Randomized Algorithms

"Fundamental Theorem"

A randomized algorithm can be viewed as a single deterministic
algorithm with a random seed

Algorithm A

\[
\text{Algorithm A: } \quad \text{input } x \quad \text{seed } r \quad \text{output } (\text{run time or memory})
\]

Starters: Assume A is "Las Vegas" (never errors)

Specifying a randomized algorithm is equivalent to specifying
a probability distribution \(p(r)\) on the seeds.

\[C_{\text{DET}} = \text{deterministic complexity of the problem (worst-case over } x)\]
\[= \min_r \max_x c(r, x)\]

Pin down seed with best worst-case complexity

\[C_{\text{RAND}} = \text{randomized (Las Vegas) complexity}\]
\[= \min_r \max_x E_p(c(R, x)) \quad \text{R is a random variable with prob distribution } p(r)\]

\[C_{\text{DIST}} = \text{distributional (Las Vegas) complexity}\]
\[= \max_Q \min_r E_q(c(r, X))\]

\(Q = \text{distributions on input space of } x\)

Fix the best seed for each \(x\), then take the worst-case
(expected perf of best det. alg. under worst input distribution)

\[C_{\text{DET}} \geq C_{\text{RAND}} = C_{\text{DIST}}\]

First inequality immediate since \(p\) deterministic distr. on \(r\)
is one of the distributions that can be used in \(C_{\text{RAND}}\)

Proof that \(C_{\text{RAND}} \geq C_{\text{DIST}}\): (Equality: linear prog. duality)

Let \(p \) be a distr. achieving \(C_{\text{RAND}}\)
\[q \] achieving \(C_{\text{DIST}}\)
then \( \text{CRAND} = \max_x E_p(c(R,x)) \geq E_{p,q}(c(R,X)) \geq \min_x E_q(c(r,X)) = \text{CDIST} \)

**Linear Program**

**Primal vars:**

<table>
<thead>
<tr>
<th>( W )</th>
<th>( Pr_1 )</th>
<th>( \cdots )</th>
<th>( Pr_m )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(-c(r_1,x_i))</td>
<td>(\cdots)</td>
<td>(-c(r_m,x_i))</td>
</tr>
<tr>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
</tr>
<tr>
<td>1</td>
<td>(-c(r_1,x_n))</td>
<td>(\cdots)</td>
<td>(-c(r_m,x_n))</td>
</tr>
</tbody>
</table>

\[ \geq 0 \]

\[ \sum \text{dot product of these constants and these probabilities} \] \[ \geq 0 \]

\[ \forall i \quad \sum Pr_i \cdots Pr_m \text{ must be a distr.} \]

**Goal:** minimize

\[ 1 \quad 0 \quad \cdots \quad 0 \]

"standard form" all variables \( \geq 0 \)

\( W, P_1, \ldots, P_m \geq 0 \)

\( \uparrow \) complexity (worst-case)

\( \downarrow \) distribution on seeds

**Dual program:** again standard form

\( \alpha, q_1 x_1, \ldots, q_m x_m \geq 0 \)

\( \uparrow \) lower bound on distributional complexity

\[ \begin{array}{ccc}
W & 0 & \cdots \\
Pr_1 & \vdots & \vdots \\
\vdots & \vdots & \vdots \\
Pr_m & 0 & 0 \\
\end{array} \]

\[ \begin{array}{cccc}
1 & -c(r_1,x_i) & \cdots & -c(r_m,x_i) \\
\vdots & \vdots & \ddots & \vdots \\
1 & -c(r_1,x_n) & \cdots & -c(r_m,x_n) \\
0 & 1 & \cdots & 1 \\
\end{array} \]

\( \sum \text{dot product of row of } c_i^r \text{ values and col. of } q_i x_i \) \( \leq 0 \)

\( \sum q_i x_i = 0 \)

\[ \sum q_i x_i = 0 \]

\[ \forall i \quad \sum q_i x_i \cdots q_m x_i \text{ must be a distr.} \]

**w = value of the primal program = CRAND**

**\( \alpha = \text{dual} = \text{CDIST} \)**

**Easy LP duality** \( w \geq \alpha \)

**Hard LP duality** \( w = \alpha \)
In a single picture...

\[
\begin{align*}
\text{primal} & \\
\min & \quad \begin{bmatrix} 1 & 0 & \cdots & 0 \\ v_1 & v_1 & \cdots & v_1 \end{bmatrix} \\
& \quad \begin{bmatrix} w \quad p_1 & \cdots & p_m \end{bmatrix} \\
& \quad \begin{bmatrix} 1 & -c(r, x_1) & \cdots & -c(r, x_m) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & -c(r, x_n) & \cdots & -c(r, x_n) \end{bmatrix} \\
& \quad \begin{bmatrix} 0 & 1 & \cdots & 1 \end{bmatrix}
\end{align*}
\]

\[q x_i \geq 0 \]
\[q x_n \geq 0 \]
\[\alpha \geq 1 \]

\[w = \text{value of primal program} = \text{Grand} \]
\[\alpha = \text{dual} = \text{Coint} \]

Simple game: Hide & seek

- Child has \( n \) possible locations to hide
- Child wants to maximize time until found
- You know those same \( n \) locations
- You want to maximize time until found

Child chooses distribution over locations
You choose distribution over orders for visiting hiding locations
Monte Carlo Version

\[ C_{\text{rand}}, \lambda = \inf_{p} \max_{x} E_{p} (c(R, x)) \]

where \( p \) ranges over all distributions whose worst-case (over \( x \)) prob. of error \( \leq \lambda \)

**Note:** much broader class of algorithms here since need not always be correct... allowed \( \lambda \) error probability

\[ C_{\text{dist}}, \lambda = \sup_{q} \min_{r} E_{q} (c(r, X)) \]

where \( r \) ranges over all algorithms that have probability of error \( \leq \lambda \) on dist \( q \)

**Theorem (Yao):**

\[ C_{\text{rand}}, \lambda \leq \frac{1}{2} C_{\text{dist}}, 2\lambda \]

**Proof:** Fix any \( \varepsilon > 0 \).

Let \( p_{\varepsilon} \) be a distribution on \( r \)

s.t. \( \forall x \)

\[ E_{p_{\varepsilon}} (c(R, x)) < C_{\text{rand}}, \lambda + \frac{\varepsilon}{2} \quad (*) \]

and \( p_{\varepsilon} \) achieves worst-case Perror \( \leq \lambda \) \( \quad \text{(**)} \)

Let \( q_{\varepsilon} \) be an arbitrary distribution on \( x \).

Set

\[ y = E_{p_{\varepsilon}, q_{\varepsilon}} (c(R, X)) \]

\[ s = \text{Perror}_{p_{\varepsilon}, q_{\varepsilon}} \]

Then

\[ y \leq C_{\text{rand}}, \lambda + \frac{\varepsilon}{2} \quad \text{since (**) holds} \forall x \]

\[ s \leq \lambda \quad \text{since (**).} \]

Let

\[ \mathcal{K} = \{ r : E_{q}(c(r, X)) > 2y \} \]

\[ \mathcal{L} = \{ r : \text{Perror}_{q} > 2s \} \]

(2 tables)

\[ r \]

<table>
<thead>
<tr>
<th>( x )</th>
<th>( c(r, x) ) values</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r )</td>
<td>( x )</td>
</tr>
</tbody>
</table>

| \( x \) | \( 1/2 \neq 0 \) for \( \text{error} \) and \( x \) |

\[ y \text{ and } s \text{ are expectations over rows} \]

\( \mathcal{K} \text{ and } \mathcal{L} \text{ pick out high-valued rows} \]
\[ P_{\text{pe}}(K) < \frac{1}{2} \]
\[ P_{\text{pe}}(L) < \frac{1}{2} \]

For every distribution \( q \), \( \exists r \) s.t.

\[ P_q(\text{alg r errs}) \leq 2\lambda \]
\[ E_q(c(r, X)) \leq 2C_{\text{RAND}, \lambda} + \varepsilon \]

\[ \Rightarrow CDIST_{2\lambda} \leq 2C_{\text{RAND}, \lambda} + \varepsilon \]

True \( \forall \varepsilon > 0 \), so

\[ CDIST_{2\lambda} \leq 2C_{\text{RAND}, \lambda} \]