Uncertainty in Combinatorial Optimization

Jian Li

Institute of Interdisiplinary Information Sciences

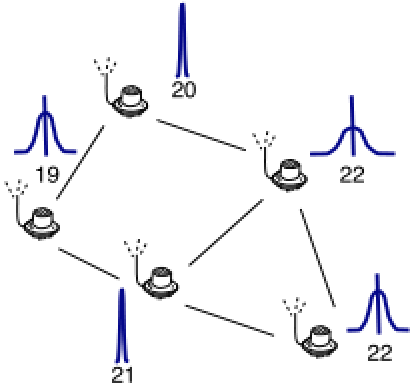
Tsinghua University

Aug. 2012

lijian83@mail.tsinghua.edu.cn

Uncertain Data

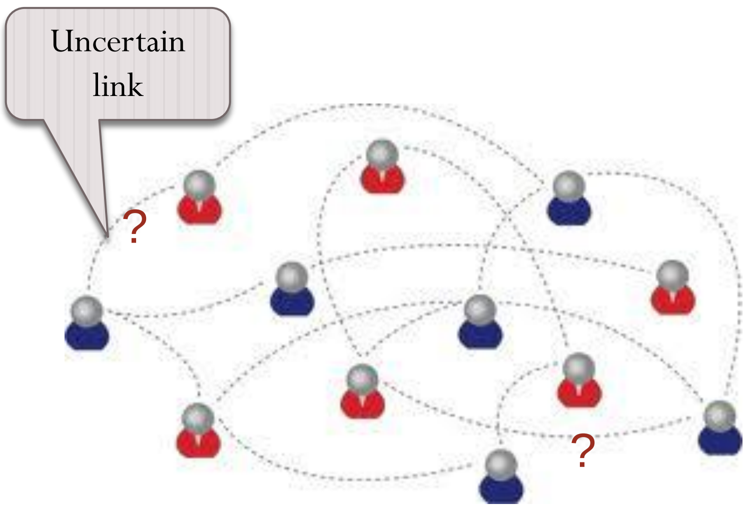
- Uncertain data is ubiquitous
  - Data Integration and Information Extraction
  - Sensor Networks; Information Networks



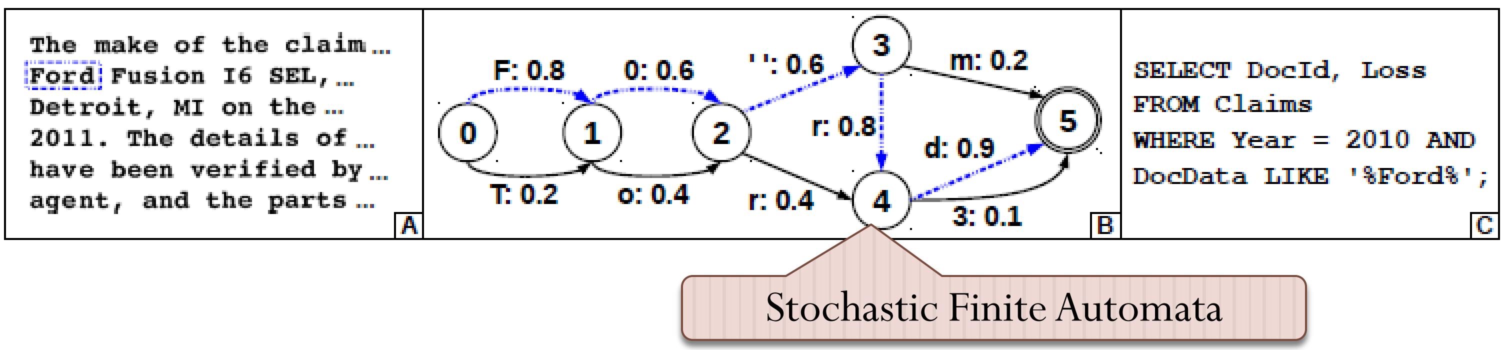
Sensor network

Sensor ID	Temp.
1	Gauss(40,4)
2	Gauss(50,2)
3	Gauss(20,9)
…	…

Uncertain Data



Social network



OCR (Optical Character Recognition) data.

Uncertain Data

- Future data is destined to be uncertain

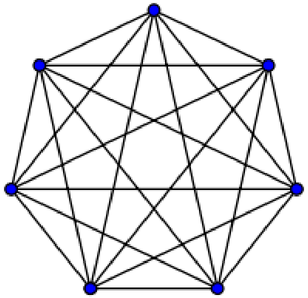
Dealing with Uncertainty

- Handling uncertainty is a very broad topic that spans multiple disciplines
  - Economics / Game Theory
  - Finance
  - Operation Research
  - Management Science
  - Probability Theory / Statistics
  - Psychology
  - Computer Science

Today: Problems in **Combinatorial Optimization**

Ignoring uncertainty is not the right thing to do

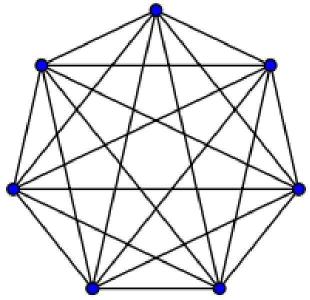
- A undirected graph with n nodes
- The length of each edge: i.i.d. Uniform[0,1]



- Question: What is E[MST]?

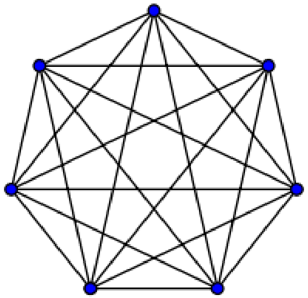
Ignoring uncertainty is not the right thing to do

- A undirected graph with n nodes
- The length of each edge: i.i.d. Uniform[0,1]



- Question: What is E[MST]?
- Ignoring uncertainty ("replace by the expected values" heuristic)
  - each edge has a fixed length 0.5
  - This gives a WRONG answer 0.5(n-1)

Ignoring uncertainty is not the right thing to do



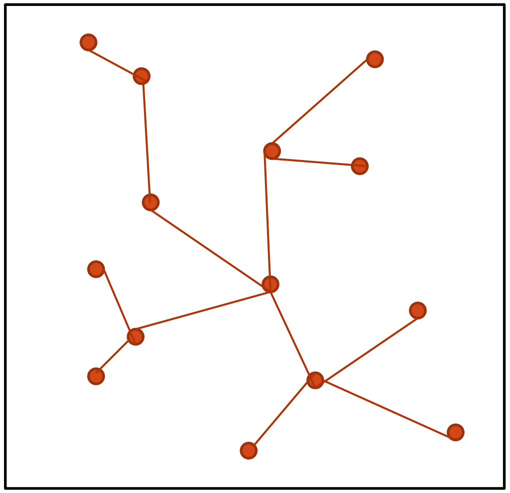
- A undirected graph with n nodes
- The length of each edge: i.i.d. Uniform[0,1]
- Question: What is E[MST]?
- Ignoring uncertainty ("replace by the expected values" heuristic)
  - each edge has a fixed length 0.5
  - This gives a WRONG answer 0.5(n-1)
- But the true answer is (as n goes to inf)

$$\zeta(3) = \sum_{i=1}^{\infty} 1/i^3 < 2$$

[McDiarmid, Dyer, Frieze, Karp, Steele, Bertsekas, Geomans]

A Similar Problem

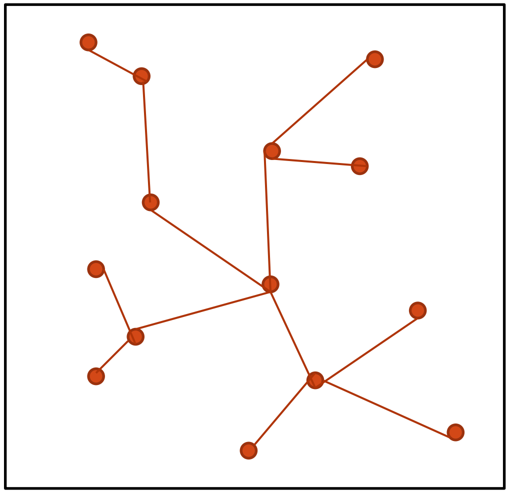
- N points: i.i.d. uniform$[0,1] \times [0,1]$



- Question: What is E[MST] ?

A Similar Problem

- N points: i.i.d. uniform$[0,1] \times [0,1]$

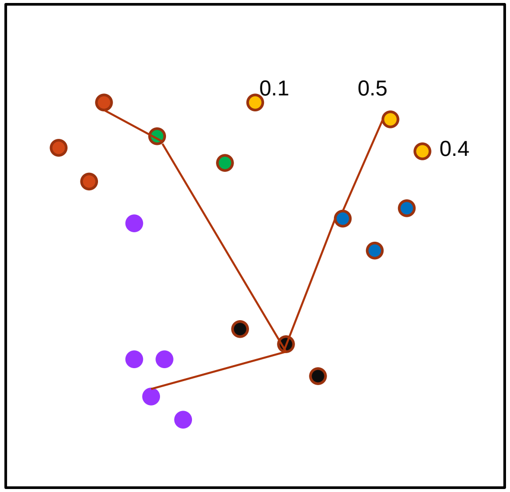


- Question: What is E[MST] ?
- Answer: $\theta(\sqrt{n})$ [Frieze, Karp, Steele, …]

The problem is similar, but the answer is not similar…………

A Generalization

- The position of each point is random (non-i.i.d)
- A model in wireless networks



- Question: What is E[MST] ?
- Of Course, there is no close-form formula
- Need efficient algorithms to compute E[MST]

[Huang, L. ArXiv 2012]

MST over Stochastic Points

- The problem is #P-hard [Kamousi, Chan, Suri. SoCG'11]
- So, let us focus on approximating the value
- Attempt one: list all realizations? (Exponentially many)
- Attempt two: Monte Carlo (variance can be very large)

PDF of a random var

MST over Stochastic Points

- Our approach: (sketch)
- Law of total expectation:

$$\mathbf{E}[X] = \sum_{y} \Pr[Y = y] \, \mathbf{E}[X \mid Y = y]$$

A carefully chosen random event Y

Hopefully, we have

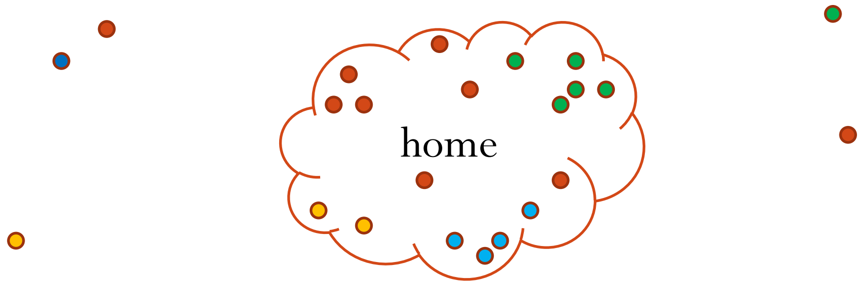
Easy to compute

Low variance

How to choose Y?

MST over Stochastic Points

- The "home set" technique:

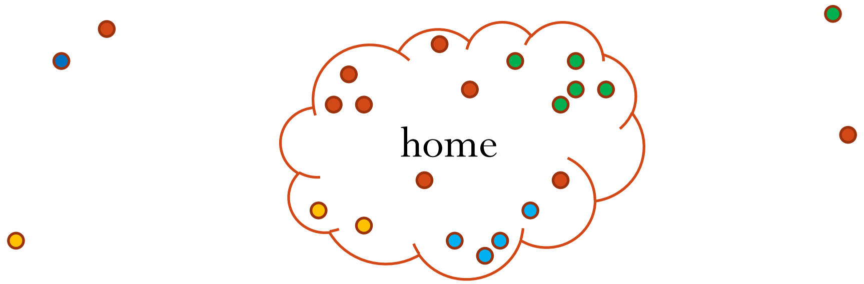


(1)$\Pr[\mathit{all\ nodes\ are\ at\ home}] \approx 1$

(2) $\mathbf{E}[\mathrm{MST} \mid \mathit{all\ node\ are\ at\ home}]$ can be estimated (due to low variance)

MST over Stochastic Points

- The "home set" technique:



(1)$\Pr[\textit{all nodes are at home}] \approx 1$

(2) $\mathbf{E}[\text{MST} \mid \textit{all node are at home}]$ can be estimated (due to low variance)

$$\mathbf{E}[MST] = \sum_{y} \Pr[y\ \textit{nodes are at home}]\, \mathbf{E}[X \mid y\ \textit{nodes are at home}]$$

$$\approx \Pr[\textit{all nodes are at home}]\, \mathbf{E}[\,X \mid \textit{all nodes are at home}\,] + \Pr[n-1\ \textit{nodes are at home}]\, \mathbf{E}[X \mid n-1\ \textit{nodes are at home}]$$

Let us start to optimize:

Online stochastic optimization

Stochastic Matching

Stochastic Matching

Given:

- Existential prob. p_e for each edge e.
- Patience level t_v for each vertex *v.*

- **Probing** *e=(u,v)*: The only way to know the existence of *e*.
  - We can probe *(u,v)* only if $t_u>0, t_v>0$.
  - If *e* indeed exists, we should add it to our matching.
  - If not, $t_u = t_u - 1$, $t_v = t_v - 1$.
- Objective: Find a probing strategy to maximize the expected weight of the matching

[Bansal, Gupta, L, Mestre, Nagarajan, Rudra. ESA'10]

Stochastic Matching

Stochastic Matching

Given:

- Existential prob. p_e for each edge e.
- Patience level t_v for each vertex v.

- **Probing** $e=(u,v)$: The only way to know the existence of e.
  - We can probe (u,v) only if $t_u>0, t_v>0$.
  - If e indeed exists, we should add it to our matching.
  - If not, $t_u = t_u - 1$, $t_v = t_v - 1$.
- Objective: Find a probing strategy to maximize the expected weight of the matching
- Our Results: we give constant approx. algo. for the weighted version, resolving an open question posed in previous work

[Bansal, Gupta, L, Mestre, Nagarajan, Rudra. ESA'10]

Stochastic Matching

Motivation: **Online dating**

- Existential prob. p_e : estimation of the success prob. based on users' profiles.

Stochastic Matching

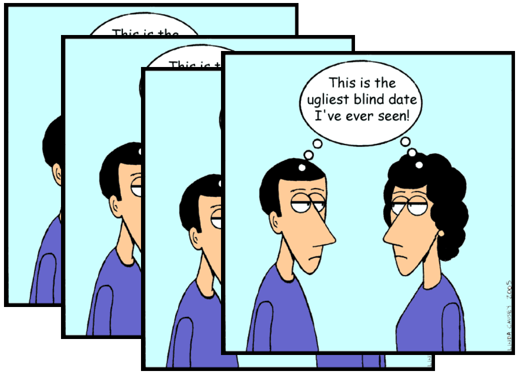
Motivation: **Online dating**

- Existential prob. p_e : estimation of the success prob. based on users' profiles.
- Probing edge $e=(u,v)$: u and v are sent to a date.

Stochastic Matching

Motivation: **Online dating**

- Existential prob. p_e : estimation of the success prob. based on users' profiles.
- Probing edge $e=(u,v)$: u and v are sent to a date.
- Patience level: obvious.



- Other motivations: Kidney exchange, online ad assignment

A LP Upper Bound

- Variable y_e : Prob. that any algorithm probes *e*.

$$\text{maximize} \quad \sum_{e\in E} w_e \cdot x_e$$

$$\text{subject to} \quad \sum_{e\in\partial(v)} x_e \leq 1 \quad \forall v \in V$$

At most 1 edge in $\partial(v)$ is matched

$$\sum_{e\in\partial(v)} y_e \leq t_v \quad \forall v \in V$$

At most t_v edges in $\partial(v)$ are probed

$$x_e = p_e \cdot y_e \quad \forall e \in E$$

x_e: Prob. e is matched

$$0 \leq y_e \leq 1 \quad \forall e \in E$$

The LP value is an upper bound of the optimal expected value

A Simple 8-Approximation

An edge (u,v) is *safe* if $t_u>0, t_v>0$ and neither u nor v is matched

Algorithm:

- Pick a permutation π on edges uniformly at random
- For each edge e in the ordering π, do:
  - If e is not safe then do not probe it.
  - If e is safe then probe it w.p. y_e/α.

A Simple 8-Approximation

An edge (u,v) is *safe* if $t_u>0, t_v>0$ and neither u nor v is matched

Algorithm:

- Pick a permutation π on edges uniformly at random
- For each edge *e* in the ordering π, do:
  - If *e* is not safe then do not probe it.
  - If *e* is safe then probe it w.p. y_e/α.

- If *e* is always safe, we can recover the LP value $\sum_e w_e y_e p_e$
- We can show this algorithm can recover 1/8 of the LP value by proving *Pr[e is safe]>=1/8*

A Simple 8-Approximation

Analysis:

Lemma: For any edge *(u,v)*, at the point when *(u,v)* is considered under π, *Pr(u loses its patience)* $\leq 1/2\alpha$.

Proof: Let *U* be #probes incident to *u* and before *e*.

A Simple 8-Approximation

Analysis:

Lemma: For any edge *(u,v)*, at the point when *(u,v)* is considered under π, *Pr(u loses its patience) ≤1/2α* .

Proof: Let *U* be #probes incident to *u* and before *e*.

$$\mathbb{E}[U] = \sum_{e\in\partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi \text{ AND } e \text{ is probed}]$$

$$= \sum_{e\in\partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi \text{ AND } e \text{ is safe}] \cdot \frac{y_e}{\alpha}$$

$$\leq \sum_{e\in\partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi] \cdot \frac{y_e}{\alpha}$$

$$= \sum_{e\in\partial(u)} \frac{1}{2} \cdot \frac{y_e}{\alpha} \quad \leq \quad \frac{t_u}{2\alpha}.$$

$$\sum_{e\in\partial(v)} y_e \leq t_v$$

A Simple 8-Approximation

Analysis:

Lemma: For any edge *(u,v)*, at the point when *(u,v)* is considered under π, *Pr(u loses its patience) ≤1/2α* .

Proof: Let *U* be #probes incident to *u* and before *e*.

$$\begin{aligned}\mathbb{E}[U] &= \sum_{e\in\partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi \text{ AND } e \text{ is probed}]\\ &= \sum_{e\in\partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi \text{ AND } e \text{ is safe}] \cdot \tfrac{y_e}{\alpha}\\ &\le \sum_{e\in\partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi] \cdot \tfrac{y_e}{\alpha}\\ &= \sum_{e\in\partial(u)} \tfrac{1}{2} \cdot \tfrac{y_e}{\alpha} \quad \le \quad \tfrac{t_u}{2\alpha}.\end{aligned}$$

$$\sum_{e\in\partial(v)} y_e \le t_v$$

By the Markov inequality: $\Pr[U \ge t_u] \le \dfrac{\mathbb{E}[U]}{t_u} \le \dfrac{1}{2\alpha}.$

A Simple 8-Approximation

Analysis:

Lemma: For any edge e=*(u,v)*, at the point when *(u,v)* is considered under π, *Pr(u is matched) ≤1/2α* .

Proof: Let *U* be #matched edges incident to u and before *e*.

$$\mathbb{E}[U] = \sum_{e \in \partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi \text{ AND } e \text{ is matched}]$$

$$= \sum_{e \in \partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi \text{ AND } e \text{ is safe}] \cdot \frac{y_e}{\alpha} \cdot p_e$$

$$\leq \sum_{e \in \partial(u)} \Pr[\text{edge } e \text{ appears before } (u,v) \text{ in } \pi] \cdot \frac{y_e}{\alpha} \cdot p_e$$

$$= \sum_{e \in \partial(u)} \frac{1}{2} \cdot \frac{y_e}{\alpha} \cdot p_e \quad \leq \quad \frac{1}{2\alpha}.$$

$$\sum_{e \in \partial(v)} x_e \leq 1$$

By the Markov inequality: $\Pr[U \geq 1] \leq \mathbb{E}[U] \leq \frac{1}{2\alpha}$

A Simple 8-Approximation

Analysis:

Theorem: The algorithm is a 8-approximation.

Proof: When e is considered,

Pr(e is not safe) ≤ Pr(u is matched)+ Pr(u loses its patience)+
Pr(v is matched)+ Pr(v loses its patience)

≤ 2/α

A Simple 8-Approximation

Analysis:

Theorem: The algorithm is a 8-approximation.

Proof: When e is considered,

Pr(e is not safe) ≤ Pr(u is matched)+ Pr(u loses its patience)+ Pr(v is matched)+ Pr(v loses its patience)

≤ 2/α

Therefore, $$\mathbb{E}[\text{Our Solution}] = \sum_e w_e \Pr(e \text{ is safe}) \frac{y_e}{\alpha} p_e$$

$$\geq (1 - \frac{2}{\alpha}) \frac{1}{\alpha} \sum_e w_e y_e p_e$$

$$\geq \frac{1}{8} OPT \qquad (\alpha = 4)$$

Recall $\Sigma_e\, w_e\, y_e\, p_e$ is an upper bound of *OPT*

A Simple 8-Approximation

Analysis:

Theorem: The algorithm is a 8-approximation.

Proof: When e is considered,

Pr(e is not safe) ≤ Pr(u is matched)+ Pr(u loses its patience)+ Pr(v is matched)+ Pr(v loses its patience)

≤ 2/α

Therefore, $$\mathbb{E}[\text{Our Solution}] = \sum_e w_e \Pr(e \text{ is safe}) \frac{y_e}{\alpha} p_e$$

$$\geq (1 - \frac{2}{\alpha}) \frac{1}{\alpha} \sum_e w_e y_e p_e$$

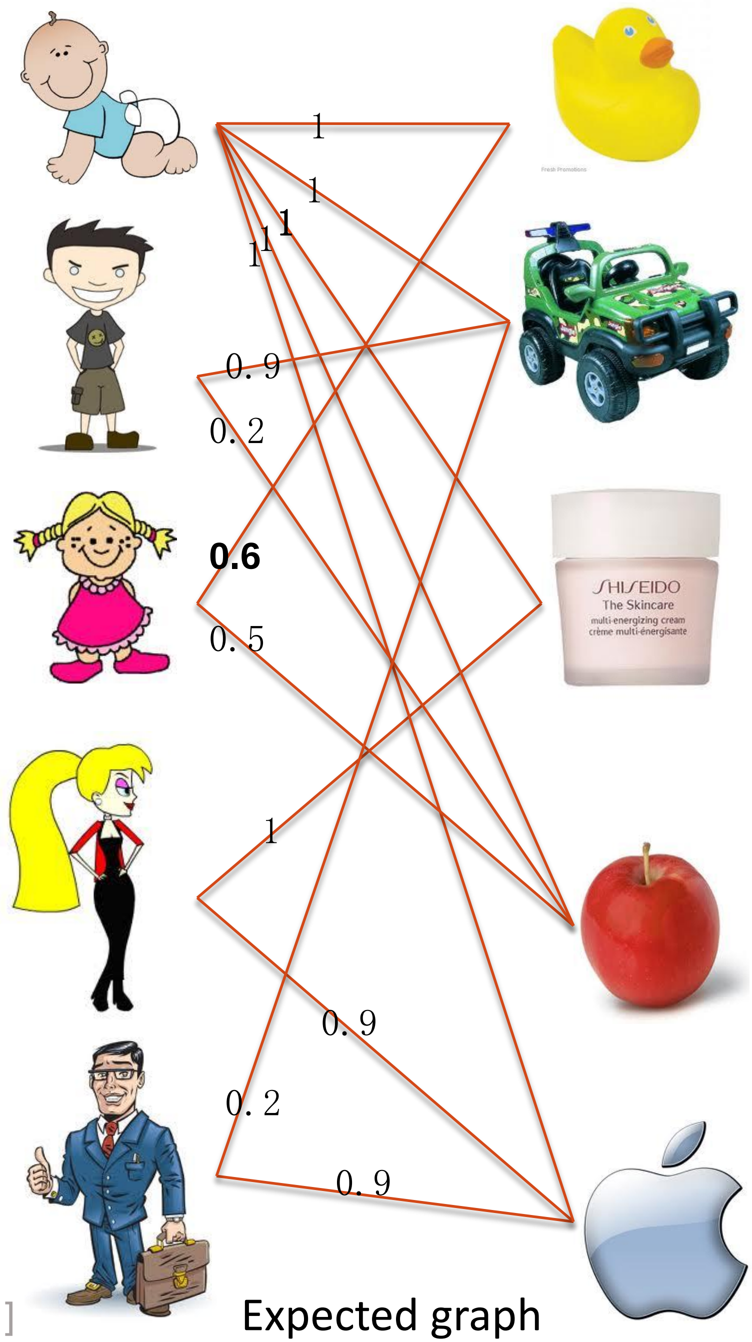
$$\geq \frac{1}{[illegible]} OPT \qquad (\alpha = 4)$$

Can be improved to a 3-approximation with a more careful algorithm

Recall $\Sigma_e \, w_e \, y_e \, p_e$ is an upper bound of *OPT*

Stochastic online matching

- A set of items and a set of buyer types. A buyer of type *b* likes item *a* with probability p_{ab}.
  - G(buyer types, items): Expected graph)
- The buyers arrive online.
  - Her type is an i.i.d. r.v. .
- The algorithm shows the buyer (of type *b*) at most *t* items one by one.
- The buyer buys the first item she likes or leaves without buying.
- Goal: Maximizing the expected number of satisfied users.



[Bansal, Gupta, L, Mestre, Nagarajan, Rudra. Algorithmica '11]

Bayesian Online Selection Problem

- A knapsack of capacity C
- A set of items.
- Known: Prior distr of (size, profit) of each item.
- Items arrive one by one
- Can see the actually size and profit of an item. But have to decide whether to accept the item immediately
- Knapsack constraint: The total size of accepted items <= C

- Goal: maximize E[Profit]

✓ Generalization of the Prophet inequalities in optimal control
✓ Application in multi-parameter mechanism design

[L, Yuan. ArXiv 2012]

Bayesian Online Selection Problem

- We can get a constant approx using the same LP technique (simple exercise)

We can get a $1+\epsilon$ –approximate optimal policy

We developed a new technique, called Poisson approximation technique

The technique can be used in many other problems:

Stochastic knapsack problem

Stochastic Bin Packing Problem

Stochastic Shortest Path …….

[L, Yuan. ArXiv 2012]

A More Fundamental Issue

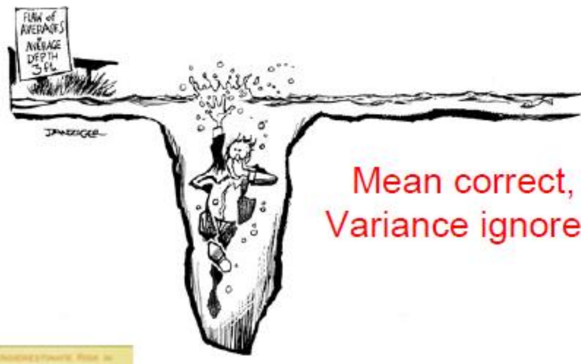
Inadequacy of Expected Value

- Stochastic Optimization
  - Most common objective: Optimizing the **expected value**
- Inadequacy of expected value:
  - Unable to capture risk-averse or risk-prone behaviors
    - Action 1: $100 VS Action 2: $200 w.p. 0.5; $0 w.p. 0.5
    - Risk-averse players prefer Action 1
    - Risk-prone players prefer Action 2 (e.g., a gambler spends $100 to play Double-or-Nothing)

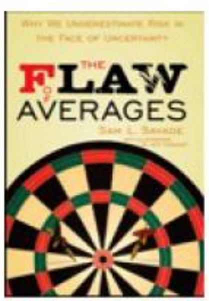
Inadequacy of Expected Value

- Be aware of risk!

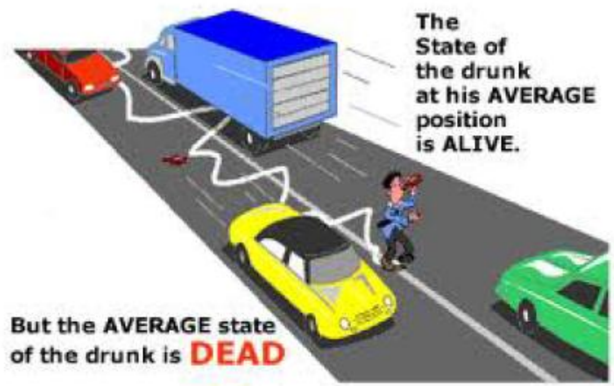
Flaw of averages (weak form):



Mean correct,
Variance ignored



Flaw of averages (strong form):



Wrong value of mean:
$f(E[X]) \neq E[f(X)]$

Inadequacy of Expected Value

- St. Petersburg paradox
  - You pay x dollars to enter the game
    - Repeatedly toss a fair coin until a tail appears
    - payoff=2^k where k=#heads

Inadequacy of Expected Value

- St. Petersburg paradox
  - You pay x dollars to enter the game
    - Repeatedly toss a fair coin until a tail appears
    - payoff=2^k where k=#heads
  - How much should x be?
    - Expected payoff =1x(1/2)+2x(1/4)+4x(1/8)+......= infinity
    - Few people would pay even $25 [Martin '04]

Expected Utility Maximization Principle

Remedy: Use a utility function

$\mu : R \rightarrow R$: The utility function: value (profit/cost)-> utility

Expected Utility Maximization Principle: the decision maker should choose the action that maximizes the **expected utility**

$$\text{maximize. } \mathbb{E}[\mu(\text{profit})]$$

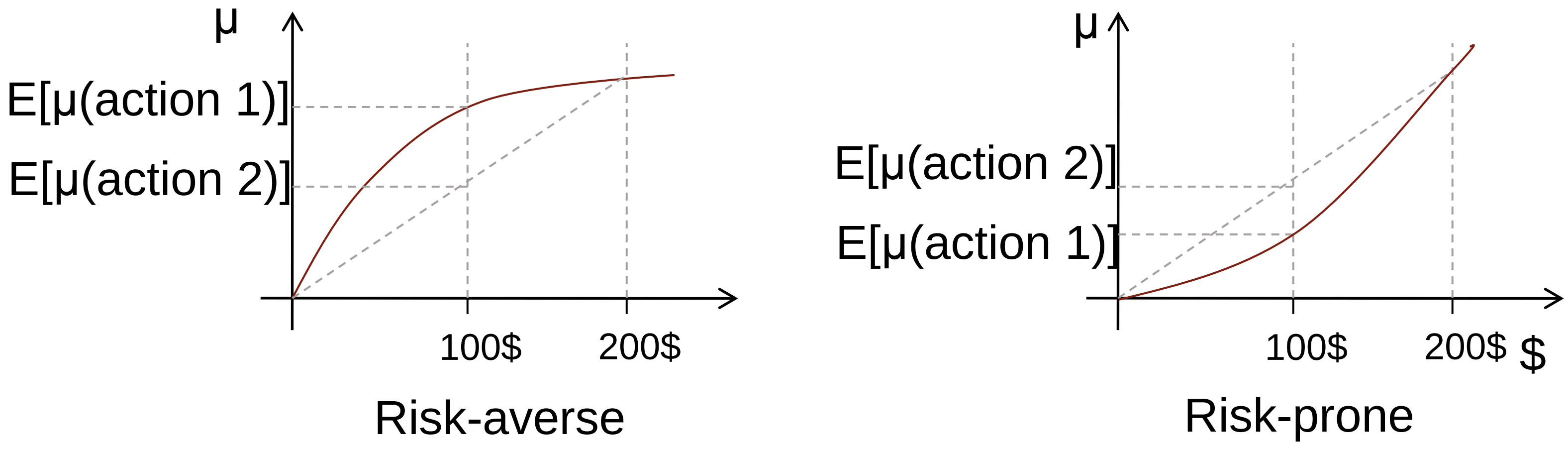
- Proved quite useful to explain some popular choices that seem to contradict the expected value criterion
- An *axiomatization* of the principle (known as von Neumann-Morgenstern expected utility theorem).

Expected Utility Maximization Principle

$u \colon R \to R$:   The utility function: profit-> utility

Expected Utility Maximization Principle: the decision maker should choose the action that maximizes the **expected utility**

- Action 1: \$100
- Action 2: \$200 w.p. 0.5; \$0 w.p. 0.5



Problem Definition

- Deterministic version:
  - A set of element $\{e_i\}$, each associated with a weight w_i
  - A solution S is a subset of elements (that satisfies some property)
  - **Goal:** Find a solution S such that the total weight of the solution $w(S)=\Sigma_{i\in S} w_i$ is minimized
  - E.g. shortest path, minimal spanning tree, top-k query, matroid base
- Stochastic version:
  - w_is are independent positive random variable
  - $\mu()$: $R^+ \rightarrow R^+$ is the utility function (assume $\lim_{x\rightarrow\infty}\mu(x)=0$)
  - **Goal:** Find a solution S such that the expected utility $E[\mu(w(S))]$ is maximized

[L. , Deshpande. FOCS'11]

Our Results

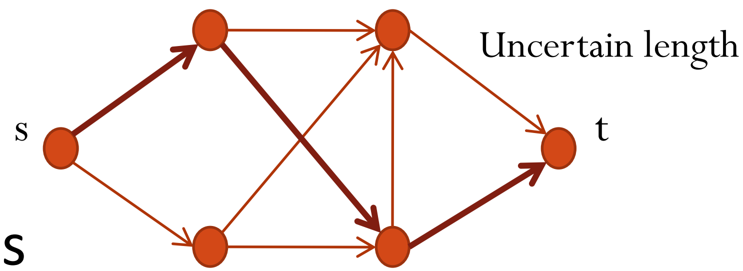
- **THM:** If the following two conditions hold
  - (1) there is a pseudo-polynomial time algorithm for the **exact version**of deterministic problem, and
  - (2) μ is bounded by a constant and satisfies *Hőlder condition* $|\mu(x)-\mu(y)|\leq C|x-y|^{\alpha}$ for constant C and $\alpha\geq 0.5$,

  then we can obtain in polynomial time a solution S such that $E[\mu(w(S))]\geq OPT-\varepsilon$, for any fixed $\varepsilon>0$

- Exact version: find a solution of weight exactly K
- Pseudo-polynomial time: polynomial in K
- Problems satisfy condition (1): shortest path, minimum spanning tree, matching, knapsack.

Our Results

- **Stochastic shortest path** : find an s-t path P such that *Pr*[*w*(*P*)<*1*] is maximized



- Previous results
  - Many heuristics
  - Poly-time approximation scheme (PTAS) if (1) all edge weights are normally distributed r.v.s (2) *OPT>0.5*[Nikolova, Kelner, Brand, Mitzenmacher. ESA'06] [Nikolova. APPROX'10]
  - Bicriterion PTAS for exponential distributions [Nikolova, Kelner, Brand, Mitzenmacher. ESA'06]
- Our result
  - Bicriterion PTAS ($Pr[w(P)<1+\delta]>(1\text{-eps})OPT$) if *OPT= Const*

Our Results

- **Stochastic knapsack**: find a collection S of items such that $Pr[w(S)<1]>\gamma$ and the total profit is maximized

Each item has a deterministic profit and a (uncertain) size

Knapsack, capacity=1

- Previous results
  - $log(1/(1-\gamma))$-approximation [Kleinberg, Rabani, Tardos. STOC'97]
  - Bicriterion PTAS for exponential distributions [Goel, Indyk. FOCS'99]
  - PTAS for Bernouli distributions if $\gamma = Const$ [Goel, Indyk. FOCS'99] [Chekuri, Khanna. SODA'00]
  - Bicriterion PTAS if $\gamma = Const$ [Bhalgat, Goel, Khanna. SODA'11]
- Our result
  - Bicriterion PTAS if $\gamma = Const$ (with a better running time than Bhalgat et al.)
  - Stochastic partial-ordered knapsack problem with tree constraints

- Research interests:

  Algorithms: Approx Algo for NP-hard problems
  Graph problems
  Scheduling Problems
  Data structures
  Stochastic Optimization

  also interested in Databases, Game theory, Networking, Machine Learning….

- lijian83@mail.tsinghua.edu.cn

Thanks

lijian83@mail.tsinghua.edu.cn