The Query and Communication Complexity of Cake Cutting

Simina Brânzei and Noam Nisan
Hebrew University of Jerusalem

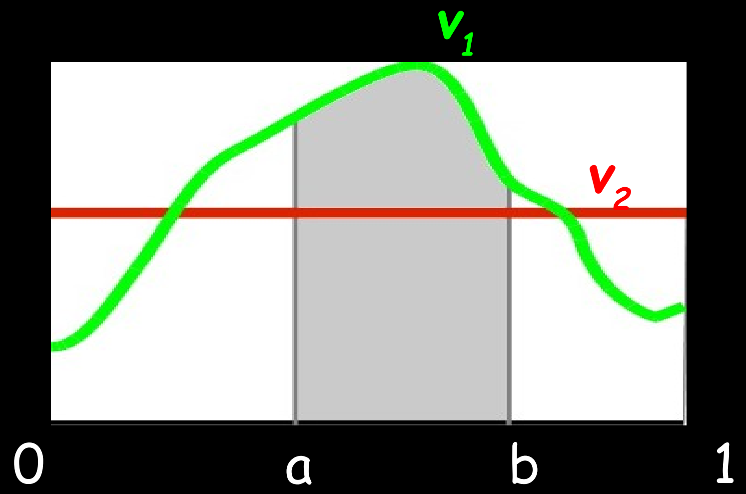
August 9, 2017
Higher School of Economics, St Petersburg

Cake Cutting: metaphor for fair division

The cake is the interval [0, 1]

Interested parties (players) N = {1,..., n}

Each player i has a private (non-atomic) value density function v_i. Valuation of a piece: integral of the value density



Can be seen as the limit of a model of indivisible goods when number of goods goes to infinity.

Goal : Find allocation A = (A_1, ..., A_n), i.e. assignment of (disjoint) pieces to players, where a piece is a union of intervals

Fairness

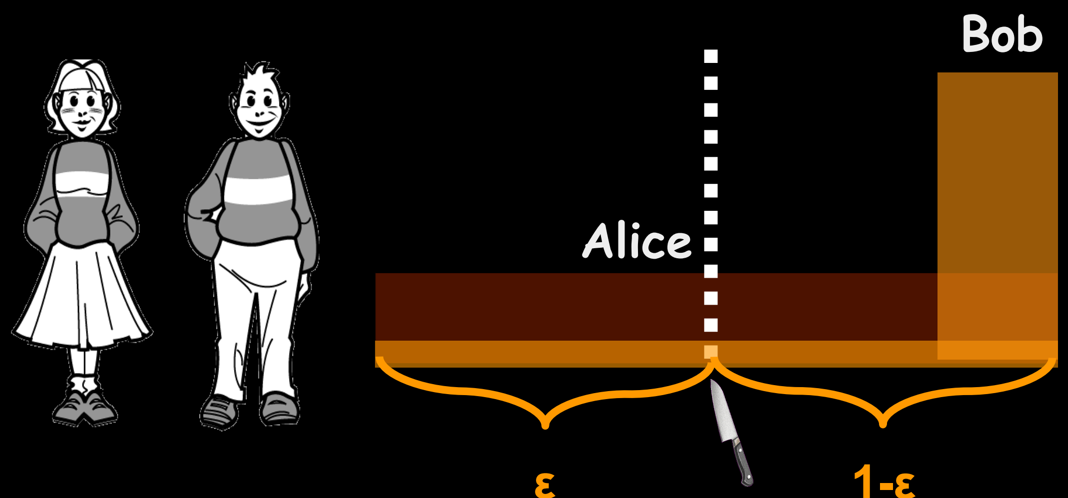
Proportional: Each player i gets their minimum fair share: $V_i(A_i) \geq 1/n$

Envy-Free: Nobody prefers anyone else's piece to its own: $V_i(A_i) \geq V_i(A_j)$

Equitable: All the players are equally happy with their piece : $V_i(A_i) = V_k(A_k)$

Perfect: Each player values every piece at exactly 1/n : $V_i(A_k) = 1/n$

Cut-and-Choose : Alice cuts the cake in two pieces of equal value to her. Bob chooses his favorite piece, and Alice takes the remainder.



Query Model

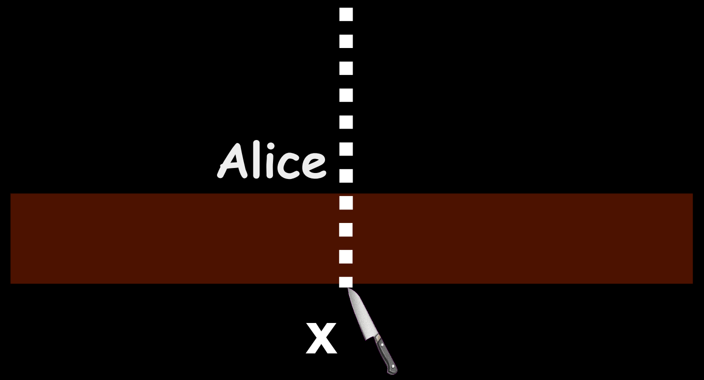
Private valuations : center interacts with the players; needs to extract enough information to output a fair allocation. The standard (RW) query model :

CUT_i (v) : Player i cuts at point x where $V_i(0,x) = v$; x becomes a cut point

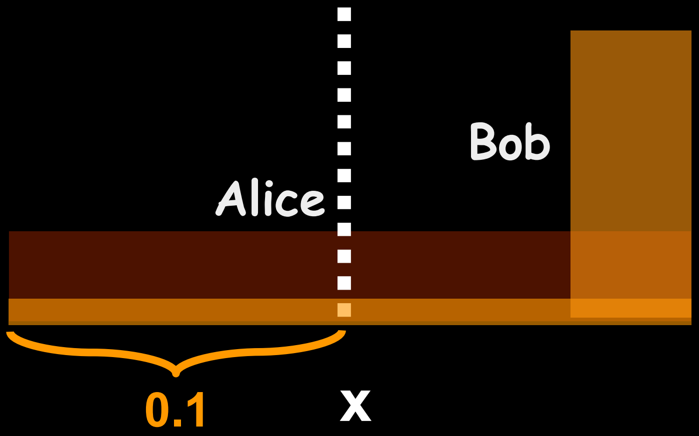
$EVAL_i$ (x) : Player i returns value v so that $V_i(0, x) = v$, where x is a cut point

Example :

- Ask Alice $CUT_A(0.5)$: Alice cuts the cake in half



- Ask Bob $EVAL_B(x)$: Bob evaluates the left piece demarcated by Alice



Query Complexity

The center can ask the players to discretize the cake in many cells, each worth at most ε/n^2, then assemble an ε-fair allocation offline

→ high communication + high fragmentation.

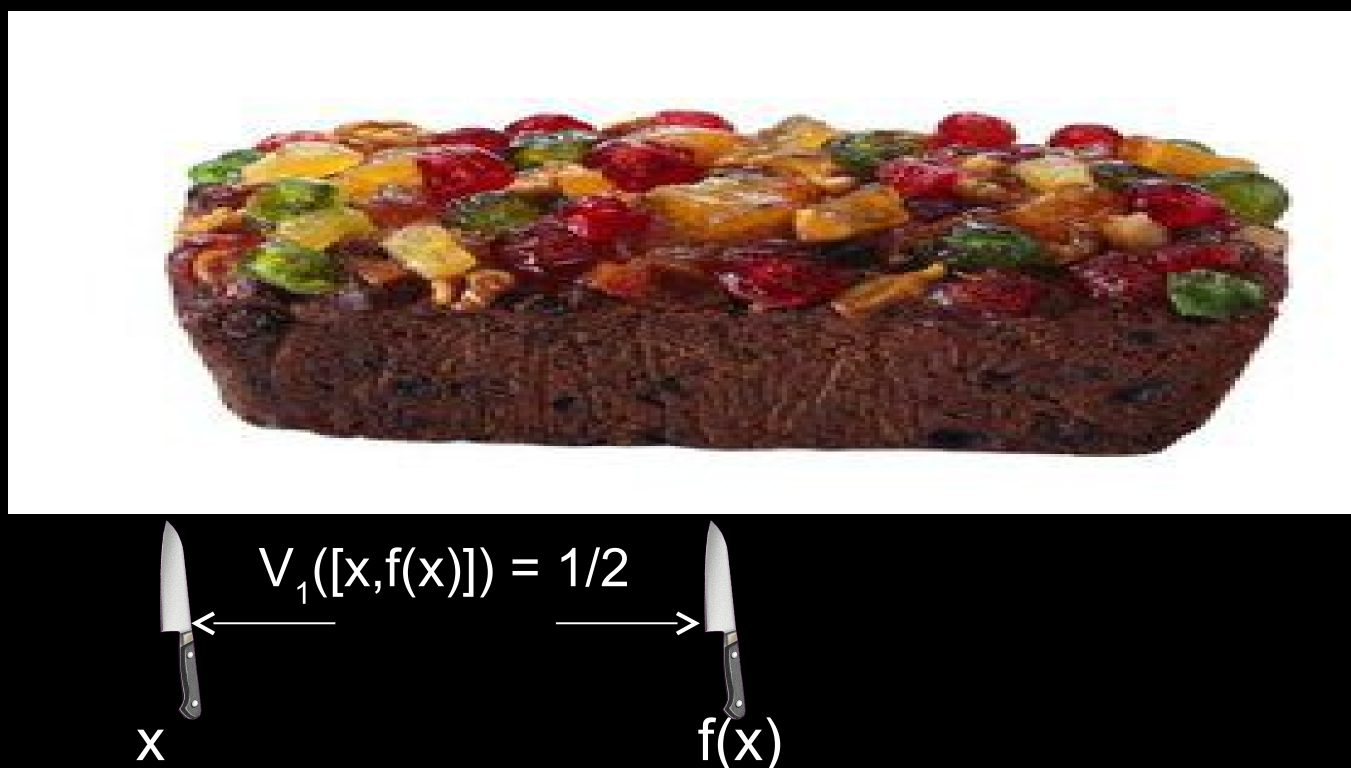
The problem of fair division is much more interesting when spatial structure matters – e.g. aim for connected pieces (or minimize number of cuts).

- Proportional, envy-free, and equitable allocations with connected pieces exist for all n; perfect allocations exist with n(n-1) cuts.
- Via some fixed point theorem (Sperner, Borsuk-Ulam)

Query Complexity: Summary

Fairness notion	Number of players	Upper bound	Lower bound
ϵ-envy-free (connected)	$n = 2$	1	1
	$n = 3$	$O(\log \epsilon^{-1})$ $(*)$	$\Omega(\log \epsilon^{-1})$ $(*)$
	$n \geq 4$	$O(n/\epsilon)$ $(*)$	$\Omega\left(\log \epsilon^{-1}\right)$ $(*)$
ϵ-perfect (minimum cuts)	$n = 2$	$O(\log \epsilon^{-1})$ $(*)$	$\Omega(\log \epsilon^{-1})$ $(*)$
	$n \geq 3$	$O\left(n^3/\epsilon\right)$ [BM15]	$\Omega\left(\frac{\log \epsilon^{-1}}{\log\log \epsilon^{-1}}\right)$ [PW17]
ϵ-equitable (connected)	$n = 2$	$O(\log \epsilon^{-1})$ [CP12]	$\Omega(\log \epsilon^{-1})$ $(*)$
	$n \geq 3$	$O\left(n\left(\log n + \log \epsilon^{-1}\right)\right)$ [CP12]	$\Omega\left(\frac{\log \epsilon^{-1}}{\log\log \epsilon^{-1}}\right)$ [PW17]
envy-free (exact)	$n \geq 2$	$O\left(n^{n^{n^{n^{n^{n}}}}}\right)$ [AM16]	$\Omega(n^2)$ [Pro09]
proportional (exact)	$n \geq 2$	$O\left(n \log n\right)$ [EP84]	$\Omega(n \log n)$ [WS07, EP06b]

Perfect Allocations: Austin's procedure



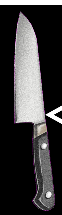
Perfect Allocations: Austin's procedure

$V_1([x,f(x)]) = 1/2$

$V_2([x,f(x)]) >? 1/2$

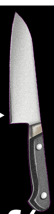
x f(x)

Perfect Allocations: Austin's procedure



$V_1([x,f(x)]) = 1/2$

$V_2([x,f(x)]) = 1/2$



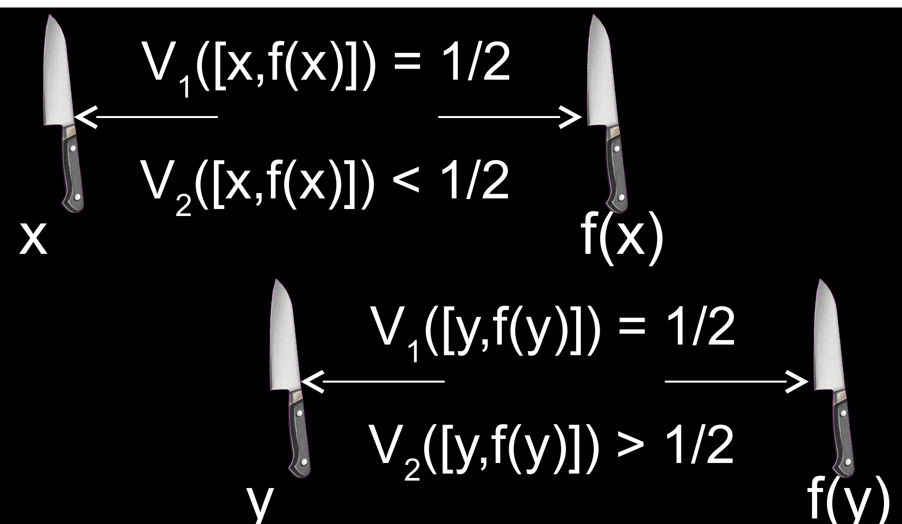
x

f(x)

Perfect Allocations: n=2 players

Theorem: Computing an ε-perfect allocation for n=2 players with two cuts in the (RW) query model takes $\Theta(\log(1/\varepsilon))$ queries.

(Proof) Upper bound: binary search on the position of the first knife.



Perfect Allocations: n=2 players

(Proof) Lower bound: Maintain throughout execution 2 intervals I and J:

- the protocol has not made any cuts inside I and J,
- any allocation obtained with cuts outside I and J is far from perfect, and
- the distance to a perfect allocation cannot decrease much with any Cut query

	$[0, x]$	$[x, x+a]$	$[x+a, y]$	$[y, y+a]$	$[y+a, 1]$
V_1	x	a	$0.5-a$	a	b
V_2	c	d	$0.5-2d$	$3d$	e

Perfect Allocations: n=2 players

(Proof) Lower bound (cont).

	$[0,x]$	$[x,x+a]$	$[x+a,y]$	$[y,y+a]$	$[y+a,1]$
V_1	x	a	$0.5-a$	a	b
V_2	c	d	$0.5-2d$	$3d$	e

If a Cut query falls outside [x,x+a] or [y,y+a], answer consistent with history.

Else, say player 1 gets $Cut_1(\alpha)$:

- **Case 1: $\alpha \in [c, c + d/2]$.** Let m = x + a/2, n = x + 51a/100, p = y+a/2, q = y + 51a/100.

	$[0,x]$	$[x,m]$	$[m,n]$	$[n,x+a]$	$[x+a,y]$	$[y,p]$	$[p,q]$	$[q,y+a]$	$[y+a,1]$
V_1	x	$a/2$	$a/100$	$49a/100$	$0.5-a$	$a/2$	$a/100$	$49a/100$	b
V_2	c	$d/2$	$d/8$	$3d/8$	$0.5-2d$	$11d/8$	$3d/8$	$10d/8$	e

Perfect Allocations: n=2 players

(Proof) Lower bound (cont).

	$[0, x]$	$[x, x+a]$	$[x+a, y]$	$[y, y+a]$	$[y+a, 1]$
V_1	x	a	$0.5-a$	a	b
V_2	c	d	$0.5-2d$	$3d$	e

- **3 more cases:** α ∈ [c+d/2,c+d], [0.5+c-d, 0.5+c+d/2], [0.5+c+d/2,0.5+c+2d]

Starting configuration:

	$[0, 0.2]$	$[0.2, 0.3]$	$[0.3, 0.7]$	$[0.7, 0.8]$	$[0.8, 1]$
V_1	0.2	0.1	0.4	0.1	0.2
V_2	0.15	0.1	0.3	0.3	0.15

Connected Equitable Allocations: n=2 players

Theorem: Computing a connected ε-equitable allocation for n=2 players takes Θ(log(1/ε)) queries.

(Proof) Upper bound: Cechlarova and Pillarova 2012.

Lower bound: Maintain throughout execution an interval I such that

- the protocol has not made any cut inside I
- the distance to an equitable allocation by cutting outside I is high, and
- the interval I cannot be diminished by much with any single Cut query

	$[0, x]$	$[x, y]$	$[y, 1]$
V_1	$0.5 + a$	$b - a$	$0.5 - b$
V_2	$0.5 - b$	$b - a$	$0.5 + a$

* 0 < a < b < 0.5

Connected Equitable Allocations: n=2 players

(Proof) Lower bound (cont):

	$[0, x]$	$[x, y]$	$[y, 1]$
V_1	$0.5 + a$	$b - a$	$0.5 - b$
V_2	$0.5 - b$	$b - a$	$0.5 + a$

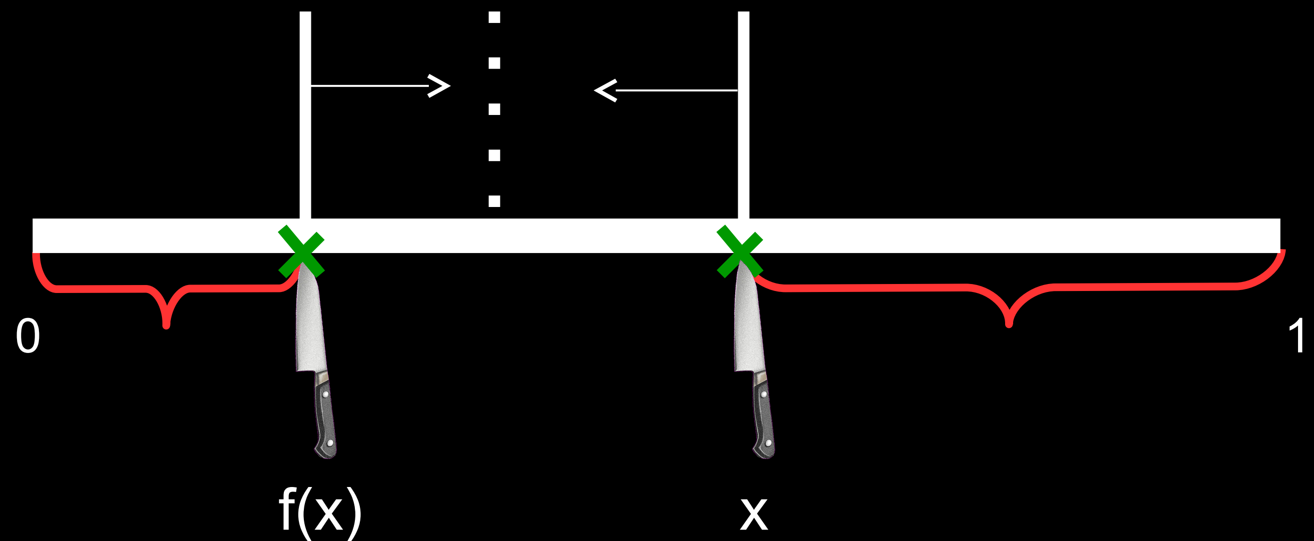
Starting configuration: a = 0.05 and b = 0.06

	$[0, 0.4]$	$[0.4, 0.6]$	$[0.6, 1]$
V_1	0.55	0.01	0.44
V_2	0.44	0.01	0.55

Connected Envy-free Allocations: n=3 players

Theorem: Computing a connected ε-envy-free allocation for n=3 players takes Θ(log(1/ε)) queries.

(Proof) Upper bound: We simulate a moving knife procedure due to Barbanel and Brams in the RW model.



Connected Envy-free Allocations: n=3 players

(Proof) Lower bound: Use valuations drawn from class of "generalized rigid measure systems":

- the density of each measure is bounded: $1/\sqrt{2} < v_i(x) < \sqrt{2}$, for each player i
- there exist points $x, y \in [0, 1]$, such that for each player i there exist $0 < s_i < 1/3 < t_i < 1/2$ and the matrix of valuations satisfies the constraints in the table:

	$[0, x]$	$[x, y]$	$[y, 1]$
V_1	t_1	t_1	s_1
V_2	s_2	t_2	t_2
V_3	t_3	s_3	t_3

**Stromquist first introduced a variant of rigid measure systems to show an impossibility for RW protocols.*

Connected Envy-free Allocations: n=3 players

(Proof) Lower bound (cont): Maintain throughout execution two intervals I, J:

- there are no cut points inside I and J, and any allocation that does not use cuts in I and J has high envy
- the intervals I, J cannot be diminished much with a single Cut query

	$[0,x]$	$[x,x+k]$	$[x+k,y]$	$[y,y+k]$	$[y+k,1]$
V_1	ℓ_1	k	ℓ_1	k	m_1
V_2	m_2	k	ℓ_2	k	ℓ_2
V_3	ℓ_3	k	m_3	k	ℓ_3

Starting configuration:

	$[0,0.34]$	$[0.34,0.35]$	$[0.35,0.67]$	$[0.67,0.68]$	$[0.68,1]$
V_1	0.35	0.01	0.35	0.01	0.28
V_2	0.28	0.01	0.35	0.01	0.35
V_3	0.35	0.01	0.28	0.01	0.35

Moving Knife Protocols

Moving Knife Step: devices 1 ... K (“knives” and “triggers”) move along the cake as time proceeds from α to ω. The value of each device j, x_j, is a function of time, of the values of devices 1…j-1, and of the valuations of the players for pieces demarcated by knives at that time.

- value of knife: its position
- value of trigger: arbitrary.

A moving knife step ends when a trigger “fires”, i.e. when $x_j(t) = 0$ for some j, t.

Outcome of a step:

- **index of a trigger j with $x_j(\alpha) * x_j(\omega) \leq 0$**
- **a time t where $x_j(t) = 0$**
- **values of all other devices at this time**.

Moving Knife Protocols

Moving Knife Protocol: has finite number of steps, each of which is either an RW query or a moving knife step.

Example: Austin's procedure can be cast a single moving knife step, with 3 Devices:

- Knife 1: position x_1 = time
- Knife 2: position x_2 depends on time and valuation of player 1
- Trigger: value $x_3 = V_1(\text{knife}_1, \text{knife}_2) - 0.5$

Theorem (informal): Fair Moving Knife protocols with a constant number of steps can be simulated approximately with $O(\log(1/\varepsilon))$ queries.

Moving Knife Protocols

Main open question: super-logarithmic query complexity lower bound for computing connected ε-envy-free allocations for $n \geq 4$ players or perfect allocations for $n \geq 3$ players.

- This would imply no moving knife protocol can exist.

Beyond infinite precision models: A few words on communication complexity

We need bounded density: $v_i(x) < D$, for some constant D.

This is the correct interpretation of no-atoms in the communication model

For simplicity n is arbitrary but fixed.

Communication complexity: Each player knows its own input v_i. An F-fair protocol is a tree that on every input $v = (v_1, \ldots, v_n)$ reaches a leaf marked with an allocation that is F-fair for v.

Beyond infinite precision models: A few words on communication complexity

The **deterministic communication complexity** of F, D(F) :

- the number of bits sent on the worst case input by the best communication protocol that computes F-fair allocations.

The **randomized communication complexity** of F, $R_\varepsilon(F)$:

- the worst case number of bits sent by the best randomized protocol that computes F-fair allocations with probability 1 – ε.

  (error probability taken over the random choices of the protocol on the worst case input).

Communication complexity

3 classes:

"Easy" problems: Admit bounded protocols in the RW model.

Theorem **(upper bound)**: The following problems have communication protocols with a constant number of rounds of communication $O(\log(1/\varepsilon))$ per round:

- For any fixed n, a connected ε-proportional allocation among n players.
- For any fixed n, for some constant C that depends on n, an ε-envy-free allocation with at most C cuts, for n players.

Theorem **(lower bound)**: Every (deterministic or randomized) protocol for computing a (not necessarily connected) ε-proportional allocation among $n \geq 2$ players requires $\Omega(\log(1/\varepsilon))$ bits of communication.

Communication complexity

“Medium” problems: Admit moving knife protocols:

Theorem **(upper bound)**: The deterministic communication complexity of the following problems is $O(\log^2 \varepsilon^{-1})$:

- ε-perfect allocation with 2-cuts between n = 2 players,
- a connected ε-equitable allocation between n = 2 players,
- a connected ε-envy-free allocation among n = 3 players.

The randomized communication complexity of these problems is $O(\log \varepsilon^{-1} \log \log \varepsilon^{-1})$.

Communication complexity

“Medium” problems: Admit moving knife protocols:

Theorem **(lower bound)**: Any (deterministic or randomized) protocol for finding

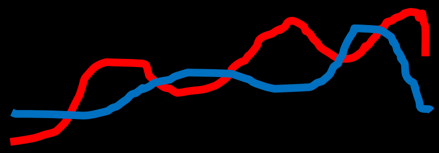
- an ε-perfect allocation with 2-cuts between $n = 2$ players
- a connected ε-equitable allocation between $n = 2$ players

using rounds of communication of polylog(ε^{-1})-bits each requires $\Omega(\log \varepsilon^{-1} / \log \log \varepsilon^{-1})$ rounds of communication.

Communication complexity

Medium problems are intuitively equivalent to the **Crossing Problem**:

Alice gets sequence of numbers $x_0, x_1, \ldots, x_m$ with $0 \leq x_i \leq m$ and Bob gets $y_0, y_1, \ldots, y_m$ with $0 \leq x_i \leq m$, where $x_0 \leq y_0$ and $x_m \geq y_m$.
Goal: find an index i such that either both $x_{i-1} \leq y_{i-1}$ and $x_i \geq y_i$ or that both $x_{i-1} \geq y_{i-1}$ and $x_i \leq y_i$.



Bounds on the communication complexity of the crossing problem
\+ reductions between the fair division problems and crossing.

Communication complexity

"Hard" problems: No moving knife protocols known.

Natural candidates:

- connected ε-envy-free allocation for $n \geq 4$ players
- perfect for $n \geq 3$ players

Main open question: Separate the "hard" from "medium" → Show super-polylogarithmic lower bounds on the communication complexity of these problems.

THANK YOU