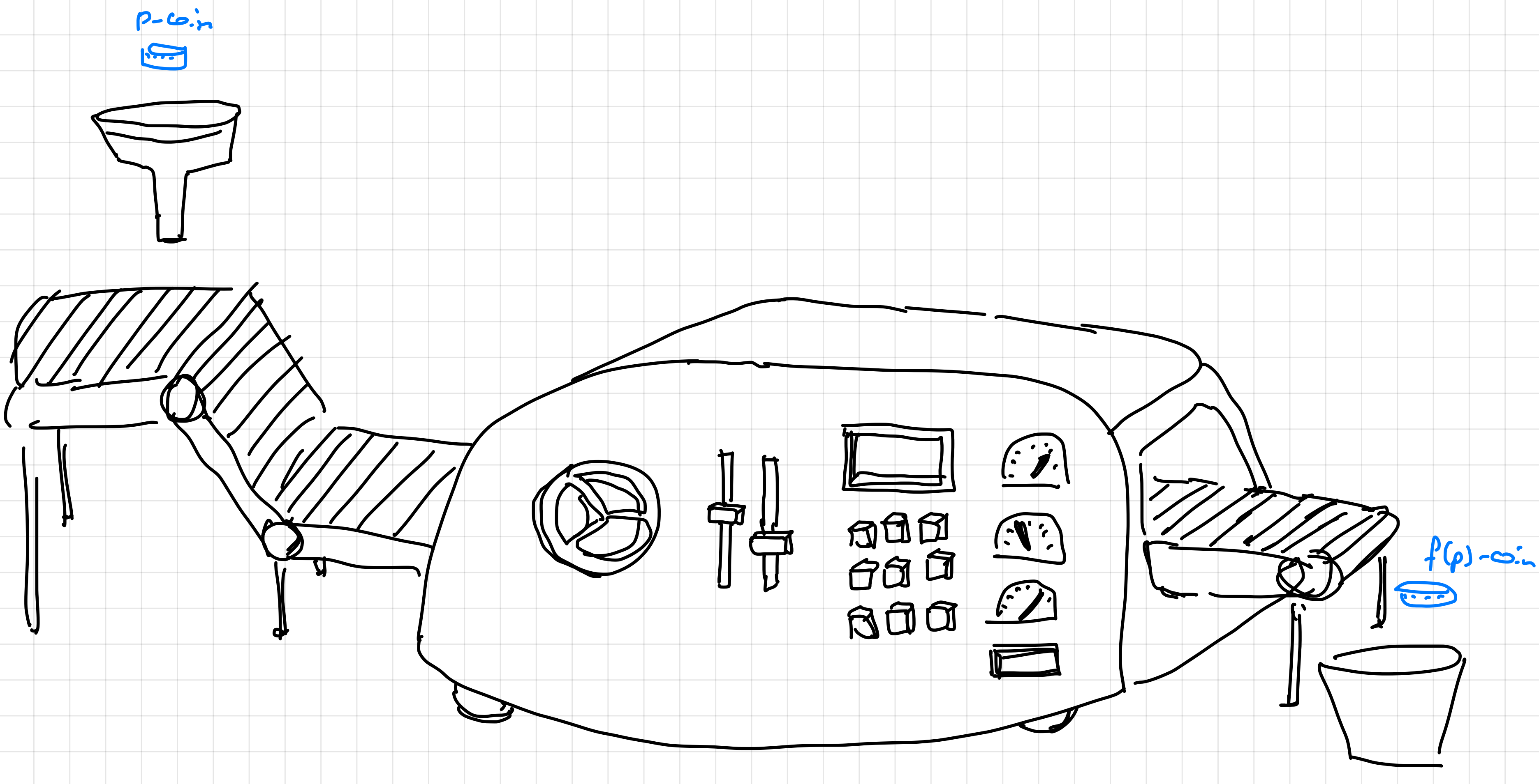
Bernoulli Factories

Renato Paes Leme
(Google Research)

www.renatoppl.com/bernoulli

(Bernoulli wearing a factory hat)

Bernoulli Factories



Examples of Simple Factories

① $f(p) = p^2$

② $f(p) = p(1-p)$

③ Bernstein monomials:

Single Parameter Bernoulli Factories

Given a function $f : S \subseteq (0,1) \to [0,1]$,

design an algorithm (decision tree)

that samples an $f(p)$-coin from a p-coin

(of unknown bias).

[Keane O'Brien 1994]

Von Neumann's Procedure

Sample an unbiased coin from a biased one

4 $\quad f(p) = \frac{1}{2} \quad \text{for } p \in (0,1)$

Von Neumann's Procedure

Sample a coin of bias x from a biased one

5 $f(p) = x$ for some fixed $x \in [0,1)$

Bernstein Polynomials

(6) Bernstein polynomials $f(p) = \cdots$

(7) $f(p) = \dfrac{p}{2-p}$

Other Examples

(f) Moment Generating Functions, e.g. $f(p) = e^{p-1}$

Complete Characterization

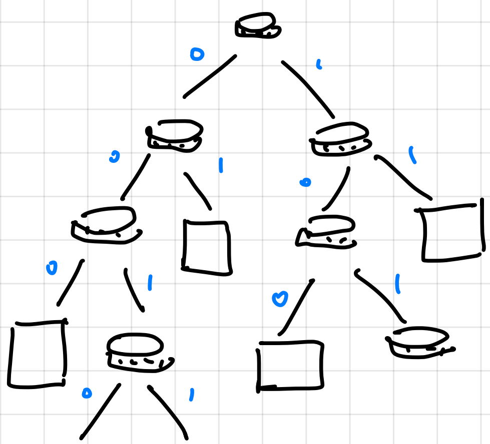
Which functions $f : S \subseteq (0,1) \longrightarrow [0,1]$

admit Bernoulli factories ?

1. What is a general factory ?

2. Examples of functions that don't admit factories

Generic Factory



Factory = Decision Tree

Nodes:

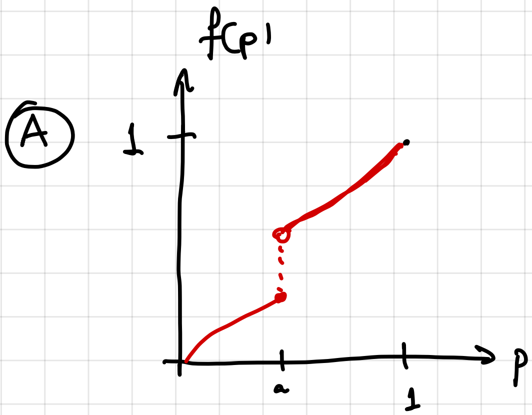
- p (p-coin)
- c (helper coin)
- $\boxed{0}$ $\boxed{1}$ (output)

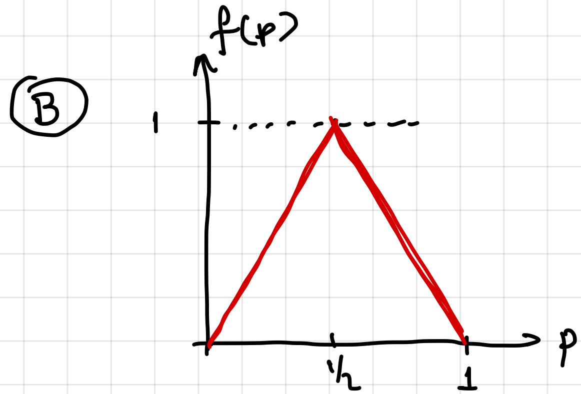
- Finite vs Infinite:

- Termination:

- $\mathbb{P}[\mathcal{F}(p)=1] =$

Functions Without Factories

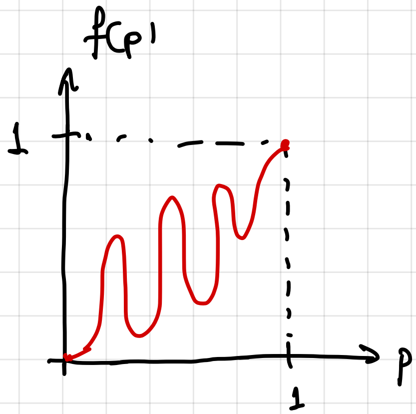




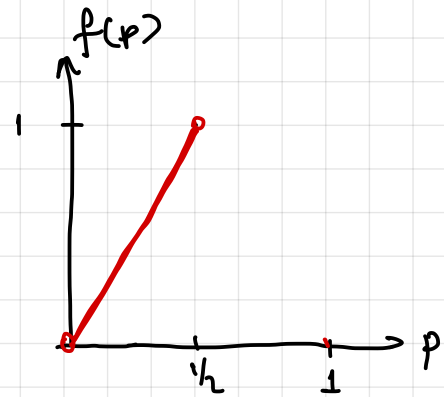
Functions Without Factories

Test Your Intuition. Which of these functions has a factory?

C



D



$f : (0, 1/2) \to (0, 1)$

Theorem (Keane, O'Brien)

Function $f: S \subseteq (0,1) \to (0,1)$ admits a factory if:

(1) it is continuous

(2) f is constant or poly-bounded

$\exists n$ s.t. $\min\left[f(p), 1-f(p)\right] \geq \min\left(p^n, (1-p)^n\right)$

Proof: Necessity:

Theorem (Keane, O'Brien)

Function $f: S \subseteq (0,1) \to (0,1)$ admits a factory if:

(1) it is continuous

(2) f is constant or poly-bounded

$\exists n$ s.t. $\min[f(p), 1-f(p)] \geq \min(p^n, 1-p^n)$

<u>Lemma</u>: Given f as in Thm, there is a function g implementable by a finite factory s.t.

$$0 \leq f(p) - \frac{1}{4} g(p) \leq \frac{3}{4}$$

s.t. $\frac{4}{3}\left(f(p) - \frac{1}{4} g(p)\right)$ is poly-bounded.

Lemma $\Rightarrow$ Thm

Theorem (Keane, O'Brien)

Function $f : S \subseteq (0,1) \to (0,1)$ admits a factory if:

(1) it is continuous

(2) f is constant or poly-bounded

$\exists n$ s.t. $\min[f(p), 1-f(p)] \geq \min(p^n, 1-p^n)$

<u>Lemma</u>: Given f as in Thm, there is a function g implementable by a finite factory s.t.

$$0 \leq f(p) - \frac{1}{4} g(p) \leq \frac{3}{4}$$

s.t. $\frac{4}{3}\left(f(p) - \frac{1}{4} g(p)\right)$ is poly-bounded.

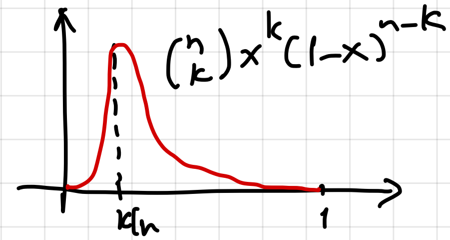
Lemma (Proof by picture)

Alternative Proof (Nacu-Peres)

① Bernstein approximation: given continuous function $f : [0,1] \to \mathbb{R}$

define $Q_n[f](x) = \sum_{k=0}^{n} f\left(\frac{k}{n}\right)\binom{n}{k} x^k (1-x)^{n-k}$

then: $Q_n[f](x) \to f(x)$ uniformly.



② Polya's Thm: If $g(x,y)$ is a real homogeneous polynomial such that.

$g(x,y) > 0 \;\; \forall x, y > 0$ then $\exists n$ s.t all coefficients of $(x+y)^n g(x,y)$

are non-negative.

Idea: If $f : (0,1) \to [\varepsilon, 1-\varepsilon]$ use ① to construct approximators and

② to stitch the functions together in a series.

Fast Simulation

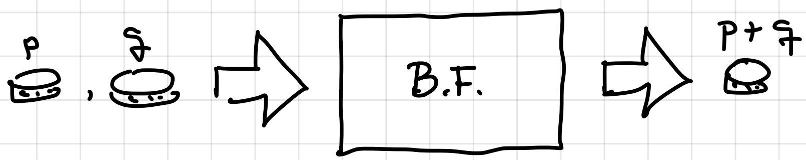
(Mossel-Peres 2002)
(Nacu-Peres 2005)

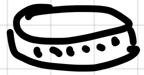
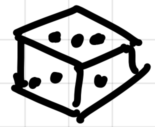
N = # coins tossed until output (random variable)

Property of f		Property of N
continuous	$\Leftrightarrow$	$N < \infty$ a.s. (i.e. it terminates)
real-analytic	$\Leftrightarrow$	$\mathbb{P}(N \leq n) \leq O(\rho^n)$ exponential tail. $\rho = \rho(p) < 1$.
Lipschitz	$\Leftarrow$	$\mathbb{E}[N] < \infty$
rational	$\Leftrightarrow$	N via finite automata

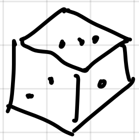
Problem: Sum of Two Coins

Given two coins p, q with the promise that $p+q \leq 1-\varepsilon$, sample from a $(p+q)$-biased coin.



Coins  to Dice 

(Mossel - Peres 2002)
(Dughmi et al 2017)
(Morina et al 2020)

 $\longrightarrow$ 

p_1 , p_2 , … p_n

$$X = \begin{cases} 1 & \text{w.p. } p_1 \\ 2 & \text{w.p. } p_2 \\ \dots & \\ n & \text{w.p. } p_n \end{cases}$$

(Bernoulli Race).

Rational Functions

Factory for $f(p) = \dfrac{\sum_{i=0}^{k} a_i p^i}{\sum_{i=0}^{k} b_i p^i}$ $f: (0,1) \to (0,1)$.

① Bernoulli race between two coins  and 

Rational Functions

Factory for $f(p) = \dfrac{\sum_{i=0}^{k} a_i p^i}{\sum_{i=0}^{k} b_i p^i}$ $\quad f: (0,1) \to (0,1)$.

② <u>Polya's Thm:</u> If $q(x,y)$ is a real homogeneous polynomial such that $q(x,y) > 0 \;\; \forall x,y > 0$ then $\exists n$ s.t. all coefficients of $(x+y)^n q(x,y)$ are non-negative.

$a(x) = \sum_i a_i x^i$ $\qquad$ $A(x,y) =$

$b(x) = \sum_i b_i x^i$ $\qquad$ $B(x,y) =$

Rational Functions

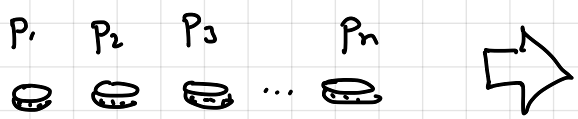
$$\mathbb{P}(X=i) = \frac{f_i(x_1 \dots x_n)}{g_i(x_1 \dots x_n)}$$ rational.

Polya's Thm: If $g(x_1 \dots x_m)$ is a real homogeneous polynomial such that $g(x_1, \dots x_m) > 0$ for $x_i > 0 \ \forall i$ then $\exists n$ s.t. all coefficients of $(x_1 + \dots + x_m)^n g(x_1 \dots x_m)$ are positive.

Part II :

Combinatorial Factories

(Niazadeh, PL, Schneider 2021)



random combinatorial object

Random Combinatorial Objects

(1) k - Subset

(2) Matchings

Random Combinatorial Objects

(3) spanning-trees

(4) s-t-flows

Random Combinatorial Objects

(*) Vertices of a polytope

Polytope $\mathcal{P}$ contained in $[0,1]^n$

V = vertices of $\mathcal{P}$

Given coins $p_1 \cdots p_n$ s.t. $p = (p_1 \cdots p_n) \in \mathcal{P} \cap (0,1)^n$

output a vertex $v \in V$ s.t. $\mathbb{E}[v] = p$.

General Bernoulli Factory

Output set : V

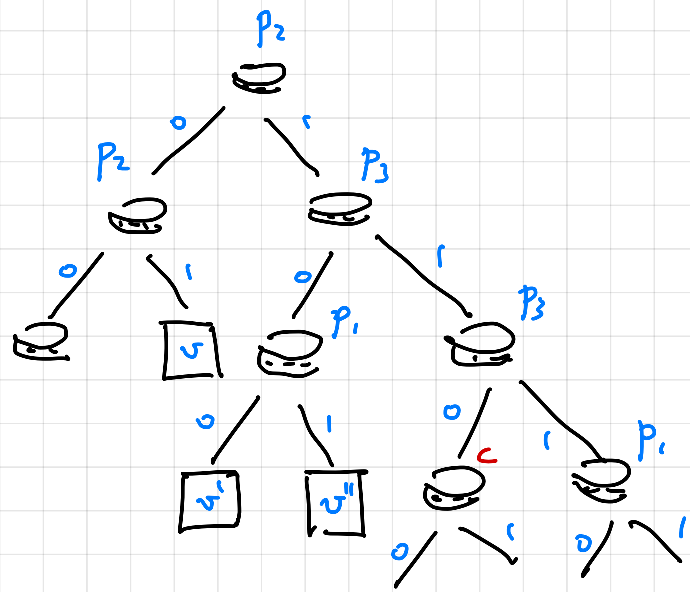
Input coins: p_1 p_2 ... p_3

<u>Nodes:</u>

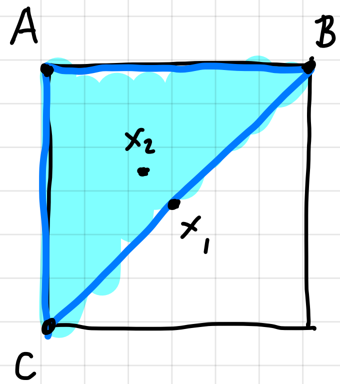
p_i - coin

helper coin c (constant c)

output node $\boxed{v}$ $(v \in V)$



Necessary Conditions

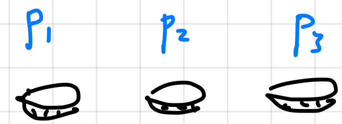


Output at x_1 :

Output at x_2 :

Factory for k-Subset

Natural Algorithms:



Factory for k-Subset

p_1 p_2 p_3

Sampford Sampling

$$\mathbb{P}[\text{sample } S] = \prod_{i \in S} p_i \prod_{i \notin S} (1-p_i) \cdot$$

<u>Algorithm</u>

Flip each coin X_i

$S = \{i \,;\, X_i = 1\}$

if $|S| \neq k$ restart

output S

A Guess

For every polytope $P = \{x \in [0,1]^n ; Wx = b\}$ we can find polynomials $f_v(p_1 \cdots p_n)$ such that:

(1) $\sum_{v \in V} f_v(p_1 \cdots p_n) \cdot (p - v) = 0 \quad \forall p \in P.$

(2) $f_v(p_1 \cdots p_n)$ are implementable by a Bernoulli factory

$$f_v(p) =$$

Solve a Program

$f_v(p) =$

$$\sum_v f_v(p) \cdot (p - v) = 0 \quad \forall p \quad \text{s.t.} \quad Wp = b$$

sage

(Manipulate polynomials)

SciPy

(Solve linear programs)

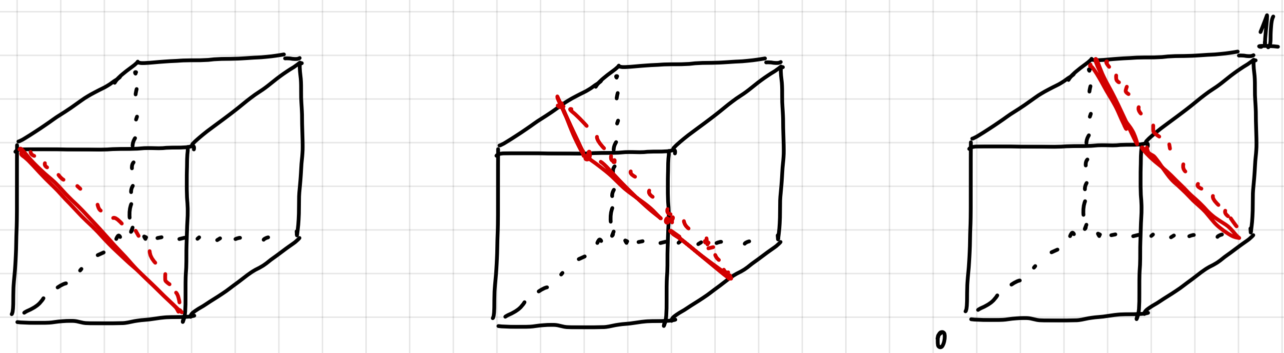
Next Simplest Example

Towards co-dimension 1: $P = \{x \in [0,1]^n ;\ w^T x = b\}$.

Simplest case we didn't understand: $\sum_i x_i = \alpha \quad \alpha \notin \mathbb{Z}$.

Example: $n = 3 \qquad \alpha = 1.5$

What are the vertices?



$\alpha = 1$	$\alpha = 1.5$	$\alpha = 2$
$\lvert V \rvert = 3$	$\lvert V \rvert = 6$	$\lvert V \rvert = 3$
permutations of $(1,0,0)$	permutations of $(1,\frac{1}{2},0)$	permutations of $(1,1,0)$

Next Simplest Example

Towards co-dimension 1: $P = \{x \in [0,1]^n ;\ w^T x = b\}$.

Simplest case we didn't understand: $\sum_i x_i = \alpha \qquad \alpha \notin \mathbb{Z}$.

In general: $n\ ,\ \alpha = k - \varepsilon \qquad 0 < \varepsilon < 1$

$V =$ permutations of $(\underbrace{1,1,\dots,1}_{k-1}, 1-\varepsilon, \underbrace{0,\dots 0}_{n-k})$

Start with $n = 3 \quad k = 1 + \varepsilon$ (simplest unknown example).

If polynomials exist we should be able to find it. After sterms of the solutions for very long:

$f_v(p) =$

Co-dimension One

$P = \{ x \in [0,1]^n ; \; w^T x = b \}.$

Generic hyperplane $\Longrightarrow$ Each vertex has one special index.
avoiding $\{0,1\}^n$

$v_i \in (0,1)$

$$f_v(p) = |w_i| \cdot p_i (1-p_i) \prod_{j : v_j = 1} p_j \prod_{j : v_j = 0} (1-p_j)$$

Non-generic hyperplanes:

Co-dimension One

$P = \{x \in [0,1]^n ;\ w^T x = b\}.$ $\quad f_v(x) = |w_i|\ x_i(1-x_i) \prod_{j: v_j = 1} x_j \prod_{j: v_j = 0} (1-x_j)$

Proof idea: $\sum_v (v - x) f_v(x) = 0$

$w(A) = \sum_{j \in A} w_j$

Vertices $v \in V \leftrightarrow$ pairs (A, i) s.t. $\dfrac{b - w(A)}{w_i} \in (0,1)$

$v = (A, i)$

Define $P_A(x) = \prod_{i \in A} x_i \prod_{i \notin A} (1 - x_i)$ s.t. $f_{(A,i)}(x) = |w_i| \cdot x_i\, P_A(x)$

Look at coordinate 1 of $\sum_{(A,i)} (v_1 - x_1) f_v(x) = 0$

$$\sum_{\substack{(A,i)\\ 1 \notin A\\ 1 \neq i}} (0 - x_i) f_{(A,i)}(x) + \sum_{\substack{(A,i)\\ 1 \in A}} (1 - x_i) f_{(A,i)}(x) + \sum_{\substack{(A,i)\\ i = A}} \left(\frac{b - w(A)}{w_i} - x_1 \right) f_{(A,i)}(x) = 0$$

Any dimension

$P(x) = \{ x \in \mathbb{R}^n ; Wx = b \}$ W: $k \times n$ matrix of rank k.

Vertex v of P corresponds to a partition: (A, S, B)

$A = \{ i ; v_i = 0 \}$

$B = \{ i ; v_i = 1 \}$

$S = \{ i ; 0 < v_i < 1 \}$

$$W = \begin{array}{ccc} \overbrace{}^{A} & \overbrace{}^{S} & \overbrace{}^{B} \\ W_A & W_S & W_B \end{array}$$

$v_A = 0_A \quad v_B = \mathbb{1}_B$

$v_S = W_S^{-1} (b - W_B \mathbb{1})$

Generic subspace $Ax = b$:

$$f_v(p) = \boxed{} \prod_{i \in A} p_i \prod_{i \in B} (1 - p_i) \prod_{i \in S} p_i (1 - p_i)$$

Factory for Matching

Matching is a bijection $\pi : [n] \to [n]$.

$$f_\pi(p) = \prod_{i=1}^{n} p_{i\pi(i)} \sum_{T \in Arb_1} \prod_{(u,v)\in T} x_{u,\pi(v)}$$

Algorithm:

Choose a random matching π.

Sample $p_{i,\pi(i)}$ $\forall i$. If any 0, restart.

Pick random spanning tree on K_n.

Orient edges of T towards 1.

For each $(i,j) \in T$, sample $p_{i,\pi(j)}$

If any 0, restart.

Output matching π.

Application to Mechanism Design

Mechanism Design Setup: n agents + space of outcomes X.

- allocations of items to agents
- flows/paths in a graph
- scheduling

Preference/type is a mapping: $v_i : X \to \mathbb{R}$. (private to agents).

Designer wants to optimize welfare $\sum_{i=1}^{n} v_i(x)$.

Two problems: (1) How to compute optimal allocation?

(2) How to incentivize agents to report truthfully?

Is there any reduction from (1) to (2)? Assume $A(v_1 \cdots v_n)$.

- Yes! If A is the optimal algorithm (Vickrey-Clarke-Groves 60s-70s)
- No, otherwise, without any prior about the valuations.

Application to Mechanism Design

However if $v_i \sim F_i$ i.i.d. it is possible to come up with "BIC"-reductions:

- Yes! If types are single-parameter (Hartline - Lucier 2010)
- Yes*, in general (Hartline - Kleinberg - Malekian 2011)

  Algorithm A $\Rightarrow$ Mechanism $\mathcal{M}$ that is ε-BIC and

  $$\mathbb{E}[\text{welfare}(M)] \geq \mathbb{E}[\text{welfare}(A)] - \varepsilon.$$

  Technique: Replica - Surrogate Matching

- Yes, in general (Dughmi - Hartline - Kleinberg - Niazadeh 2017)

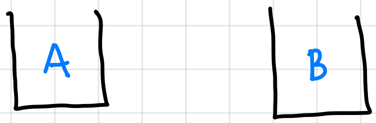
  Extra Ingredient is a Bernoulli Factory

Simple Mechanism Design Problem

- One agent with valuation $v: X \to \mathbb{R}$
- k urns, each with a distribution $F_1, \dots F_k$ on X.
- Designer has only sample access wants to maximize $\mathbb{E}_{x \sim F_i}[v(x)]$.

- Example: $X = \{1, 2, 3\}$ $\quad v(1) = 1 \quad v(2) = 2 \quad v(3) = 3.$

  Two urns $(k=2)$.



$\mathbb{P}(x=2) = 1$

$\mathbb{P}(x=1) = \frac{1}{2} + \varepsilon$

$\mathbb{P}(x=3) = \frac{1}{2} - \varepsilon$

- Alternative: Choose an urn with probability proportional to $\mathbb{E}_{x \sim F_j} \exp(\lambda (v(x) - 1))$

  + implicit payment computation (Babaioff, Kleinberg, Slivkins 2013)

Some Open Problems

(1) What are the conditions for $f:(0,1)^n \to (0,1)$ to admit a factory?

- how does the notion of poly-bounded generalize?

$f(p) \geq \min(p_1, 1-p_1, p_2, 1-p_2, \dots, p_n, 1-p_n)^n$ for some n ?

- Nacu-Peres proof probably works for $f:(0,1)^n \to (\varepsilon, 1-\varepsilon)$.

(2) Remove "bounded-variation" from Keane-O'Brien proof.

Some Open Problems

(3) Dealing with $\{0,1\}$-values. E.g extend results to $f:[0,1]\to[0,1]$.

E.g. Can we extend Sampford sampling to the boundary?

Not with an exponential tail.

Some Open Problems

(4) Sample complexity: how many samples to obtain a matching or a k-subset? How optimized are the current factories with respect to $\mathbb{E}[N]$?

For combinatorial factories $\mathbb{E}[N] \sim \dfrac{n|V|}{\sum_v P_v(p)}$

(5) Explicit factories for other polytopes: e.g. flows/circulations. Non-limit construction for generic polytopes.

(6) Other applications to Game Theory / Mechanism Design (e.g. Cai et al on revenue-preserving reductions)

Slides, Lecture Notes, References:

www.renatoppl.com/bernoulli

Thanks!