CS4618 Artificial Intelligence I

Today: Black-Box Complexity of Unimodal Functions Analysing Mutation

Thomas Jansen

November 14th

Plans for Today

Remember

We want to prove $\forall\delta$ with $0<\delta<1$ constant: $B_{\mathcal{U}} > 2^{n^\delta}$

- $\mathcal{U}$ is set of all unimodal functions
- f unimodal $\Leftrightarrow \forall x\colon\ x$ either optimal or has better Hamming neighbour
- $f_P(x) := \begin{cases} n+i & \text{if } x=p_i \text{ and } x\neq p_j \text{ for all } j>i,\\ \textsc{OneMax}(x) & \text{if } x\notin P\end{cases}$
- $P := (p_1, p_2, \ldots, p_{l(n)})$ with $p_1 = 1^n$, all other neighbours selected uniformly at random
- for proof concessions made towards optimal deterministic algorithm
  1. letting it know which functions have probability 0.
  2. giving away for free the knowledge about any p_i with $f(p_i)\le f(p_j)$ once p_j is sampled,
  3. giving away for free the knowledge about $p_{j+1},\ldots p_{j+n}$ if p_j is the current known best path point and some point not on the path is sampled,
  4. giving away for free the knowledge about $p_{l(n)}$ (the global optimum) once p_{j+n} is sampled while p_j is the current known

Distance Between Points on the Path

Lemma

$\forall \beta > 0$ *constant*: $\exists \alpha(\beta) > 0$ *constant*: $\forall i \leq l(n) - \beta n$:
$\forall j \geq \beta n$: $\mathit{Prob}\left(\mathsf{H}(p_i, p_{i+j}) \leq \alpha(\beta) n\right) = 2^{-\Omega(n)}$

Proof Due to symmetry:
Considering $i = 1$ and some $j \geq \beta n$ suffices.

$H_t := \mathsf{H}(p_1, p_t)$

We want to prove: $\text{Prob}\left(H_j \leq \alpha(\beta) n\right) = 2^{-\Omega(n)}$

We choose $\alpha(\beta) := \min\{1/50, \beta/5\}$.

Due to the random path construction:

- $H_{t+1} \in \{H_t - 1, H_t + 1\}$
- $\text{Prob}\left(H_{t+1} = H_t + 1\right) = 1 - H_t/n$
- $\text{Prob}\left(H_{t+1} = H_t - 1\right) = H_t/n$

Proof of Lemma Continued

Define $\gamma := \min\{1/10, j/n\}$.

Observations

- $\gamma \leq 1/10$
- $\gamma \geq 5\alpha(\beta)$
- γ bounded below and above by positive constants

Consider the last γn steps towards p_j.
Let t be the first of these steps.

Note $(\gamma \leq j/n) \Rightarrow (\gamma n \leq j)$

Case 1 $H_t \geq 2\gamma n$

Clearly, $H_j \geq \underbrace{2\gamma n}_{\text{in the beginning}} - \overbrace{\gamma n}^{\text{number of steps}} = \gamma n > \alpha(\beta)n.$

Proof of Lemma Continued

Case 2 $H_t < 2\gamma n$

Clearly, $H_i < 3\gamma n$ for all $i \in \{t, \dots, j\}$.

Therefore, $\text{Prob}\left(H_i = H_{i-1} + 1\right) \geq 1 - 3\gamma \geq 7/10$,
$\text{Prob}\left(H_i = H_{i-1} - 1\right) \leq 3/10$.

Define independent random variable $S_t, S_{t+1}, \dots, S_j \in \{0, 1\}$ with $\text{Prob}\left(S_k = 1\right) = 7/10$.

Define $S := \sum_{k=t}^{j} S_k$.

Observation $\text{Prob}\left(S \geq (3/5)\gamma n\right) \leq \text{Prob}\left(H_j \geq (1/5)\gamma n\right)$

Since

1. $H_t \geq 0$
2. $\text{Prob}\left(H_i = H_{i-1} + 1\right) \geq \text{Prob}\left(S_i = 1\right)$
3. $\geq (3/5)\gamma n$ increasing steps $\Rightarrow \leq (2/5)\gamma n$ decreasing steps
4. $H_j \geq (3/5)\gamma n - (2/5)\gamma n$

Proof of Lemma Continued

We have γn independent random variable $S_t, S_{t+1}, \ldots, S_j \in \{0,1\}$ with $\mathrm{Prob}\left(S_k = 1\right) = 7/10$ and $S := \sum_{k=t}^{j} S_k$.

Apply Chernoff Bounds

$\mathrm{E}\left(S\right) = (7/10)\gamma n$

$$\mathrm{Prob}\left(S < \tfrac{3}{5}\gamma n\right)$$
$$= \mathrm{Prob}\left(S < \left(1 - \tfrac{1}{7}\right)\tfrac{7}{10}\gamma n\right)$$
$$< e^{-(7/10)\gamma n(1/7)^2/2} = e^{-(1/140)\gamma n} = 2^{-\Omega(n)}$$ □

True Path Length

Lemma with $\beta = 1$ yields:
$\text{Prob}\left(\text{return to path after } n \text{ steps}\right) = 2^{-\Omega(n)}$

$\text{Prob}\left(\text{return to path after} \geq n \text{ steps happens anywhere}\right)$
$= 2^{n^{\varepsilon}} \cdot 2^{-\Omega(n)} = 2^{-\Omega(n)}$

$\text{Prob}\left(l'(n) \geq l(n)/n\right) = 1 - 2^{-\Omega(n)}$

We can prove at best lower bound of
$\frac{l'(n)-n+1}{n} > \frac{l(n)}{n^2} - 1 > 2^{n^{\delta}}$.

An Optimal Deterministic Algorithm

Let N denote the points known not to belong to P.
Let p_i denote the best currently known point on the path.

Initially, $N = \emptyset$, $i \geq 1$.

Algorithm decides to sample x as next point.

Case 1 $\mathrm{H}(p_i, x) \leq \alpha(1)n$

$\mathrm{Prob}\,(x = p_j \text{ with } j \geq n) = 2^{-\Omega(n)}$

Case 2 $\mathrm{H}(p_i, x) > \alpha(1)n$

Consider random path construction starting in p_i.

Similar to Lemma
$\mathrm{Prob}\,(\text{hit } x) = 2^{-\Omega(n)}$

Later steps

$N \neq \emptyset$

Partition N

$N_{\text{far}} := \{y \in N \mid \mathsf{H}(y, p_i) \geq \alpha(1/2)n\}$
$N_{\text{near}} := N \setminus N_{\text{far}}$

Case 1 $N_{\text{near}} = \emptyset$

Consider random path construction starting in p_i.

A: path hits x
E: path hits no point in N_{far}

Clearly, optimal deterministic algorithm avoid N_{far}.

Thus, we are interested in $\text{Prob}\left(A \mid E\right)$
$= \frac{\text{Prob}(A \cap E)}{\text{Prob}(E)} \leq \frac{\text{Prob}(A)}{\text{Prob}(E)}$.

Clearly, $\text{Prob}\left(E\right) = 1 - 2^{-\Omega(n)}$.

Thus, $\text{Prob}\left(A \mid E\right) \leq \left(1 + 2^{-\Omega(n)}\right) \text{Prob}\left(A\right) = 2^{-\Omega(n)}$.

Later Steps With Close Known Points

Case 2 $N_{\text{near}} \neq \emptyset$

Knowing points near by can increase $\text{Prob}(A)$.

Ignore the first $n/2$ steps of path construction; consider $p_{i+n/2}$.

$\text{Prob}(N_{\text{near}} = \emptyset \text{ now}) = 1 - 2^{-\Omega(n)}$

Repeat Case 1. □

Mutation Operators

Remember different randomised search heuristics

local 'mutation'

- randomised local search
- Metropolis algorithm
- simulated annealing

global mutation

- evolutionary algorithms, e. g., (1+1) EA

local 'mutation'

Local Mutation: Pick from 1-Bit Neighbourhood

1. Select $l \in \{1, 2, \ldots, n\}$ uniformly at random.
2. For i := 1 To n do
3. If $i = l$
4. Then $y[i] := 1 - x[i]$
5. Else $y[i] := x[i]$.

global mutation

Global Mutation: Standard Bit Mutation w. $p_m = 1/n$

1. For i := 1 To n do
2. With probability $1/n$
3. $y[i] := 1 - x[i]$
4. Else $y[i] := x[i]$.

Local and Global Mutation

local 'mutation'

Local Mutation: Pick from 1-Bit Neighbourhood

1. Select $l \in \{1, 2, \ldots, n\}$ uniformly at random.
2. For i := 1 To n do
3. If $i = l$
4. Then $y[i] := 1 - x[i]$
5. Else $y[i] := x[i]$.

global mutation

Global Mutation: Standard Bit Mutation w. $p_m = 1/n$

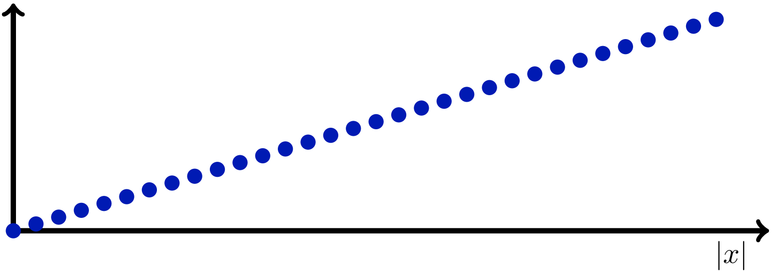
1. For i := 1 To n do
2. With probability $1/n$
3. $y[i] := 1 - x[i]$
4. Else $y[i] := x[i]$.

	local	global
Observation	$\mathrm{H}(x, y) = 1$	$\mathrm{H}(x, y) \in \{0, 1, \ldots, n\}$
	$\mathrm{E}(\mathrm{H}(x, y)) = 1$	$\mathrm{E}(\mathrm{H}(x, y)) = n \cdot (1/n) = 1$

Similar? Perhaps even the same?

An Introductory Example: OneMax

$$\textsc{OneMax}(x) = \sum_{i=1}^{n} x[i]$$



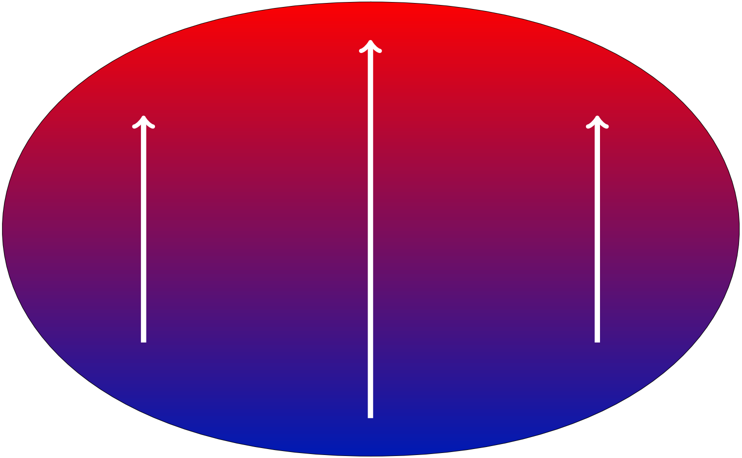
Theorem

$$E\left(T_{\mathsf{RLS},\textsc{OneMax}}\right) = O(n \log n)$$

OneMaxin the Search Space

$$\text{OneMax}(x) = \sum_{i=1}^{n} x[i]$$

$|x| = n$



$|x| = 0$

RLS on OneMax

Theorem

$$\mathrm{E}\left(T_{\mathsf{RLS},\textsc{OneMax}}\right) = O(n \log n)$$

Proof

$$\mathrm{E}\left(T_{\mathsf{RLS},\textsc{OneMax}}\right) = \sum_{i=0}^{n} \mathrm{E}\left(T_{\mathsf{RLS},\textsc{OneMax}} \mid |x_0| = i\right) \cdot \mathrm{Prob}\left(|x_0| = i\right)$$

$$= \sum_{i=0}^{n} \mathrm{E}\left(T_{\mathsf{RLS},\textsc{OneMax}} \mid |x_0| = i\right) \cdot \frac{\binom{n}{i}}{2^n}$$

$$= \sum_{i=0}^{n} \frac{\binom{n}{i}}{2^n} \cdot \sum_{j=i}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right) \leq \sum_{i=0}^{n} \frac{\binom{n}{i}}{2^n} \cdot \sum_{j=0}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right)$$

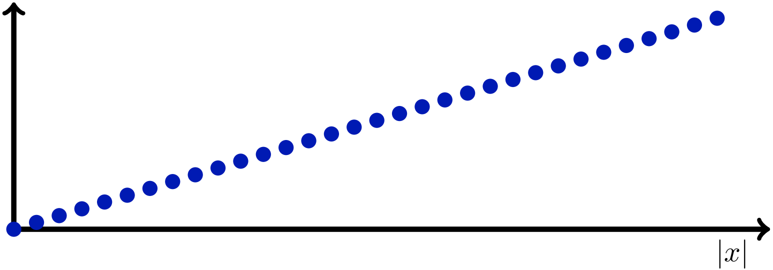
$$= \sum_{j=0}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right) \cdot \sum_{i=0}^{n} \frac{\binom{n}{i}}{2^n} = \sum_{j=0}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right)$$

$$= \sum_{j=0}^{n-1} \frac{n}{n-j} = n \cdot \sum_{j=1}^{n} \frac{1}{j} = n \cdot H_n < n \cdot (\ln(n) + 1) = O(n \log n)$$

□

The Example OneMaxand the (1+1) EA

$$\text{OneMax}(x) = \sum_{i=1}^{n} x[i]$$



Theorem

$$\mathrm{E}\left(T_{\text{(1+1) EA},\text{OneMax}}\right) = O(n \log n)$$

(1+1) EA on OneMax

Theorem

$$\mathrm{E}\left(T_{\text{(1+1) EA},\textsc{OneMax}}\right) = O(n \log n)$$

Proof

$$\mathrm{E}\left(T_{\text{(1+1) EA},\textsc{OneMax}}\right) = \sum_{i=0}^{n} \mathrm{E}\left(T_{\text{(1+1) EA},\textsc{OneMax}}\right) \mid |x_0| = i \cdot \mathrm{Prob}\left(|x_0| = i\right)$$

$$= \sum_{i=0}^{n} \mathrm{E}\left(T_{\text{(1+1) EA},\textsc{OneMax}} \mid |x_0| = i\right) \cdot \frac{\binom{n}{i}}{2^n}$$

$$\leq \sum_{i=0}^{n} \frac{\binom{n}{i}}{2^n} \cdot \sum_{j=i}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right) \leq \sum_{i=0}^{n} \frac{\binom{n}{i}}{2^n} \cdot \sum_{j=0}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right)$$

$$= \sum_{j=0}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right) \cdot \sum_{i=0}^{n} \frac{\binom{n}{i}}{2^n} = \sum_{j=0}^{n-1} \mathrm{E}\left(\text{time } j \rightsquigarrow j+1\right)$$

$$= \leq \sum_{j=0}^{n-1} \frac{n}{n-j} \cdot \left(1 - \frac{1}{n}\right)^{-(n-1)} \leq en \cdot \sum_{j=1}^{n} \frac{1}{j} = en \cdot H_n < en \cdot (\ln(n) + 1)$$

$$= O(n \log n) \quad \square$$

Interlude: A General Proof Method (Part 1 of 3)

Definition (Fitness-Based Partitions)

Let $f\colon \{0,1\}^n \to \mathbb{R}$, $k \in \mathbb{N}$.
$L_0, L_1, \ldots, L_k$ are called f-based partition iff

1. $L_0, L_1, \ldots, L_k$ are a partition of $\{0,1\}^n$.
   $\left(\bigcup_{i=0}^{k} L_i = \{0,1\}^n, \forall i \neq j\colon L_i \cap L_j = \emptyset\right)$
2. Fitness values increase with increasing index.
   $(\forall i \neq j\colon \forall x \in L_i, y \in L_j\colon (i < j) \Rightarrow (f(x) < f(y)))$
3. L_k is the set of global optima.
   $(L_k = \{x \in \{0,1\}^n \mid f(x) = \max\{f(y) \mid y \in \{0,1\}^n\}\})$

Interlude: A General Proof Method (Part 2 of 3)

Remember fitness-based partitions $L_0, L_1, \dots, L_k$

Definition (Probabilities for Improvement)

Let $L_0, L_1, \dots, L_k$ be an f-based partition.
For the (1+1) EA with mutation probability p_m we call

$$s_i := \min_{x \in L_i} \left\{ \sum_{j=i+1}^{k} \sum_{y \in L_j} p_m^{\mathsf{H}(x,y)} \cdot (1-p_m)^{n-\mathsf{H}(x,y)} \right\}$$

the probability for improvement from level L_i.

Observation s_i is lower bound on
probability to leave L_i in one mutation
(pessimistic since minimising over $x \in L_i$)

Interlude: A General Proof Method (Part 3 of 3)

Remember fitness-based partitions $L_0, L_1, \ldots, L_k$
probabilities for improvement $s_0, s_1, \ldots, s_{k-1}$

Theorem

Let $f\colon \{0,1\}^n \to \mathbb{R}$, $L_0, L_1, \ldots, L_k$ *an* f*-based partition,* $s_0, s_1, \ldots, s_{k-1}$ *the probabilities for improvement.*

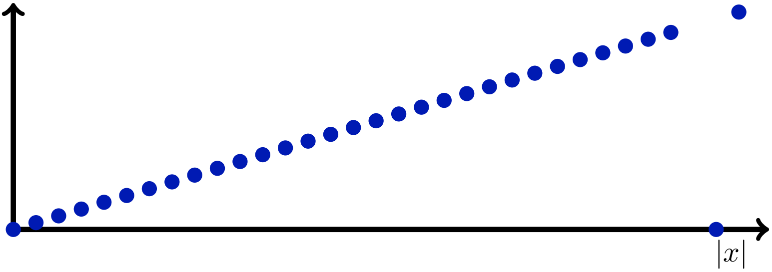
$$E\left(T_{\text{(1+1) EA},f}\right) \leq \sum_{i=0}^{k-1} \frac{1}{s_i}$$

Observations

- works without change for RLS
- works for (1+λ) EA if s_i are adapted

Local Optima

$$f_1(x) = \begin{cases} \textsc{OneMax}(x) & \text{if } |x| \neq n-1 \\ 0 & \text{otherwise} \end{cases}$$

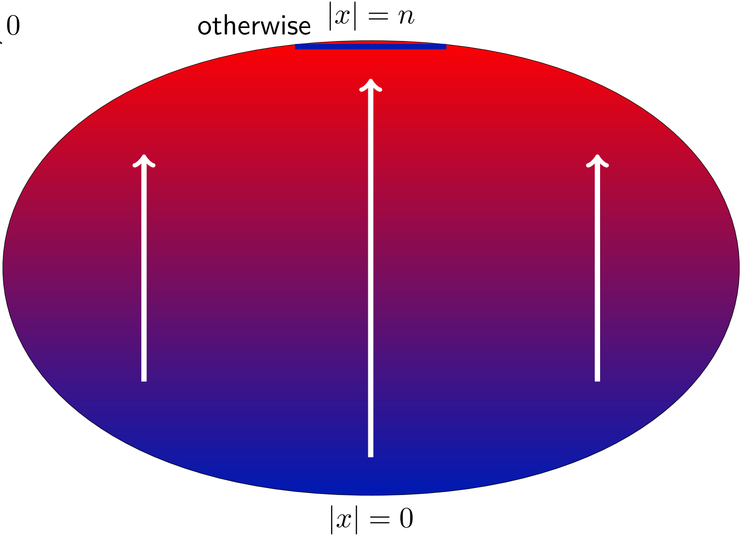


Theorem

$\mathrm{E}\left(T_{(1+1)\ \mathrm{EA}, f_1}\right) = O(n^2)$

f_1 in the Search Space

$$f_1(x) = \begin{cases} \textsc{OneMax}(x) & \text{if } |x| \neq n-1 \\ 0 & \text{otherwise} \end{cases}$$



(1+1) EA on f_1

$$f_1(x) = \begin{cases} \text{OneMax}(x) & \text{if } |x| \neq n-1 \\ 0 & \text{otherwise} \end{cases}$$

Theorem

$$\mathrm{E}\left(T_{\text{(1+1) EA},f_1}\right) = O(n^2)$$

Proof

Observe difference only for $|x| = n-2$

$$\mathrm{E}\left(\text{time } n-2 \rightsquigarrow n\right) = \left(\left(\frac{1}{n}\right)^2 \cdot \left(1 - \frac{1}{n}\right)^{n-2}\right)^{-1}$$

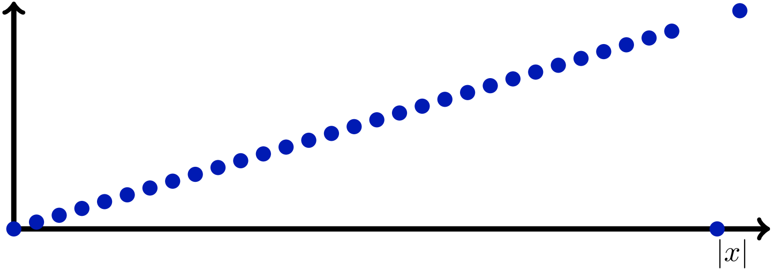
$$\leq en^2 = O(n^2)$$

Thus $\mathrm{E}\left(T_{\text{(1+1) EA},f_1}\right) = O(n \log n) + O(n^2) = O(n^2)$

□

Example function f_1 and RLS

$$f_1(x) = \begin{cases} \textsc{OneMax}(x) & \text{if } |x| \neq n-1 \\ 0 & \text{otherwise} \end{cases}$$



Theorem

$\text{Prob}\left(T_{\mathsf{RLS},f_1} < \infty\right) = 2^{-n+1}$

RLS and the Local Optimum

$$f_1(x) = \begin{cases} \textsc{OneMax}(x) & \text{if } |x| \neq n-1 \\ 0 & \text{otherwise} \end{cases}$$

Theorem

$\text{Prob}\left(T_{\text{RLS},f_1} < \infty\right) = 2^{-n+1}$

Proof

Observation RLS avoids the local optimum iff

(A) $|x_0| = n$ or

(B) $|x_0| = n-1$ and $|x_1| = n$

Observation

$\text{Prob}\left(\text{(A)}\right) = 2^{-n}$

$\text{Prob}\left(\text{(B)}\right) = \text{Prob}\left(|x_0| = n-1\right)$
$\cdot \text{Prob}\left(\text{mutate } `n-1' \rightsquigarrow `n'\right) = n \cdot 2^{-n} \cdot \frac{1}{n} = 2^{-n}$

Thus $\text{Prob}\left(\text{RLS avoid local opt.}\right) = \text{Prob}\left(T_{\text{RLS},f_1} < \infty\right)$
$= 2^{-n} + 2^{-n} = 2 \cdot 2^{-n} = 2^{-n+1}$ □

Summary & Take Home Message

Things to remember

- black-box complexity of unimodal problems
- Local mutations fail in the presence of local optima.

Take Home Message

- Black-box complexity allows for meaningful general lower bounds for RSHs.
- Unimodal problems are not easy to solve.