Yao's minimax principle

How it's made: lower bounds for randomized algorithms

Taisuke Yasuda

July 27, 2018

Carnegie Mellon University

Overview

- Introduction: randomized algorithms and complexity
- Yao's minimax principle
- Applications

Sorting

- Prototypical example of a great randomized algorithm: quick sort
- Lower bound on deterministic sorting – any deterministic algorithm A sorting an n-element list L requires $\Omega(n \log n)$ comparisons in worst case
- What about for randomized algorithms?

Complexity of randomized algorithms

Let P be a computational problem.

- We view a randomized algorithm $\mathcal{R}$ as a probability distribution over a finite set of deterministic algorithms $\mathcal{A}$ solving P, where $A \in \mathcal{A}$ is an algorithm that takes inputs from a set $\mathcal{X}$
- For any measure of cost $\mathsf{cost} : \mathcal{A} \times \mathcal{X} \to \mathbb{R}^+$ for deterministic algorithms, we define a measure of expected cost for randomized algorithms via
$$\mathsf{cost}(\mathcal{R}, x) := \mathbb{E}_{A \sim \mathcal{R}} \mathsf{cost}(A, x)$$
- We define the **randomized complexity** of the problem P via
$$\min_{\mathcal{R}} \max_{x \in \mathcal{X}} \mathsf{cost}(\mathcal{R}, x)$$

Complexity of randomized algorithms

How the heck would you prove lower bounds for this thing?!

$$\min_{\mathcal{R}} \max_{x \in \mathcal{X}} \mathrm{cost}(\mathcal{R}, x)$$

Yao’s minimax principle

Yao's minimax principle

Theorem (Yao's minimax principle)

Define the average cost of a deterministic algorithm over a random distribution of inputs $\mathcal{D}$ via

$$\mathrm{cost}(A, \mathcal{D}) := \mathbb{E}_{x \sim \mathcal{D}} \mathrm{cost}(A, x).$$

Then

$$\min_{\mathcal{R}} \max_{x \in \mathcal{X}} \mathrm{cost}(\mathcal{R}, x) = \max_{\mathcal{D}} \min_{A \in \mathcal{A}} \mathrm{cost}(A, \mathcal{D}).$$

Yao's minimax principle

If we fix an input distribution $\mathcal{D}$, then

$$\min_{A \in \mathcal{A}} \text{cost}(A, \mathcal{D}) \leq \max_{\mathcal{D}'} \min_{A \in \mathcal{A}} \text{cost}(A, \mathcal{D}') = \min_{\mathcal{R}} \max_{x \in \mathcal{X}} \text{cost}(\mathcal{R}, x)$$

so it suffices to come up with a hard enough input distribution $\mathcal{D}$ and show that any deterministic algorithm has to pay a high cost!

Caution: the input distribution is known to the deterministic algorithm, meaning the deterministic algorithm only needs to solve the problem for the given input distribution.

Yao's minimax principle

Example

Let $\mathcal{R}$ be any randomized algorithm that sorts a list L of n elements. Then $\max_{x \in \mathcal{X}} \mathsf{cost}(\mathcal{R}, x) = \Omega(n \log n)$ *where* cost *is the number of comparisons.*

Yao's minimax principle

Proof. Let

$$C := \max_{x \in \mathcal{X}} \operatorname{cost}(\mathcal{R}, x)$$

and consider the cost

$$\operatorname{cost}'(A, x) := \begin{cases} 1 & \operatorname{cost}(A, x) \geq 10C \\ 0 & \text{otherwise} \end{cases}.$$

First note by Markov's inequality,

$$\begin{aligned} \max_{x \in \mathcal{X}} \mathbb{E}_{A \sim \mathcal{R}} \operatorname{cost}'(A, x) &= \max_{x \in \mathcal{X}} \Pr_{A \sim \mathcal{R}} (\operatorname{cost}(A, x) \geq 10C) \\ &\leq \max_{x \in \mathcal{X}} \frac{\mathbb{E}_{A \sim \mathcal{R}} \operatorname{cost}(A, x)}{10C} \leq \frac{1}{10}. \end{aligned}$$

Yao's minimax principle

Now let $\mathcal{D}$ be the uniform distribution on all $n!$ permutations of L. We then have by Yao's minimax principle that

$$\min_{A \in \mathcal{A}} \Pr_{x \sim \mathcal{D}} (\mathrm{cost}(A, x) \geq 10C) = \min_{A \in \mathcal{A}} \mathbb{E}_{x \sim \mathcal{D}} \mathrm{cost}'(A, x) \leq \max_{x \in \mathcal{X}} \mathbb{E}_{A \sim \mathcal{R}} \mathrm{cost}'(A, x).$$

If A is an algorithm achieving the LHS minimum, then

$$\Pr_{x \sim \mathcal{D}} (\mathrm{cost}(A, x) \geq 10C) \leq \frac{1}{10}$$

so A can output at most 2^{10C} permutations, for $n!(9/10)$ distinct inputs. Because A must correctly sort L, we have

$$2^{10C} \geq \frac{9n!}{10} \implies C = \Omega(\log n!) = \Omega(n \log n).$$

$\square$

Yao’s minimax principle: proof

Notations/definitions

- m-dimensional simplex:

$$\Delta^m := \left\{ \mathsf{p} \in \mathbb{R}^m : \mathsf{p}_i \geq 0, \sum_{i \in [m]} \mathsf{p}_i = 1 \right\}$$

- e_i: ith standard basis vector

Finite two-player zero-sum game

The setting...

- Player 1 has $m \in \mathbb{N}$ possible actions, player 2 has $n \in \mathbb{N}$ possible actions (finite, two-player)
- **Payoff matrix** $\mathsf{A} \in \mathbb{R}^{m \times n}$: when player 1 plays action $i \in [m]$ and player 2 plays action $j \in [n]$, player 1 receives payoff A_{ij}, player 2 receives payoff $-\mathsf{A}_{ij}$ (zero-sum)
- **Pure strategy**: playing a single action with probability 1
- **Mixed strategy**: playing action i with probability x_i for some $\mathsf{x} \in \Delta^m$ (drawn independently)

Finite two-player zero-sum game

Payoff matrix A for a game of rock paper scissors where winner gets 1 point, loser gets -1 points, and a draw results in 0 points for both players:

	Rock	Paper	Scissors
Rock	0	-1	1
Paper	1	0	-1
Scissors	-1	1	0

Finite two-player zero-sum game

- **Expected payoff** (for player 1) for a pair of strategies:

$$V(\mathsf{p}, \mathsf{q}) := \sum_{i \in [m]} \sum_{j \in [n]} \Pr(i, j)\, \mathsf{A}_{ij} = \sum_{i \in [m]} \sum_{j \in [n]} \mathsf{p}_i \mathsf{A}_{ij} \mathsf{q}_j = \mathsf{p}^\top \mathsf{A} \mathsf{q}$$

- **Equilibrium point** $(\hat{\mathsf{p}}, \hat{\mathsf{q}})$:

$$V(\mathsf{p}, \hat{\mathsf{q}}) \leq V(\hat{\mathsf{p}}, \hat{\mathsf{q}}) \qquad \text{for all } \mathsf{p} \in \Delta^m$$

and

$$V(\hat{\mathsf{p}}, \mathsf{q}) \geq V(\hat{\mathsf{p}}, \hat{\mathsf{q}}) \qquad \text{for all } \mathsf{q} \in \Delta^n$$

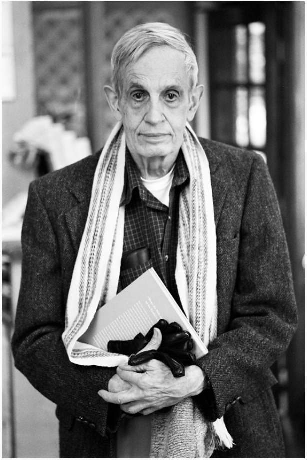
Von Neumann’s minimax theorem

Von Neumann’s minimax theorem

Theorem (Von Neumann’s minimax theorem)

Every finite two-player zero-sum game has an equilibrium point.

Von Neumann’s minimax theorem



Von Neumann's minimax theorem: proof

Theorem (Brouwer's fixed point theorem)

Let $K \subseteq \mathbb{R}^d$ *be compact and convex set. Then if* $f : K \to K$ *is a continuous function, then there exists a fixed point* $\hat{\mathrm{x}} \in K$ *such that* $f(\hat{\mathrm{x}}) = \hat{\mathrm{x}}$*.*

Von Neumann's minimax theorem: proof

John von Neumann on Nash's proof of the Nash equilibrium (more powerful version of minimax theorem):

> *That's trivial, you know. That's just a fixed point theorem.*

Von Neumann's minimax theorem: proof

Idea: define a function $\mathsf{T} : \Delta^m \times \Delta^n \to \Delta^m \times \Delta^n$ such that its fixed points are exactly the equilibrium points $(\hat{\mathsf{p}}, \hat{\mathsf{q}})$

Von Neumann's minimax theorem: proof

For $i \in [m]$ and $j \in [n]$, define

$$\begin{aligned} \varphi_i(\mathbf{p}, \mathbf{q}) &:= \max\left\{V(\mathbf{e}_i, \mathbf{q}) - V(\mathbf{p}, \mathbf{q}), 0\right\} \\ \psi_j(\mathbf{p}, \mathbf{q}) &:= \max\left\{V(\mathbf{p}, \mathbf{q}) - V\left(\mathbf{p}, \mathbf{e}_j\right), 0\right\} \end{aligned}.$$

Now define

$$\mathsf{T}(\mathbf{p}, \mathbf{q}) := (\boldsymbol{\Phi}(\mathbf{p}, \mathbf{q}), \boldsymbol{\Psi}(\mathbf{p}, \mathbf{q}))$$

where the $[m] \ni i$th component of $\boldsymbol{\Phi}$ and $[n] \ni j$th component of $\boldsymbol{\Psi}$ are given by

$$\begin{aligned} \boldsymbol{\Phi}(\mathbf{p}, \mathbf{q})_i &:= \frac{\mathbf{p}_i + \varphi_i(\mathbf{p}, \mathbf{q})}{1 + \sum_{i' \in [m]} \varphi_{i'}(\mathbf{p}, \mathbf{q})} \\ \boldsymbol{\Psi}(\mathbf{p}, \mathbf{q})_j &:= \frac{\mathbf{q}_j + \psi_j(\mathbf{p}, \mathbf{q})}{1 + \sum_{j' \in [n]} \psi_{j'}(\mathbf{p}, \mathbf{q})} \end{aligned}.$$

Von Neumann's minimax theorem: proof

Proposition

If $(\hat{p}, \hat{q})$ *is an equilibrium pair, then it is a fixed point.*

Von Neumann's minimax theorem: proof

Proof. For an equilibrium pair, we have

$$\begin{aligned} \varphi_i(\hat{\mathsf{p}}, \hat{\mathsf{q}}) &= \max\left\{V(\mathsf{e}_i, \hat{\mathsf{q}}) - V(\hat{\mathsf{p}}, \hat{\mathsf{q}}), 0\right\} = 0 \\ \psi_j(\hat{\mathsf{p}}, \hat{\mathsf{q}}) &= \max\left\{V(\hat{\mathsf{p}}, \hat{\mathsf{q}}) - V\left(\hat{\mathsf{p}}, \mathsf{e}_j\right), 0\right\} = 0 \end{aligned}.$$

Thus,

$$\begin{aligned} \mathbf{\Phi}(\hat{\mathsf{p}}, \hat{\mathsf{q}})_i &= \frac{\hat{\mathsf{p}}_i + \varphi_i(\hat{\mathsf{p}}, \hat{\mathsf{q}})}{1 + \sum_{i' \in [m]} \varphi_{i'}(\hat{\mathsf{p}}, \hat{\mathsf{q}})} = \hat{\mathsf{p}}_i \\ \mathbf{\Psi}(\hat{\mathsf{p}}, \hat{\mathsf{q}})_j &= \frac{\hat{\mathsf{q}}_j + \psi_j(\hat{\mathsf{p}}, \hat{\mathsf{q}})}{1 + \sum_{j' \in [n]} \psi_{j'}(\hat{\mathsf{p}}, \hat{\mathsf{q}})} = \hat{\mathsf{q}}_j \end{aligned}$$

so $(\hat{\mathsf{p}}, \hat{\mathsf{q}})$ is a fixed point. $\square$

Von Neumann's minimax theorem: proof

Proposition

If $(\hat{p}, \hat{q})$ *is a fixed point, then it is an equilibrium point.*

Von Neumann's minimax theorem: proof

Proof. Note that

$$\sum_{k \in [m]} \mathsf{p}_k V(\mathsf{p}, \mathsf{q}) = V(\mathsf{p}, \mathsf{q}) = \sum_{k \in [m]} \mathsf{p}_k V(\mathsf{e}_k, \mathsf{q}).$$

Thus, it is cannot be that $V(\hat{\mathsf{p}}, \hat{\mathsf{q}}) < V(\mathsf{e}_k, \hat{\mathsf{q}})$ for all $\hat{\mathsf{p}}_k > 0$. Thus there exists k^* such that $\hat{\mathsf{p}}_{k^*} > 0$ and

$$V(\hat{\mathsf{p}}, \hat{\mathsf{q}}) \geq V(\mathsf{e}_{k^*}, \hat{\mathsf{q}}) \implies \varphi_{k^*}(\hat{\mathsf{p}}, \hat{\mathsf{q}}) = 0.$$

Since $(\hat{\mathsf{p}}, \hat{\mathsf{q}})$ is a fixed point,

$$\hat{\mathsf{p}}_{k^*} = \Phi(\hat{\mathsf{p}}, \hat{\mathsf{q}})_{k^*} = \frac{\hat{\mathsf{p}}_{k^*} + \varphi_{k^*}(\hat{\mathsf{p}}, \hat{\mathsf{q}})}{1 + \sum_{i' \in [m]} \varphi_{i'}(\hat{\mathsf{p}}, \hat{\mathsf{q}})} = \frac{\hat{\mathsf{p}}_{k^*}}{1 + \sum_{i' \in [m]} \varphi_{i'}(\hat{\mathsf{p}}, \hat{\mathsf{q}})}.$$

Von Neumann's minimax theorem: proof

$$\hat{\mathsf{p}}_{k^*} = \frac{\hat{\mathsf{p}}_{k^*}}{1 + \sum_{i' \in [m]} \varphi_{i'}(\hat{\mathsf{p}}, \hat{\mathsf{q}})}$$

Then,

$$\begin{aligned}\sum_{i' \in [m]} \varphi_{i'}(\hat{\mathsf{p}}, \hat{\mathsf{q}}) = 0 &\implies \max\{V(\mathsf{e}_{i'}, \hat{\mathsf{q}}) - V(\hat{\mathsf{p}}, \hat{\mathsf{q}}), 0\} = 0 \\ &\implies V(\mathsf{e}_{i'}, \hat{\mathsf{q}}) \leq V(\hat{\mathsf{p}}, \hat{\mathsf{q}})\end{aligned}$$

so for any mixed strategy $\mathbf{p} \in \Delta^m$,

$$V(\mathsf{p}, \hat{\mathsf{q}}) = \sum_{k \in [m]} \mathsf{p}_k V(\mathsf{e}_k, \hat{\mathsf{q}}) \leq \sum_{k \in [m]} \mathsf{p}_k V(\hat{\mathsf{p}}, \hat{\mathsf{q}}) = V(\hat{\mathsf{p}}, \hat{\mathsf{q}}).$$

$\square$

Von Neumann’s minimax theorem: proof

Proof of von Neumann’s minimax theorem. Let T be as defined above and note that it is a continuous function defined from a compact convex set to itself. Then by the Brouwer’s theorem, there exists a fixed point $(\hat{\mathsf{p}}, \hat{\mathsf{q}})$ of T and thus there exists an equilibrium pair $(\hat{\mathsf{p}}, \hat{\mathsf{q}})$. $\square$

Yao's minimax principle: proof

Corollary

$$\max_{\mathbf{p} \in \Delta^m} \min_{\mathbf{q} \in \Delta^n} V(\mathbf{p}, \mathbf{q}) = \min_{\mathbf{q} \in \Delta^n} \max_{\mathbf{p} \in \Delta^m} V(\mathbf{p}, \mathbf{q})$$

Yao's minimax principle: proof

Proof. For all $\mathbf{p} \in \Delta^m$, we have that

$$\min_{\mathbf{q}\in\Delta^n} V(\mathbf{p},\mathbf{q}) \leq \min_{\mathbf{q}\in\Delta^n} \max_{\mathbf{p}\in\Delta^m} V(\mathbf{p},\mathbf{q})$$

so maximizing over $\mathbf{p} \in \Delta^m$ on both sides yields

$$\max_{\mathbf{p}\in\Delta^m} \min_{\mathbf{q}\in\Delta^n} V(\mathbf{p},\mathbf{q}) \leq \min_{\mathbf{q}\in\Delta^n} \max_{\mathbf{p}\in\Delta^m} V(\mathbf{p},\mathbf{q}).$$

Yao's minimax principle: proof

On the other hand, let $(\hat{\mathsf{p}}, \hat{\mathsf{q}})$ be an equilibrium pair. Then,

$$\begin{aligned}\min_{\mathsf{q}\in\Delta^n}\max_{\mathsf{p}\in\Delta^m} V(\mathsf{p},\mathsf{q}) &\leq \max_{\mathsf{p}\in\Delta^m} V(\mathsf{p},\hat{\mathsf{q}}) \leq V(\hat{\mathsf{p}},\hat{\mathsf{q}}) \\ &\leq \min_{\mathsf{q}\in\Delta^n} V(\hat{\mathsf{p}},\mathsf{q}) \leq \max_{\mathsf{p}\in\Delta^m}\min_{\mathsf{q}\in\Delta^n} V(\mathsf{p},\mathsf{q}).\end{aligned}$$

$\square$

Yao's minimax principle: proof

$$\min_{\mathcal{R}} \max_{x \in \mathcal{X}} \mathrm{cost}(\mathcal{R}, x) = \max_{\mathcal{D}} \min_{A \in \mathcal{A}} \mathrm{cost}(A, \mathcal{D}).$$

Proof. Consider the finite two-player zero-sum game where player 1's actions are the inputs $\mathcal{X}$, player 2's actions are the deterministic algorithms $\mathcal{A}$, and the payoff for actions $x \in \mathcal{X}$ and $A \in \mathcal{A}$ is given by $\mathrm{cost}(A, x)$. Then von Neumann's minimax theorem yields the above result. $\square$

Applications

Example: Property testing

Let $\varepsilon > 0$ and consider the problem P of taking an n-bit string $x \in \{0,1\}^n$ and correctly outputting whether it has less than an ε fraction of 0s or not, with probability at least $2/3$. Then for any n, any randomized algorithm solving P requires $\Omega(1/\varepsilon)$ queries.

Example: Property testing

Proof. We define an input distribution $\mathcal{D}$ as follows. Divide n into $1/\varepsilon$ blocks of size εn each. Then for each $i \in [1/\varepsilon]$, define the n-bit string y_i that is all 1s everywhere except on the ith block:

$$y_i = \underbrace{1\,1\;\ldots\;1}_{\text{block } 1}\underbrace{1\,1\;\ldots\;1}_{\text{block } 2}\ldots\underbrace{0\;0\;\ldots\;0}_{\text{block } i}\ldots\underbrace{1\,1\;\ldots\;1}_{\text{block } 1/\varepsilon}.$$

Note that on input y_i, the algorithm should output `NO`. We then draw from $\mathcal{D}$ as follows:

$$\mathcal{D} := \begin{cases} 1^n & \text{with probability } 1/2 \\ y_i & \text{with } i \in [1/\varepsilon] \text{ drawn uniformly with probability } \varepsilon/2 \end{cases}.$$

Example: Property testing

Fix a deterministic algorithm A making Q queries and solving the problem with probability at least 2/3.

- If A doesn't output YES on input 1^n, then it is already incorrect with probability 1/2. Then since A is deterministic, A outputs YES if it reads all 1s.
- A deterministically queries at most Q blocks, so with probability

$$\left(\frac{1}{\varepsilon} - Q\right) \cdot \frac{\varepsilon}{2} = \frac{1}{2} - \frac{Q\varepsilon}{2}$$

it outputs YES when it should have said NO.

Example: Property testing

It cannot be that $Q < 1/(3\varepsilon)$, since otherwise

$$\frac{1}{2} - \frac{Q\varepsilon}{2} > \frac{1}{2} - \frac{1}{6} = \frac{1}{3}$$

and so the failure rate is too high. Then by Yao's minimax principle, for any randomized algorithm $\mathcal{R}$ with expected query complexity $< 1/(3\varepsilon)$, there exists an input x such that the probability that $\mathcal{R}$ fails with probability at least $1/3$. $\square$

Example: Solving a system of equations

Consider the problem P of reading a matrix $\mathsf{A} \in \mathbb{R}^{n \times d}$ and a vector $\mathsf{b} \in \mathbb{R}^{n}$ and outputting a vector $\mathsf{x}' \in \mathbb{R}^{d}$ such that

$$\|\mathsf{A}\mathsf{x}' - \mathsf{b}\|_2 \leq 2 \min_{\mathsf{x} \in \mathbb{R}^d} \|\mathsf{A}\mathsf{x} - \mathsf{b}\|_2 \,,$$

with probability at least $2/3$. Then for each m, there exists A_m and b_m such that any randomized algorithm solving P reads $\Omega(m)$ in expectation.

Example: Solving a system of equations

Proof (sketch). We construct an input distribution where half the time, the algorithm must output $\mathsf{x}' = 1^d$ and the rest of the time, we place a single very large R in a random entry of A so that $\mathsf{x}' = 1^d$ fails to be a successful output. The algorithm must read $\Omega(m)$ entries to determine whether R is in the matrix or not with constant probability, so we conclude by Yao's minimax principle. $\square$

References

1. Kuhn, Harold William. *Lectures on the Theory of Games (AM-37)*. Princeton University Press, 2009.
2. Motwani, Rajeev, and Prabhakar Raghavan. *Randomized algorithms*. Chapman & Hall/CRC, 2010.
3. CSE 598A Sublinear Algorithms Spring 2012 (Penn State University, Prof. Sofya Raskhodnikova)
4. 15-895 Algorithms for Big Data Fall 2017 (Carnegie Mellon University, Prof. David Woodruff)

Questions?