Μεταπτυχιακό Πρόγραμμα «Αλγόριθμοι, Λογική και Διακριτά Μαθηματικά» – 2016

$\propto \wedge \mu \forall$

FPRAS for the Permanent

Iro Oikonomou

July 5, 2022

Counting Complexity

Outline

Introduction

Permanent

Definition 1

The permanent of an $n \times n$ nonnegative matrix $A = (a(i,j))$ is defined as

$$\text{per}(A) = \sum_{\sigma} \prod_{i} a(i, \sigma(i)), \tag{1.1}$$

where the sum is over all permutations σ of $\{1, 2, \ldots, n\}$.

Permanent

Definition 1

The **permanent** of an $n \times n$ nonnegative matrix $A = (a(i,j))$ is defined as

$$\text{per}(A) = \sum_{\sigma} \prod_{i} a(i, \sigma(i)), \tag{1.2}$$

where the sum is over all permutations σ of $\{1, 2, \ldots, n\}$.

Remark 1

Let $G_A = (V_1, V_2, E)$ be a bipartite graph, and A the corresponding 0,1-adjacency matrix. The permanent of A equals the number of perfect matchings in G_A.

FPRAS

Definition 2

A **fully polynomial randomized approximation scheme (fpras)** for a counting problem $f : \Sigma^* \to \mathbb{N}$ is a randomized algorithm that takes as input an instance $x \in \Sigma^*$, an error tolerance $0 < \epsilon < 1$, and $0 < \delta < 1$, and outputs a number $\widehat{f(x)} \in \mathbb{N}$ such that

$$\Pr[(1-\epsilon)f(x) \leq \widehat{f(x)} \leq (1+\epsilon)f(x)] \geq 1-\delta. \tag{1.3}$$

The algorithm must run in time polynomial in $|x|,\ 1/\epsilon$ and $\log(1/\delta)$

FPRAS

Remark 2

A **fully polynomial randomized approximation scheme (fpras)** for a **the permanent** is a randomized algorithm that takes as input an $n \times n$ nonnegative matrix A, an error tolerance $0 < \epsilon < 1$, and $0 < \delta < 1$, and outputs a number Z such that

$$\Pr[(1-\epsilon)Z \leq \text{per}(A) \leq (1+\epsilon)Z] \geq 1-\delta. \quad (1.4)$$

The algorithm must run in time polynomial in $|x|,\ 1/\epsilon$ and $\log(1/\delta)$

Main Theorem

Theorem 1

There exists a fully polynomial randomized approximation scheme for the permanent of an arbitrary $n \times n$ matrix A with nonnegative entries.

Approach

Construct a **fully-polynomial almost uniform sampler (fpaus) for perfect matchings**, namely a randomized algorithm which, given as inputs an $n \times n$ $0,1$-matrix A and a bias parameter $\delta \in (0,1]$, outputs a random perfect matching in G_A from a distribution $\mathcal{U}'$ that satisfies

$$d_{\text{tv}}(\mathcal{U}',\mathcal{U}) \leq \delta,$$

where $\mathcal{U}$ is the uniform distribution on perfect matchings of G_A and d_{tv} denotes the total variation distance.

$\rightarrow$ The sampler will be based on simulation of a suitable Markov chain.

Markov Chain

Markov Chain Revisited

We consider a Markov Chain (MC)

- Ergodic MC(Ω, P) $\Rightarrow$ unique stationary distribution π
- Satisfies the detailed balance conditions for all $M,\ M' \in \Omega$, that is,

$$\pi(M)P(M,M') = \pi(M')P(M',M) =: Q(M,M'),$$

then the chain is said to be **time-reversible** and π **is a stationary distribution**.

Markov Chain Revisited

We consider a Markov Chain (MC)

- Ergodic MC(Ω, P) $\Rightarrow$ unique stationary distribution π
- Satisfies the detailed balance conditions for all $M,\ M' \in \Omega$, that is,

$$\pi(M)P(M,M') = \pi(M')P(M',M) =: Q(M,M'),$$

  then the chain is said to be **time-reversible** and π **is a stationary distribution**.
- The mixing time (from state M)

$$\tau(\delta) = \tau_M(\delta) = \min\{t : d_{\text{tv}}(P^t(M,\cdot),\pi) \leq \delta\}$$

  When the MC is used as a random sampler, the mixing time determines the number of simulation steps needed before a sample is produced.

Markov Chain

The running time of the random sampler is determined by the mixing time of the Markov chain.

Definition

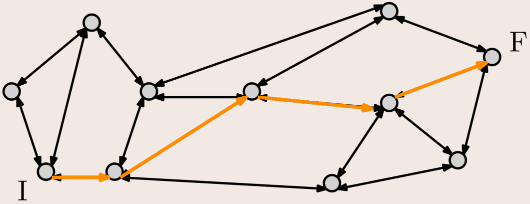
A Markov chain is called **rapidly mixing** (from initial state x) if, for any fixed $\delta > 0$, $\tau(\delta)$ is bounded above by a polynomial function of n.

MC Congestion

- The graph $G_P = (\Omega, E_P)$, where $E_P = \{(M, M') : P(M, M') > 0\}$.
- For all ordered pairs $(I, F) \in \Omega \times \Omega$ of "initial" and "final" states, let $\mathcal{P}_{I,F}$ denote a collection of simple directed paths in G_P from I to F
- $f_{I,F} : \mathcal{P}_{I,F} \to \mathbb{R}^+$ is a **flow** from I to F if the following holds:

$$\sum_{p \in \mathcal{P}_{I,F}} f_{I,F}(p) = \pi(I)\pi(F)$$

- The flow of the Markov chain is $f = \{f_{I,F} : I, F \in \Omega\}$



MC congestion

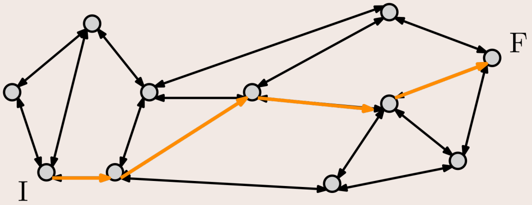
- The **congestion**, ϱ, of a flow f is defined as

$$\varrho = \varrho(f) = \max_{t=(M,M')\in E_P} \varrho_t, \tag{2.5}$$

where

$$\varrho_t = \frac{1}{Q(t)} \sum_{I,F\in\Omega} \sum_{p:t\in p\in\mathcal{P}_{I,F}} f_{I,F}(p)|p|, \tag{2.6}$$

and $Q(t) = Q(M,M') = \pi(M)P(M,M')$.



MC congestion

Theorem

For an ergodic, time reversible Markov chain with self-loop probabilities $P(M,M) \geq 1/2$ for all states M, and any initial state $M_0 \in \Omega$,

$$\tau_{M_0}(\delta) \leq \varrho(\ln \pi(M_0)^{-1} + \ln \delta^{-1})$$

To prove rapid mixing, it suffices to demonstrate a flow with an upper bound of the form **poly(n)** on its congestion for our Markov chain on matchings.

[Sin92] Alistair Sinclair. “Improved Bounds for Mixing Rates of Markov Chains and Multicommodity Flow”. In: *Combinatorics, Probability and Computing* 1.4 (1992), pp. 351–370. doi: 10.1017/S0963548300000390

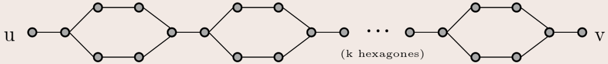
Sampling Algorithm

We consider a Markov Chain (MC)

- $G = (V1, V2, E)$ be a bipartite graph on $n + n$ vertices.
- $\mathcal{M}$: the set of perfect matchings in G
- $\mathcal{M}(u, v)$: the set of near-perfect matchings with holes only at the $u \in V1$ and $v \in V2$.
- $\Omega = \mathcal{M} \cup \bigcup_{u,v} \mathcal{M}(u, v)$.

Problem

$|\mathcal{M}|/|\Omega|$ must be bounded below by a inverse polynomial in n.



The matchings have insufficient weight in the stationary distribution!

Weights

Let the quantities:

- $\lambda(u,v)$ is a positive weight , which we call its activity.
- $\lambda(M) = \prod_{(u,v)\in M} \lambda(u,v)$is the activity of the matching M
- $\lambda(\mathcal{S}) = \sum_{M\in\mathcal{S}} \lambda(M)$ is the activity of the set $\mathcal{S}$

We consider complete graph on $n+n$ vertices, where

- $\lambda(e) = 1$ for $e \in E$
- $\lambda(e) = \xi \approx 0$ for $e \notin E$

Stationary distribution

This will be the distribution π over Ω defined by $\pi(M) \propto \Lambda(M)$, where

$$\Lambda(M) = \begin{cases} \lambda(M)w(u,v) & \text{, if } M \in \mathcal{M}(u,v) \text{ for some } u,v \\ \lambda(M) & \text{, if } M \in \mathcal{M} \end{cases} \tag{2.7}$$

where $w : V_1 \times V_2 \to \mathbb{R}^+$ is a weight function for holes.

Transitions

(1) If $M \in \mathcal{M}$, choose an edge $e = (u, v)$ uniformly at random from M; set $M' = M \backslash e$.

(2) If $M \in \mathcal{M}(u, v)$, choose z uniformly at random from $V_1 \cup V_2$.

(i) if $z \in \{u, v\}$ and $(u, v) \in E$, let $M' = M \cup (u, v)$
(ii) if $z \in V_2$, $(u, z) \in E$ and $(x, z) \in M$, let $M' = M \cup (u, z) \backslash (x, z)$;
(iii) if $z \in V_1$, $(z, v) \in E$ and $(z, y) \in M$, let $M' = M \cup (z, v) \backslash (z, y)$
(iv) otherwise, let $M' = M$

(3) With probability $\min\{1, \Lambda(M')/\Lambda(M)\}$ go to M'; otherwise, stay at M. (Metropolis rule)

Transitions

Markov chain properties?

- In steps (1) and (2) for selecting the candidate matching M' are symmetric, being $1/n$ in the case of moves between perfect and near-perfect matchings, and $1/2n$ between near-perfect matchings.
- Combined with the Metropolis rule for accepting the move to M' applied in step (3)
  - $\Rightarrow$ Markov chain is reversible with $\pi(M) \propto \Lambda(M)$ as its stationary distribution
- We add a self-loop probability of $1/2$ to every state

Weight function

Ideally $w = w^*$ equals:

$$w^*(u,v) = \frac{\lambda(\mathcal{M})}{\lambda(\mathcal{M}(u,v))} \tag{2.8}$$

for each pair of holes u, v with $\mathcal{M}(u,v) \neq \emptyset$.

With this choice of weights, any hole pattern is equally likely under the distribution π since there are at most $n^2 + 1$ such patterns. When sampling from the distribution π a perfect matching is generated with probability at least $1/(n^2 + 1)$.

Weight function

We will not know w^* exactly but we will calculate weights w satisfying

$$\frac{w^*(u,v)}{2} \leq w(u,v) \leq 2w^*(u,v), \tag{2.9}$$

with very high probability.

Theorem

Assuming the weight function w satisfies inequality (2.9) for all $(u,v) \in V_1 \times V_2$ with $\mathcal{M}(u,v) \neq \emptyset$, then the mixing time of the Markov chain MC is bounded above by $\tau_M(\delta) = O(n^6 g(\log(\pi(M)^{-1}) + \log \delta^{-1}))$.

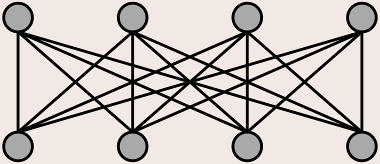
Weight function

Since the weight of an invalid matching is at most $1/n!$ and there are at most $n!$ possible matchings, the combined weight of all invalid matchings is at most 1.

$\Rightarrow$ "target" activities are $\lambda_G(e) = 1$ for all $e \in E$, and $\lambda_G(e) = 1/n!$ for all other e.

We are working with the complete graph, the initial choice is to set $\lambda(e) = 1$ for all e.

Phases:



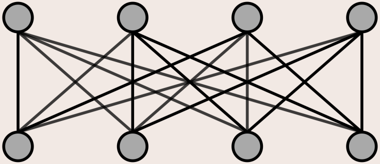
Weight function

Since the weight of an invalid matching is at most $1/n!$ and there are at most $n!$ possible matchings, the combined weight of all invalid matchings is at most 1.

$\Rightarrow$ "target" activities are $\lambda_G(e) = 1$ for all $e \in E$, and $\lambda_G(e) = 1/n!$ for all other e.

We are working with the complete graph, the initial choice is to set $\lambda(e) = 1$ for all e.

Phases:



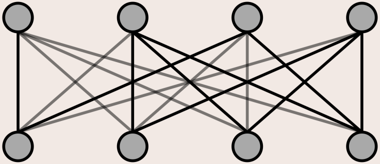
Weight function

Since the weight of an invalid matching is at most $1/n!$ and there are at most $n!$ possible matchings, the combined weight of all invalid matchings is at most 1.

- $\Rightarrow$ “target” activities are $\lambda_G(e) = 1$ for all $e \in E$, and $\lambda_G(e) = 1/n!$ for all other e.

We are working with the complete graph, the initial choice is to set $\lambda(e) = 1$ for all e.

Phases:



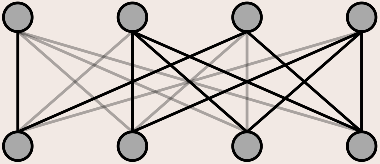
Weight function

Since the weight of an invalid matching is at most $1/n!$ and there are at most $n!$ possible matchings, the combined weight of all invalid matchings is at most 1.

$\Rightarrow$ "target" activities are $\lambda_G(e) = 1$ for all $e \in E$, and $\lambda_G(e) = 1/n!$ for all other e.

We are working with the complete graph, the initial choice is to set $\lambda(e) = 1$ for all e.

Phases:



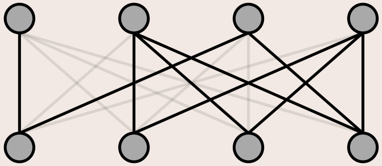
Weight function

Since the weight of an invalid matching is at most $1/n!$ and there are at most $n!$ possible matchings, the combined weight of all invalid matchings is at most 1.

$\Rightarrow$ "target" activities are $\lambda_G(e) = 1$ for all $e \in E$, and $\lambda_G(e) = 1/n!$ for all other e.

We are working with the complete graph, the initial choice is to set $\lambda(e) = 1$ for all e.

Phases:



Weight function

For some vertex v, the activities $\lambda(e)$ for all non-edges $e \notin E$ which are incident to v are updated to

$$\exp(-1/2)\lambda(e) \leq \lambda'(e) \leq \exp(1/2)\lambda(e)$$

We must consolidate our position by finding, for each pair (u, v), a better approximation to $w^*(u, v)$: one that is within ratio c for some $1 < c < 2$. For this purpose, we may use the identity

$$\frac{w(u,v)}{w^*(u,v)} = \frac{\pi(\mathcal{M}(u,v))}{\pi(\mathcal{M})}, \tag{2.10}$$

since $w(u, v)$ is known to us and the probabilities on the right hand side may be estimated to arbitrary precision by taking sample averages

Weight function

- Sample, in polynomial time, from a distribution $\hat{\pi}$, $d_{\mathrm{TV}}(\pi, \hat{\pi}) \leq \delta$.
- We generate S samples from $\hat{\pi}$, and for each pair $(u, v) \in V_1 \times V_2$ we consider the proportion $a(u, v)$ of samples with hole pair u, v, together with the proportion a of samples that are perfect matchings. Clearly,

$$\mathbb{E}_a(u, v) = \hat{\pi}(\mathcal{M}(u, v)), \text{ and } \mathbb{E}_a = \hat{\pi}(\mathcal{M}) \tag{2.11}$$

- We denote by $\hat{\eta}$ the failure probability.
- Provided the sample size S is large enough, $a(u, v)$ approximates $\hat{\pi}(\mathcal{M}(u, v))$ within ratio $c^{1/4}$, with probability at least $1 - \hat{\eta}$

Weight function

Condition (3.13) entails

$$\mathbb{E}_a(u,v) = \hat{\pi}(\mathcal{M}(u,v)) \geq \pi(\mathcal{M}(u,v)) - \delta \geq \frac{1}{4(n^2+1)} - \delta.$$

Assuming $\delta \leq 1/8(n^2+1)$, it follows from any of the standard Chernoff bounds, that $O(n^2 \log(1/\hat{\eta}))$ samples from $\hat{\pi}$ suffice to estimate $\mathbb{E}_a(u,v) = \hat{\pi}(\mathcal{M}(u,v))$ within ratio $c^{1/4}$ with probability at least $1-\hat{\eta}$.

Taking $c = 6/5$ and using $S = O(n^2 \log(1/\hat{\eta}))$ samples, we obtain refined estimates $w(u,v)$ satisfying

$$5w^*(u,v)/6 \leq w(u,v) \leq 6w^*(u,v)/5 \qquad (2.12)$$

with probability $1-(n^2+1)\hat{\eta}$.

Algorithm for weight estimation

Initialize $\lambda(u,v) \leftarrow 1$ for all $(u,v) \in V_1 \times V_2$.
Initialize $w(u,v) \leftarrow n$ for all $(u,v) \in V_1 \times V_2$.
While there exists a pair y, z with $\lambda(y,z) > \lambda_G(y,z)$ do:
  Take a sample of size S from MC with parameters λ, w,
    using a simulation of T steps in each case.
  Use the sample to obtain estimates $w'(u,v)$ satisfying
    condition (3.13), for all u, v, with high probability.
  Set $\lambda(y,v) \leftarrow \max\{\lambda(y,v)\exp(-1/2), \lambda_G(y,v)\}$, for all $v \in V_2$,
    and $w(u,v) \leftarrow w'(u,v)$ for all u, v.
Output the final weights $w(u,v)$.

Weight function

- $O(n^2 \log n)$ phases
- $O(n \log n)$ to reduce the activities of edges incident at each of these vertices to their final values
- $O(n^9 \log n \log(1/\hat{\eta}))$ the running time for each phase

  $\Rightarrow O(n^{11}(\log n)^2(\log n + \log \eta^{-1}))$
- Set $\hat{\eta} = O(\eta/(n^4 \log n))$, since there are $O(n^4 \log n)$ individual estimates to make in total.

Weight function

Lemma

The algorithm finds approximations $w(\cdot,\cdot)$ within a constant ratio of the ideal weights $w^*_G(\cdot,\cdot)$ associated with the desired activities λ_G in time $O(n^{11}(\log n)^2(\log n + \log \eta^{-1}))$, with failure probability η.

We need to set η so that the overall failure probability is strictly less than δ, e.x. $\eta = \delta/2$.

Congestion Bound

Reacap

Theorem

Assuming the weight function w satisfies inequality

$$\frac{w^*(u,v)}{2} \leq w(u,v) \leq 2w^*(u,v), \tag{3.13}$$

for all $(u,v) \in V_1 \times V_2$ with $\mathcal{M}(u,v) \neq \emptyset$, then the mixing time of the Markov chain MC is bounded above by $\tau_M(\delta) = O(n^6 g(\log(\pi(M)^{-1}) + \log \delta^{-1}))$.

Theorem

For an ergodic, reversible Markov chain with self-loop probabilities $P(M,M) \geq 1/2$ for all states M, and any initial state $M_0 \in \Omega$,

$$\tau_{M_0}(\delta) \leq \varrho(\ln \pi(M_0)^{-1} + \ln \delta^{-1})$$

Canonical Paths

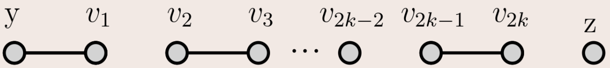
- **canonical path** $\gamma_{I,F}$ from each state $I \in \Omega$ to every other state $F \in \Omega$
- **flow** $f_{I,F}$ for all ordered pairs (I,F), $f_{I,F}(\gamma_{I,F}) = \pi(I)\pi(F)$.
- canonical paths for states $I \in \mathcal{N} = \Omega \backslash \mathcal{M}$ to states in $F \in \mathcal{M}$
- $\Gamma = \gamma_{I,F} : (I,F) \in \mathcal{N} \times \mathcal{M}$ and bound its congestion
- The canonical paths are defined by $I \oplus F$.
  - $\rightarrow$ alternating cycles
  - $\rightarrow$ single alternating path from y to z.

Canonical Paths

The alternating path $y = v_0 \sim \cdots \sim v_{2k+1} = z$ is unwound by:

(i) successively, for each $0 \leq i \leq k-1$, exchanging the edge (v_{2i}, v_{2i+1}) for the edge (v_{2i+1}, v_{2i+2})

(ii) adding the edge (v_{2k}, v_{2k+1})

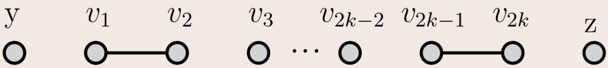


Canonical Paths

The alternating path $y = v_0 \sim \cdots \sim v_{2k+1} = z$ is unwound by:

(i) successively, for each $0 \leq i \leq k-1$, exchanging the edge (v_{2i}, v_{2i+1}) for the edge (v_{2i+1}, v_{2i+2})

(ii) adding the edge (v_{2k}, v_{2k+1})

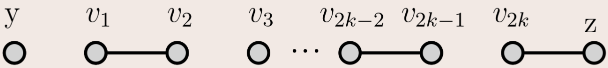


Canonical Paths

The alternating path $y = v_0 \sim \cdots \sim v_{2k+1} = z$ is unwound by:

(i) successively, for each $0 \leq i \leq k-1$, exchanging the edge (v_{2i}, v_{2i+1}) for the edge (v_{2i+1}, v_{2i+2})

(ii) adding the edge (v_{2k}, v_{2k+1})



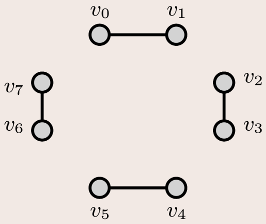
Canonical Paths

A cycle $v_0 \sim v_1 \sim \cdots \sim v_{2k} = v_0$, where we assume without loss of generality that the edge (v_0, v_1) belongs to I, is unwound by:

(i) removing the edge (v_0, v_1)

(ii) successively, for each $1 \leq i \leq k-1$, exchanging the edge $(v_{2i}-1, v_{2i})$ for the edge (v_{2i}, v_{2i+1})

(iii) adding the edge (v_{2k-1}, v_{2k})



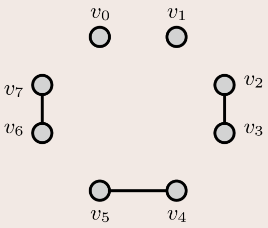
Canonical Paths

A cycle $v_0 \sim v_1 \sim \cdots \sim v_{2k} = v_0$, where we assume without loss of generality that the edge (v_0, v_1) belongs to I, is unwound by:

(i) removing the edge (v_0, v_1)

(ii) successively, for each $1 \leq i \leq k-1$, exchanging the edge $(v_{2i}-1, v_{2i})$ for the edge (v_{2i}, v_{2i+1})

(iii) adding the edge (v_{2k-1}, v_{2k})



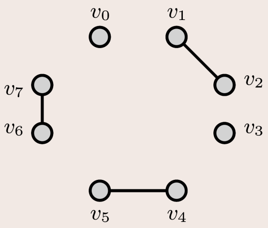
Canonical Paths

A cycle $v_0 \sim v_1 \sim \cdots \sim v_{2k} = v_0$, where we assume without loss of generality that the edge (v_0, v_1) belongs to I, is unwound by:

(i) removing the edge (v_0, v_1)

(ii) successively, for each $1 \leq i \leq k-1$, exchanging the edge $(v_{2i}-1, v_{2i})$ for the edge (v_{2i}, v_{2i+1})

(iii) adding the edge (v_{2k-1}, v_{2k})



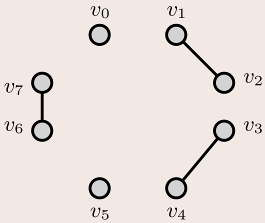
Canonical Paths

A cycle $v_0 \sim v_1 \sim \cdots \sim v_{2k} = v_0$, where we assume without loss of generality that the edge (v_0, v_1) belongs to I, is unwound by:

(i) removing the edge (v_0, v_1)

(ii) successively, for each $1 \leq i \leq k-1$, exchanging the edge $(v_{2i} - 1, v_{2i})$ for the edge (v_{2i}, v_{2i+1})

(iii) adding the edge (v_{2k-1}, v_{2k})



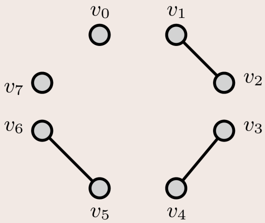
Canonical Paths

A cycle $v_0 \sim v_1 \sim \cdots \sim v_{2k} = v_0$, where we assume without loss of generality that the edge (v_0, v_1) belongs to I, is unwound by:

(i) removing the edge (v_0, v_1)

(ii) successively, for each $1 \leq i \leq k-1$, exchanging the edge $(v_{2i}-1, v_{2i})$ for the edge (v_{2i}, v_{2i+1})

(iii) adding the edge (v_{2k-1}, v_{2k})



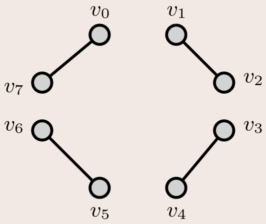
Canonical Paths

A cycle $v_0 \sim v_1 \sim \cdots \sim v_{2k} = v_0$, where we assume without loss of generality that the edge (v_0, v_1) belongs to I, is unwound by:

(i) removing the edge (v_0, v_1)

(ii) successively, for each $1 \leq i \leq k-1$, exchanging the edge $(v_{2i} - 1, v_{2i})$ for the edge (v_{2i}, v_{2i+1})

(iii) adding the edge (v_{2k-1}, v_{2k})



Canonical Paths

For each transition t, denote by

$$\mathrm{cp}(t) = \{(I, F) : \gamma_{I,F} \text{ contains t as a transition}\}$$

The set of canonical paths using that transition.

We define the **congestion** of Γ as

$$\varrho(\Gamma) = \max_t \left\{ \frac{L}{Q(t)} \sum_{(I,F) \in \mathrm{cp}(t)} \pi(I)\pi(F) \right\}, \qquad (3.14)$$

where L is an upper bound on the length $|\gamma_{I,F}|$ of any canonical path, and t ranges over all transitions.

Canonical Paths

Lemma

Assuming the weight function w satisfies inequality (3.13) for all $(u, v) \in V_1 \times V_2$, then $\varrho(\Gamma) \leq 48n^4$.

Congestion bound

Lemma

Denoting by $\mathcal{N} = \Omega \backslash \mathcal{M}$ the set of near-perfect matchings, there exists a flow f in MC with congestion

$$\varrho(f) \leq \left[2 + 4 \left(\frac{\pi(\mathcal{N})}{\pi(\mathcal{M}} + \frac{\pi(\mathcal{M})}{\pi(\mathcal{N})}) \right) \right] \varrho(\Gamma)$$

Theorem

For an ergodic, time reversible Markov chain with self-loop probabilities $P(M,M) \geq 1/2$ for all states M, and any initial state $M_0 \in \Omega$,

$$\tau_{M_0}(\delta) \leq \varrho(\ln \pi(M_0)^{-1} + \ln \delta^{-1})$$

Also: $\pi(\mathcal{N})/\pi(\mathcal{M}) = \Theta(n^2)$

Theorem

For an ergodic, time reversible Markov chain with self-loop probabilities $P(M,M) \geq 1/2$ for all states M, and any initial state $M_0 \in \Omega$,

$$\tau_{M_0}(\delta) \leq \varrho(\ln \pi(M_0)^{-1} + \ln \delta^{-1})$$

Also:$\pi(\mathcal{N})/\pi(\mathcal{M}) = \Theta(n^2)$

Theorem

Assuming the weight function w satisfies inequality (2.9) for all $(u,v) \in V_1 \times V_2$ with $\mathcal{M}(u,v) \neq \emptyset$, then the mixing time of the Markov chain MC is bounded above by $\tau_M(\delta) = O(n^6 g(\log(\pi(M)^{-1}) + \log \delta^{-1}))$.

Estimate the Permanent

Suppose G is a bipartite graph on $n+n$ vertices and that we want to estimate the number of perfect matchings in G within ratio $e^{\pm\varepsilon}$, for some specified $\varepsilon > 0$.

- Λ_i: the weight function associated with the pair (λ_i, w_i) in the i-th phase

Suppose G is a bipartite graph on $n+n$ vertices and that we want to estimate the number of perfect matchings in G within ratio $e^{\pm\varepsilon}$, for some specified $\varepsilon > 0$.

- Λ_i: the weight function associated with the pair (λ_i, w_i)

$$\Lambda(M) = \begin{cases} \lambda(M)w(u,v) & \text{, if } M \in \mathcal{M}(u,v) \text{ for some } u,v \\ \lambda(M) & \text{, if } M \in \mathcal{M} \end{cases}$$

Suppose G is a bipartite graph on $n+n$ vertices and that we want to estimate the number of perfect matchings in G within ratio $e^{\pm\varepsilon}$, for some specified $\varepsilon > 0$.

- Λ_i: the weight function associated with the pair (λ_i, w_i)
- $\Lambda_i(\Omega) = \sum_{M\in\Omega} \Lambda_i(M)$ is a "partition function" for weighted matchings after the ith phase.
- $\Lambda_0(\Omega) = (n^2+1)n!$
- At termination, $\Lambda_r(\Omega)$ is roughly n^2+1 times the number of perfect matchings in G.

Considering the “telescoping product”

$$\Lambda_r(\Omega) = \Lambda_0(\Omega) \times \frac{\Lambda_1(\Omega)}{\Lambda_0(\Omega)} \times \frac{\Lambda_2(\Omega)}{\Lambda_1(\Omega)} \times \cdots \times \frac{\Lambda_r(\Omega)}{\Lambda_{r-1}(\Omega)}, \tag{4.15}$$

we see that we may obtain a rough estimate for the number of perfect matchings in G by estimating in turn each of the ratios $\Lambda_{i+1}/\Lambda_i(\Omega)$.

Considering the "telescoping product"

$$\Lambda_r(\Omega) = \Lambda_0(\Omega) \times \frac{\Lambda_1(\Omega)}{\Lambda_0(\Omega)} \times \frac{\Lambda_2(\Omega)}{\Lambda_1(\Omega)} \times \cdots \times \frac{\Lambda_r(\Omega)}{\Lambda_{r-1}(\Omega)}, \tag{4.15}$$

we see that we may obtain a rough estimate for the number of perfect matchings in G by estimating in turn each of the ratios $\Lambda_{i+1}/\Lambda_i(\Omega)$.

The rule for updating the activities from λ_i to λ_{i+1}, together with the constraints on the weights, ensure

$$\frac{1}{4e} \leq \frac{\Lambda_{i+1}(M)}{\Lambda_i(M)} \leq 4e, \quad \text{for all } M \in \Omega$$

Thus we are in "good shape" to estimate the various ratios in (4.15) by Monte Carlo sampling.

Let π_i denote the stationary distribution of the Markov chain used in phase $i+1$, so that $\pi_i(M) = \Lambda_i(M)/\Lambda_i(\Omega)$. Let Z_i denote the random variable that is the outcome of the following experiment:

By running the Markov chain MC with parameters $\Lambda = \Lambda_i$
and $\delta = \varepsilon/80e^2r$, obtain a sample matching M from a distribution
that is within variation distance $\varepsilon/80e^2r$ of π_i.
Return $\Lambda_{i+1}(M)/\Lambda_i(M)$.

If we had sampled M from the exact stationary distribution π_i instead of an approximation, then the resulting modified random variable Z_i' would have satisfied

$$\mathbb{E}Z_i' = \sum_{M \in \Omega} \frac{\Lambda_i(M)}{\Lambda_i(\Omega)} \frac{\Lambda_{i+1}(M)}{\Lambda_i(M)} = \frac{\Lambda_{i+1}(\Omega)}{\Lambda_i(\Omega)}$$

We must settle for

$$\exp\left(-\frac{\varepsilon}{4r}\right)\frac{\Lambda_{i+1}(\Omega)}{\Lambda_i(\Omega)} \leq \mathbb{E}Z_i \leq \exp\left(\frac{\varepsilon}{4r}\right)\frac{\Lambda_{i+1}(\Omega)}{\Lambda_i(\Omega)}$$

Suppose s independent trials are conducted for each i using the above experiment, and denote by $\bar{Z}_i$ the sample mean of the results.

Then $\mathbb{E}\bar{Z}_i = \mathbb{E}Z_i$, and

$$\exp\left(-\frac{\varepsilon}{4}\right)\frac{\Lambda_r(\Omega)}{\Lambda_0(\Omega)} \leq \mathbb{E}(\bar{Z}_0\bar{Z}_1 \ldots \bar{Z}_{r-1}) \leq \exp\left(\frac{\varepsilon}{4}\right)\frac{\Lambda_r(\Omega)}{\Lambda_0(\Omega)} \tag{4.16}$$

$s = \Theta(r\varepsilon^{-1})$ sufficiently large

By Chebyshev's inequality,

$$Pr[\exp(-\varepsilon/4)\mathbb{E}(\bar{Z}_0 \cdots \bar{Z}_{r-1}) \leq (n^2+1)n!\bar{Z}_0 \cdots \bar{Z}_{r-1} \leq \exp(\varepsilon/4)\mathbb{E}(\bar{Z}_0 \cdots \bar{Z}_{r-1})] \geq \frac{11}{12}$$

By Chebyshev's inequality,

$$Pr[\exp(-\varepsilon/4)\mathbb{E}(\bar{Z}_0\cdots\bar{Z}_{r-1}) \leq (n^2+1)n!\bar{Z}_0\cdots\bar{Z}_{r-1} \leq \exp(\varepsilon/4)\mathbb{E}(\bar{Z}_0\cdots\bar{Z}_{r-1})] \geq \frac{11}{12}$$

Combining the inequalities with the fact that $\Lambda_0(\Omega) = (n^2+1)n!$

$$Pr[\exp(-\varepsilon/2)\Lambda_r(\Omega) \leq (n^2+1)n!\bar{Z}_0\cdots\bar{Z}_{r-1} \leq \exp(\varepsilon/2)\Lambda_r(\Omega)] \geq \frac{11}{12}$$

Denote by $\mathcal{M}_G \subset \mathcal{M}$ the set of perfect matchings in the graph G. The inequality provides an effective estimator for $\Lambda_r(\mathcal{M}_G)$, already yielding a rough estimate for $|\mathcal{M}_G|$.

Consider the following experiment:

By running the Markov chain MC with parameters $\Lambda = \Lambda_r$ and $\delta = \varepsilon/80e^2$, obtain a sample matching M from a distribution that is within variation distance $\varepsilon/80e^2$ of π_r.
Return 1 if $M \in \mathcal{M}_G$, and 0 otherwise.

Y: the outcome of the experiment.
$\bar{Y}$: denotes the sample mean of $s' = \Theta(n^2\varepsilon^{-1})$ independent trials.

Permanent Estimator

From the experiments we obtain:

$$Pr[\exp(-\varepsilon)|\mathcal{M}_G| \leq (n^2+1)n!\bar{Y}\bar{Z}_0\cdots\bar{Z}_{r-1} \leq \exp(\varepsilon)|\mathcal{M}_G|] \geq \frac{5}{6}$$

The running time is $O(\varepsilon^{-2}n^{11}(\log n)^3)$

Arbitrary Weights

Arbitrary Weights

Let an arbitrary matrix A with nonnegative entries

- $a_{\text{max}} = \max_{i,j} a(i,j)$
- $a_{\text{min}} = \min_{i,j} a(i,j)$
- $\text{per}(A) > 0 \ \Rightarrow \text{per}(A) \geq (a_{\text{min}})^n$
- Rounding zero entries $a(i,j)$ to $(a_{\text{min}})^n/n!$,

Generalization

Initialize $\lambda(u,v) \leftarrow a_{\max}$ for all $(u,v) \in V_1 \times V_2$.
Initialize $w(u,v) \leftarrow n a_{\max}$ for all $(u,v) \in V_1 \times V_2$.
While there exists a pair y,z with $\lambda(y,z) > a(y,z)$ do:
    Take a sample of size S from MC with parameters λ, w,
        using a simulation of T steps in each case.
    Use the sample to obtain estimates $w'(u,v)$ satisfying
        condition (3.13), for all u,v, with high probability.
    Set $\lambda(y,v) \leftarrow \max\{\lambda(y,v)\exp(-1/2), a(y,v)\}$, for all $v \in V_2$,
        and $w(u,v) \leftarrow w'(u,v)$ for all u,v.
Output the final weights $w(u,v)$.

!The running time of this algorithm is polynomial in n and $\log(a_{\max}/a_{\min})$.

Conclusion

Conclusion

- We obtain a FPRAS for the Permanent
- Interesting techniques
- The disproportionate number of near-perfect matchings is resolved.
- Several counting problems are reducible via approximation-preserving reductions to the $0, 1$ permanent,

Conclusion

- We obtain a FPRAS for the Permanent
- Interesting techniques
- The disproportionate number of near-perfect matchings is resolved.
- Several counting problems are reducible via approximation-preserving reductions to the $0, 1$ permanent,
  - $\rightarrow$ computing the number of labeled subgraphs of G with a specified degree sequence of an arbitrary bipartite graph G.

Conclusion

- We obtain a FPRAS for the Permanent
- Interesting techniques
- The disproportionate number of near-perfect matchings is resolved.
- Several counting problems are reducible via approximation-preserving reductions to the $0, 1$ permanent,
  - → computing the number of labeled subgraphs of G with a specified degree sequence of an arbitrary bipartite graph G.
  - → counting the number of 0, 1-flows of an arbitrary directed graph G

Thank you!

References

A Z Broder. "How Hard is It to Marry at Random? (On the Approximation of the Permanent)". In: *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*. STOC '86. Berkeley, California, USA: Association for Computing Machinery, 1986, pp. 50–58. isbn: 0897911938. doi: 10.1145/12130.12136.

M. Jerrum. *Counting, Sampling and Integrating: Algorithms and Complexity*. Springer, 2004. isbn: 978-3-0348-8005-3. doi: https://doi.org/10.1007/978-3-0348-8005-3.

- Mark Jerrum, Alistair Sinclair, and Eric Vigoda. "A Polynomial-Time Approximation Algorithm for the Permanent of a Matrix with Nonnegative Entries". In: *J. ACM* 51.4 (July 2004), pp. 671–697. issn: 0004-5411. doi: 10.1145/1008731.1008738.
- Alistair Sinclair. "Improved Bounds for Mixing Rates of Markov Chains and Multicommodity Flow". In: *Combinatorics, Probability and Computing* 1.4 (1992), pp. 351–370. doi: 10.1017/S0963548300000390.