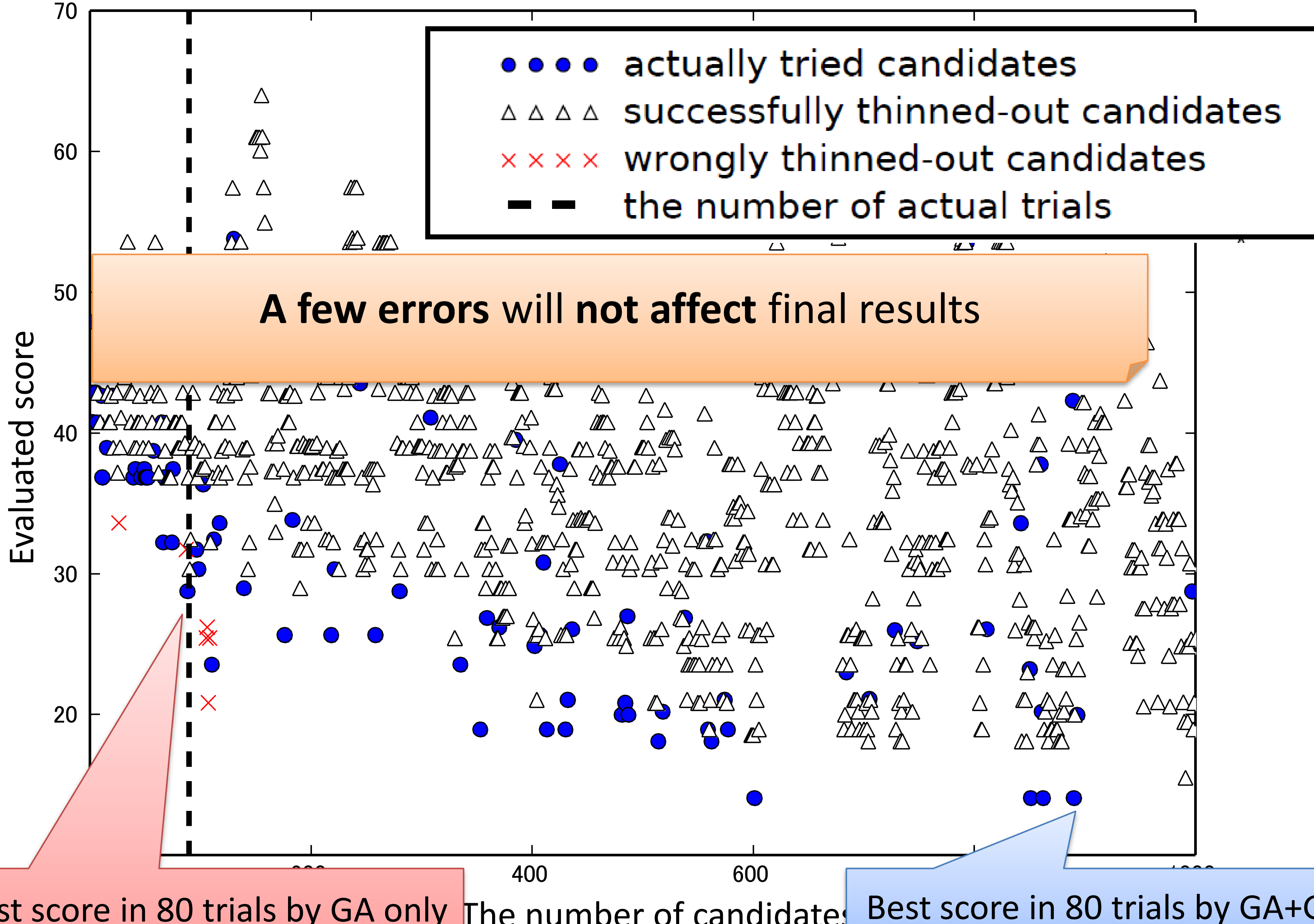
Trial rate = #trials / #candidates × 100
Error rate = #(wrongly thinned-out candidates) / #(thinned-out candidates) × 100

Minimization process with thinning-out



(1,000 candidates by GA+GD in 2 dimensional Rastrigin function)

Minimization process with thinning-out



A few errors will **not affect** final results

Best score in 80 trials by GA only

Best score in 80 trials by GA+GD

(1,000 candidates by GA+GD in 2 dimensional Rastrigin function)

Minimization results

Minimization results of GA only, GA+MG, and GA+GD

Function	Min score by GA only	Min score by GA+MG	Min score by GA+GD
Rastrigin	24	13	19
Schwefel	712	435	439
Griewank	43	32	33
Rosenbrock	418	330	296
Ridge	11,542,427	8,233,764	8,878,178
Ackley	19	18	18

(The average over 100 experiments using 50 trials in 2 dimensions)

Performance evaluation by the dimension size of a function

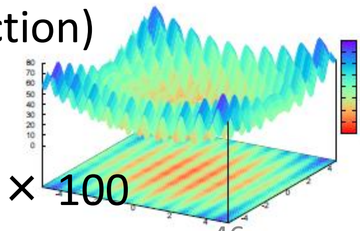
Trial rate and error rate of GA+MG and GA+GD

Dimension size	GA+MG		GA+GD	
	Trial rate (%)	Error rate (%)	Trial rate (%)	Error rate (%)
2	17.03	0.41	22.56	2.44
5	40.17	0.16	22.74	4.08
10	54.48	0.12	25.92	5.09
50	64.77	0.19	29.16	6.89
100	64.30	0.13	31.11	6.12

(the average over 100 experiments using 100 candidates in Rastrigin function)

Trial rate = #trials / #candidates × 100

Error rate = #(wrongly thinned-out candidates) / #(thinned-out candidates) × 100



(a) Rastrigin

Performance evaluation by the dimension size of a function

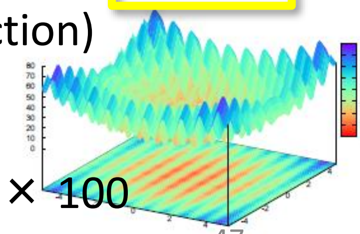
In high dimensions
MG has the **advantage of error rate**
GD has the **advantage of trial rate**

Trial

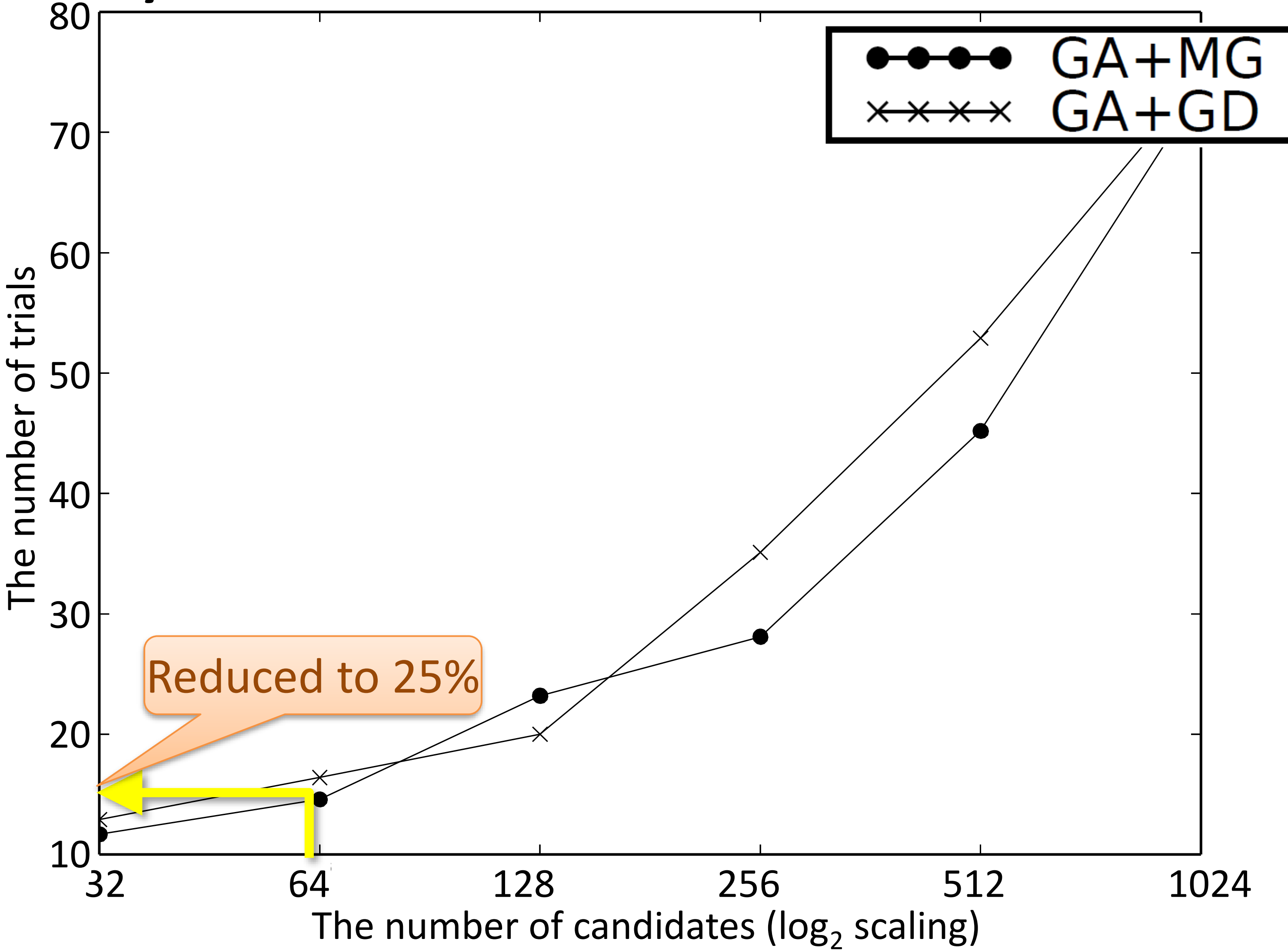
Dimension size	GA+MG		GA+GD	
	Trial rate (%)	Error rate (%)	Trial rate (%)	Error rate (%)
2	17.03	0.41	22.56	2.44
5	40.17	0.16	22.74	4.08
10	54.48	0.12	25.92	5.09
50	64.77	0.19	29.16	6.89
100	64.30	0.13	31.11	6.12

(the average over 100 experiments using 100 candidates in Rastrigin function)

Trial rate = #trials / #candidates × 100
Error rate = #(wrongly thinned-out candidates) / #(thinned-out candidates) × 100

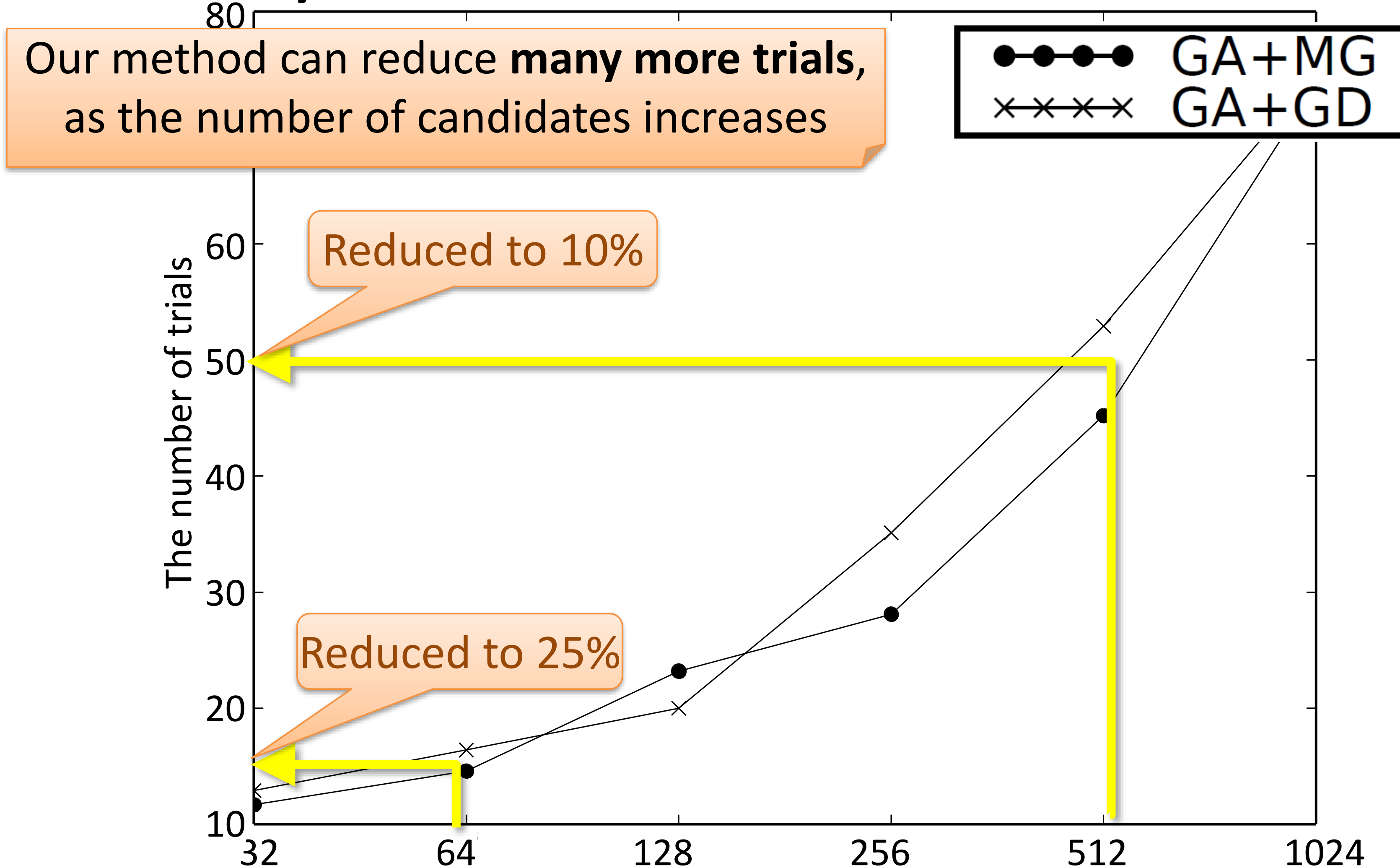

(a) Rastrigin

Performance evaluation by the number of candidates



(The average over 10 experiments in Rastrigin function in 2 dimensions)

Performance evaluation by the number of candidates



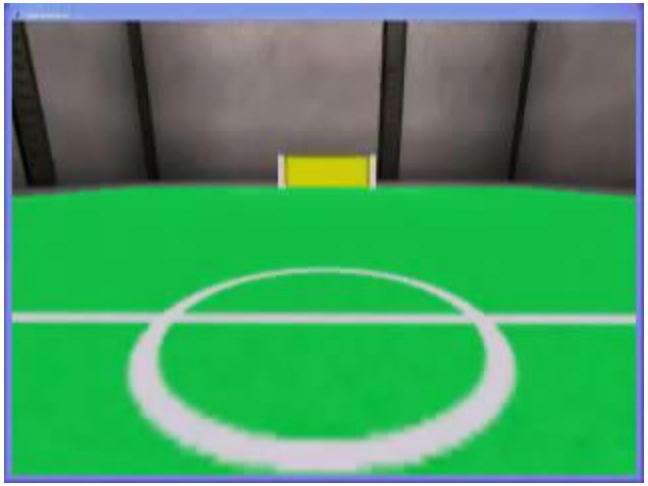
(The average over 10 experiments in Rastrigin function in 2 dimensions)

Contents

- Background
- Thinning-out for reducing trials
  – Max Gradient (MG) method
  – Gathering Differences (GD) method
- Performance evaluation by test functions
- Discovery of strong shots on virtual robots
- Conclusions and future work

Discovery of strong shot motions

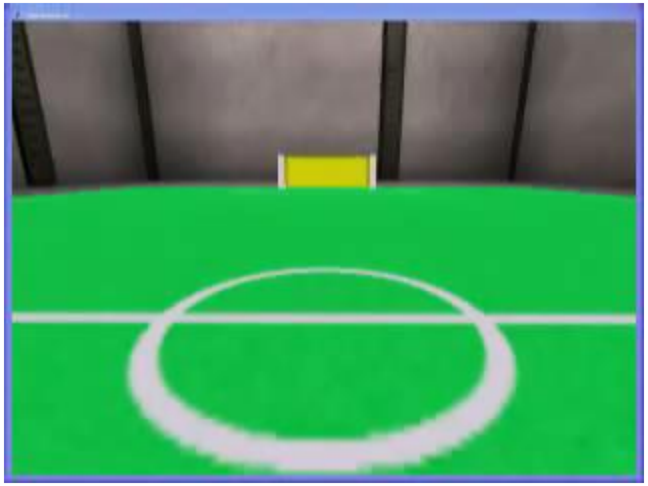
- Experiments in a simulation environment
  - Developed by Zaratti et al.(2006)



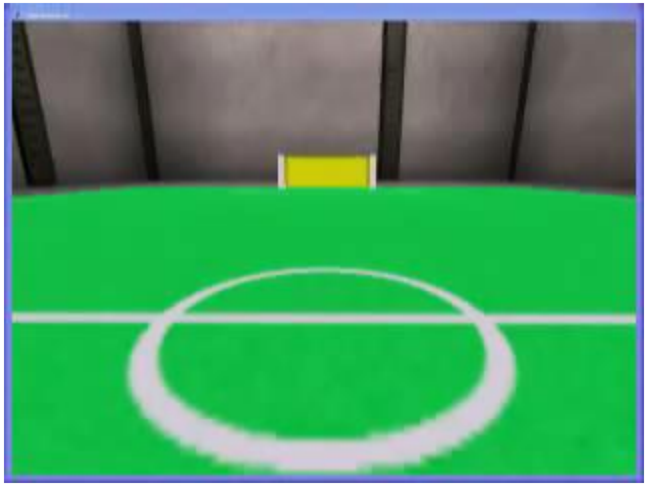
Initial motion
(The search space is 75 dimensions)
https://youtu.be/GqBj-jrEPI4
It takes dozens of hours for 1 experiment

Discovered shot motions

Score 1200

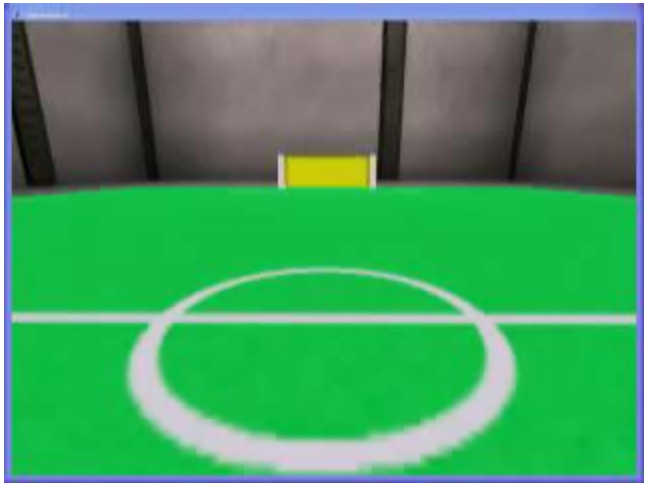


Discovered motion using its whole body
https://youtu.be/Mm30gT9oy1g



Initial motion

Score 600



Score 2000

Discovered motion using its own weight
https://youtu.be/p4OoxYc3pEs

Comparison in skill discovery

Results of maximization by GA only, GA+MG, and GA+GD

Max score by GA only	Max score by GA+MG	Max score by GA+GD
936	940	1058

(The average of 10 experiments using **50 actual trials**)

Almost the same result of GA only using about **100 actual trials**

Trial rate and error rate of GA+MG and GA+GD

GA+MG		GA+GD	
Trial rate (%)	Error rate (%)	Trial rate (%)	Error rate (%)
62.40	0.56	51.00	7.69

(The average of 10 experiments using 50 candidates)

50% trials were reduced

Trial rate = #trials / #candidates × 100

Error rate = #(wrongly thinned-out candidates) / #(thinned-out candidates) × 100

Contents

Conclusions

- Thinning-out for reducing unnecessary trials
  - Max Gradient (MG)
  - Gathering Differences (GD)
- Performance evaluation by test functions
  - MG and GD worked well in various test functions.
- Discovery of strong shot motions
  - Unexpected dynamic motions

Future work

- Exploration of more useful inferring methods
  - As many as possible
  - As correctly as possible
- Experiments in the real environment
  - Verifying that our method can treat real noise
- Theoretical analysis as a randomized algorithm
  - $O(\log n)$ trials for n candidates in random sampling

Thank you for your attention!


https://youtu.be/L7dDnJLLjv4

https://youtu.be/2-GfOOIy8Xc

Discovered poor shot motions :(